Pitch Shifting by H^∞-Optimal Variable Fractional Delay Filters

Masaaki Nagahara * **Kazutaka Hanibuchi** ** **Yutaka Yamamoto** ***

* *Graduate School of Informatics, Kyoto University, Kyoto, 606-8501, JAPAN (e-mail: nagahara@ieee.org).*
** *IHI Corporation, Tokyo, 135-8710, JAPAN(e-mail: zam0082@gmail.com).*
*** *Graduate School of Informatics, Kyoto University, Kyoto, 606-8501, JAPAN (e-mail: yy@i.kyoto-u.ac.jp).*

Abstract: Fractional delay filters are those that are designed to delay the input samples by a fractional amount of the sampling period. Since the delay is fractional, the intersample behavior of original analog signals becomes crucial. For this, we propose an optimal design via sampled-data H^∞ control theory. By this theory, our design problem is equivalently reduced to a discrete-time H^∞ optimization, and, an analytical solution is obtained under an assumption on the original analog signals. Using this analytical solution, we propose a sampling rate conversion with arbitrary conversion rate. This conversion is much faster than conventional methods using an upsampler, a digital filter, and a downsampler. We also show an application of the proposed sampling rate conversion to pitch shifting of guitar sounds. A design example is shown to illustrate the advantage of the proposed method.

Keywords: Digital signal processing, sampled-data control, sampling rate conversion, pitch shifting.

1. INTRODUCTION

Fractional delay filters are to delay the input signal by a fraction of the sampling period. Such a filter has wide applications in signal processing, including digital communications, speech processing and digital modeling of musical instruments [Laakso et al. (1996); Välimäki and Laakso (2000)].

Conventionally, fractional delay filters are designed in the discrete-time domain by assuming that the incoming continuous-time signals are fully band-limited up to the Nyquist frequency. Under this assumption, the optimal fractional delay filter is given by a delayed sinc function. Such a filter is however not realizable because of its non-causality and instability, and hence many studies have focused on approximating the ideal filter [Laakso et al. (1996); Hermanowicz (1992); Pei and Wang (2004)].

Although these studies are based on the band-limiting assumption, no real analog signals are fully band-limited, and hence the assumption is not realistic. Moreover, by their very nature, fractional delay filters should reconstruct intersample signal values. It is, therefore, necessary for designing the filters to take account of high-frequency components beyond the Nyquist frequency and the intersample behavior of input analog signals.

For such problems, *sampled-data control theory* provides an optimal platform. Based on this theory, the design problem of fractional delay filters has been formulated as a sampled-data H^∞ optimization [Nagahara and Yamamoto (2003, 2005)]. In particular, the analytical expression for the H^∞ optimal fractional delay filter is given under the assumption that the underlying frequency characteristic of the continuous-time input signal is governed by a low-pass filter of first order.

By this analytical solution, we propose a method for fast sampling rate conversion. Conventionally, sampling rate conversion is executed by an upsampler, a digital filter and a downsampler [Vaidyanathan (1993)]. This scheme is effective if the conversion rate is a fraction of small integers (e.g., 2/3). It however seriously increases the computation load when both integers become very large, e.g., 44100/48000= 147/160 as in CD to DAT conversion [Zölzer (2008)]. Moreover, if the rate is an irrational number, this scheme cannot be used. On the other hand, by using the fractional delay filters, we can convert digital signals with arbitrary positive real rate [Ramstad (1984)]. In addition, while conventional design of the digital filter in sampling rate conversion depends on the band-limiting assumption mentioned above, our design can take account of analog characteristic of input signals.

By using this sampling rate converter, we propose a new pitch shifting method. Pitch shifting is a technique for raising or lowering the original pitch of audio signals. This is often used in synthesizing musical tone from a recorded signal of a musical instrument with a fixed fundamental frequency [Roads (1996)]. A naive method of pitch shifting does not conserve the signal length, for example, if the pitch is made higher, the length becomes

