80558 MULTIRATE SIGNAL PROCESSING

Tapio Saramäki

Part III: Polynomial-Based Interpolation for DSP Applications

- This pile of lecture notes is mainly based on the research work done by Jussi Vesma and the lecturer during the last three years.
- Many thanks to Jussi Vesma for his help in preparing this pile of lecture notes.
- If there is some interest to get the pile of lecture notes of the overall course, please contact Tapio Saramäki using his home e-mail adress: saram@vip.fi.
- The overall course consists of the following parts:
- Part I: Basics of Multirate DSP
- Part II: Design and Implementation of Efficient Decimators and Interpolators
- Part III: Polynomial-Based Interpolation for DSP Applications
- Part IV: Design of FIR Filters Using Multirate DSP and Complementary Filtering
- Part V: Multirate Filter Banks Including Discrete-Time Wavelet Banks

POLYNOMIAL-BASED INTERPOLATION FILTERS FOR DSP APPLICATIONS

DESIGN, IMPLEMENTATION, AND APPLICATIONS

Tapio Saramäki and Jussi Vesma
Tampere University of Technology, Finland
e-mail: saram@vip.fi, or ts@cs.tut.fi
http://www.cs.tut.fi/~ts/

Contents:

Comment: The following article coming up soon: J. Vesma and T. Saramäki, "Polynomial-based Interpolation Filters–Part I: Filter Synthesis; Part II: Filter Properties and Applications.

Interpolation Filters

- In many DSP applications there is a need to know the values of the signal also between the exiting discrete-time samples $x(n)$ as shown in Figure 1.
- Special **interpolation filters** can be used to compute new sample values $y(l)=y_a(t_l)$ at **arbitrary** points $t_l=(n_l+\mu_l)T_{in}$ between the **existing samples** $x(n_l)$ and $x(n_l+1)$. Here, T_{in} is the sampling period.
- Here, $y_a(t)$ approximates either the original continuous-time signal $x_a(t)$ or the signal obtained with the aid of the existing discrete-time samples $x(n)$ using the sinc interpolation.
- The output sample time is determined by $\boldsymbol{n_l T_{in}}$, the location of the **preceding existing sample**, and the **fractional interval** $\boldsymbol{\mu_l}\in[0,1)$, the difference between t_l and $n_l T_{in}$ as a fraction of T_{in}.

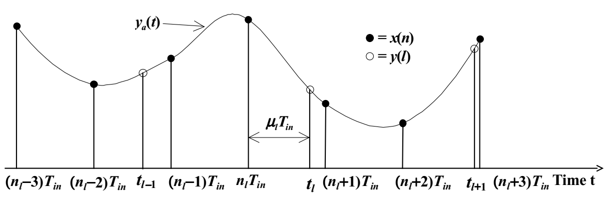


Fig. 1. Interpolation in the time domain.

Applications for Interpolation Filters

- Timing adjustment in all-digital receivers (symbol synchronization)
- Time delay estimation
- Conversion between arbitrary sampling frequencies
- Echo cancellation
- Phased array antenna systems
- Speech coding and synthesis
- Derivative approximation of discrete-time signals
- Computer simulation of continuous-time systems
- ML symbol timing recovery in digital receivers

Discrete-Time Interpolation

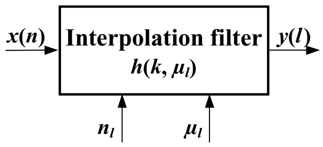


Fig. 2. Simplified block diagram for the interpolation filter.

- We concentrate on performing the interpolation problem stated on Page 3 using the system of Figure 2.
- The input parameters n_l and μ_l are used to determine the time instant t_l for the l^{th} output sample $y(l)=y_a(t_l)$ as $t_l=(n_l+\mu_l)T_{in}$.
- Given t_l, these parameters are determined by

$$n_l = \lfloor t_l / T_{in} \rfloor \text{ and } \mu_l = t_l / T_{in} - \lfloor t_l / T_{in} \rfloor \qquad (1)$$

- After knowing n_l and μ_l, the interpolation filter calculates $y(l)$ according to the following convolution:

$$y(l) = \sum_{k=-N/2}^{N/2-1} x(n_l - k) h(k, \mu_l) \qquad (2)$$

where N (even) is the filter length and $h(k, \mu_l)$ is the discrete-time impulse response of the interpolation filter.

Interpretation of the Convolution Sum

- First, the location of the existing sample preceding or occurring at the new sampling instant t_l is determined and denoted by n_l.
- Based on the location of this sample, the existing samples located at $n=n_l-N/2+1, n_l-N/2+2,\cdots, n_l+N/2$ are used.
- This means that there are $N/2$ existing discrete-time samples before and after the desired new time instant t_l.
- This gives the best results and explains why N has been selected to be an even integer.
- Second, the distance between t_l and $n_l T_{in}$ is measured as a fraction of T_{in}, giving the fractional interval μ_l.
- In the above convolution, the value of μ_l determines the impulse-response coefficient values $h(k,\mu_l)$.
- The purpose is to determine them in such a way that $y(l)$ is found according to some criterion to be discussed next.

Statement of the Interpolation Problem

Given N, find the impulse-response coefficients $h(k, \mu_l)$ for $k=-N/2+1, -N/2+2,\cdots, N/2$ to meet the following two conditions:

1. Optimize them such that $y(l)=y_a((n_l+\mu_l)T_{in})$ for all values of $\mu_l \in [0,1)$, where $y_a(t)$ approximates according to some time-domain or frequency-domain criterion the signal $x_a(t) = \sum_{n=-\infty}^{\infty} x(n) \sin[\pi(t-nT_{in})/T_{in}]/[\pi(t-nT_{in})/T_{in}]$.
2. The overall system can be implemented **digitally** using an efficient structure.

- In Condition 1, the frequency-domain criteria are usually preferred for DSP applications.

Comments

- There exist the following two trivial solutions:

1. If the ratio is an integer or a ratio of small integers, a conventional discrete-time sampling rate alteration is efficient.
2. If the ratio is not a ratio of small integers or an integer, then one can first generate a continuous-time signal with the aid of a D/A converter and a reconstruction filter and, then, resample this signal using an A/D converter and an anti-aliasing filter.

- The drawback in the second approach is the fact that if the conversion should be performed with a high accuracy, then very costly components are needed.
- Therefore, it is worth studying whether there exist an efficient technique to accomplish the same directly **digitally**.
- This is especially true since **all the information** is carried by the existing samples.

Various Approaches to Solve the Stated Problem

- The problem is now to design an interpolation filter that calculates the interpolated samples $y(l)$ for a given value of μ_l based on the above discussion.
- Furthermore, there is a need to analyze the performance of the filter, that is, to measure the error between $y_a(t)$ and $x_a(t)$ either in the time domain or frequency domain.
- This is not an easy task since the interpolation filter is a time-varying system, that is, the coefficients $h(k, \mu_l)$ depend on the value of μ_l.

Various Approaches to Solve the Stated Problem

- There exist the following three approaches to solve our problem:

1. Fractional delay (FD) filter approach.
2. Use some classical interpolation method to calculate $y(l)$, e.g., Lagrange or B-spline interpolation (**time-domain approach**).
3. Utilize the analog model for the interpolation filter (**frequency-domain approach**).

Fractional Delay (FD) Filter Approach

- There exist several synthesis methods for designing FIR filters with the transfer function of the form (N is even)

$$H(z,\mu) = \sum_{k=0}^{N} h(k,\mu) z^{-k}$$

for various values of μ such that in the given passband region $[0, \omega_p]$

1. The phase delay approximates $D_{\mathrm{int}}-\mu$, where $D_{\mathrm{int}}=N/2$ is the integer delay and $\mu \in [0,1)$ is the fractional delay.
2. The amplitude response approximates unity.

- This approach provides a good solution **if μ is fixed.**
- **Otherwise**, several FD FIR filters have to be designed for numerous values of μ and coefficients have stored in a lookup table.
- This means that if several values of μ are needed to provide a good interpolation, **a very large memory** is needed.
- Furthermore, the analysis of the overall interpolation process is not so easy to perform.
- The next page shows example FD FIR filters.

Example FD FIR Filters

- N=8, passband region =[0, 0.75π], ten filters having $4-\mu$ with μ=0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6 ,0.7, 0.8 ,0,9, and (1; not needed)

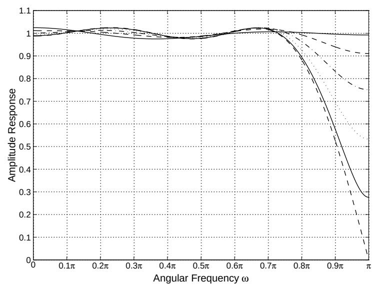


Fig. 3. Amplitude responses for the example FD filters.

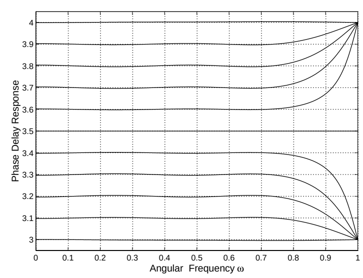


Fig. 4. Phase delay responses.

