4 Fractional Delay Waveguide Filters

The fractional delay filters studied in Chapter 3 are essential in digital waveguide models. This is a consequence of the fact that waveguide models deal with propagation delays that are generally not integral multiples of the sampling interval. When FD approximation is used, *spatially continuous* discrete-time simulation of a physical waveguide is achieved.

In practice fractional delays are needed for two purposes in waveguide models:

1) for *controlling the length* of a digital waveguide, and

2) for *connecting* two or more waveguides at arbitrary points.

The former case, which involves interpolation of the signal value of delay lines, has been used for several years. Jaffe and Smith (1983) used a first-order allpass filter to tune a waveguide string model. Karjalainen and Laine (1991) showed how Lagrange interpolation can be applied to the same problem.

The latter approach was introduced recently by Välimäki *et al.* (1993a). In that paper it was shown how two or three digital waveguides with different impedances can be connected so that their junction is located between two sampling points. This technique can be employed, e.g., for modeling the human vocal tract or for modeling the finger holes of woodwind instruments.

In this chapter, several structures for fractional delay waveguide filters are discussed. Section 4.1 discusses deinterpolation that is the inverse operation of interpolation. In Section 4.2 it is shown how fractional delay waveguide systems can be realized by utilizing the operations of interpolation and deinterpolation. Two specific fractional delay junction structures, an interpolated junction of two and three acoustic tubes, are tackled. In Sections 4.3 and 4.4 they are implemented using FIR FD filters whereas Sections 4.5–4.7 introduce the corresponding allpass FD structures.

4.1 Deinterpolation

Another basic operation employed in fractional delay waveguide models side by side with interpolation is called *deinterpolation*. It is needed when connecting digital waveguides at fractional points. Deinterpolation was first introduced in Välimäki *et al.* (1993a). Earlier, Karjalainen and Laine (1991) had been using this technique for modeling a vibrating string, but the novelty of this technique was not recognized at that time.

4.1.1 Introduction to Deinterpolation

Interpolation is used for estimating the value of the signal between known samples. In contrast, deinterpolation is used for computing the sample values when the amplitude of

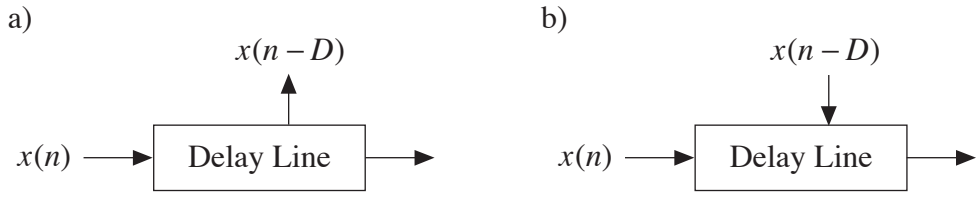

Fig. 4.1 a) Interpolation from a delay line. b) Deinterpolation into a delay line.

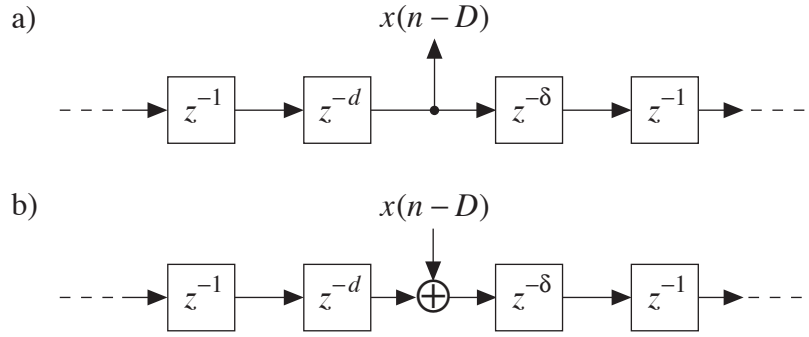

Fig. 4.2 More detailed diagrams of a) interpolation and b) deinterpolation showing the fractional delay elements.

the signal between sampling points is known. Thus, deinterpolation can be understood as an *inverse operation to interpolation*† in the sense that it is an operation used for computing the input of a system while interpolation estimates the value of the output of a system.

In Fig. 4.1a, an output is located at an arbitrary point on the delay line, and the output value is generally obtained by bandlimited interpolation. In Fig. 4.1b, an input is located at an arbitrary point along the delay line. If the input point does not correspond to one of the sample points of the delay line but is directed between them, *bandlimited deinterpolation* must be applied to spread the incoming signal to the adjacent signal samples of the delay line.

More detailed versions of the block diagrams given in Fig. 4.1 are presented in Fig. 4.2 showing the individual unit and fractional delay elements. One of the unit delay elements of the delay line has been split into two pieces, that is

$$z^{-1} = z^{-(d+\delta)} = z^{-d} z^{-\delta} \tag{4.1}$$

where d is the fractional delay ($0 < d < 1$) and δ is the *complementary fractional delay* defined by

$$\delta = 1 - d \tag{4.2}$$

By examining the two cases presented in Figs. 4.2a and 4.2b, it can be understood

† The name 'deinterpolation' has been given to this operation instead of the more obvious 'inverse interpolation' because in mathematics, inverse interpolation denotes the problem of finding an estimate for the argument x when the value of the function $y = f(x)$ is given (see, e.g., Hildebrand, 1974, pp. 68–70 or Kreyszig, 1988, p. 969). Deinterpolation is thus not the same operation as inverse interpolation. Note also that the operation of decimation, which is used in multirate signal processing as an inverse operation of interpolation, is yet another technique.

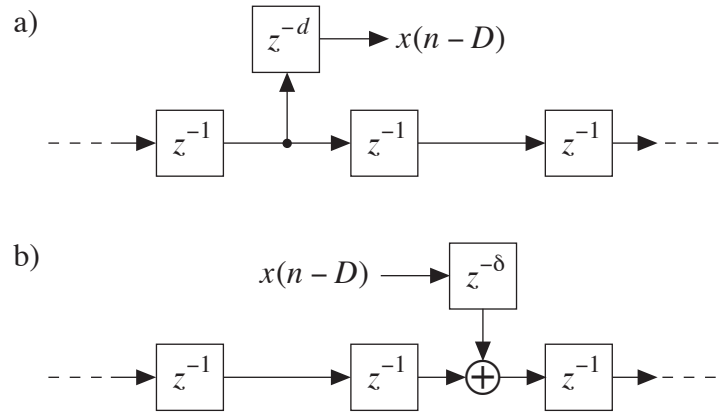

Fig. 4.3 The operations of a) interpolation and b) deinterpolation can equivalently be presented by pulling the fractional delay elements out from the delay lines.

that in interpolation it is needed to produce a fractional delay of d whereas in deinterpolation the complementary fractional delay δ is required. Thus, configurations of Fig 4.2 can be equivalently presented as shown in Fig. 4.3. Now the fractional delays have been pulled out from the delay line. Consequently, the delay line is again constructed of a chain of unit delays. Note that the deinterpolator is connected to the delay line one unit delay to the right from the point where the corresponding interpolator is connected (see Fig. 4.3).

4.1.2 Deinterpolation Versus Interpolation

In the following we discuss the relation between nonrecursive (or FIR) interpolation and *nonrecursive deinterpolation*. The latter may be interpreted as an inverse operation of the former. The inverse nature of these operations is essentially reflected by the fact that the FIR interpolator approximates the total delay D while the FIR deinterpolator approximates the complementary total delay Δ defined by

$$\Delta = N - D \tag{4.3}$$

where N is the order of the FIR interpolator. The fractional part δ of Δ is related to d according to Eq. (4.2). The relation of D and Δ is illustrated in Fig. 4.4 for an even filter order N.

The ideal FIR interpolator and the corresponding ideal FIR deinterpolator can thus be presented as systems with transfer functions z^{-D} and $z^{-\Delta}$, respectively. Were an Nth-order FIR interpolator and the corresponding Nth-order FIR deinterpolator cas-

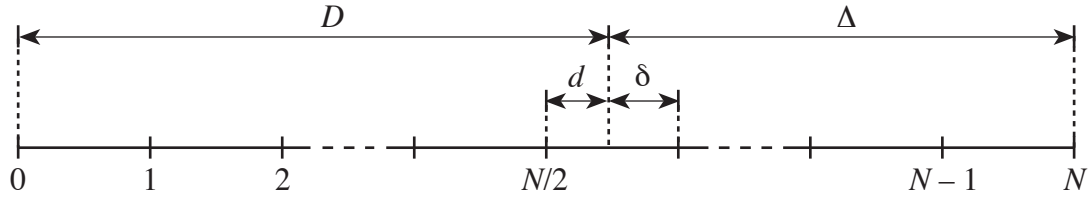

Fig. 4.4 A nonrecursive interpolator approximates the delay D but a nonrecursive deinterpolator the complementary delay Δ. Also shown are the fractional delay d and the complementary fractional delay δ.

caded, the resulting system would merely cause an incoming sequence $x(n)$ to be delayed by N samples.

If the impulse response of an FIR interpolating filter approximating delay D is reversed in time, the resulting filter approximates the complementary delay Δ. This statement is easily confirmed by considering a hypothetical Nth-order linear-phase interpolating filter with the frequency response

$$H(e^{j\omega}) = \left|H(e^{j\omega})\right| e^{-j\omega D} \tag{4.4}$$

The inversion in time corresponds to inverting the sign of the phase function. To make the filter causal, it is necessary to add a delay of N samples. The resulting time-reversed and delayed interpolator has a frequency response

$$\begin{aligned} H_{\text{rev}}(e^{j\omega}) &= e^{-j\omega N} H(e^{-j\omega}) = e^{-j\omega N}\left|H(e^{j\omega})\right| e^{j\omega D} = \left|H(e^{j\omega})\right| e^{-j\omega(N-D)} \\ &= \left|H(e^{j\omega})\right| e^{-j\omega\Delta} \end{aligned} \tag{4.5}$$

where Δ is the complementary total delay defined by (4.3). This clarifies the relation of the nonrecursive interpolator and deinterpolator. In Section 3.3.3 it was shown that for Lagrange interpolation the coefficients for the delay $N - D$ are the same as for D, but in reversed order.

4.1.3 Implementing Deinterpolation Using an FIR Filter

Nonrecursive deinterpolation can be thought of as a superposition of the signal values onto the delay line using the *transposed FIR filter structure* with interpolating coefficients (Välimäki *et al.*, 1993a). The transpose of a digital filter structure is obtained by changing the direction of all branches by replacing all branch nodes with summation nodes (i.e., adders) and adders with branch nodes, and by interchanging the input and output nodes of the system (see, e.g., Jackson, 1989, pp. 74–76).

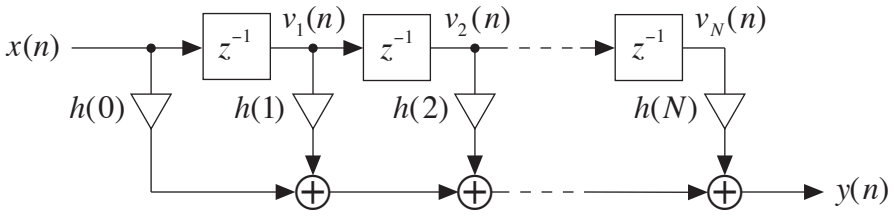

Fig. 4.5 The standard direct-form FIR filter structure that can be used for nonrecursive interpolation when interpolating coefficients $h(n)$ are used.