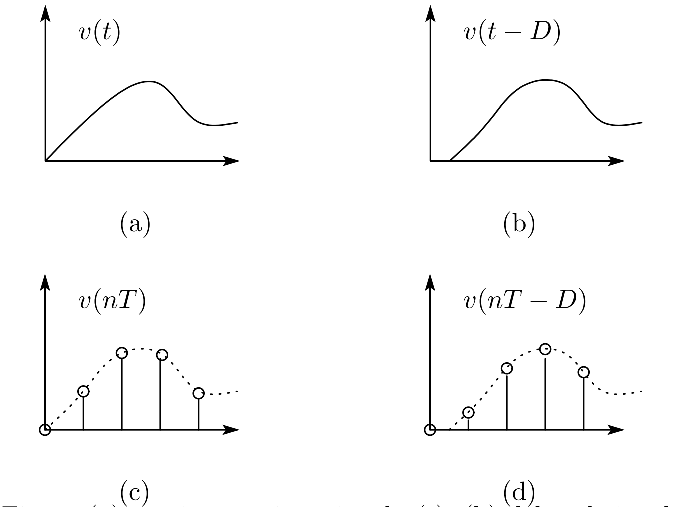


Fig. 1. (a) continuous-time signal $v(t)$, (b) delayed signal $v(t-D)$, (c) sampled signal $v(nT)$, (d) delayed and sampled signal $v(nT-D)$

short. To conserve the length, we also propose a method by adding (or cutting) the shifted signal.

The article is organized as follows. Section 2 defines fractional delay filters, and formulates the design problem as a sampled-data H^∞ optimization and show its solution. In Section 3, we propose a new sampling rate conversion method based on sampled-data H^∞ optimal fractional delay filters, and propose a method for conservation of the signal length. Numerical examples are shown in Section 4 to illustrate the superiority of the proposed method. Section 5 makes a conclusion.

2. DESIGN OF FRACTIONAL DELAY FILTERS

2.1 Fractional delay filters

Consider a continuous-time signal $v(t)$, $t \in \mathbb{R}_+$ as shown in Fig. 1 (a). Delaying the signal $v(t)$ by the continuous-time delay operator e^{-Ds} $(D > 0)$, we obtain the delayed signal $v(t-D)$ shown in Fig. 1 (b). Then the signal $v(t-D)$ is sampled with period T and becomes a discrete-time signal $v(nT-D)$, $n \in \mathbb{Z}_+$ as shown in Fig. 1 (d).

On the other hand, consider the sampled signal $v(nT)$, as shown in Fig. 1 (c). Then we define the ideal fractional delay filter as follows:

Definition 1. The *ideal fractional delay filter* K^{id} with delay time D is defined by

$$K^{\mathrm{id}} : v(nT) \mapsto v(nT-D).$$

Note that if the delay D is an integer multiple of the sampling period T, that is, $D = mT$, $m \in \mathbb{Z}_+$, the ideal filter K^{id} is the discrete-time delay z^{-m}. Moreover, if the input analog signal $v(t)$ is fully band-limited up to the Nyquist frequency $\omega_N := \pi/T$, that is, the Fourier transform $V(j\omega)$ of $v(t)$ vanishes at $|\omega| \geq \omega_N$, the impulse response of the ideal fractional delay filter is obtained as follows [Laakso et al. (1996)]:

$$k^{\mathrm{id}}[n] = \frac{\sin \pi(nT-D)/T}{\pi(nT-D)/T} = \mathrm{sinc}\left[\frac{(nT-D)}{T}\right], \qquad (1)$$
$$n = 0, \pm 1, \pm 2, \ldots, \quad \mathrm{sinc}\, t := (\sin \pi t)/(\pi t).$$

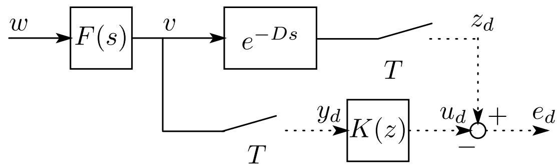


Fig. 2. Error system $\mathcal{E}$ for designing fractional delay filter $K(z)$

The frequency response of this ideal filter is derived by the Fourier transform:

$$K^{\mathrm{id}}(e^{j\omega T}) = e^{-j\omega D}, \quad \omega \leq \omega_N. \qquad (2)$$

Since the impulse response (1) does not vanish at $n = -1, -2, \ldots$ and is not absolutely summable, the ideal filter must be noncausal and unstable, and hence the ideal filter cannot be realized. Conventional designs thus aim at approximating (1) or (2) to a causal and stable filter via a window method, maximally-flat FIR approximations, weighted least-squares approximation, and so forth [Laakso et al. (1996)].