Lagrange Interpolation

- For the Lagrange interpolation, the approximating continuous-time signal $y_a(t)$ is formed as follows:

$$y_a(kT_{in}) = x(k) \equiv x_a(kT_{in}) \text{ for } -N/2+1 \le k \le N/2. \quad (3)$$

- The resulting approximating $y_a(t)$ is a polynomial of t and its degree is $M = N - 1$.
- The next page illustrates how to use the linear ($N = 2$) and the cubic Lagrange interpolation ($N = 4$) for generating a new sample value at an arbitrary point between the existing samples $x(n_l)$ and $x(n_l+1)$.
- From the curiosity, also the zero-order hold is included.
- It should be pointed out that the approximation error $|x_a(t) - y_a(t)|$ becomes the smallest in the interval between the existing samples $x(n_l)$ and $x(n_l+1)$ if equally many sample values are used before and after the new sample instant.
- This is why it is preferred to select N to be an even integer.
- After introducing a hybrid analog/digital model to be mimicked digitally, we consider in more details how to realize the Lagrange interpolation using efficient digital implementations.

Example: Lagrange Interpolation: How to generate $y(l)$?

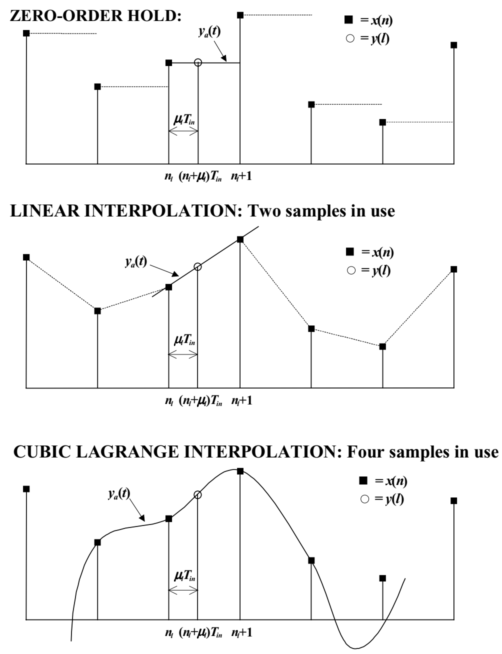


Fig. 5. Lagrange interpolation.

Hybrid Analog/Digital Model to be Mimicked

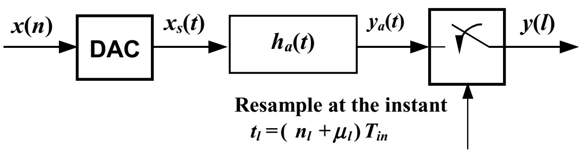


Fig. 6. Analog model for the interpolation filter.

- Interpolation is a reconstruction problem where the approximating signal $y_a(t)$ is reconstructed based on the existing discrete-time samples $x(n)$.
- Therefore, a useful way of modeling discrete-time interpolation filters is to use **the analog system** shown in Fig. 6.
- In this system, $x(n)$ is first converted with the aid of **the ideal D/A converter** to the sequence of weighted and shifted analog impulses $x_s(t) = \sum_{n=-\infty}^{\infty} x(n)\delta_a(t - nT_{in})$.
- $x_s(t)$ is then filtered using **an analog reconstruction filter** with impulse response $h_a(t)$ resulting in the following convolution:

$$y_a(t) = \int_{-\infty}^{\infty} x_s(\tau) h_a(t-\tau) d\tau = \sum_{k=-\infty}^{\infty} x(k) h_a(t - kT_{in}). \quad (4)$$

Hybrid Analog/Digital Model to be Mimicked

- Assuming that $h_a(t)$ is **zero outside the interval $-NT_{in}/2 \le t < NT_{in}/2$**, $y(l)$ obtained by sampling $y_a(t)$ at t_l is given by

$$y(l) = y_a(t_l) = \sum_{k=-N/2}^{N/2-1} x(n_l - k) h_a((k + \mu_l)T_{in}). \quad (5)$$

- By comparing Equations (2) and (5), it can be seen that the impulse responses of the analog and discrete-time filters are related as follows:

$$h(k, \mu_l) = h_a((k + \mu_l)T_{in}) \quad (6)$$

for $k = -N/2, -N/2+1, \cdots, N/2-1$.

- In the causal case, $h_a(t)$ is delayed by $NT_{in}/2$, i.e., the impulse response is given by $h_a(t - NT_{in}/2)$.
- In this case, $y(l)$ obtained $NT_{in}/2$ time units later is given by

$$y(l) = \sum_{k=0}^{N-1} x(n_l + N/2 - k)\, h_a\big((k + \mu_l - N/2)T_{in}\big). \quad (7)$$

- In the sequel, the non-causal $h_a(t)$ [Equation (5)] and the causal $h_a(t - NT_{in}/2)$ [Equation (7)] are used for the design and interpolation purposes, respectively.

Why to Use the Analog Model?

- Interpolation is generally considered as a time-domain problem of fitting polynomial through the existing samples, which is not very practical approach for DSP applications.
- These include the Lagrange and B-spline interpolations
- This is because the time-domain characteristics of the input sequence $x(n)$ are not usually known. What is usually known is the frequency-domain performance of the signal.
- It should be pointed out that recently Atanas Gotchev, Karen Egiazarian, and Tapio Saramäki have improved the performance of B-splines in interpolation problems, especially in the case of images, by using modified B-splines consisting of a weighted sum of odd-order B-splines. Contact: saram@vip.fi (home e-mail address of Saramäki).
- The main idea is to determine the weights in such a manner that the resulting filter effectively preserves the baseband of interest and attenuates the corresponding images.

Why to Use the Analog Model?

- The use of the analog model converts the interpolation problem from the time-domain to the frequency domain in a manner to be considered next.

Synthesis problem for interpolation: Determine $h_a(t)$ such that

1. It provides the desired filtering performance.
2. The overall system of Fig. 6 can be implemented **digitally** using an efficient structure.

Frequency-Domain Criteria

- For the overall system of Fig. 6, the Fourier transform of $y_a(t)$ is related to that of the sequence $x(n)$ or equivalently to that of the signal $x_s(t) = \sum_{n=-\infty}^{\infty} x(n)\delta_a(t - nT_{in})$ through

$$
\begin{aligned}
Y_a(j2\pi f) &= H_a(j2\pi f)X(e^{j2\pi f/F_{in}}) = \\
&= H_a(j2\pi f)F_{in}\sum_{k=-\infty}^{\infty} X_a(j2\pi(f - kF_{in}))
\end{aligned}
\qquad (8)
$$

where $F_{in} = 1/T_{in}$ is the sampling rate of the input signal and $H_a(j2\pi f)$ is the Fourier transform of the reconstruction filter with impulse response $h_a(t)$.

- The last form of Equation (8) is for the case where $x(n) = x_a(nT_{in})$ are samples of a continuous-time signal $x_a(t)$ with $X_a(j2\pi f)$ being its Fourier transform.

Role of $h_a(t)$ in the Frequency Domain

As shown below, the role of the reconstruction filter with impulse response $h_a(t)$ is to attenuate the extra images of $x_s(t) = \sum_{n=-\infty}^{\infty} x(n)\delta_a(t - nT_{in})$ and to preserve the signal components only in the original baseband $[0, F_{in}/2]$.

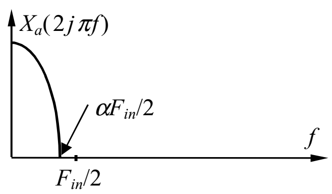


Fig. 7. The spectrum of the original continuous-time signal bandlimited to $|f| \le \alpha F_{in}$. The sequence is formed as $x(n) = x_a(nT_{in})$.

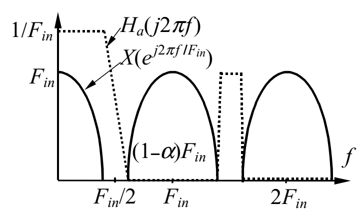


Fig. 8. The spectrum of $x_s(t) = \sum_{n=-\infty}^{\infty} x(n)\delta_a(t - nT_{in})$, denoted by $X(e^{j2\pi f/F_{in}})$ and the frequency response of the reconstruction filter with impulse response $h_a(t)$, denoted by $H_a(j2\pi f)$.

Criteria for the Uniform Sampling: Interpolation and Decimation

- If $y(l)$ is generated at the time instants $t_l = lT_{out}$, then

$$Y(e^{j2\pi f/F_{out}}) = F_{out}\sum_{k=-\infty}^{\infty} Y_a(j2\pi(f-kF_{out})), \qquad (9)$$

where $F_{out} = 1/T_{out}$ is the sampling rate of the output signal $y(l)$ and the baseband of interest is $[0, F_{out}/2]$.

- The case $\beta = F_{out}/F_{in} > 1$ corresponds to **the interpolation.**
- The case $\beta = F_{out}/F_{in} < 1$ corresponds to **the decimation.**
- In both cases, the ideal response for $H_a(j2\pi f)$ avoiding both imaging and aliasing is given by

$$D(f) = \begin{cases} 1/F_{in} & \text{for } 0 \le f \le F_C/2 \\ 0 & \text{for } f > F_C/2, \end{cases} \qquad (10a)$$

where

$$F_C = \min(F_{in}, F_{out}). \qquad (10b)$$

- Note that in the interpolation case, it is enough to attenuate the images of $X(e^{j2\pi f/Fin})$.
- In the decimation case, $X(e^{j2\pi f/Fin})$ should be band-limited into the range $[0, F_{out}/2]$, that is, the region $[F_{out}/2, F_{in}/2]$ should be attenuated in order to avoid aliasing.