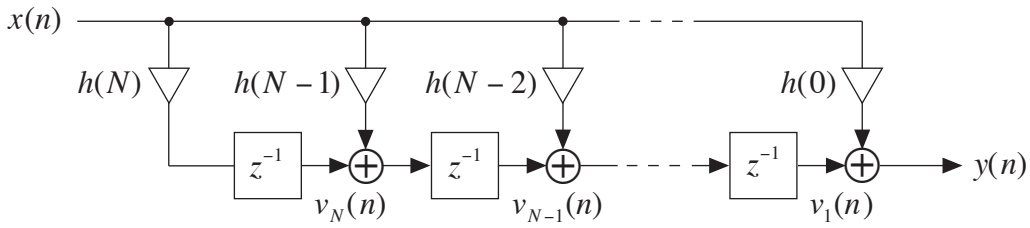

Fig. 4.6 The transposed direct-form FIR filter structure.

The standard direct-form FIR filter is shown in Fig. 4.5 and its transpose in Fig. 4.6. Note that the transposed structure in Fig. 4.6 has been vertically and horizontally mirrored to have the signal propagate from left to right or downwards.

The direct-form FIR filter computes a discrete convolution between its input sequence $x(n)$ and the coefficients $h(n)$ whereas the transposed FIR structure copies each input sample, multiplies each of them by one coefficient $h(n)$ and adds the products to the delay line, which is represented by the state variables $v_k(n)$ for $k = 1, 2, 3, ..., N$. The output of the transpose structure is computed as the sum of the input sample $x(n)$ multiplied by one of the weighting coefficients $h(n)$ and the value of the state variable $v_1(n)$. This can be expressed using the state-variable description which is a standard tool in discrete-time signal processing. The *next-state equation* is given by

$$\mathbf{v}(n+1) = \mathbf{F}^T\mathbf{v}(n) + \mathbf{g}x(n) \tag{4.6a}$$

and the *output equation* by

$$y(n) = v_1(n) + h(0)x(n) \tag{4.6b}$$

where we have defined the state vector (i.e., the delay line) as

$$\mathbf{v}(n) = \begin{bmatrix} v_N(n) & v_{N-1}(n) & \cdots & v_1(n) \end{bmatrix}^T \tag{4.6c}$$

the $N \times N$ matrix $\mathbf{F}$ as

$$\mathbf{F} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & & 0 \\ 0 & 0 & 0 & \ddots & \vdots \\ \vdots & & & \ddots & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix} \tag{4.6d}$$

and the coefficient vector $\mathbf{g}$ as

$$\mathbf{g} = \begin{bmatrix} h(N) & h(N-1) & \cdots & h(1) \end{bmatrix}^T \tag{4.6e}$$

Note that vector $\mathbf{g}$ excludes the first filter coefficient $h(0)$.

When the coefficients $h(n)$ of the standard direct-form and the transpose structure are the same, the two filters have *identical impulse responses and transfer functions*. The proof of this is quite straightforward (see, e.g., Jackson, 1989, pp. 74–76, or Välimäki, 1994b, Appendix B). For another proof based on Tellegen's theorem, see Fettweis (1971) or Oppenheim and Schafer (1975, pp. 173–178).

Thus the transpose of an FIR interpolator does not implement deinterpolation; its impulse response has to be *reversed* in time. Now we can write the state-variable description for the nonrecursive deinterpolator as

$$\mathbf{v}(n+1) = \mathbf{F}^T\mathbf{v}(n) + \mathbf{g}_{\text{rev}}x(n) \tag{4.7a}$$

$$y(n) = v_1(n) + h(N)x(n) \tag{4.7b}$$

where we have defined the coefficient vector $\mathbf{g}_{\text{rev}}$ as

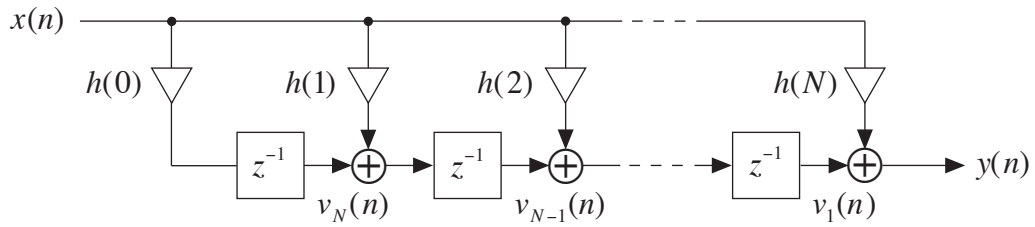

Fig. 4.7 Nonrecursive deinterpolation is implemented by the transpose FIR filter structure with a time-reversed impulse response.

$$\mathbf{g}_{\text{rev}} = \begin{bmatrix} h(0) & h(1) & \cdots & h(N-1) \end{bmatrix}^T \tag{4.7c}$$

The vector $\mathbf{v}(n)$ and matrix $\mathbf{F}$ are defined as in Eqs. (4.6c) and (4.6d) above.

Equivalently, one may imagine that both the delay line of the transpose structure as well as the order of state variables $v_k(n)$ have to be reversed to obtain the FIR realization of deinterpolation. Figure 4.7 illustrates the implementation of nonrecursive deinterpolation by the transpose nonrecursive structure with a reversed impulse response. Apart from the ordering of coefficients this filter is equivalent to the transpose structure of Fig. 4.6.

Based on the above discussion it is understood that the transfer function of the FIR deinterpolator can be written as

$$H_{\text{d}}(z) = \sum_{n=0}^{N} h_{\text{d}}(n) z^{-n} = \sum_{n=0}^{N} h(N-n) z^{-n} = H_{\text{rev}}(z) \tag{4.8}$$

This implies that if the transfer function $H(z)$ of the FIR interpolator approximates the transfer function z^{-D}, then $H_{\text{d}}(z)$ approximates the z-transform $z^{-\Delta}$ as discussed earlier.

4.1.4 Adding a Signal to a Fractional Point of a Delay Line Using an FIR Deinterpolator

Consider addition of a signal to an arbitrary fractional point on a delay line as illustrated in Fig. 4.1b. The implementation of this system by means of nonrecursive deinterpolation is presented in Fig. 4.8. Note that now the deinterpolator does not have independent state variables as was assumed in the preceding section, but employs the unit delays of the delay line.

When a signal $w(n)$ is deinterpolated onto the delay line, the resulting signal may be expressed as

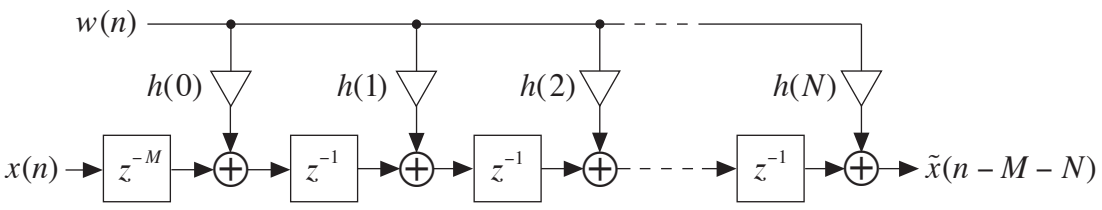

Fig. 4.8 Adding a signal to a fractional point of a delay line using a deinterpolator.

$$\tilde{x}(n-M-k) = x(n-M-k) + h_{\text{rev}}(k)w(n) \quad \text{for } k = 0,1,2,\ldots,N \tag{4.9}$$

where $h_{\text{rev}}(n)$ are the coefficients of the deinterpolation filter, and $x(k)$ and $\tilde{x}(k)$ are the signals in the delay line before and after deinterpolation, respectively. For the next sampling cycle, the updated signal values are shifted so that we can write

$$x(n-M-k-1) = \tilde{x}(n-M-k) \quad \text{for } k = 0,1,2,\ldots,N-1 \tag{4.10}$$

Equation (4.9) can be expressed in vector form as

$$\tilde{\mathbf{x}}_n = \mathbf{x}_n + \mathbf{h}_{\text{rev}} w(n) \tag{4.11a}$$

where $\tilde{\mathbf{x}}_n$ is the modified version of vector $\mathbf{x}_n$ defined by

$$\mathbf{x}_n = [x(n-M) \quad x(n-M-1) \quad \cdots \quad x(n-M-N)]^T \tag{4.11b}$$

and $\mathbf{h}_{\text{rev}}$ is the coefficient vector of the FIR deinterpolator containing the coefficients $h_{\text{rev}}(n)$ that are the same as those used with interpolation but in reversed order, i.e., $h_{\text{rev}}(n) = h(N-n)$ for $n = 0,1,2,\ldots,N$:

$$\begin{aligned} \mathbf{h}_{\text{rev}} &= [h_{\text{rev}}(0) \quad h_{\text{rev}}(1) \quad \cdots \quad h_{\text{rev}}(N)]^T \\ &= [h(N) \quad h(N-1) \quad \cdots \quad h(0)]^T \end{aligned} \tag{4.11c}$$

The vector notation shows that the result of the deinterpolation appears in the delay line itself.

4.1.5 Discussion

The same number of additions and multiplications is needed for the implementation of interpolation and deinterpolation. However, in deinterpolation, cumulative additions to a delay line are needed and in software implementation they usually require more than one instruction, e.g., a read from memory, an addition, and a save to memory. Thus, in practice deinterpolation may be computationally more expensive than interpolation.

It can be argued that deinterpolation could be implemented just as well by a direct-form FIR interpolator that approximates the delay of Δ samples instead of D. This is true and seems to make the definition of a new technique such as deinterpolation quite unnecessary. However, the main advantage of deinterpolation can be clearly observed by examining addition to the delay line as illustrated in Fig. 4.8. If this system were realized by a direct-form FIR interpolator, N extra unit delays would be required. Also, the implementation would not be as intuitive, since the fractional delay would have to be processed separately and the result be added to the delay line signal at a point located about $N/2$ samples before the fractional point D. In the following, other benefits of deinterpolation will become clear. Especially the implementation of fractional delay junctions of waveguides is straightforward using nonrecursive deinterpolation .

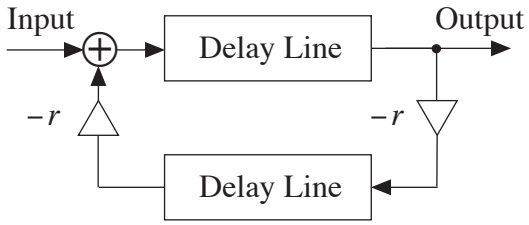

Fig. 4.9 Block diagram of the waveguide model for an ideal string.

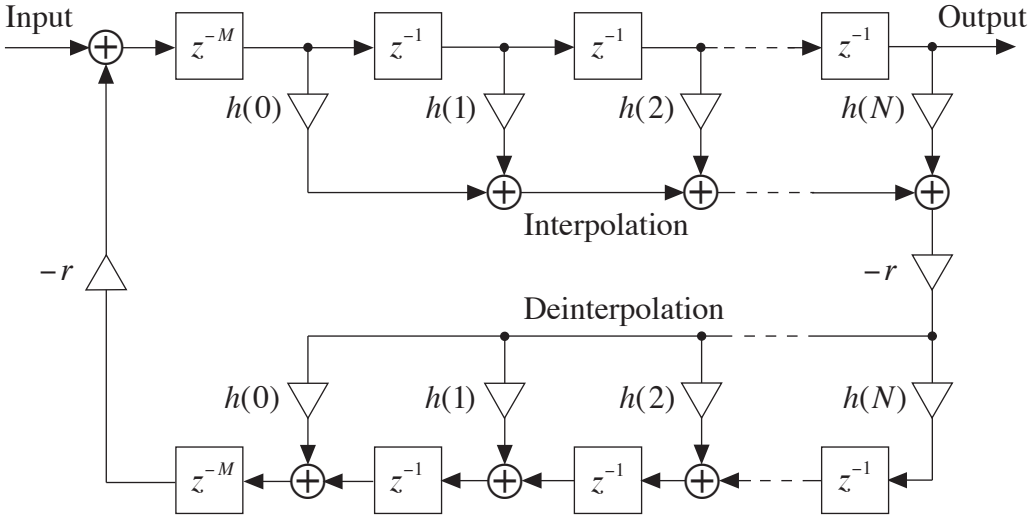

Fig. 4.10 FD waveguide filter for modeling a vibrating string of arbitrary length.