These methods are based upon the band-limiting assumption as mentioned above. In practice, however, real analog signals always contain some frequency components beyond the Nyquist frequency. In what follows, we formulate the design problem of fractional delay filters without such an assumption by using the sampled-data H^∞ optimization.

2.2 Design problem of fractional delay filters

Consider the block diagram Fig. 2. In this diagram, $F(s)$ governs the frequency-domain characteristic [1] of the input signal $w \in L^2$. Then, the upper path of the diagram is the ideal process of the fractional delay filter (the process (a) → (b) → (d) in Fig. 1), that is, the continuous-time signal v is delayed by the continuous-time delay e^{-Ds}, then sampled with period T, and becomes a discrete-time signal $z_d \in \ell^2$. On the other hand, the lower path is the real process ((a) → (c) → (d) in Fig. 1), that is, the continuous-time signal v is sampled with period T, filtered by $K(z)$ to be designed, and becomes a discrete-time signal $u_d \in \ell^2$.

Put $e_d := z_d - u_d$ (the difference between the ideal output z_d and the real output u_d), and let $\mathcal{E}$ denote the system from $w \in L^2$ to $e_d \in \ell^2$. Then our problem is to find the filter $K(z)$ which minimizes the H^∞ norm of the error system $\mathcal{E}$.

Problem 1. Given a stable, strictly proper $F(s)$, a delay time $D > 0$, and a sampling period $T > 0$, find the digital filter $K(z)$ which minimizes

$$\|\mathcal{E}\|_\infty := \sup_{\substack{w \in L^2 \\ w \neq 0}} \frac{\|e_d\|_{\ell^2}}{\|w\|_{L^2}}. \qquad (3)$$

2.3 Design of Optimal Filters

Assume the delay D is decomposed by

$$D = mT + d,$$

[1] Conventionally, $F(s)$ is assumed to be an ideal filter such that $F(j\omega) = 0$, $|\omega| \geq \omega_N$ (the Nyquist frequency).

where m is a non-negative integer and $0 < d \le T$. Assume also that the filter $F(s)$ is a first-order low-pass filter with cutoff frequency $\omega = \omega_c$, that is,

$$F(s) = \frac{\omega_c}{s + \omega_c}.$$

Then the H^∞ optimal filter $K(z)$ which minimizes (3) can be obtained with an analytical representation [Nagahara and Yamamoto (2003, 2005)]:

Theorem 1. The optimal filter $K(z)$ is obtained by

$$K(z) = a_0(d)z^{-m} + a_1(d)z^{-m-1}, \tag{4}$$

where

$$a_0(d) := \frac{\sinh(\omega_c(T-d))}{\sinh(\omega_c T)}, \quad a_1(d) := e^{-\omega_c T}(e^{\omega_c d} - a_0). \tag{5}$$

Moreover, the optimal value of $\|\mathcal{E}\|_\infty$ is

$$\|\mathcal{E}\|_\infty := \sqrt{\frac{\omega_c \sinh(\omega_c d)\sinh(\omega_c(T-d))}{\sinh(\omega_c T)}}. \tag{6}$$

Remark 2. When $d = 0$, the formula (5) gives $a_0(0) = 1$, $a_1(0) = 0$ and the optimal filter in (4) becomes $K(z) = z^{-m}$. Also when $d = T$, we have $a_0(T) = 0$, $a_1(T) = 1$, and the optimal filter becomes $K(z) = z^{-m-1}$. In both cases, the optimal value of $\|\mathcal{E}\|_\infty$ given in (6) becomes 0, that is, perfect reconstruction.

Remark 3. Since $\sinh(x) \approx x$ and $e^{-x} \approx 1 - x$ when x is sufficiently small, we have

$$a_0(d) \approx \frac{\omega_c(T-d)}{\omega_c T} = \frac{T-d}{T},$$
$$a_1(d) \approx 1 - \omega_c(T-d) - (1-\omega_c T)\cdot\frac{T-d}{T} = \frac{d}{T},$$

when $\omega_c T$ is sufficiently small, that is, when the frequency ω_c is much smaller than the sampling frequency $1/T$. Then the optimal filter becomes

$$K(z) = \frac{T-d}{T}z^{-m} + \frac{d}{T}z^{-m-1}.$$

This is the linear interpolation between two points $y_d[m]$ and $y_d[m+1]$. In other words, if the original signals contain very few high-frequency components relative to the sampling frequency $1/T$ and the decay of the Fourier transform is the first order, then the linear interpolation is approximately optimal.