Practical Criteria

- Like for conventional digital interpolators and decimators, the criteria can be stated as

$$1-\delta_p \le F_{in}|H_a(j2\pi f)| \le 1+\delta_p \quad \text{for} \quad f \in [0, f_p] \qquad (11a)$$

$$F_{in}|H_a(j2\pi f)| \le \delta_s \quad \text{for} \quad f \in \Omega_s, \qquad (11b)$$

where $f_p < F_C/2$ and

$$\Omega_s = \begin{cases} [F_C/2, \infty) & \text{for Type A} \\ [F_C - f_p, \infty) & \text{for Type B} \\ \bigcup_{k=1}^{\infty}[kF_C - f_p, kF_C + f_p] & \text{for Type C.} \end{cases} \qquad (11c)$$

- For Type A, no aliasing or imaging are allowed.
- For Type C decimation case, aliasing is allowed into the transition band $[f_p, F_{out}/2]$. For Type B, aliasing into this band is allowed only from band $[F_{out}/2, F_{out}-f_p]$.
- In the interpolation case, Types B and C are useful if most of the energy of the incoming signal is in the range $[0, f_p]$.

Polynomial-Based Interpolation Filters

- In order to arrive at an efficient digital implementation of the system of Fig. 6, it is required that $h_a(t)$ is expressible as a polynomial of t in each interval $[kT_{in}, (k+1)T_{in}]$ for $k=-N/2,\cdots, N/2-1$.
- This is achieved by expressing $h_a(t)$ as follows:

$$h_a((k+\mu_l)T_{in}) = \sum_{m=0}^{M}\hat{c}_m(k)\mu_l^{\,m} \qquad (12)$$

for $k=-N/2,\cdots, N/2-1$. Here, the $\hat{c}_m(k)$'s are the coefficients and M is the degree of the polynomials.

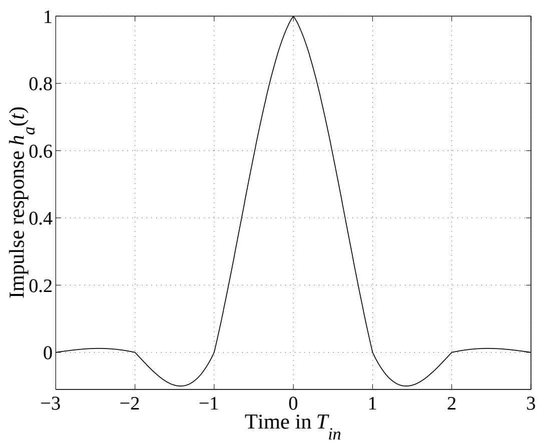
- When μ_l varies from 0 to 1, then $h_a(t)$ takes in each interval $[kT_{in}, (k+1)T_{in}]$ for $k=-N/2,\cdots, N/2-1$ the following form:

$$h_a(t) = \sum_{m=0}^{M}\hat{c}_m(k)\left(\frac{t-kT_{in}}{T_{in}}\right)^m, \qquad (13)$$

as is desired.

- An example impulse response $h_a(t)$ is shown in Fig. 9 for $N=6$ and $M=5$.

Fig. 9. An example of the impulse response of a polynomial-based interpolation filter for $N=6$ and $M=5$ (5^{th} order Lagrange). In each interval $[kT_{in}, (k+1)T_{in}]$ for $k=-N/2,\cdots, N/2-1$ $h_a(t)$ is a polynomial of degree 5.



Generation an Efficient Digital Implementation: Original Farrow Structure

- By using the analog model, the digital implementation structure for polynomial-based interpolation filters can be derived by substituting Equation (12) into Equation (7) giving

$$y(l) = \sum_{k=0}^{N-1} x(n_l - k + N/2) \sum_{m=0}^{M} \hat{c}_m (k - N/2) \mu_l{}^m . \qquad (14)$$

- Alternatively, this equation can be expressed as

$$y(l) = \sum_{m=0}^{M} v_m (n_l) \mu_l{}^m , \qquad (15a)$$

where

$$\begin{aligned} v_m(n_l) &= \sum_{k=0}^{N-1} x(n_l - k + N/2) \hat{c}_m (k - N/2) \\ &= \sum_{k=0}^{N-1} \hat{c}_m (k - N/2) x(n_l + N/2 - k). \end{aligned} \qquad (15b)$$

Interpretation of the Above Equation

- According to Equation (15), the lth output sample $y(l)$ at the time instant $t = (n_l + \mu_l)T_{in}$ can be generated based on the N existing samples $x(n)$ being located at $n = n_l - N/2+1$, $n_l - N/2+2, \cdots, n_l + N/2$ in the following three steps:
- First, these samples are filtered using M+1 FIR filters having the transfer functions of the form

$$\hat{C}_m(z) = \sum_{k=0}^{N-1} \hat{c}_m (k - N/2) z^{-k} \text{ for } m = 0,1,\cdots,M. \qquad (16)$$

- Second, the outputs of these filters, denoted by $v_m(n_l)$, are multiplied with constants $\mu_l{}^m$.
- Third, the multiplication results are added, leading to the so-called original Farrow structure shown in Figure 10.
- In this figure, the input to both structures is denoted by $x(n_l + N/2)$ to emphasize the fact that this is the last existing sample value when evaluating $y(l)$.
- If desired, the transfer functions $\hat{C}_m(z)$ can share the delay elements.

Original Farrow structure for a polynomial-based interpolation filter.

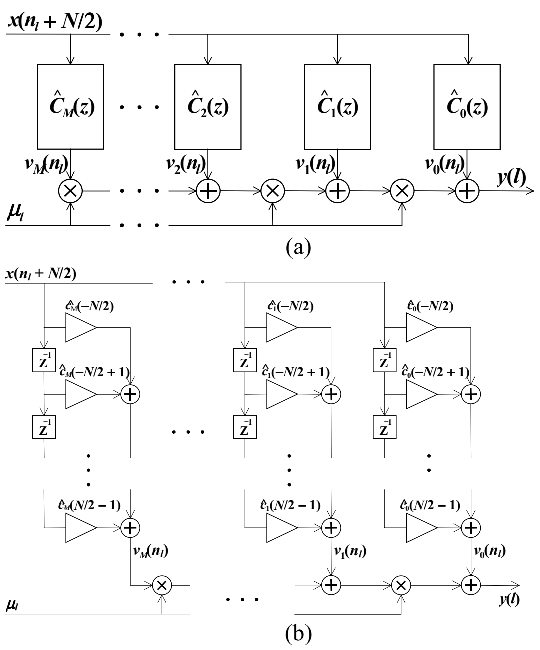


Fig. 10. Farrow structure for a polynomial-based interpolation filter. (a) Basic structure. (b) Details.

Characteristics of the Farrow Structure

- The number of FIR filters is M+1.
- The length of these filters is N.
- Filter coefficients are directly determined by the polynomial coefficients of the impulse response $h_a(t)$.
- The main advantage of the Farrow structure is that all the filter coefficients are fixed.
- The only changeable parameters are the fractional interval μ_l as well as n_l that depend on the lth output sampling instant.
- The control of μ_l is easier during the computation than in the implementation based on the FD filters.
- The resolution of μ_l is limited only by the precision of the arithmetic not by the size of the memory or lookup table.
- These characteristics of the Farrow structure make it a very attractive structure to be implemented using a VLSI circuit or a signal processor.

What Do We Have Up to Now?

1. The fundamental idea is that the $y_a(t)$, the output of our analog model of Figure 6, approximates according to some criterion the following 'ideal' continuous- time function $x_a(t) = \sum_{n=-\infty}^{\infty} x(n)\sin[\pi(t-nT_{in})/T_{in}]/[\pi(t-nT_{in})/T_{in}]$.
2. By using the analog model, this approximation problem can be converted to the problem of designing the reconstruction filter $h_a(t)$.
3. If $h_a(t)$ is a piecewise polynomial, then the Farrow structure can be used for implementation.

- The remaining task is to optimize the coefficients $\hat{c}_m(k)$ for the polynomial-based impulse response $h_a(t)$. See Equations (12) and (13).

Design Strategies

- We consider only polynomial-based interpolation filters because they can be efficiently implemented using the Farrow structure.
- The following two design methods for polynomial-based interpolation filters are considered:

1. Conventional time-domain synthesis: Lagrange interpolation.
2. Frequency-domain synthesis based on the analog model of Figure 6: Minimax and least-mean-square optimization of the interpolation filters.

- These design methods will be compared with the aid of the resulting impulse and frequency responses of the corresponding reconstruction filter $h_a(t)$.

Lagrange Interpolation

- Lagrange interpolation was originally used in mathematics, not in signal processing (discovered by Joseph-Louis Lagrange 1736-1813).
  $\Rightarrow$ Does not offer a good filtering characteristics.
- Filter design is done in the time domain and the filter coefficients can be given in a closed form.
  $\Rightarrow$ No need for optimization.
- The synthesis can be accomplished as follows:

1. Choose M, the degree of the interpolation. The length of the filter is then $N=M+1$
2. Based on the time-domain conditions for $y_a(t)$ as given by Equation (3), the polynomial coefficients for $h_a(t)$ are determined from the following equation:

$$\sum_{m=0}^{M} \hat{c}_m(k)x^m = \prod_{\substack{j=-N/2+1 \\ j\neq k}}^{N/2} \frac{j-x}{-k+j} \qquad (17)$$

for $k=-N/2, -N/2+1, \cdots, N/2-1$.

- As mentioned earlier, it is desired to use an even value for N. This means that M should be an odd integer.
- The following page show both the time- and frequency-domain responses corresponding to the linear ($M=1$, $N=2$) and cubic ($M=3$, $N=4$) Langrange interpolation.