4.2 Fractional Delay Waveguide Systems

Our main interest in this work is not to operate with single delay lines but with digital waveguides that are *bidirectional* delay lines. With the operation of deinterpolation it is possible to implement a variable-length digital waveguide or connect waveguides at arbitrary points.

4.2.1 Variable-Length Digital Waveguide

Karjalainen and Laine (1991) used nonrecursive interpolation and deinterpolation for implementing a waveguide model for a variable-length guitar string. Figure 4.9 illustrates the system. The two delay lines represent waves that travel in opposite directions along the string. The ends of the vibrating string are fixed and the waves reflect with inverted phase. In this simple model the end reflection is assumed to be purely resistive, i.e., independent of the frequency.

The length of the string has to be continuously variable to produce tones on some musical scale. The actual implementation of this model by means of a fractional delay waveguide filter is presented in Fig. 4.10. Fractional delay is only needed at one end of the delay lines. The length of the upper delay line is controlled by an FIR interpolator. The interpolated result is multiplied by $-r$ and the produced reflected signal is superimposed onto the lower line using an FIR deinterpolator. The filter coefficients $h(n)$ of the deinterpolator are the same as those of the interpolator. Note, however, that the coefficients of the deinterpolator are in the reverse order with respect to those of the

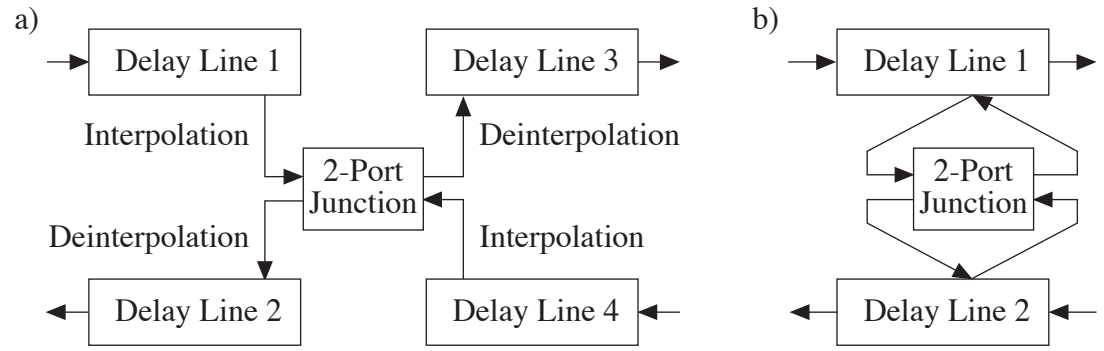

Fig. 4.11 Two alternative configurations for the implementation of a fractional delay junction of two digital waveguides.

interpolator.

In certain cases the two delay lines that form the waveguide may be combined and a more efficient realization results. For example, the system of Fig. 4.10 could be implemented by a single delay line with a reflection coefficient r^2 at one end. A single fractional delay filter could then be used to control its length. Only the delay between the input and output points of the one-delay-line version is different from that of the implementation of Fig. 4.10. This is usually acceptable. Other properties of the modified waveguide system remain the same.

4.2.2 Connecting Waveguides at Arbitrary Points

Deinterpolation proves to be an extremely useful operation in the context of fractional delay junctions of digital waveguides. An example of the simplest FD junction is the fractional delay endpoint shown in Fig. 4.10 above. Other elementary cases are

- a junction of two waveguides,
- a junction of a side branch connected to a waveguide, and
- a junction of three waveguides.

A junction indicates a change of impedance. A junction of two waveguides disappears or reduces to pure transmission if the impedances of the connected waveguides are equal. When more waveguides are coupled, the signal inside each one of them sees that the input impedance of the junction is equal to the parallel connection of the impedances of the other waveguides. Thus scattering occurs. Part of the wave is transmitted through the junction to the other waveguides while the rest is reflected back.

Figure 4.11 depicts two ways of connecting two digital waveguides at fractional points. In Fig. 4.11a the two waveguides are represented by four distinct delay lines: delay lines 1 and 2 form one digital waveguide and lines 3 and 4 the other. In contrast, in Fig. 4.11b the left-going and right-going delay lines of the two waveguides have been combined. In both cases, the signals entering the two-port scattering junction are interpolated out from the delay lines and the outcoming (transmitted) signals are deinterpolated onto the delay lines.

In the case of Fig. 4.11a some extra unit delays have to be reserved for each delay line when FIR FD filters are used for realizing interpolation and deinterpolation. This is because FIR interpolators need to make use of signal samples at both sides of the inter-

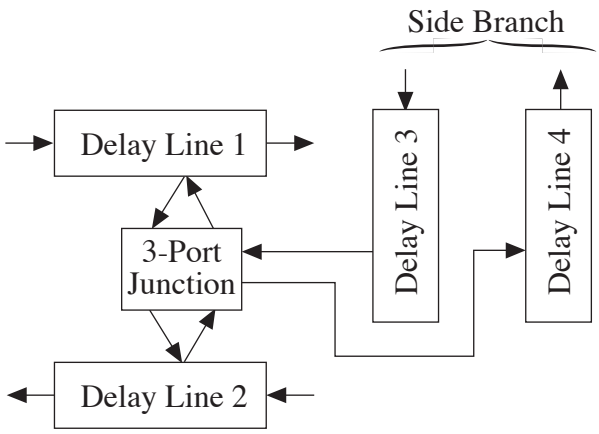

Fig. 4.12 An example of a fractional delay junction of three digital waveguides: a side branch in a waveguide.

polation point. The system of Fig. 4.11b does not usually require these extra data, since there are unit samples on both sides of the junction anyway. It is, nevertheless, important to make sure that the junction is not located nearer than about $N/2$ unit delays away from either end of the delay lines. Otherwise the accuracy of the interpolators has to be reduced by lowering the order of the FD filter.

The above remarks are valid for FD junctions of three or more waveguides as well. A special case of a junction of three waveguides is illustrated in Fig. 4.12. In this example a side branch represented by delay lines 3 and 4 is connected to a waveguide. The three-port junction determines how much of each input signal is reflected back and how much is transmitted to the other waveguides. The input signals of the junction are obtained by interpolation and the output signals are superimposed onto the delay lines by deinterpolation. Later in this chapter this structure is applied to modeling of a finger hole of a woodwind instrument. It is also useful for implementing the junction of the vocal and nasal tract in speech production models.

4.3 Interpolated Waveguide Tube Model

An inherent limitation of the traditional Kelly–Lochbaum model is that the scattering junctions can only be located at integral points of the waveguide. Consequently, the total delay length of the model is quantized to a multiple of the unit delay. These problems, and earlier attempts to overcome them, were discussed in Section 2.2.8. In this section we demonstrate how it is possible to eliminate these limitations using fractional delay waveguide modeling techniques that were introduced above.

Figure 4.13 illustrates the basic idea of the enhanced tube model that is developed in this work. An acoustic tube has been split into a number of parts each of which is modeled by a cylindrical tube section like in the traditional approach, or by a conical tube section, as described in Section 2.4. The length of each tube section can be a real number independent of the lengths of other sections.

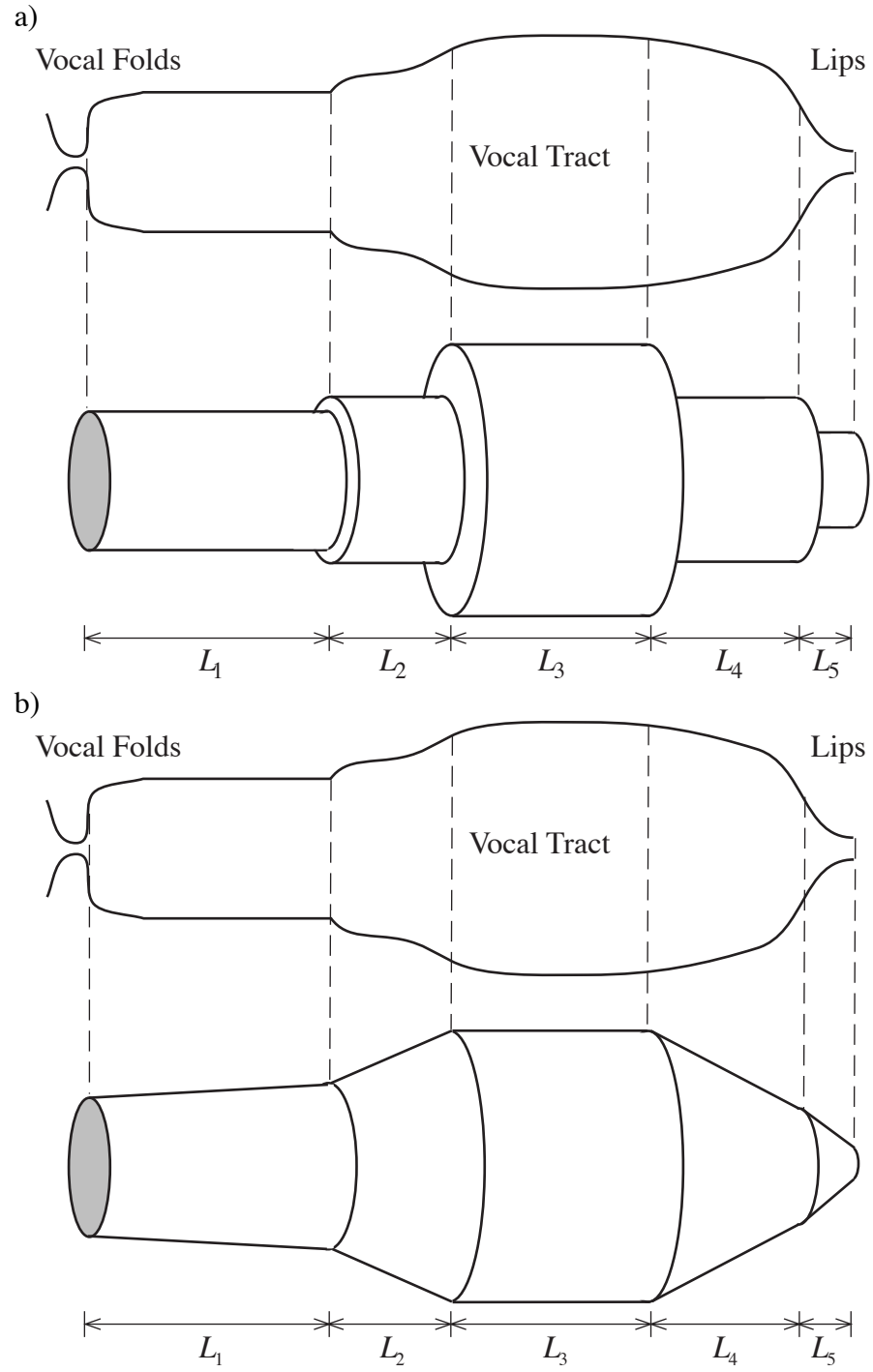

Fig. 4.13 The profile of the vocal tract and its approximation by a) cylindrical and b) conical tube sections of different length.

4.3.1 Interpolated Scattering Junction

As discussed earlier in this chapter, the length of a waveguide may be adjusted by applying interpolation and deinterpolation techniques. With the help of these operations it is moreover possible to locate a scattering junction at a fractional point of a waveguide. The block diagram of a single fractional delay, or interpolated, two-port junction is presented in Fig. 4.14. The input signals of the scattering junction (denoted by '*S*') are now obtained using interpolation and the output signals of the junction are fed back to the delay lines using deinterpolation. The wave scattering component is computed by

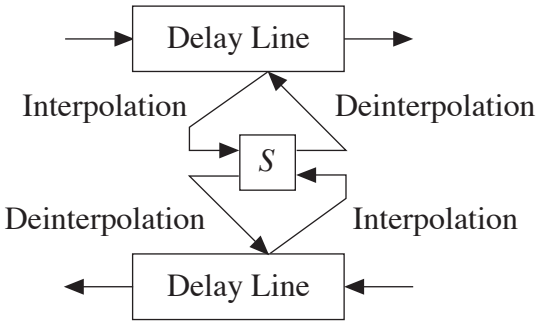

Fig. 4.14 Block diagram of the interpolated two-port scattering junction. The block denoted by '*S*' is a two-port scattering junction.

multiplying the difference of the interpolated signals with the reflection coefficient r.