3. FAST SAMPLING RATE CONVERSION AND PITCH SHIFTING

3.1 Fast sampling rate conversion

Consider a continuous-time signal $\{v(t)\}_{t\in\mathbb{R}_+}$. Assume that we are given sampled data $v[m] := v(mT)$, $m \in \mathbb{Z}_+$ where $T > 0$ is a sampling period. Then we execute sampling rate conversion on this discrete-time signal. By r, we denote the conversion rate. We assume r is a positive real number. Then sampling rate conversion aims at estimating the values of $\{v(krT)\}_{k\in\mathbb{Z}_+}\}$. Since we miss the intersample values of $\{v(t)\}_{t\in\mathbb{R}_+}$, the exact estimation is impossible.

Conventionally, this estimation is done by an upsampler, a digital filter and a downsampler [Vaidyanathan (1993); Zölzer (2008)]. Fig. 3 shows this scheme. As mentioned in Section 1, this scheme cannot be used if the rate r is irrational.

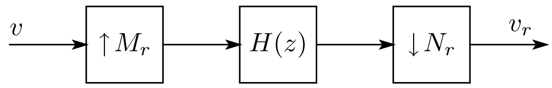


Fig. 3. Sampling rate conversion by upsampler-filter-downsampler system. The rate is $r = N_r/M_r$ where M_r and N_r are positive integers.

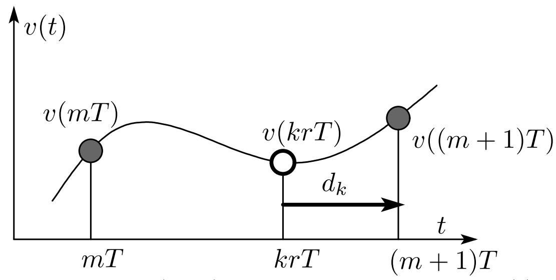


Fig. 4. The value $v(krT)$ is given by shifting $v(t)$ by $d_k = (m+1)T - krT$ and sampling at $t = (m+1)T$.

Alternatively, we adopt a sampling rate conversion by using fractional delay filters [Ramstad (1984)]. In this conversion, we use the sampled-data H^∞ optimal fractional delay filter given in Theorem 1. Let us consider estimation of the value $v(krT)$ where k is a positive integer. Assume that the time krT satisfies $mT < krT \le (m+1)T$ where m is a non-negative integer. Let $d_k := (m+1)T - krT$. Then we have

$$\begin{aligned} v(krT) &= v((m+1)T - (m+1)T + krT) \\ &= v((m+1)T - d_k) \\ &= v(t - d_k)|_{t=(m+1)T}. \end{aligned}$$

That is, the value $v(krT)$ is obtained by delaying $v(t)$ by d_k and sampling at time $t = (m+1)T$ (see Fig. 4). Therefore, the estimation $v_r[k]$ for $v(krT)$ can be obtained by the fractional delay filter given in (4) as

$$v_r[k] = a_0(d_k)v((m+1)T) + a_1(d_k)v(mT),$$

where $a_0(\cdot)$ and $a_1(\cdot)$ are given in (5). Note that this filter is a two-tap FIR filter and the estimation needs much fewer computation than the conventional scheme shown in Fig. 3. Also we emphasize that the computation load is the same for arbitrary real rate r while that of the conventional scheme increases the load when N_r and M_r are large. This is an advantage over conventional methods in the case of real-time processing. In addition, while conventional design of the digital filter in sampling rate conversion depends on the band-limiting assumption mentioned above, our design can take account of analog characteristic of input signals.

The algorithm of the proposed sampling rate conversion is shown in Algorithm 1. In this algorithm, we define $v[m] := v(mT)$, $m = 0, 1, 2, \ldots$.