Filter responses corresponding to the linear and cubic Lagrange interpolation

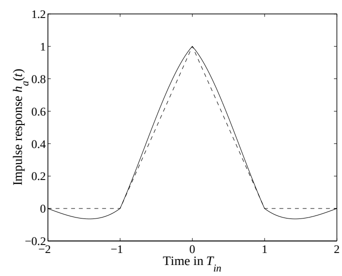


Fig. 11. Impulse response $h_a(t)$ for the cubic (solid line) and linear (dashed line) Lagrange interpolation filters.

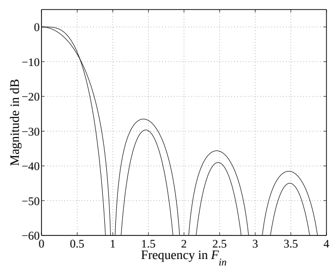


Fig. 12. Magnitude responses for the cubic (solid line) and linear (dashed line) Lagrange interpolation filters.

Disadvantages of the Lagrange interpolation filters

- The frequency response cannot be changed: stopband attenuation, passband ripple, and edge frequencies are fixed.
- If the degree of the interpolation is M, then the length of the filter is always $N=M+1$.
- The stopband attenuation cannot be improved significantly by increasing the degree of the interpolation (e.g., 5 or 7), but the number of filter coefficients increases fast.
- The characteristics of the resulting reconstruction filter are very poor, at least if the input signal contains frequency components close to half the sampling frequency.
- The attenuation of the unwanted images is very good around the multiples of F_{in}.
- Therefore, these filters can be used only if the sampling rate is increased before using them.
- For instance, the third order Lagrange interpolation filter attenuates the image frequencies at least by 60dB if the bandwidth of the input signal is $0.1F_{in}$ as shown in Fig. 13.
- This implies the use of a discrete-time conventional interpolation filter that increases the sampling rate by a factor of five.
- This makes the overall synthesis more complicated compared to the case where a single Farrow structure is used.

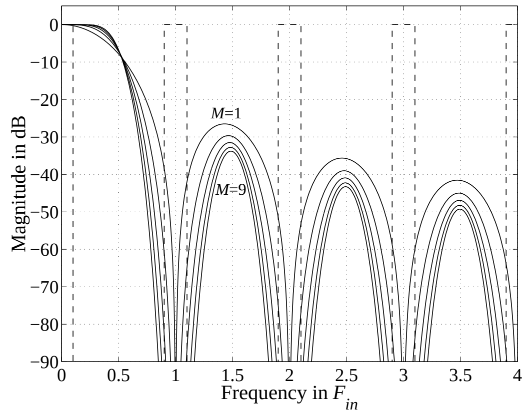


Fig. 13. The magnitude responses $|H_a(j2\pi f)|$ of the Lagrange interpolation filters with degree $M=1,3,5,7$, and 9 (solid lines) and the spectrum $X_s(j2\pi f)$ of the input signal with the bandwidth of $0.1F_{in}$ (dashed-line).

General Frequency-Domain Synthesis Technique for Interpolation Filters

- It is desired to design in the analog model of Figure 6 the reconstruction filter with impulse response $h_a(t)$ only subject to the restriction that the overall system can be implemented using the **original or modified Farrow structure**.
- In this case, M and N can be selected arbitrarily.
- The main goal to is to optimize the coefficients of the Farrow structure and the corresponding impulse response $h_a(t)$ in such a manner that the overall system provides the desired frequency-domain behavior that depends on the application.
- The purpose is to determine the coefficients in such a manner that the amplitude response of the reconstruction filter approximates **an arbitrary amplitude response** either in the least-mean square sense or in the minimax sense.
- It is also desired to allow the use of **some time-domain conditions** for the overall impulse response $h_a(t)$.
- Before developing the general synthesis scheme, **the modified Farrow structure** is introduced.

Construction of the impulse response $h_a(t)$ for the modified Farrow structure.

- In order to arrive at an efficient digital implementation, $h_a(t)$ is assumed to satisfy the following conditions:
  1) $h_a(t)$ is nonzero for $-NT_{in}/2 \le t < NT_{in}/2$.
  2) The length of the filter N is an even integer.
  3) $h_a(t)$ is a piecewise-polynomial of degree M in each interval $nT_{in} \le t < (n+1)T_{in}$ for $n=-N/2, -N/2+1, \cdots, N/2-1$.
  4) $h_a(t)$ is symmetrical, that is $h_a(-t)=h_a(t)$ except for the time instants $t=nT_{in}$ for $n=-N/2, -N/2+1, \cdots, N/2$.
- Conditions 1), 2), and 3) guarantee the corresponding causal system with impulse response $h_a(t-NT_{in}/2)$ can be implemented using an efficient digital implementation.
- The role of Conditions 4) is twofold.
  1) For the causal system, the phase response is linear.
  2) In the modified Farrow to be described later, the fixed FIR filters have either a symmetrical or anti-symmetrical impulse responses. This enables us to utilize the coefficient symmetry, reducing the number of multipliers in the implementation.

Time-domain conditions

- There exist the following time-domain constraints of interest:
  3) *Case A*: There are no time-domain conditions.
  4) *Case B*: $h_a(t)$ is continuous at $t=kT_{in}$ for $k=\pm 1, \pm 2, \cdots, \pm N/2-1$.
  5) *Case C*: $h_a(0)=1$ and $h_a(kT_{in})=0$ for $k=\pm 1, \pm 2, \cdots, \pm N/2$.
  6) *Case D*: The first derivative of $h_a(t)$ is continuous at $t=kT_{in}$ for $k=0$ and for $k=\pm 1, \pm 2, \cdots, \pm(N/2-1)$.
- *Case C* guarantees that if the new sampling instant occurs at the instant of the existing sample, then the sample value is preserved.
- *Case D* is of importance when determining the derivative of a continuous-time signal with the aid of its discrete-time samples and a generalized modified Farrow structure.

How to construct the impulse response $h_a(t)$

- Instead of using μ_l we use $2\mu_l-1$ as a basis polynomial to construct $h_a(t)$ as follows:

$$h_a((n+\mu_l)T_{in}) = \sum_{m=0}^{M} c_m(n)(2\mu_l-1)^m \qquad (18)$$

for $n=-N/2, -N/2+1, \cdots, N/2-1$ (see Fig. 14). Here, the $c_m(n)$'s are the unknown polynomial coefficients.

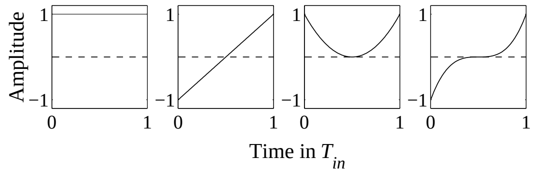


Fig. 14. Polynomials $(2\mu_l-1)^m$ for $m=0, 1, 2,$ and 3.

- When μ_l varies from 0 to 1, then $h_a(t)$ takes in each interval $[kT_{in}, (k+1)T_{in}]$ for $k=-N/2, \cdots, N/2-1$ the following form:

$$h_a(t) = \sum_{m=0}^{M} c_m(k)\left(\frac{2(t-kT_{in})-1}{T_{in}}\right)^m, \qquad (19)$$

- The symmetry property $h_a(-t)=h_a(t)$ is achieved by

$$c_m(n) = (-1)^m c_m(-n-1) \qquad (20)$$

for $m=0, 1, \cdots, M$ and $n=0, 1, \cdots, N/2-1$. This condition halves the number of unknowns.

- $h_a(t)$ can be now constructed as follows

$$h_a(t) = \sum_{n=0}^{N/2-1} \sum_{m=0}^{M} c_m(n) g(n,m,t) \qquad (21)$$

where $c_m(n)$'s are unknown coefficients and $g(n,m,t)$'s are the basis functions given by

$$g(n,m,t) = \begin{cases} \left(\dfrac{2(t-nT_{in})}{T_{in}}-1\right)^m & \text{for } nT_{in} < t \le (n+1)T_{in} \\ (-1)^m\left(\dfrac{2(t+(n+1)T_{in})}{T_{in}}-1\right)^m & \text{for } -(n+1)T_{in} \le t < -nT_{in} \\ 0 & \text{otherwise} \end{cases} \qquad (22)$$

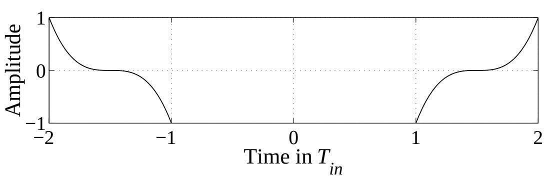


Fig. 15. The basis function $g(n,m,t)$ for $n=1$ and $m=3$.

Figure 15 shows an example basis function, whereas Fig. 16 shows how the overall impulse response can be constructed using weighted basis functions.

Example on how to construct $h_a(t)$ for $N=8$ and $M=3$.