A more detailed block diagram of a single FD two-port junction is shown in Fig. 4.15. This example is valid for pressure signals. In this figure it is seen that only the reflected and transmitted signal components have to be explicitly computed; the direct signals propagating in the two delay lines need no attention since the effect of the junction is superimposed onto the delay lines. Note that in the lower delay line, the interpolator approximates the complementary delay δ and the deinterpolator the delay d. In other words, the roles of these two operations are interchanged.

Based on the observation that only the reflected and transmitted signals have to be explicitly interpolated and deinterpolated we may simplify the block diagrams for the FD junctions. Figure 4.16 illustrates the junction computation for the volume velocity and the acoustic pressure. The signal paths starting from the delay line indicate that the signal is obtained via interpolation. Consequently, the paths leading to the delay line denote that the signal has to be deinterpolated onto the delay line. It is seen that two

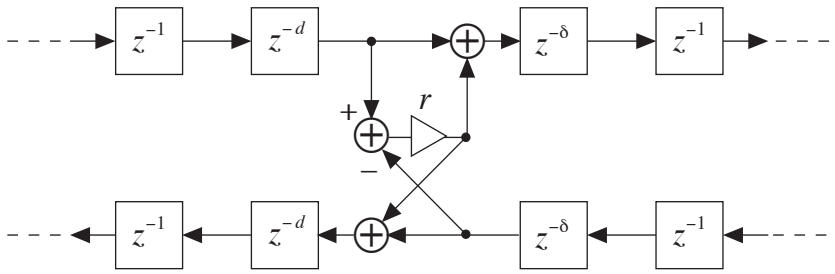

Fig. 4.15 The FD two-port scattering junction for the pressure signal.

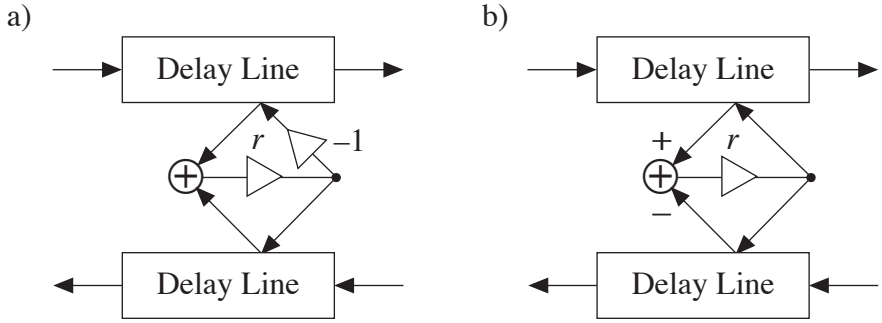

Fig. 4.16 Diagrams of the FD two-port junctions for a) volume velocity and b) pressure signals showing that only the reflected and transmitted signal components for both directions are actually computed.

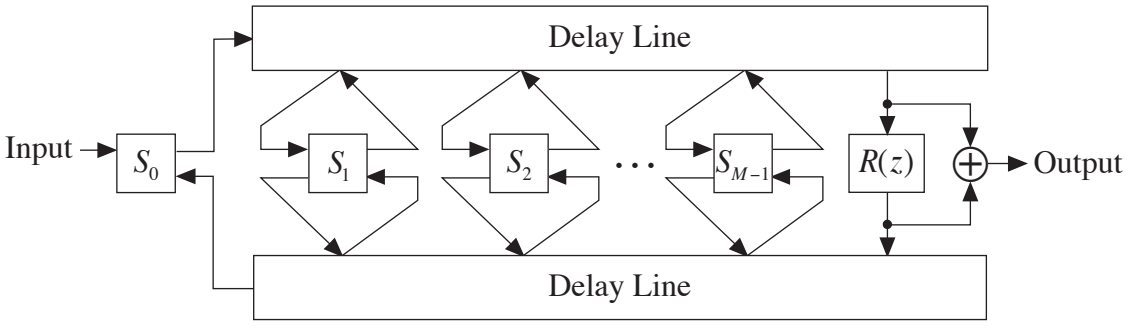

Fig. 4.17 Fractional delay M-tube model for sound pressure.

interpolations and two deinterpolations are required for the realization of an FD two-port junction, i.e., *four filters* are needed to turn a basic KL junction into a fractional delay one.

A complete fractional delay M-tube model is shown in Fig. 4.17. In principle an arbitrary number of interpolated two-port junctions may be used between the endpoints of the waveguide. One of the endpoints of the model may be fixed (the left end in Fig. 4.17) while the other has to be interpolated to enable continuous change of the total length. In principle, this model does not suffer from the restrictions of the traditional KL model that were discussed in Section 2.2.8 (see also Välimäki *et al.*, 1994b). A vocal tract model based on the FD waveguide approach can be controlled in a more straightforward way, since the scattering junctions are movable (Välimäki *et al.*, 1994a; Kuisma, 1994).

4.3.2 Implementing the FD Scattering Junction Using FIR Filters

Interpolation and deinterpolation can be implemented by means of FIR filters as described in Section 4.1. The computation of the interpolated KL junction using Nth-order FIR filters is depicted in detail in Fig. 4.18. It is assumed that the filter coefficients $h(n)$ of all the interpolating and deinterpolating FIR filters in this configuration are the same. The FIR filter coefficients $h(n)$ can be designed using one of the methods discussed in Chapter 3.

The signal value is first interpolated out from both the upper and the lower delay line, i.e.

$$p^{+}(n,m+D) = \mathbf{h}^T \mathbf{p}^{+}(n,m) \tag{4.12a}$$

$$p^{-}(n,m+D) = \mathbf{h}^T \mathbf{p}^{-}(n,m) \tag{4.12b}$$

where

$$\mathbf{h} = \begin{bmatrix} h(0) & h(1) & h(2) & \cdots & h(N) \end{bmatrix}^T \tag{4.12c}$$

$$\mathbf{p}^{+}(n,m) = \begin{bmatrix} p^{+}(n,m) & p^{+}(n,m+1) & p^{+}(n,m+2) & \cdots & p^{+}(n,m+N) \end{bmatrix}^T \tag{4.12d}$$

$$\mathbf{p}^{-}(n,m) = \begin{bmatrix} p^{-}(n,m) & p^{-}(n,m+1) & p^{-}(n,m+2) & \cdots & p^{-}(n,m+N) \end{bmatrix}^T \tag{4.12e}$$

Here n is the time index and m the spatial index as was discussed in the beginning of Chapter 2. Now it is again helpful to use the spatial variable although it is in principle

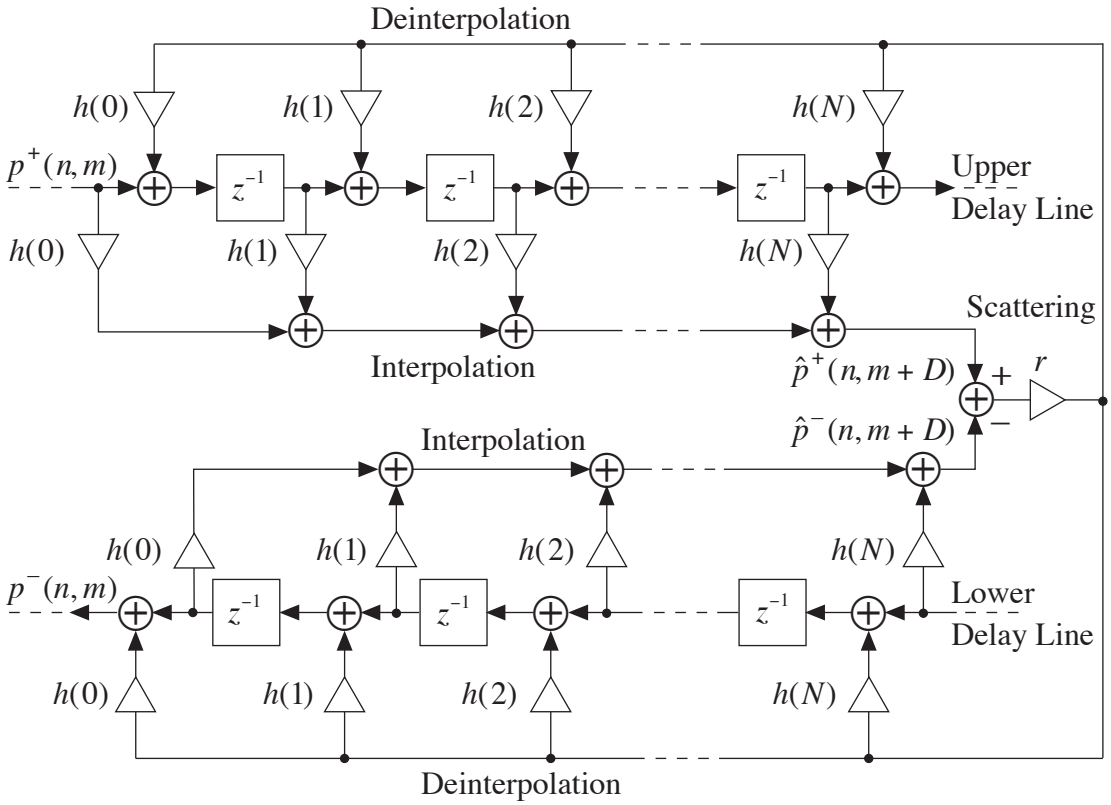

Fig. 4.18 Implementation of an FD two-port junction for pressure signals using Nth-order FIR interpolators and deinterpolators. The interpolating coefficients $h(n)$ are assumed to be the same for all the filters.

redundant. Note that the interpolation is essentially a spatial operation in this context, and it is appropriate to call D the *fractional spatial index*.