3.2 Pitch shifting

Pitch shifting is a technique for raising or lowering the original pitch of audio signals. To accomplish this, we can use a sampling rate converter; convert the sampling period T to rT with $r > 0$, and play the converted signal with the original sampling period T. By this process, we have a raised (if $r > 1$) or lowered (if $r < 1$) pitch. For

Algorithm 1 Sampling rate conversion

$v_r[0] := v[0]$
$k := 1$
for $m = 0, 1, 2, \ldots$ **do**
  while $kr \leq m + 1$ **do**
    $d := (m+1)T - krT$
    $v_r[k] := a_0(d)v[m+1] + a_1(d)v[m]$
    $k := k + 1$
  end while
end for

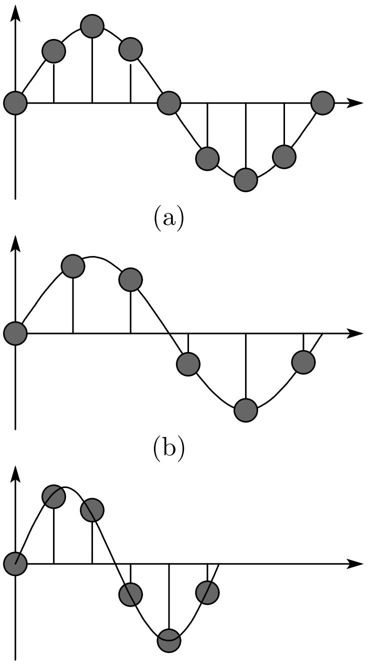


Fig. 5. Pitch shifting by sampling rate converter: (a) original digital signal, (b) sampling rate conversion, and (c) pitch sifting by processing the signal in (b) with the same sampling rate as (a).

example [2], the pitch 110 Hz, that is A_2 note is converted to D_3 note ($110 \times 2^{5/12} \approx 146.83$ Hz) with $r = 2^{-5/12}$. This is analogous to listening to music by fast- or slow-forwarding a cassette tape. This process is effectively done by the proposed algorithm (Algorithm 1) if the sound is given by digital data. We illustrate this process in Fig. 5.

A problem of this process is that the length of signal is also changed (compare (a) and (c) in Fig. 5). Let L be the length of the original signal. Then if we shift its pitch by Algorithm 1 with rate $r > 1$, the length of the converted signal will be $L/r < L$. To adjust the length, we should fill in a signal whose length is $L - L/r$. When $r < 1$, then cut out a signal whose length is $L/r - L$. To execute this, we split the original signal into *attack* and *decay* portions [Roads (1996)]. Fig. 6 shows an example of this representation. The attack is the transient response of the sound, and it is known that this portion plays the most important role in distinguishing a particular instrument [Risset and Wessel (1998)]. On the other hand, the decay is the steady-state response with decay. Fig. 7 shows a part of this stage. We can see that the signal in this portion is

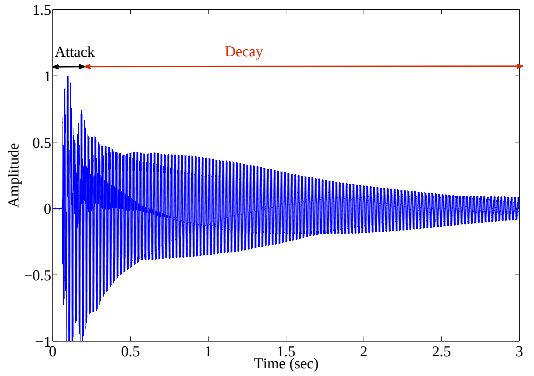


Fig. 6. Guitar sound with the fundamental frequency 110 Hz (A_2) split into attack and decay portions

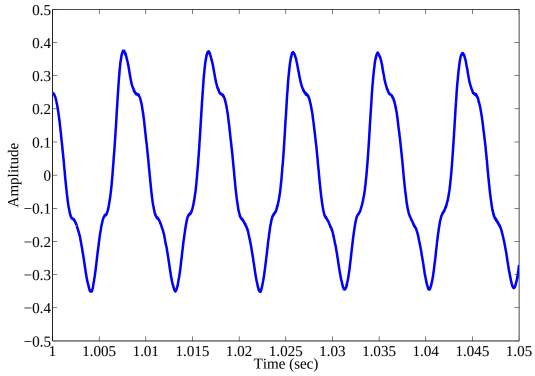


Fig. 7. Decay: locally periodic signal with the fundamental frequency 110 Hz (A_2)

locally a periodic signal. Based on this fact, we split the processing in two stages.