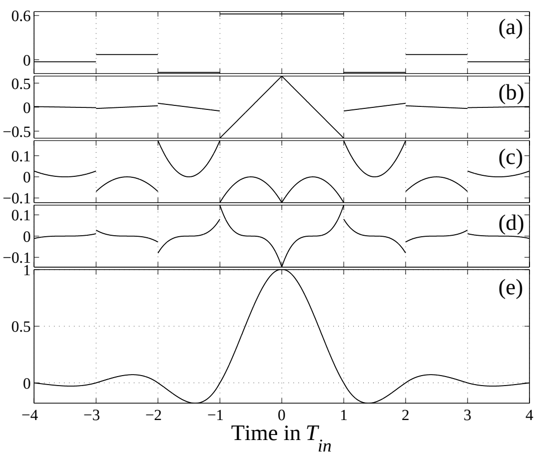


Fig. 16. Construction of the overall impulse response $h_a(t)$ for $N=8$ and $M=3$. The weighted basis functions $\sum_{n=0}^{N/2-1} c_m(n)g(n,m,t)$ for $m=0$ (a), $m=1$ (b), $m=2$ (c), and $m=3$ (d). (e) The resulting impulse response $h_a(t)$ obtained as a sum of these responses.

Modified Farrow structure

- In the above approach, the polynomial coefficients $c_m(n)$ are symmetrical according to Equation (20).
- $2\mu_l - 1$ is used, instead of μ_l.
- Therefore, interpolation filters based on this approach can be implemented using the **modified Farrow structure** shown in Fig. 17.
- This modified structure has two differences compared to the original structure.

1. The output samples $v_m(n_l)$ of the FIR filters are multiplied with $2\mu_l - 1$ instead of μ_l.
2. FIR filters with the transfer functions

$$C_m(z) = \sum_{k=0}^{N-1} c_m(k - N/2) z^{-k} \quad \text{for} \quad m = 0, 1, \cdots, M \qquad (23)$$

possess the symmetry properties given by Equation (20).

- When m is zero or even, $c_m(N/2 - 1 + k) = c_m(-N/2 - k)$ for $k=0, 1, \cdots, N/2-1$.
- For m odd, $c_m(N/2 - 1 + k) = -c_m(-N/2 - k)$ for $k=0, 1, \cdots, N/2-1$.
- When exploiting these symmetries, the number of coefficients to be implemented can be reduced from $(M+1)N$ to $(M+1)N/2$.

Modified Farrow Structure

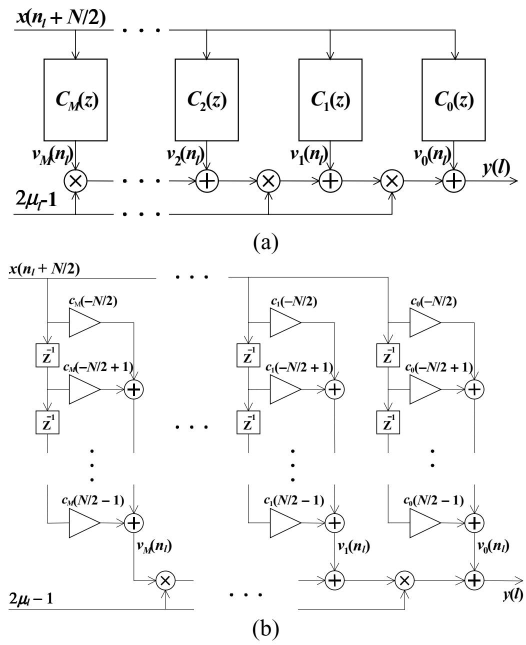


Fig. 17. Modified Farrow structure. (a) Basic structure. (b) Details.

The frequency response for the reconstruction filter with impulse response $h_a(t)$

- After some manipulations, the frequency response can be expressed as

$$H_a(j2\pi f) = \sum_{n=0}^{N/2-1} \sum_{m=0}^{M} c_m(n) G(n,m,f), \qquad (24)$$

where $G(n,m,f)$ is the frequency responses of the basis function $g(n,m,t)$.

- Since $g(n,m,t)$ is symmetrical around $t=0$, $G(n,m,f)$ is real and is given by

$$G(n,m,f) = \begin{cases} 2\cos(2\pi f T_{in}(n+\tfrac{1}{2}))\left[(-1)^{m/2} m!\Phi(m,f) + \dfrac{\sin(\pi f T_{in})}{\pi f T_{in}}\right] & \text{for } m \text{ even} \\ 2(-1)^{(m+1)/2} m! \sin(2\pi f T_{in}(n+\tfrac{1}{2}))\Phi(m,f) & \text{for } m \text{ odd,} \end{cases}$$

(25a)

where

$$\Phi(m,f) = \sum_{k=0}^{\lfloor (m-1)/2 \rfloor} (\pi f T_{in})^{2k-m} \frac{(-1)^k}{(2k)!} \left(\frac{\sin(\pi f T_{in})}{\pi f T_{in}} - \frac{\cos(\pi f T_{in})}{(2k+1)} \right).$$

(25b)

Optimization Problems

- The very attractive property of the above $H_a(j2\pi f)$ is that it is **linear with with respect its unknowns $c_m(n)$**.
- These unknowns can be easily found to minimize

$$\delta_\infty = \max_{f \in X} \left| W(f)\left(H_a(j2\pi f) - D(f)\right) \right| \qquad (26)$$

or

$$\delta_2 = \int_X \left[W(f)\left(H_a(j2\pi) - D(f)\right)\right]^2 df \qquad (27)$$

subject to the given time-domain conditions of $h_a(t)$ stated on Page 37.

- The first and second criteria correspond to the optimization in the minimax and the least-mean-square sense, respectively.
- Here $X \subset [0,\infty)$ is a compact subset and $D(f)$ is an arbitrary desired function (continuous) and $W(f)$ is an arbitrary weighting function (positive).
- Here, the approximation region X consists of a set of passband and stopband regions.
- The actual optimization can be accomplished in a manner similar to the design of linear-phase FIR filters.

- Optimization algorithms have been implemented in Matlab. For minimax problem, linear programming can be used to optimize the filter coefficients.
- For both problems, the time-domain conditions are included in the problem in such a manner that they become unconstrained problems.
- This makes the overall optimization algorithms very fast.

Parameters for the optimization

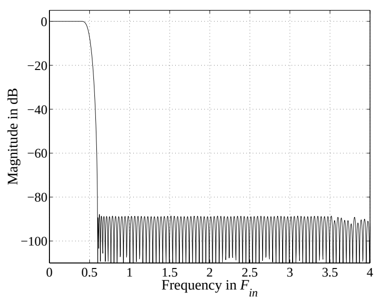
- Design parameters for the optimization programs are the following:

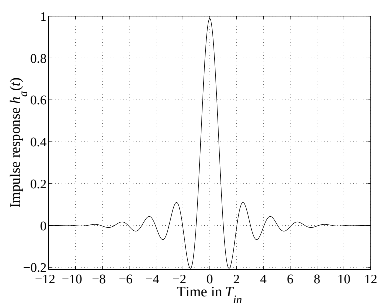
1. Edge frequencies for passband(s) and stopband(s).
2. Desired amplitude and weight for every band.
3. N, the length of the filter.
4. M, the degree of the interpolation.
5. The number of grid points.
6. Time-domain constraints:
   1) *Case A*: There are no time-domain conditions.
   2) *Case B*: h_a(t) is continuous at $t=kT_{in}$ for $k=\pm1, \pm2,\cdots, \pm N/2-1$.
   3) *Case C*: $h_a(0)=1$ and $h_a(kT_{in})=0$ for $k=\pm1, \pm2,\cdots, \pm N/2$.
   4) *Case D*: The first derivative of $h_a(t)$ is continuous at $t=kT_{in}$ for $k=0$ and for $k=\pm1, \pm2,\cdots, \pm(N/2-1)$.

- Before introducing the applications, two Case A design examples are considered.

Optimized Case A minimax design

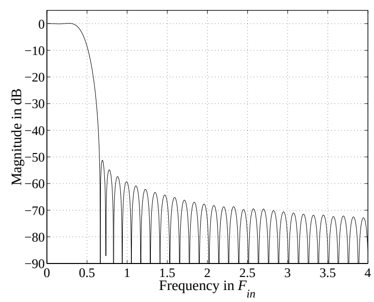
- $M=7, N=24$, $X=[0, 0.4F_{in}]\cup[0.6F_{in}, \infty)$, $D(f)$ is unity and zero on the first and second bands, whereas $W(f)$ is 0.002 and 1, respectively.

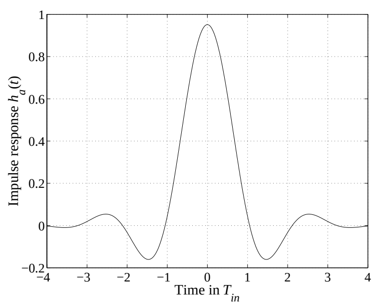




Optimized Case A least-squared design

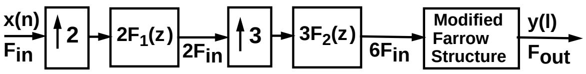
- $M=5$, $N=8$, $X=[0, 0.35F_{in}]\cup[0.75F_{in}, \infty)$, $D(f)$ is unity and zero on the first and second bands, whereas $W(f)$ is 0.02 and 1, respectively.