The difference of the interpolated signals $p^+(n,m+D)$ and $p^-(n,m+D)$ is computed and multiplied by the reflection coefficient r, that is

$$w(n) = r[p^+(n,m+D) - p^-(n,m+D)] = r\,\mathbf{h}^T[\mathbf{p}^+(n,m) - \mathbf{p}^-(n,m)] \tag{4.13}$$

The resulting signal $w(n)$ is then deinterpolated onto both delay lines. We can write the updated delay line vectors as

$$\mathbf{p}^+(n,m) = \mathbf{p}^+(n,m) + \mathbf{h}^T w(n) \tag{4.14a}$$

$$\mathbf{p}^-(n,m) = \mathbf{p}^-(n,m) + \mathbf{h}^T w(n) \tag{4.14b}$$

Since the last term $\mathbf{h}^T w(n)$ of the above equations is the same, it is economical to compute it separately and then use it in both equations. Also, as shown by Eq. (4.13) it is practical to first compute the difference of signal vectors $\mathbf{p}^+(n,m)$ and $\mathbf{p}^-(n,m)$ and only then evaluate the inner product with $\mathbf{h}^T$. Thus it is seen that by reorganizing the computation it is possible to achieve reduction of complexity. In Fig. 4.19, the two interpolators as well as the two deinterpolators have been combined. This does not reduce the number of additions (or subtractions) which is still $4N + 3$. However, in the straightforward configuration (Fig. 4.18) the number of multiplications is $4N + 5$, but in the optimized one (Fig. 4.19) only $2N + 3$. The reduction in the number of multiplica-

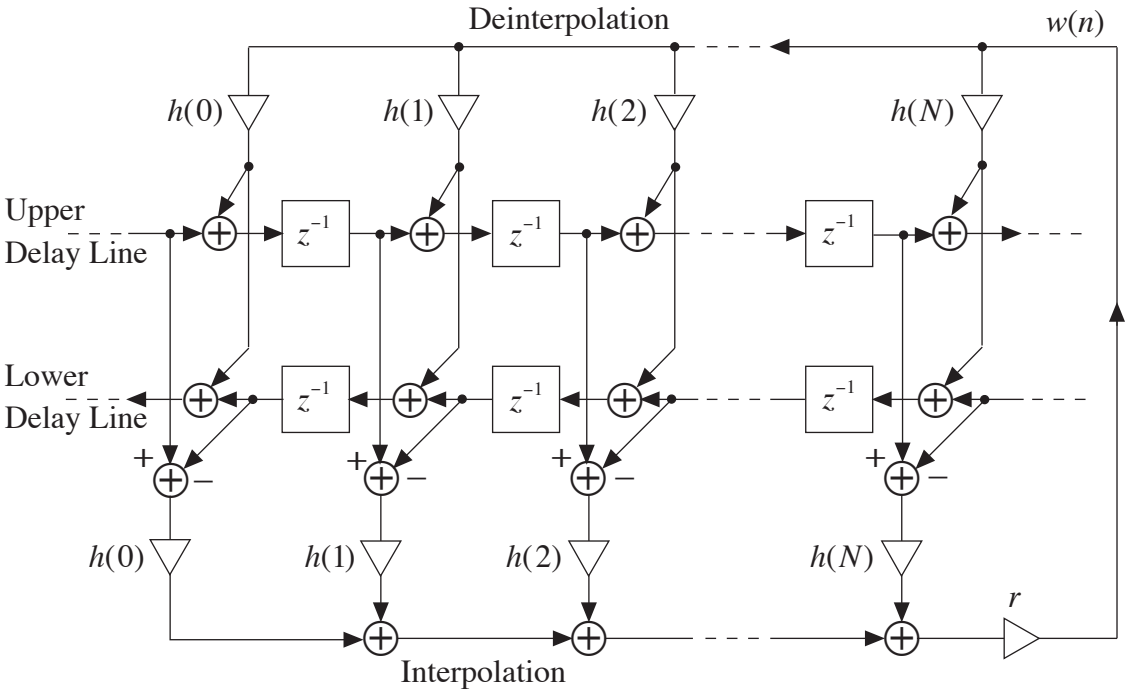

Fig. 4.19 Efficient implementation of an FD two-port junction for pressure signals.

tions is about 50%.

There are only minor differences between the FD junctions for the pressure and volume velocity signals. The junction for the volume velocity signal is obtained by replacing one of the subtractions with an addition and by inverting the sign of the signal that is to be deinterpolated into the upper delay. These differences are easily noticed in Fig. 4.16.

The FD KL model works correctly even if the FIR filters associated with contiguous FD junctions overlap, i.e., two junctions can be located so close to each other that they use the same unit delays of the waveguide for interpolation and deinterpolation. It should be remembered, however, that then it is of paramount importance to execute the operations in the following order (Välimäki *et al.*, 1993a):

1) read the signal values of the delay lines and compute the outputs of all the interpolating filters in the system,
2) compute the scattered signals $w(n)$ of all M junctions, and
3) superimpose all the results onto the delay lines using deinterpolators.

If this rule is violated, e.g., so that all the computations associated with the junctions are evaluated one after the other, the system becomes noncausal since part of the signal energy will flow through the waveguide within one cycle.

4.3.3 Transfer Functions of FIR Interpolators and Deinterpolators

To understand the functioning of the FD two-port junction discussed above, we consider the transmission and reflection functions of the junction from both sides. We assume that the coefficients of both the interpolators and the deinterpolators are the same but they may be in the same or in the reverse order. The prototype FIR transfer function that is used to assist in the comparison of the transfer functions in the follow-

ing is defined as

$$H(z) = \sum_{n=0}^{N} h(n) z^{-n} \tag{4.15}$$

It is assumed that the filter coefficients $h(n)$ have been designed so that the filter approximates the ideal bandlimited interpolation is some sense. The nominal value of the total delay of this filter is $D = \lfloor D \rfloor + d$.

The following notation is used: the superscripts '+' and '–' are used to denote that the filter operates on the signal that propagates in the positive and the negative direction, respectively; the subscripts 'i' and 'd' indicate that the filter is used for interpolation or deinterpolation, respectively.

Let us first write the transfer function for each of the four FIR filters involved in the FD junction. The signal value is interpolated from the upper delay line by the FIR interpolator with the transfer function $H_{\mathrm{i}}^{+}(z)$, that is

$$H_{\mathrm{i}}^{+}(z) = \sum_{n=0}^{N} h_{\mathrm{i}}^{+}(n) z^{-n} = \sum_{n=0}^{N} h(n) z^{-n} = H(z) \tag{4.16}$$

The deinterpolator that superimposes the scattered signal onto the upper delay line has the transfer function $H_{\mathrm{d}}^{+}(z)$. This filter approximates the complementary delay $\Delta = \lfloor \Delta \rfloor + \delta$ where δ is the complementary fractional delay. Thus the impulse response of the filter is the mirror-image of the prototype filter $H(z)$, that is, $h_{\mathrm{d}}^{+}(n) = h(N - n)$ for $n = 0, 1, ..., N$, and the transfer function may be expressed as

$$H_{\mathrm{d}}^{+}(z) = \sum_{n=0}^{N} h_{\mathrm{d}}^{+}(n) z^{-n} = \sum_{n=0}^{N} h(N-n) z^{-n} = z^{-N} \sum_{n=0}^{N} h(n) z^{n} = z^{-N} H(z^{-1}) \tag{4.17}$$

In the lower delay line the fractional delays d and δ have changed their roles as is seen, e.g., in Fig. 4.15. Thus, the interpolator $H_{\mathrm{i}}^{-}(z)$ approximates the complementary fractional delay δ and the deinterpolator $H_{\mathrm{d}}^{-}(z)$ the fractional delay d. This implies that now the impulse response of the interpolator must be reversed and the transfer function $H_{\mathrm{i}}^{-}(z)$ is written as

$$H_{\mathrm{i}}^{-}(z) = \sum_{n=0}^{N} h_{\mathrm{i}}^{-}(n) z^{-n} = \sum_{n=0}^{N} h(N-n) z^{-n} = z^{-N} H(z^{-1}) \tag{4.18}$$

The transfer function $H_{\mathrm{d}}^{-}(z)$ of the filter that deinterpolates the scattered signal onto the lower delay line is thus

$$H_{\mathrm{d}}^{-}(z) = \sum_{n=0}^{N} h_{\mathrm{d}}^{-}(n) z^{-n} = \sum_{n=0}^{N} h(n) z^{-n} = H(z) \tag{4.19}$$

It is seen that the interpolator of the upper line and the deinterpolator of the lower line have the same transfer function, that is

$$H_{\mathrm{i}}^{+}(z) = H_{\mathrm{d}}^{-}(z) = H(z) \tag{4.20}$$

The equivalence of these transfer functions is a consequence of the fact that they both

approximate the delay D. On the other hand

$$H_{\mathrm{d}}^{+}(z) = H_{\mathrm{i}}^{-}(z) = z^{-N}H(z^{-1}) \tag{4.21}$$

since both the filters $H_{\mathrm{d}}^{+}(z)$ and $H_{\mathrm{i}}^{-}(z)$ approximate the complementary delay Δ.

4.3.4 Choice of FIR FD Filter Design

In the above discussion on interpolated waveguide models, the properties of the FIR fractional delay filters used for interpolation and deinterpolation have not been taken into account. The theory of the FDWFs is independent of the choice of filter design method. Let us now consider how to choose the design technique.

The FIR FD filter should yield a good approximation of bandlimited interpolation at least at low frequencies (from about 50 Hz to some kHz). This is because the human auditory system is more sensitive at these frequencies than at higher frequencies. Strictly speaking the filter design technique should take into account the properties of human hearing. One possibility is to specify an auditory-based weighting function in the frequency domain and use that as an error weighting function in a filter design algorithm. For example, the complex LS design and minimax (Chebyshev) design algorithms allow the use of an arbitrary non-negative weighting function (Laakso *et al.*, 1994).

However, the author feels that it is unnecessarily involved to try to design FIR filters that are optimal in the auditory sense. Bandlimited approximation of ideal interpolation can be considered a more practical choice. It is apparent that oversampling by a small factor is necessary in order to obtain a satisfying approximation with a low-order FD filter. The upper part of the frequency band is left as a guard band.

FIR FD filter design methods that give good approximation on a limited low-frequency band include Lagrange interpolation, general complex least-squares design, and equiripple design (Laakso *et al.*, 1994). Lagrange interpolation has several good properties (see Section 3.3): it naturally gives a low-frequency approximation, its coefficients can be computed using a closed-form formula, and its magnitude response does not exceed unity.

In the general complex LS design (see Section 3.2.4), the approximation band can be specified (leaving the rest of the frequency band as a 'don't care' band). The filter coefficients are designed solving a matrix equation. In principle, the complexity of the design should not influence the choice of the filter since it is possible to design the coefficients beforehand and store them into a lookup table. In most practical cases, this is also the most efficient implementation technique for coefficient update.

From the viewpoint of waveguide modeling, a drawback of the general complex LS design is that its magnitude response oscillated around unity, potentially exceeding it at several frequencies. When the FD filter is used in a feedback loop, the magnitude response of the filter must not be greater than one at any frequency. Otherwise the overall system may become unstable. Hence, it is necessary to determine the maximum of the magnitude response of the filter for each delay value D, and then scale the output of the filter (or, equivalently, the filter coefficients) by the inverse of the maximum value.

Also the equiripple (minimax) design results in a filter which has an oscillating magnitude response. Thus scaling of the filter is necessary in order to restore the stability of the interpolated waveguide filter model. Oetken (1979) has proposed a simple

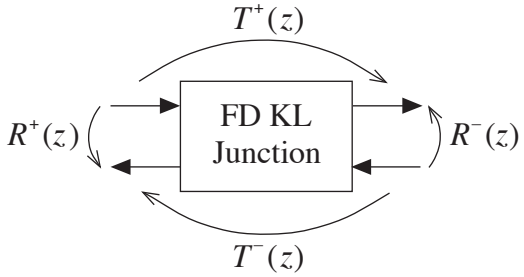

Fig. 4.20 Definitions of the transfer functions.

design technique that yields an approximately equiripple solution (see Section 3.2.5). This technique involves design of a linear-phase prototype filter, solving the frequencies where the approximation error (deviation from the ideal interpolator) is zero, and a matrix multiplication.

In order to have an FDWF system that can be implemented efficiently, it would be advantageous to use low-order FIR FD filters. Even-length (i.e., odd-order) FIR FD filters are easier to use especially in time-varying systems, since the filter taps need to be moved only when the integer part of the total delay D is changed. Based on experiments, it has been found that a third-order interpolating filter is a suitable compromise in terms of implementation costs and quality.

An easy first choice for an FIR FD filter design technique is Lagrange interpolation. If it is good enough, it is not necessary to consider the other methods. The complex general LS design and equiripple design yield smaller approximation error at middle frequencies and give a wider usable frequency band. Later in Section 4.3.7 these design methods are applied for the simulation of an acoustic tube.

4.3.5 Reflection and Transmission Functions

We proceed by studying the reflection and transmission transfer functions of the FD junction. These results were originally published in Välimäki *et al.* (1994b) but the notation differs slightly from that used here. The definitions of the transfer functions are illustrated in Fig. 4.20.