The first stage is for the attack portion. Let $L_0 > 0$ be the length of the attack signal. We convert this signal by Algorithm 1 with r, and we obtain a signal with length L_0/r. The next stage is for the decay portion. We split the decay signal (note that this signal has the length $L - L_0$) into N frames of equal length τ. Let f be the fundamental frequency of the original signal and $l := 1/f$ be the fundamental period. We choose the frame length τ to satisfy the following:

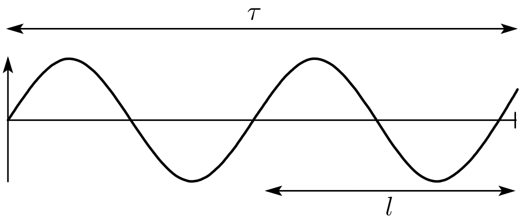
$$\tau = \frac{L - L_0}{L} \cdot \frac{l}{|r-1|} \cdot n, \tag{7}$$

where n is an integer satisfying

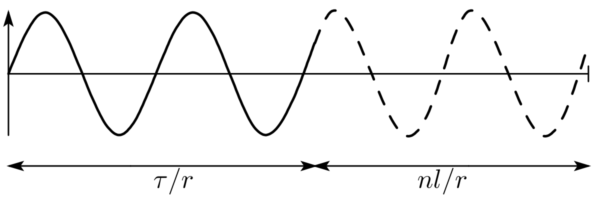
$$n \geq \frac{L}{L - L_0} \max\{r - 1, r^{-1} - 1\}. \tag{8}$$

We assume that $N := (L - L_0)/\tau$ is a positive integer (if not, we truncate the original signal). Then we convert each frame by Algorithm 1. If $r > 1$, we cut off the fundamental wave with period l/r from the converted frame, and add n pitch-shifted fundamental waves at the end of the frame. This procedure is illustrated in Fig. 8. If $r < 1$, we cut out n fundamental waves. Note that in both cases the converted frame includes at least one fundamental wave if (8) is satisfied. Then the length τ' of converted frames becomes (see Fig. 8)

[2] In this article, we consider *equal temperament* [French (2009)].



(a) Original frame



(b) Processed frame

Fig. 8. Processing of a frame in decay portion: (a) Original frame with length τ and fundamental period l, (b) Processed frame added $n = 2$ pitch-shifted fundamental waves.

$$\tau' = \frac{\tau}{r} + \frac{l}{r} \cdot n.$$

Then the equation (7) gives the length of the converted signal in the decay portion as follows

$$\begin{aligned} \tau' N &= \left(\frac{\tau}{r} \pm \frac{nl}{r}\right) \frac{L - L_0}{\tau} \\ &= \frac{L - L_0}{r} \pm \frac{nl(L - L_0)}{r\tau} \\ &= \frac{1}{r}\left(L - L_0 \pm L|r - 1|\right) \\ &= L - \frac{L_0}{r}, \end{aligned}$$

where the double sign corresponds to the two conditions $r > 1$ or $r < 1$. Since the length of the converted attack portion is L_0/r, the length of the whole converted signal becomes L, which is equal to the original length.

4. DESIGN EXAMPLES

We first demonstrate sampling rate conversion. We set the analog filter $F(s)$ as

$$F(s) = \frac{0.1}{s + 0.1},$$

that is, we set $\omega_c = 0.1$. The sampling time T is set to be 1. We consider here sampling rate conversion with irrational rate $r = 1/\sqrt{10}$. The input signal is a triangle wave, and we convert the sampling rate by Algorithm 1. For comparison, we also execute sampling rate conversion by the conventional scheme shown in Fig. 3. The number M_r and N_r are given by rational approximation of $r = 1/\sqrt{10}$ as