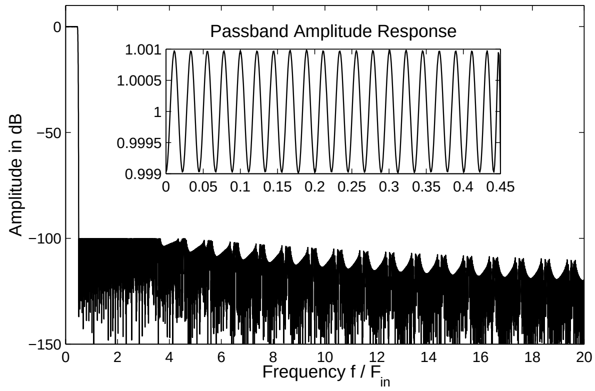
Application A: Up-Sampling Between Arbitrary Sampling Frequencies

- The Farrow structure can be directly used for providing the increse between an arbitrary input sampling rate F_{in} and an arbitrary output sampling rate F_{out}.
- It is desired that $H_a(j2\pi f)$ approximates unity for $0 \le f \le 0.45F_{in}$ with tolerance of 0.001 and zero for $f \ge 0.5F_{in}$ with tolerance of 0.00001 (100-dB attenuation).
- When using the minimax optimization, the given criteria are met by $\boldsymbol{N} = \mathbf{92}$ and $\boldsymbol{M} = \mathbf{6}$, as shown on Page 50. This implementation requires **7** fixed branch filters of length **92**.
- The implementation can be simplified using fixed linear-phase FIR interpolators before the Farrow structure, as proposed by Saramäki and Ritoniemi.
- $\boldsymbol{N} = \mathbf{4}$ and $\boldsymbol{M} = \mathbf{3}$ are required if the sampling rate is increased by a factor of six by using a two-stage fixed interpolator with interpolation factors of two and three and FIR filters of order 183 and 11, respectively. See Page 51.
- The block diagram for this multistage implementation is shown below.

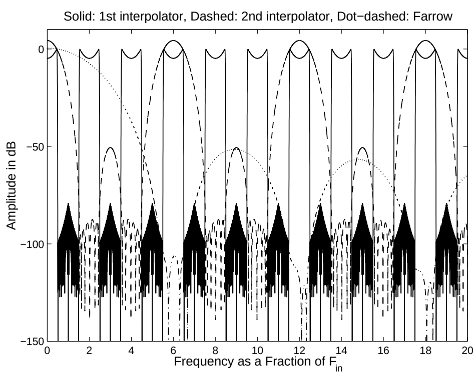


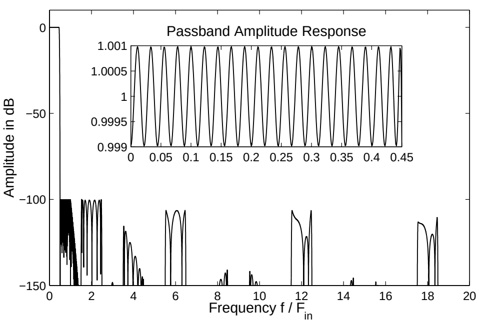
- Note that the same structure can be used for any $F_{out} > F_{in}$.

Direct Design



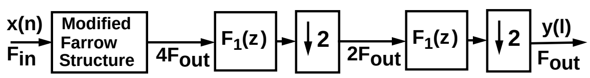
Design with fixed interpolators before the Farrow structure: Simultaneous optimization has been used.





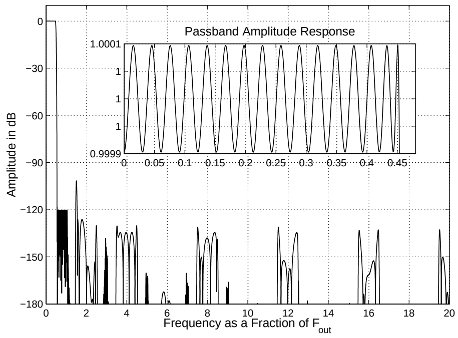
Application B: Down-Sampling Between Arbitrary Sampling Frequencies: First Alternative

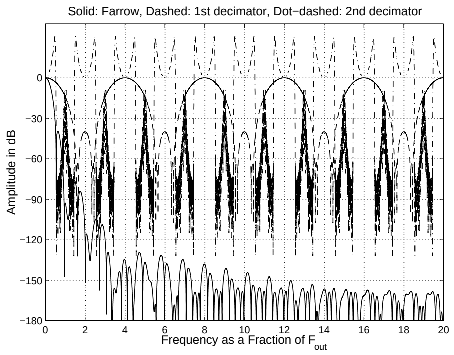
- There exist two alternatives to perform down-sampling.
- The first alternative is to increase the sampling rate to a multiple of the output sampling rate and then to decimate to the desired output sampling rate.
- As an example, consider sampling rate reduction from 48 kHz to 44.1 kHz using the a structure shown below



- The passband edge is 20 kHz and aliasing is allowed into the band between 20 kHz and 44.1/2 kHz.
- The passband ripple is 0.0001 and the minimum stopband attenuation is 120 dB.
- To meet these criteria L = 4 and N = 6 are required by the modified Farrow structure, whereas the orders of the first and second decimator are 4 and 126, respectively.
- The resulting responses are shown on the next page.

Three -stage Decimator using the Modified Farrow structure: Simultaneous optimization has been used.





Application B: Down-Sampling Between Arbitrary Sampling Frequencies: Second Alternative

- The second alternative is to use the transposed modified Farrow structure together with fixed decimators.
- Due to the lack of time, this alternative being more efficient is not included here. This is because the transposed structure is a very new idea.
- An up-to-date-version of this pile including the above will be found in two months in http://www.cs.tut.fi/~ts/ called POLYINT.
- There is an article: Djordje Babic, Jussi Vesma, Tapio Saramäki, and Markku Renfors, "Implementation of the transposed Farrow structure", submitted to ISCAS 2002.
- The main advantage of this structure is that the same structure can be used for any input sampling rate $F_{in} > F_{out}$.

Application C: Continuous-Time Signal Processing

- The Farrow structure can be easily generalized to process digitally the reconstructed signal $y_a(t)$.
- These applications include, among others, determining the derivative or the integral of $y_a(t)$.
- The derivative is widely utilized, for example, in finding the location of the maximum or minimum of the signal.
- The integral can be used to calculate the energy of the signal over the given time interval.
- We concentrate on determining the derivative of $y_a(t)$.

Generalized Farrow Structure for Determining the Derivative of $y_a(t)$

- In the intervals $nT_{in} \le t < (n+1)T_{in}$ for $n=0, 1, 2, \cdot$ $y_a(t)$ can be expressed as

$$y_a(t)\Big|_{t=(n+\mu)T_{in}} = p(n,\mu) = \sum_{m=0}^{M} v_m(n)[2\mu-1]^m \tag{28}$$

where the $v_m(n)$'s are the output samples of the FIR branch filters in the modified Farrow structure.

- The derivative of $y_a(t)$ in the intervals is thus given by

$$\frac{dy_a(t)}{dt}\Big|_{t=(n+\mu)T_{in}} = \frac{dp(n,\mu)}{d\mu} = \sum_{m=0}^{M} v_m(n)2m[2\mu-1]^{m-1}. \tag{29}$$

- The derivative of $y_a(t)$ at $t=(n+\mu)T_{in}$ can be determined by multiplying the $v_m(n)$'s by $2m(2\mu-1)^{m-1}$, instead of $(2\mu-1)^m$ in the modified Farrow structure.
- For estimating the derivative, it is desired that $H_a(j2\pi f)$ approximates $1/(2\pi f)$ in the passband with the weighting equal to $2\pi f$.
- In the stopband, it is desired to approximate zero with a constant weight.

Example on the Derivative Approximation

- It is desired to estimate the continuous-time derivative of a discrete-time ECG signal shown in Fig. 18(b).
- Figures 18(b) and 18(c) show the continuous-time interpolated signal and the derivative signal, respectively.
- For $h_a(t)$ used for determining the derivative signal, N=8, M=5, and the passband and stopband edges are located at $0.35F_s$ and $0.65F_s$, respectively.
- When using the minimax optimization criterion with weigting equal to 0.035 in the passband and equal to unity in the stopband, we end up with $h_a(t)$ with the amplitude and impulse responses shown in Fig. 19.

Continuous-time processing of an ECG signal

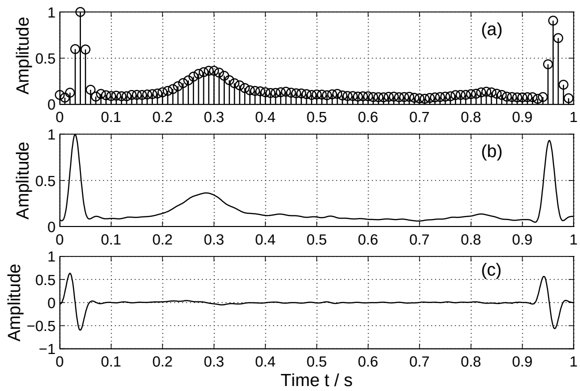
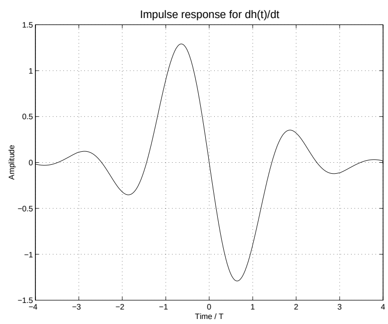
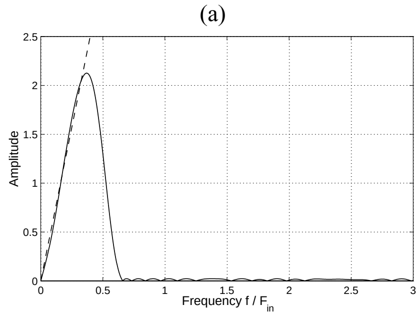


Fig.18. Continuous-time processing of an ECG signal. (a) Discrete-time ECG signal. (b) Interpolated continuous-time signal. (c) Continuous-time derivative.