The transfer function through the FD junction in the positive direction, i.e., $T^+(z)$ in Fig. 4.20, may then be expressed as[†]

$$\begin{aligned} T^+(z) &= z^{-N} + rH_\mathrm{i}^+(z)H_\mathrm{d}^+(z) = z^{-N}\Big[1 + rH(z)H(z^{-1})\Big] \\ &= z^{-N}\Big[1 + r|H(z)|^2\Big] \end{aligned} \tag{4.22}$$

It can be seen from this z-transform that the impulse response (IR) through the port consists of a delayed unit impulse plus the convolution of $h(n)$ and the time-reversed (and delayed) IR $h_\mathrm{rev}(n)$ scaled by r. The length of the IR is thus $2N + 1$. Note that when the transfer functions $H_\mathrm{i}^+(z)$ and $H_\mathrm{d}^+(z)$ are cascaded, their phase functions cancel each other out and a *linear-phase IR* results.

Similarly, a linear-phase transfer function is obtained for transmission in the negative direction (see Fig. 4.20)

[†] Strictly speaking the last equality in (4.20) is true only when $z = e^{j\omega}$ but we use this notation for convenience.

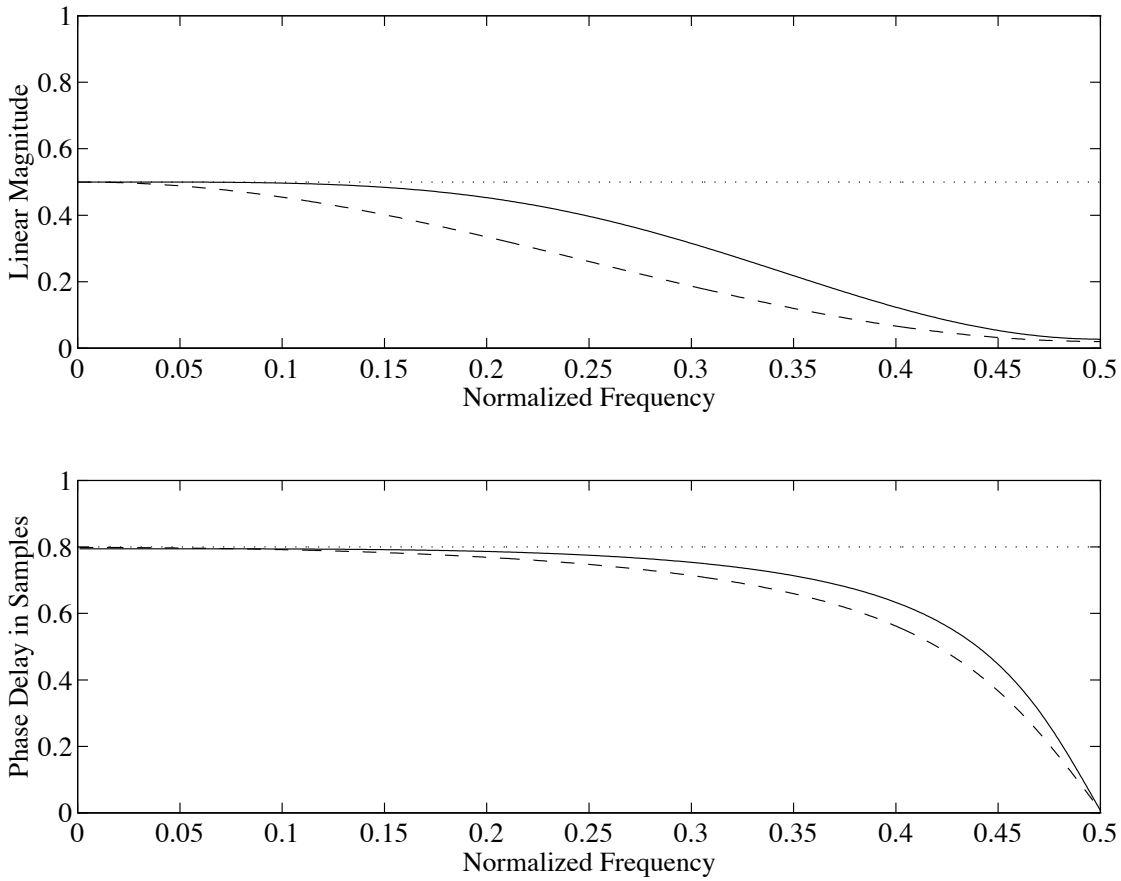

Fig. 4.21 The magnitude and phase delay responses of the *reflection function* $R^+(e^{j\omega})$ of the FD two-port junction. In the upper part the magnitude of the reflection function using first (dashed line) and third-order (solid line) interpolators is illustrated. The corresponding phase delay curves are shown in the lower part. The dotted line corresponds to the desired curve. In this example $r = 0.5$ and $d = 0.4$.

$$T^-(z) = z^{-N} - rH_\mathrm{i}^-(z)H_\mathrm{d}^-(z) = z^{-N}\Big[1 - r|H(z)|^2\Big] \tag{4.23}$$

The reflection function $R^+(z)$ in Fig. 4.20 can be written as

$$R^+(z) = rH_\mathrm{i}^+(z)H_\mathrm{d}^-(z) = r\big[H(z)\big]^2 \tag{4.24}$$

Again we see that a finite IR of length $2N + 1$ is obtained. It is the result of convolution of the IRs of the interpolation and deinterpolation filter (scaled by r), and does not have linear phase unless both $H_\mathrm{i}^+(z)$ and $H_\mathrm{d}^-(z)$ are linear-phase transfer functions or they are complementary.

The reflection function $R^-(z)$ (see Fig. 4.20) is given by

$$R^-(z) = -rH_\mathrm{i}^-(z)H_\mathrm{d}^+(z) = -rz^{-N}H(z^{-1})z^{-N}H(z^{-1}) = -rz^{-2N}\big[H(z)\big]^2 \tag{4.25}$$

Here the transfer functions of both the interpolation and deinterpolation filters appear in the time-reversed and delayed form. Also this transfer function does not in general have a linear phase.

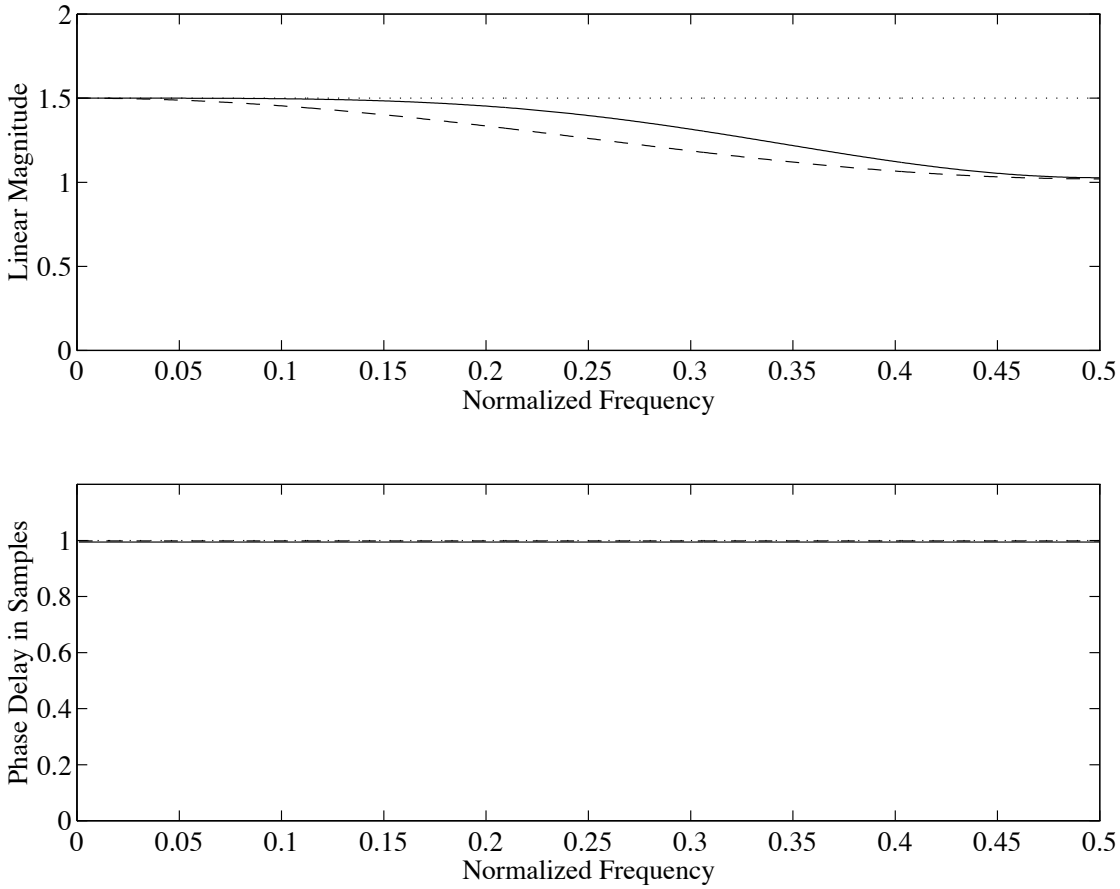

Fig. 4.22 The magnitude (upper) and phase delay (lower) responses of the *transmission function* $T^+(e^{j\omega})$ of the FD KL junction. The dashed lines correspond to the linear interpolator and the solid lines to the third-order Lagrange interpolator. The dotted line corresponds to the desired ideal curve. In this example $r = 0.5$ and $d = 0.4$.

By setting $z = e^{j\omega}$ in the above transfer functions it is possible to study the behavior of a single FD two-port junction. In the following examples we use first and third-order *Lagrange interpolators* and show the magnitude response and phase delay in two cases. In these examples $r = 0.5$ corresponding to a configuration where two tubes with diameters A_1 and $A_2 = 3A_1$ are connected. The fractional delay is $d = 0.4$, which is nearly the worst case†.

In Fig. 4.21 the reflection function $R^+(z)$ is studied. In the upper part of Fig. 4.21 the magnitude response $\left|R^+(e^{j\omega})\right|$ obtained from Eq. (4.24) is shown for the cases where the first ($N = 1$) and third-order ($N = 3$) Lagrange interpolators are used for the implementation of the FD two-port junction. Both magnitude responses have lowpass characteristics, which follows from the lowpass nature of the Lagrange interpolators. Remember that the error due to FIR FD approximation is applied twice in the reflection function as demonstrated by Eq. (4.24).

In the lower part of Fig. 4.21 the phase delay response is plotted for both cases. We have subtracted a delay of 2 samples from the phase delay of the third-order interpolator

† In the worst case, i.e., when the fractional delay is 0.5, the odd-order Lagrange interpolators have linear phase, and the effect of phase error would then not be shown in the figures. For this reason we use a delay value that is nearly but not exactly 0.5.

to more clearly view the difference of the two responses. The ideal phase delay $2d$ is represented by a dotted straight line. The phase delay of both responses is identical with the ideal one at low frequencies, but at high frequencies the curves approach zero.

The transmission function $T^+(z)$ of the two-port FD junction is examined in Fig. 4.22. The magnitude response $\left|T^+(e^{j\omega})\right|$ obtained from Eq. (4.22) is again shown for both the linear and the third-order interpolator (see upper part of Fig. 4.22). At low frequencies both curves coincide with the ideal result (dotted straight line), but at high frequencies they approach unity because the magnitude responses of the Lagrange interpolators approach zero. The third-order interpolators yield a slightly better result. The phase delay of both transmission functions is constant corresponding to a linear phase function. This is due to the cancellation of the phase responses of the interpolators and deinterpolators, as mentioned earlier.

From these examples we may derive some general properties of the FD KL junction implemented using Lagrange interpolation.

- The reflected signal is a lowpass filtered version of the input signal. The phase of the reflected signal can be severely distorted at high frequencies.
- Transmission through the junction attenuates high frequencies of a signal but does not cause any phase distortion.