$$M_r = 19, \quad N_r = 6.$$

With this approximation, the approximation error is bounded by

$$\frac{1}{\sqrt{10}} - 6/19 < 10^{-3},$$

where we used the MATLAB function

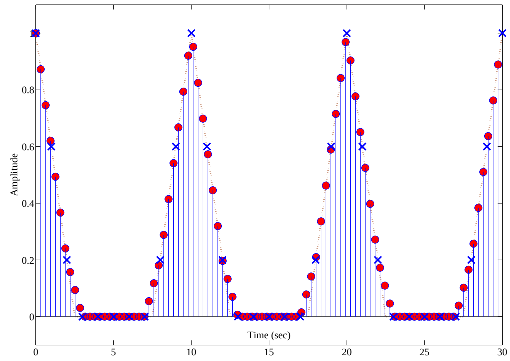


Fig. 9. Sampling rate conversion by the proposed method: original signal (dash), sampled-data (×), and converted data (∘)

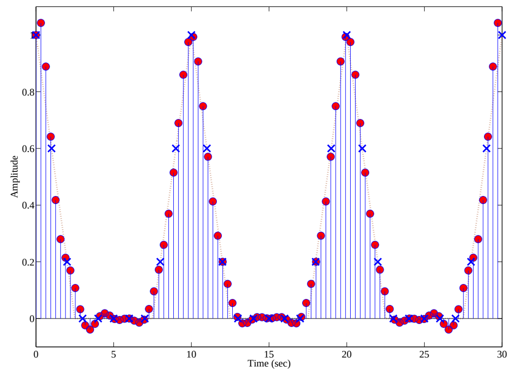


Fig. 10. Sampling rate conversion by the conventional method: original signal (dash), sampled-data (×), and converted data (∘)

```
rat(1/sqrt(10), 1e-3)
```

for M_r and N_r. The sampling rate conversion by the scheme shown in Fig. 3 is easily executed by the MATLAB function

```
resample(v,19,6)
```

We also obtain the filter $H(z)$ in Fig. 3 by the same MATLAB function, which is a 381-tap FIR filter.

The converted signals are shown in Fig. 9 (proposed method) and Fig. 10 (conventional method). We can see that the proposed method produces much better reconstruction than the conventional method. In fact, the ℓ^2 norm of the reconstruction error is 0.2530 for the proposed method and 0.4820 for the conventional one. That is, our method is about 50% better than the conventional method in ℓ^2 norm of the reconstruction error.

Moreover, our method requires much fewer computation than the conventional method. The computation time for computation by the proposed method is 0.000143 (sec), while that by the conventional method is 0.001788 (sec). That is, our method is about 10 times faster than the conventional method. This shows the effectiveness of the proposed method.

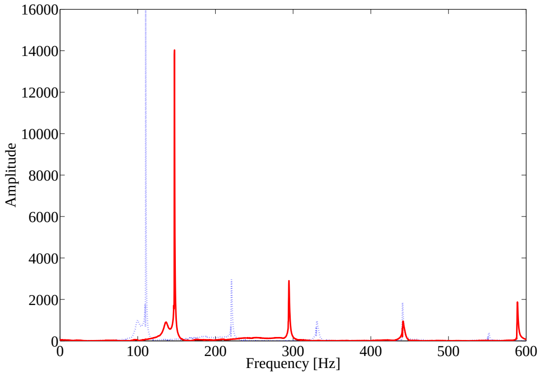


Fig. 11. Frequency response: original signal with fundamental frequency 110 Hz (dots) and pitch-shifted signal with fundamental frequency $110 \times 2^{5/12} \approx 146.83$ Hz (solid).

Then by using the proposed sampling rate converter, we shift the pitch of a guitar sound with A_2 (110 Hz) shown in Fig. 6 to D_3 note ($110 \times 2^{5/12} \approx 146.83$ Hz). For this, we take $r = 2^{-5/12}$. Fig. 11 shows the frequency responses of the original signal with fundamental frequency 110 Hz (dots) and the pitch-shifted signal. The frequency response of the shifted signal shows that the processed signal has a valid fundamental frequency $110 \times 2^{5/12} \approx 146.83$ Hz and the harmonics with $110 \times 2^{5/12} \times n$, $n = 2, 3, 4$.