Characteristics of the differentiator



(a)



(b)

Fig.19. Optimized differentiator. (a) Impulse response. (b) Amplitude response

Application C: Symbol Synchronization in Digital Receivers

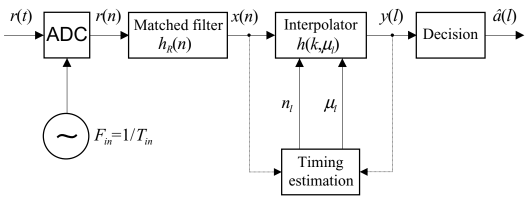


Fig. 20. Digital receiver with non-synchronized sampling.

- The sampling of the received signal is performed by a fixed sampling clock, and thus, sampling is not synchronized to the received symbols.
  - ⇒ timing adjustment must be done by digital methods after sampling.
- Can be done by using interpolation filter.
- Advantages of nonsynchronized sampling:
  - separates the analog and digital parts
  - easy to change the sampling rate
  - sampling frequency does not have to be a multiple of the symbol frequency (only high enough to avoid aliasing)
  - no need for complex PLL circuit.

Practical Case

- When deriving the frequency-domain specifications for the anti-imaging filter $h_a(t)$, it is assumed that
  1) The pulse shape of the transmitted signal has the excess bandwidth of α and the ratio between the sampling rate F_{in} and the symbol rate is R.
  2) In order to avoid aliasing, it is required that $R \geq (1+\alpha)$.
- Based on these assumptions, the input signal of the interpolator $x(n)$ contains the desired component in the frequency range $[0, \beta F_{in}]$, where $\beta=(1+\alpha)/R/2$ and undesired images in the bands $[(k-\beta)F_{in}, (k+\beta)F_{in}]$ for $k=1, 2,\cdots$.
- Therefore, the desired function for $H_a(j2\pi f)$ is specified by

$$D(f) = \begin{cases} 1/F_{in} & \text{for } 0 \leq f \leq \beta F_{in} \\ 0 & \text{for } f \in \Omega_s, \end{cases} \tag{30}$$

where the stopband region, denoted by Ω_s, can be selected as

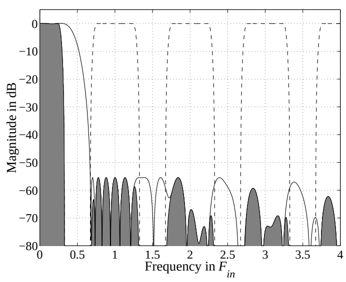
$$\Omega_s = [(1-\beta)F_{in}, \infty) \tag{31a}$$

or

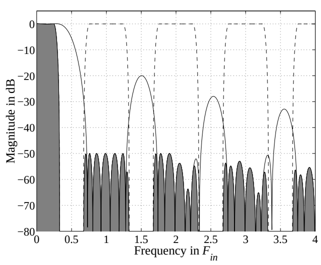
$$\Omega_s = \bigcup_{k=1}^{\infty} [(k-\beta)F_{in}, (k+\beta)F_{in}] \tag{31b}$$

- Two polynomial-based interpolators have been designed in the minimax sense to illustrate the use of the above-mentioned specifications.
- It is assumed that the received signal has a raised cosine pulse shape with the excess bandwidth of α=0.15 and the oversampling ratio is R=1.75.
- The passband edge for both filters is $f_p=\beta F_{in}=23/70F_{in}$ ($\approx 0.33F_{in}$)
- Furthermore, it is required that the passband distortion is less than δ_p=0.01 and the minimum stopband attenuation is A_s=50dB.
- The first filter has a uniform stopband .In order to meet the specifications, ***N*=8** and ***M*=3** are required, as shown in Figure 21(a) on the next page
- The second filter has a non-uniform stopband as given by Eq. (31b) and the spectrum of the raised cosine pulse shape is used as a weighting function. In this case ***N*=6** and ***M*=3** meets the requirements giving A_s=50.0dB and δ_p=0.01, as shown in Figure 21(b) on the next page.

Fig.21. The magnitude response of the interpolation filter (solid line), the spectrum for the raised cosine pulse (dashed line) and for the reconstructed signal $y_a(t)$ (dark area). (a)With uniform stopband. (b) With non-uniform stopbands having the raised cosine weighting.



(a)



(b)

Properties of Minimax Case A designs

- Case A: The minimum even value of N can be estimated by

$$N = 2\left\lceil \frac{-20\log_{10}\left(\sqrt{\delta_p \delta_s}\right) - 8.4}{7.6\left(f_s - f_p\right)/F_{in}} \right\rceil \tag{32}$$

where δ_p and δ_s are the maximum deviations of the amplitude response from unity for $f \in [0, f_p]$ and the maximum deviation from zero for $f \in [f_s, \infty)$.

- Here, $\lceil x \rceil$ stands for the smallest integer that is larger or equal to x.
- It has been observed that in most cases the above estimation formula is rather accurate with only a 2 % error.
- The next problem is to find the minimum value of M to meet the criteria.
- To illustrate this the following specifications are considered:

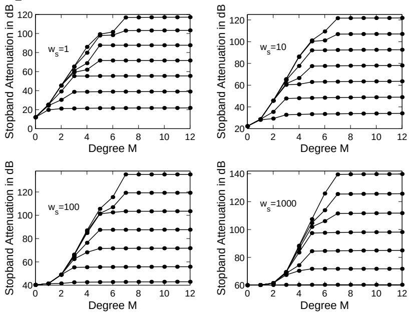
  Specifications 1: The passband and stopband edges are at $f_p=0.25F_{in}$ and at $f_s=0.75F_{in}$.
  Specifications 2: The passband and stopband edges are at $f_p=0.25F_{in}$ and at $f_s=0.5F_{in}$.
  Specifications 3: The passband and stopband edges are at $f_p=0.375F_{in}$ and at $f_s=0.675F_{in}$.
  Specifications 4: The passband and stopband edges are at $f_p=0.375F_{in}$ and at $f_s=0.5F_{in}$.

Properties of Minimax Case A designs

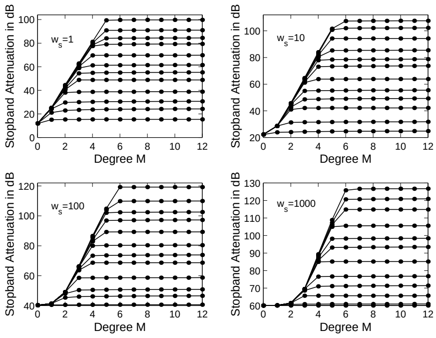
- In each case, several filters have been designed in the minimax sense with the passband weighting equal to unity and stopband weightings of $W_s = 1$, $W_s = 10$, $W_s = 100$, and $W_s = 1000$.
- M, the degree of the polynomial in each subinterval, varies from 0 to 12. N, the number of intervals varies from 2 to the smallest integer for which the stopband ripple for the amplitude response is less than or equal to 0.0001 (100 dB) for $W_s = 1$.
- For Specifications 1, 2, 3, and 4,, the corresponding smallest values of N are 12, 24, 24, and 48, respectively. Recall that N is an even integer.
- The following two pages give the results for Case A.
- For other cases, N is either the same or should be incrased by two.
- For Case C the passband and stopband edges satisfy
- $$f_p = (1-\rho)F_{in}/2, \quad f_s = (1+\rho)F_{in}/2.$$

Specifications 1 and 2:

Specifications 1: N=2, 4,⋯, 14

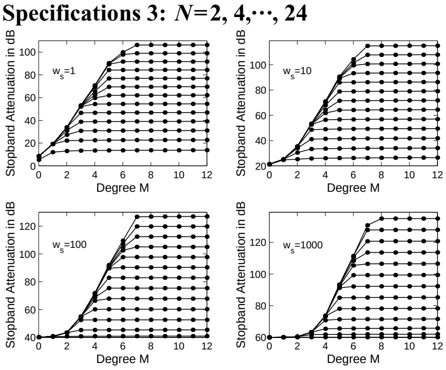


Specifications 2: N=2, 4,⋯, 24

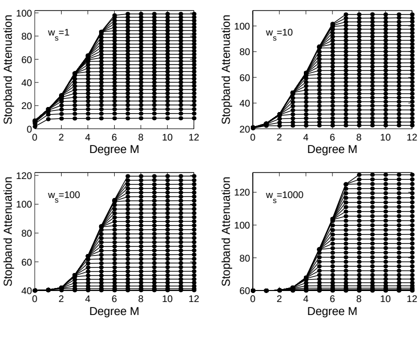


Specifications 3 and 4:

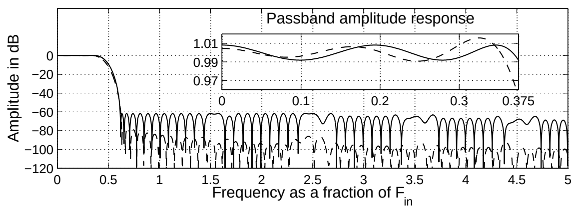
Specifications 3: N=2, 4,⋯, 24

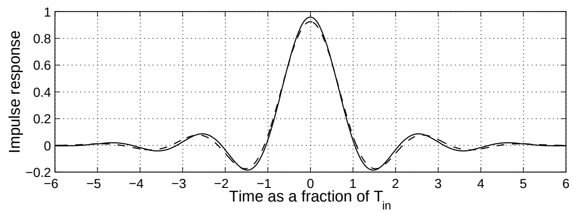


Specifications 2: N=2, 4,⋯, 48

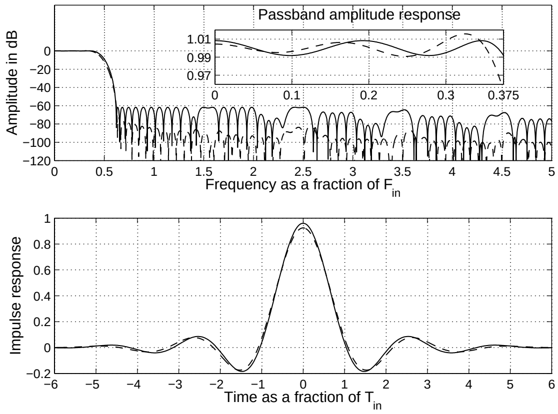


Case A: f_s = 0.625F_{in}, f_p = 0.375F_{in}, δ_p = 0.01, δ_s = 0.001: *N*=12; Minimax (solid line): *M*= 4; Least-squared (dashed line): *M*= 5

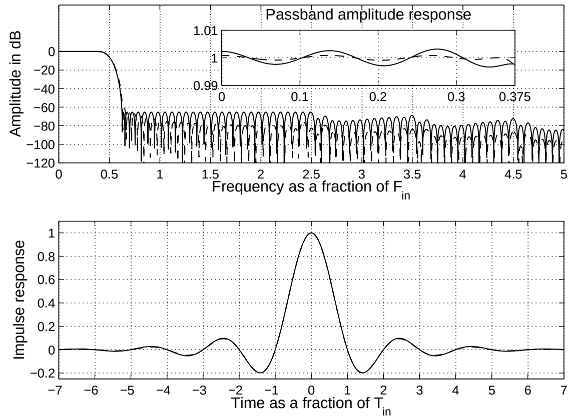




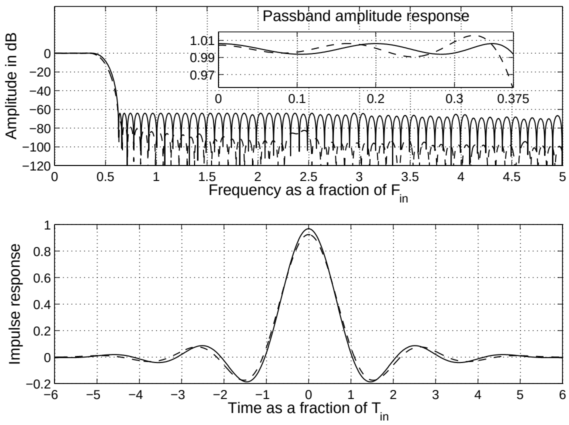
Case B: $f_s = 0.625F_{in}$, $f_p = 0.375F_{in}$, $\delta_p = 0.01$, $\delta_s = 0.001$: N=12; Minimax (solid line): M= 4; Least-squared (dashed line): M= 5



Case C: $f_s = 0.625F_{in}$, $f_p = 0.375F_{in}$, $\delta_p = 0.01$, $\delta_s = 0.001$: N=14; Minimax (solid line): M= 5; Least-squared (dashed line): M= 5



Case D: $f_s = 0.625F_{in}$, $f_p = 0.375F_{in}$, $\delta_p = 0.01$, $\delta_s = 0.001$: N=12; Minimax (solid line): M= 5; Least-squared (dashed line): M= 5



Conclusion

- An efficient approach has been described for interpolating new sample values between the existing discrete-time samples.
- This approach has the following advantages:
    - Design directly in the frequency domain is straightforward.
    - The overall system has an efficient implementation form.
    - The analysis of the system performance is easy.
    - There exist several DSP applications.

References

[1] T. I. Laakso, V. Välimäki, M. Karjalainen, and U. K. Laine, "Splitting the unit delay," *IEEE Signal Processing Magazine*, vol. 13, pp. 30-60, Jan. 1996.

[2] C. W. Farrow, "A continuously variable digital delay element," in *Proc. IEEE Int. Symp. Circuits & Syst.*, Espoo, Finland, June 1988, pp. 2641-2645.

[3] F. M. Gardner, "Interpolation in digital modems - Part I: Fundamentals," *IEEE Trans. Commun.*, vol. 41, pp. 501-507, Mar. 1993.

[4] L. Erup, F. M. Gardner, and R. A. Harris, "Interpolation in digital modems - Part II: Implementation and performance," *IEEE Trans. Commun.*, vol. 41, pp. 998-1008, June 1993.

[5] D. Kincaid and W. Cheney, *Numerical Analysis*. Pacific Grove, 1991.

[6] J. Vesma and T. Saramäki, "Interpolation filters with arbitrary frequency response for all-digital receivers," in *Proc. IEEE Int. Symp. Circuits & Syst.*, Atlanta, GA, May 1996, pp. 568-571.

[7] J. Vesma, M. Renfors, and J. Rinne, "Comparison of efficient interpolation techniques for symbol timing recovery," in *Proc. IEEE Globecom 96*, London, UK, Nov. 1996, pp. 953-957.

[8] J. Vesma and T. Saramäki, "Optimization and efficient implementation of FIR filters with adjustable fractional delay," in *Proc. IEEE Int. Symp. Circuits & Syst.*, Hong Kong, June 1997, pp. 2256-2259.

[9] H. Ridha, J. Vesma, T. Saramäki, and M. Renfors, "Derivative approximations for sampled signals based on polynomial interpolation," in *Proc. 13th Int. Conf. on Digital Signal Processing*, Santorini, Greece, July 1997, pp. 939-942.

[10] H. Ridha, J. Vesma, M. Renfors, and T. Saramäki, "Discrete-time simulation of continuous-time systems using generalized interpolation techniques," in *Proc. 1997 Summer Computer Simulation Conference*, Arlington, Virginia, USA, July 1997, pp. 914-919.

[11] V. Tuukkanen, J. Vesma, and M. Renfors, "Combined interpolation and maximum likelihood symbol timing recovery in digital receivers," to be presented in *1997 IEEE Int. Conference on Universal Personal Communications*, San Diego, CA, USA, Oct. 1997.

[12] T. Saramäki and M. Ritoniemi, "An efficient approach for conversion between arbitrary sampling frequencies," in *Proc. IEEE Int. Symp. Circuits & Syst.*, Atlanta, GA, May 1996, pp. 285-288.

[13] J. Vesma, R. Hamila, T. Saramäki, and M. Renfors, "Design of polynomial interpolation filters based on Taylor series," in *Proc. IX European Signal Processing Conf.*, Rhodes, Greece, Sep. 1998, pp. 283-286.

[14] J. Vesma, R. Hamila, M. Renfors, and T. Saramäki, "Continuous-time signal processing based on polynomial approximation," in *Proc. IEEE Int. Symp. on Circuits and Systems*, Monterey, CA, USA, May 1998, vol. 5, pp. 61-65.

[15] D. Fu and A. N. Willson, Jr., "Interpolation in timing recovery using a trigonometric polynomial and its implementation," in *IEEE Globecom 1998 Communications Mini* Conference Record, Sydney, Australia, Nov. 1998, pp. 173–178.

[16] f. harris, "Performance and design considerations of Farrow filter used for arbitrary resampling," in *Proc. 13th Int. Conf. on Digital Signal Processing*, Santorini, Greece, July 1997, pp. 595–599.

[17] G. Oetken, "A new approach for the design of digital interpolation filters," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP–27, pp. 637–643, Dec. 1979.

[18] T. A. Ramstad, "Digital methods for conversion between arbitrary sampling frequencies," *IEEE Trans. Acoust. Speech, Signal Processing*, vol. ASSP–32, pp. 577–591, June 1984.

[19] T. A. Ramstad, "Fractional rate decimator and interpolator design," in *Proc. IX European Signal Processing Conf.*, Rhodes, Greece, Sep. 1998, pp. 1949–1952.

[20] R. W. Schafer and L. R. Rabiner, "A digital signal processing approach to interpolation," *Proc. IEEE*, vol. 61, pp. 692–702, June 1973.

[21] M. Unser, A. Aldroubi, and M. Eden, "Fast B-spline transforms for continuous image representation and interpolation," *Trans. Pat. Anal., Mach. Int.*, vol. 13, pp. 277–285, Mar. 1991.

[22] M. Unser, A. Aldroubi, and M. Eden, "Polynomial spline signal approximations: Filter design and asymptotic equivalence with Shannon's sampling theorem," *IEEE Trans. Information Theory*, vol. 38, pp. 95–103, Jan. 1992.

[23] J. Vesma, *Timing Adjustment in Digital Receivers Using Interpolation*. M.Sc. Thesis, Tampere, Finland: Tampere University of Tech., Department of Information Technology, Nov. 1995.

[24] V. Välimäki, *Discrete-Time Modeling of Acoustic Tubes Using Fractional Delay Filters*. Doctoral thesis, Espoo, Finland: Helsinki University of Technology, Dec. 1995.

[25] S. R. Dooley and A. K. Nandi, "On explicit time delay estimation using the Farrow structure," *Signal Processing*, vol. 72, pp. 53–57, Jan. 1999.

[26] J. Vesma, "A frequency-domain approach to polynomial-based interpolation and the Farrow structure," to appear *IEEE Trans. on Circuits and Systems II*, March 2000.

[27] J. Vesma, Optimization and Applications of Polynomial-Based Interpolation Filters. Dr. Tech. Thesis, Tampere, Finland: Tampere University of Tech., Department of Information Technology, May 1999