4.3.6 Analysis of Errors Due to FD Approximation

The approximation errors of the FD filters degrade the performance of the waveguide model. The interpretation illustrated in Fig. 4.23 helps in the error analysis (Välimäki *et al.*, 1994b). The blocks $e^{-j\omega D}$ and $e^{-j\omega\Delta}$ denote the frequency responses of the ideal fractional delay and ideal complementary fractional delay, respectively. Here we have normalized the sampling rate to 1, i.e., the sample interval T is also equal to 1. The complementary total delay Δ in Fig. 4.23 is defined by Eq. (4.3) as

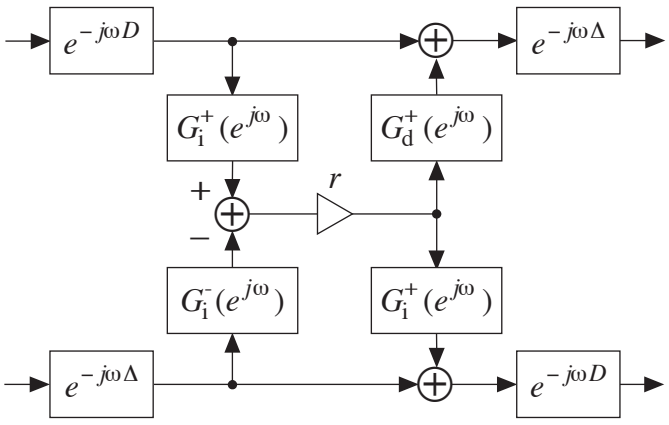

Fig. 4.23 Temporal flow diagram of the FD KL junction for pressure waves with ideal fractional delays of lengths D and Δ. The frequency response functions $G_i^+(e^{j\omega})$ and $G_i^-(e^{j\omega})$ include the interpolation error and $G_d^+(e^{j\omega})$ and $G_d^-(e^{j\omega})$ the deinterpolation error, and r is the reflection coefficient. Note that in this figure the direction of the lower delay line has been reversed to avoid crossing of signal paths.

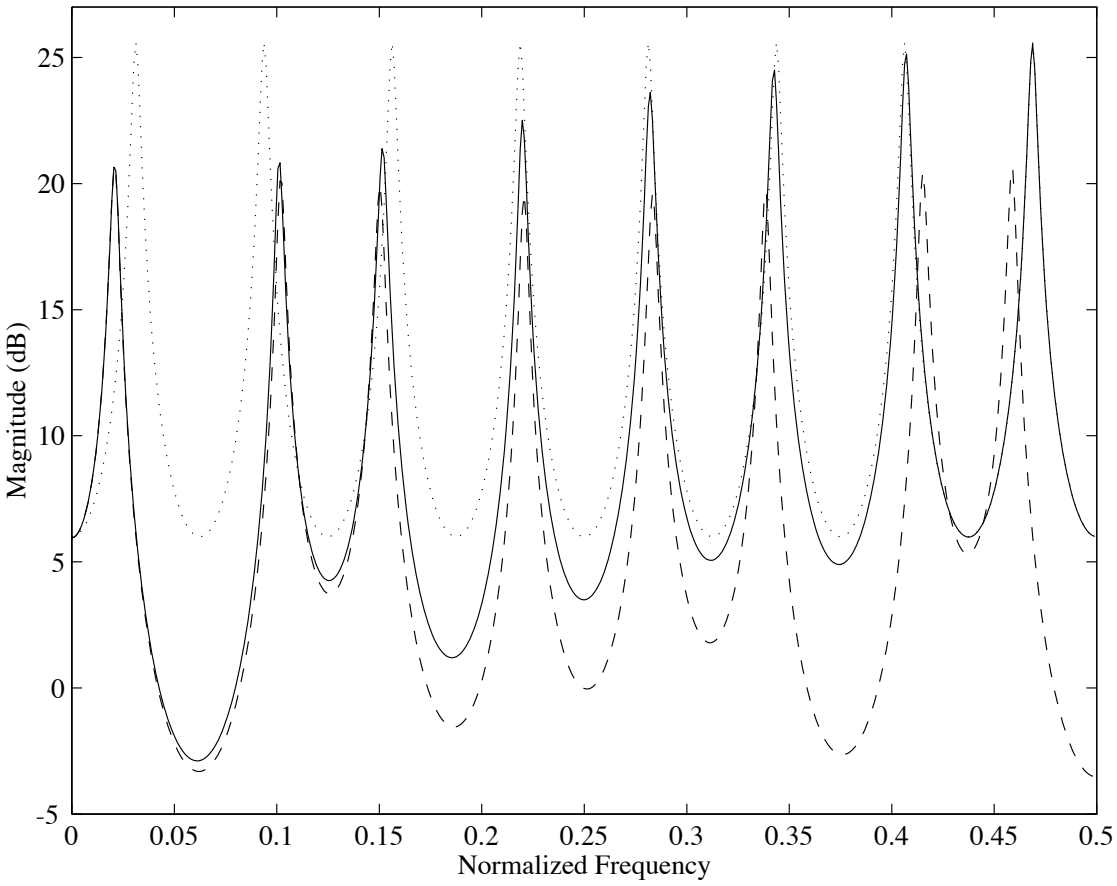

Fig. 4.24 The magnitude responses of the FD two-tube model with *linear interpolation* (solid line) and ideal interpolation (dashed line). The dotted line shows the magnitude response of a uniform tube.

$$\Delta = N - D \tag{4.26}$$

The frequency response functions $G(e^{j\omega})$ include the approximation error due to the fractional delay filters. Were ideal interpolating filters used, the frequency responses $G(e^{j\omega})$ would be equal to unity. These functions are defined as

$$G_{\mathrm{i}}^{+}(e^{j\omega}) = e^{j\omega D} H_{\mathrm{i}}^{+}(e^{j\omega}) = 1 - e^{j\omega D} E_{\mathrm{i}}^{+}(e^{j\omega}) \tag{4.27a}$$

$$G_{\mathrm{d}}^{+}(e^{j\omega}) = e^{j\omega \Delta} H_{\mathrm{d}}^{+}(e^{j\omega}) = 1 - e^{j\omega \Delta} E_{\mathrm{d}}^{+}(e^{j\omega}) \tag{4.27b}$$

$$G_{\mathrm{i}}^{-}(e^{j\omega}) = e^{j\omega \Delta} H_{\mathrm{i}}^{-}(e^{j\omega}) = 1 - e^{j\omega \Delta} E_{\mathrm{i}}^{-}(e^{j\omega}) \tag{4.27c}$$

$$G_{\mathrm{d}}^{-}(e^{j\omega}) = e^{j\omega D} H_{\mathrm{d}}^{-}(e^{j\omega}) = 1 - e^{j\omega D} E_{\mathrm{d}}^{-}(e^{j\omega}) \tag{4.27d}$$

where the error functions E represent the difference between the frequency response of the ideal delay element and the approximation. They are defined by

$$E_{\mathrm{i}}^{+}(e^{j\omega}) = e^{-j\omega D} - H_{\mathrm{i}}^{+}(e^{j\omega}) \tag{4.28a}$$

$$E_{\mathrm{d}}^{+}(e^{j\omega}) = e^{-j\omega \Delta} - H_{\mathrm{d}}^{+}(e^{j\omega}) \tag{4.28b}$$

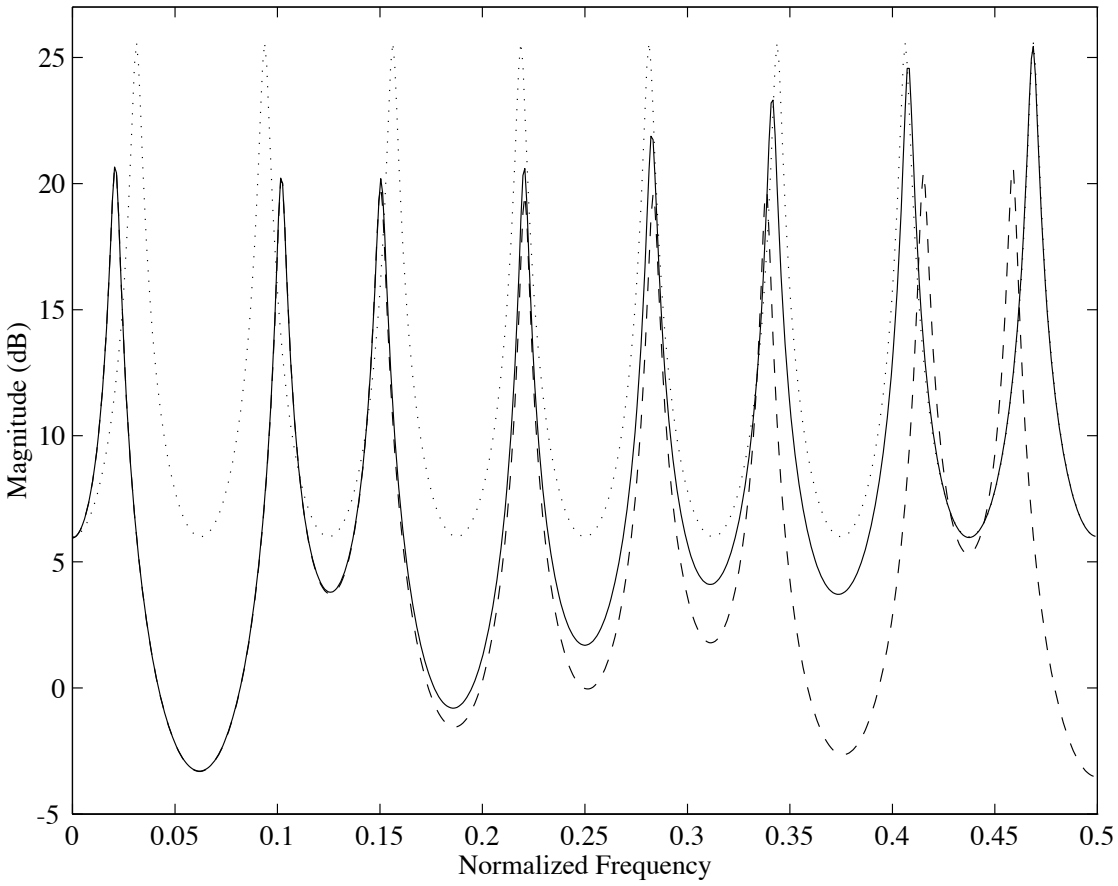

Fig. 4.25 The magnitude responses of the FD two-tube model with *third-order Lagrange interpolation* (solid line) and ideal interpolation (dashed line). The dotted line presents the magnitude response of a uniform tube.

$$E_{\mathrm{i}}^{-}(e^{j\omega}) = e^{-j\omega\Delta} - H_{\mathrm{i}}^{-}(e^{j\omega}) \tag{4.28c}$$

$$E_{\mathrm{d}}^{-}(e^{j\omega}) = e^{-j\omega D} - H_{\mathrm{d}}^{-}(e^{j\omega}) \tag{4.28d}$$

By using the above definitions it is possible to study the effects of the FD approximation errors due to the two-port junction. Setting all $E(e^{j\omega}) = 0$ or, equivalently, all $G(e^{j\omega}) = 1$, results in the ideal frequency response.

4.3.7 Simulation of a Two-Tube System Using FIR FD Filters

As an example we shall investigate the frequency response of a two-tube model implemented with an interpolation waveguide model. The total length of the tube corresponds to eight unit delays. The lengths of the two tube sections are 3.5 and 4.5 samples and the scattering junction is thus located half way between unit delays. This can be considered the 'worst case' for FIR FD filters since their approximation error is largest when $d = 0.5$. The reflection coefficient at the junction of the two tubes is –0.5. One end of the tube is assumed to be closed and the other one open, but in order to simplify the example the terminations have been approximated with constant reflection coefficients $r_1 = 0.9$ and $r_2 = -0.9$.