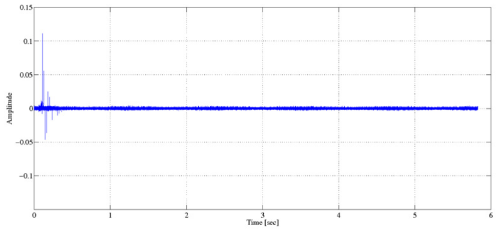
We show another example of pitch shifting. We shift the note A_2 (110Hz) used above to A_3 (220 Hz). Then the shifted signal is again shifted down to the original note A_2. We measure the error between the original sound and the processed one. Figure 12(a) shows the reconstruction error by the proposed method. For comparison, we processed by the phase vocoder [Portoff (1976); Ellis (2002)], which is widely used in pitch shifting. Figure 12(b) shows the error of the phase vocoder. We can see that the phase vocoder produces much larger errors than the proposed method. This shows effectiveness of our method.

One can listen to the processed sounds at the following web page:

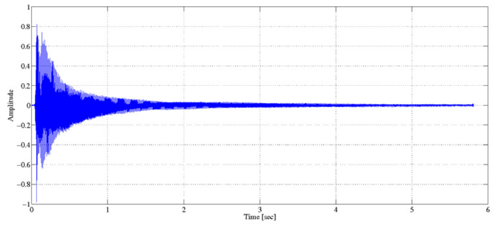
`http://www-ics.acs.i.kyoto-u.ac.jp/~nagahara/ps/`

5. CONCLUSION

We have presented a new method of designing variable fractional delay filters via sampled-data H^∞ optimization. We have given the H^∞ optimal filter having delay time variable D as an adaptive variable, when the frequency distribution of the input analog signal is modeled as a first-order low-pass filter. Moreover, by using this optimal filter, we propose a fast sampling rate conversion and a new pitch shifting which conserves the signal length. A design example shows that the proposed sampling rate conversion exhibits a much more satisfactory performance than conventional ones, and also the effectiveness of the proposed sampling rate conversion. We have also shown an example of the proposed pitch shifting. Future work may include the design of H^∞-optimal fractional delay filters for second or higher order low-pass filters.



(a) Proposed method



(b) Phase vocoder

Fig. 12. Reconstruction error

REFERENCES

Ellis, D.P.W. (2002). A phase vocoder in Matlab. http://www.ee.columbia.edu/~dpwe/resources/matlab/pvoc/.

French, R.M. (2009). *Engineering the Guitar*. Springer.

Hermanowicz, E. (1992). Explicit formulas for weighting coefficients of maximally flat tunable FIR delayers. *Electronics Letters*, 28, 1936–1937.

Laakso, T.I., Välimäki, V., Karjalainen, M., and Laine, U.K. (1996). Splitting the unit delay. *IEEE Signal Processing Mag.*, 13, 30–60.

Nagahara, M. and Yamamoto, Y. (2003). Optimal design of fractional delay filters. In *Proc. of 39th Conf. on Decision and Control*, 6539–6544.

Nagahara, M. and Yamamoto, Y. (2005). Optimal design of fractional delay FIR filters without band-limiting assumption. In *IEEE ICASSP'05*, 221–224.

Pei, S.C. and Wang, P.H. (2004). Closed-form design of all-pass fractional delay filters. *IEEE Signal Processing Lett.*, 11, 788–791.

Portoff, M.R. (1976). Implementation of the digital phase vocoder using the fast fourier transform. *IEEE Trans. Acoust., Speech, Signal Processing*, 24(3), 243–248.

Ramstad, T.A. (1984). Digital methods for conversion between arbitrary sampling frequencies. *IEEE Trans. Acoust., Speech, Signal Processing*, 32(3), 577–591.

Risset, J.C. and Wessel, D.L. (1998). Exploration of timbre by analysis and synthesis. In D. Deutsch (ed.), *The Psychology of Music*. Academic Press.

Roads, C. (1996). *The Computer Music Tutorial*. The MIT Press.

Vaidyanathan, P.P. (1993). *Multirate Systems and Filter Banks*. Prentice Hall.

Välimäki, V. and Laakso, T.I. (2000). Principles of fractional delay filters. In *IEEE ICASSP'00*, 3870–3873.

Zölzer, U. (2008). *Digital Audio Signal Processing*. Wiley, 2nd edition.