Figures 4.24 and 4.25 show the magnitude responses of this simulated system with first and third-order Lagrange interpolation, respectively. The solid line in both figures

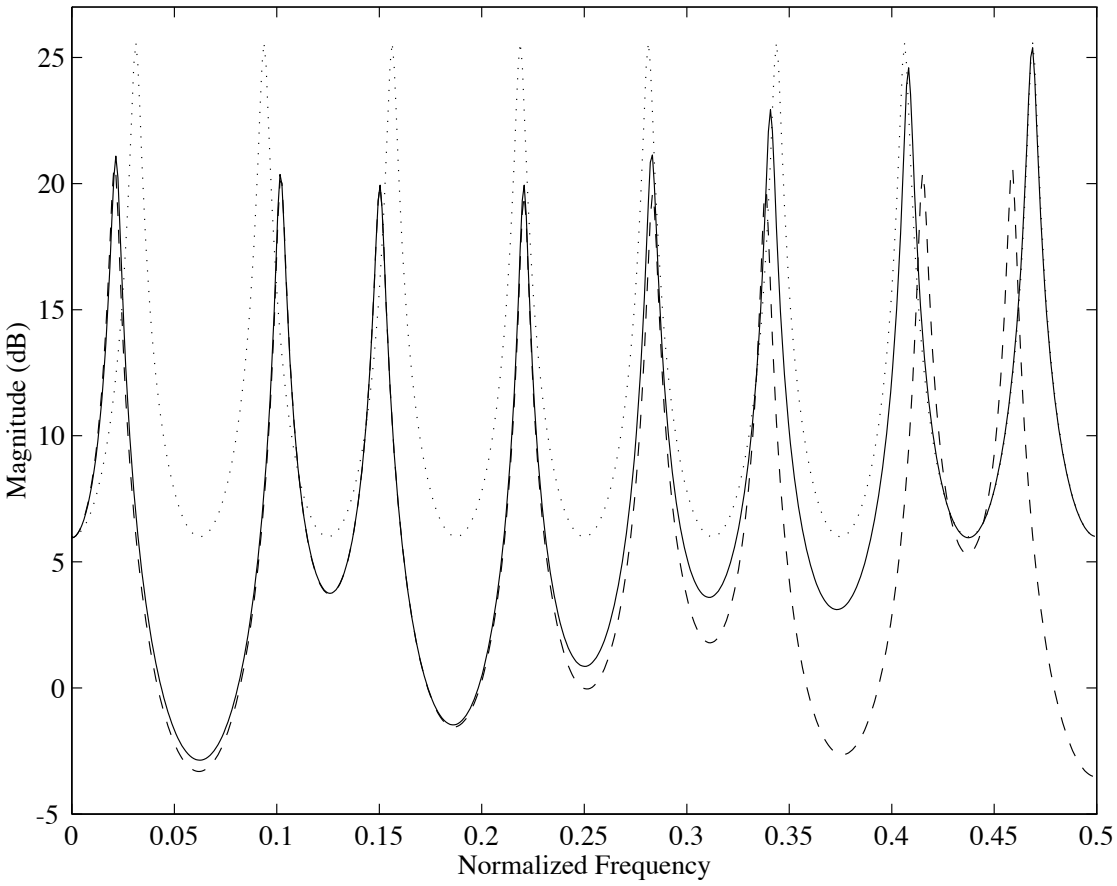

Fig. 4.26 The magnitude responses of the FD two-tube model with *general LS FIR design* ($N = 3$, $\alpha = 0.5$) (solid line) and ideal interpolation (dashed line). The dotted line presents the magnitude response of a uniform tube.

is obtained by substituting the sampled frequency response of the interpolating filter while the dashed line results from setting the error frequency response to zero as proposed above. This corresponds to the case of ideal interpolation, or equivalently, an infinite sampling rate. The dotted line in figures 4.24 and 4.25 shows the magnitude response of a *uniform tube* that is eight unit delays long. (The uniform tube is obtained when the reflection coefficient is zero in the two-tube model.) The formants of the uniform tube are equally spaced along the linear frequency axis.

It is seen in both cases that at low frequencies the magnitude response of the FD model coincides with the ideal one, whereas at high frequencies it joins the magnitude response of the uniform tube. This is because the interpolators are lowpass filters that have a transmission zero at the Nyquist frequency, and thus the reflection coefficient of the FD junction effectively approaches zero (i.e., the uniform case) when the frequency is increased. The third-order interpolator (Fig. 4.25) yields a much better result at middle frequencies than the linear interpolator (Fig. 4.24).

For comparison, the two-tube system was simulated using general LS (GLS) and equiripple FIR FD filters. These filter design methods were discussed in Sections 3.2.4 ad 3.2.5, respectively. The magnitude responses of these two-tube simulations are presented in Figs. 4.26 and 4.27. Also in these figures the solid line gives the result of the FIR filter approximation, the dashed line the ideal result, and the dotted line the magnitude response of a uniform tube. The bandwidth of approximation for the GLS and

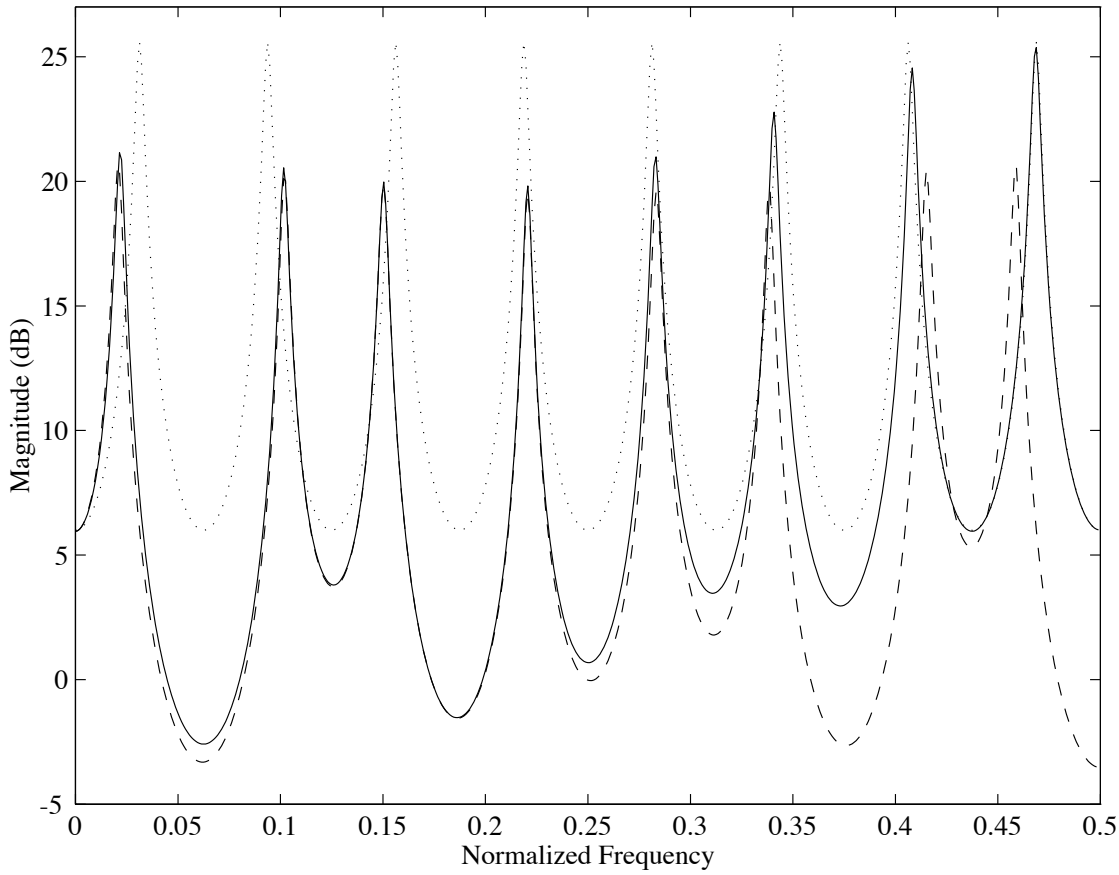

Fig. 4.27 The magnitude responses of the FD two-tube model with *equiripple (Oetken) FIR interpolation* ($N = 3$, $\alpha = 0.5$) (solid line) and ideal interpolation (dashed line). The dotted line presents the magnitude response of a uniform tube.

equiripple filters was $\alpha = 0.5$, i.e., the results are optimal up to the normalized frequency 0.25 in Figs. 4.26 and 4.27. The filter coefficients of the GLS and the equiripple filter were divided by 1.0157 and 1.0224, respectively, in order to force the maximum of the magnitude response to be equal to one.

To enable comparison of the four simulated results, the errors in the amplitude of the formants were computed in each case. The results are collected in Table 4.1. The number in parentheses after the name of the filter design method indicates the order N of the FIR filter approximation. The leftmost column gives the formant number and the sec-

Table 4.1 Formant amplitude errors (dB) in the two-tube simulations (see text).

Formant	f_k/f_s	Lagrange (1)	Lagrange (3)	GLS (3)	Oetken (3)
1	0.021	0.147	0.000551	–0.456	0.410
2	0.10	–1.06	–0.0777	–0.169	–0.340
3	0.15	2.26	0.737	0.00322	0.0798
4	0.22	3.59	0.667	0.241	0.164
5	0.28	3.29	2.86	0.710	0.642
6	0.34	5.45	4.03	4.35	3.94
7	0.42	3.68	2.82	3.39	3.53
8	0.46	5.55	5.30	5.16	5.13

ond column the normalized center frequency of each formant. The error in each case is the difference of the peak magnitude of the approximation and the ideal magnitude response. Note that there are also remarkable errors in the center frequencies of formants at high frequencies in Figs. 4.24–4.27 but they are not presented in Table 4.1.

It is seen that the magnitude error of the Lagrange interpolators increases with frequency. The errors of the GLS and Oetken (equiripple) filter approximations are less than 1 dB for the five lowest formants. At high frequencies the errors are several dB in all cases.

It can be concluded that the use of FIR FD filters for the realization of an interpolated waveguide model leads to a *low-frequency approximation* of a tube system. Implementation of a vocal tract model, or other acoustic tube simulation, thus requires *oversampling* and possibly a high-order interpolator to achieve a small enough approximation error up to a desired frequency. In Välimäki *et al.* (1994a) it was proposed that the use of a sampling rate of 22 kHz and a third-order Lagrange interpolation yields an effective bandwidth of about 5 kHz, which is quite adequate for high-quality speech synthesis.

4.3.8 Conclusion

In this section we tackled simulation of an acoustic tube system, such as the human vocal tract, with fractional delay filters. Combined use of nonrecursive interpolation and deinterpolation enabled us to derive a structure for an interpolated scattering junction which can be located at an arbitrary point along a digital waveguide. The main motivation for this new structure is the elimination of the most severe limitation of the KL vocal tract model—the use of equal-length tube sections. In the proposed new model, the locations of the junctions are not restricted by sampling of the waveguide system.

Realization of the interpolated scattering junction using FIR-type fractional delay filters was considered. The effects of approximation error on the reflection and transmission functions of the scattering junction, as well as on the frequency response of a simulated tube system were discussed. FIR interpolation yields a low-frequency approximation of a waveguide system. The application of Lagrange interpolation, general LS FIR design, and Oetken's almost-equiripple FIR FD design were considered for the implementation of interpolated waveguide tube models. High-quality simulation requires oversampling by a factor of two and at least four-tap FD filters for interpolation and deinterpolation.

In Section 4.6 we consider implementation of acoustic tube models using allpass FD filters. Next we tackle the simulation of an interpolated three-way scattering junction.

4.4 Interpolated Finger-Hole Model

The position of the finger holes is a critical question in the design of woodwind instruments. Also in a waveguide model of a woodwind instrument, the place of each finger hole has to be defined carefully. This is conceivable with the aid of FD filtering techniques introduced earlier in this chapter. This approach to finger-hole modeling was first described in Välimäki *et al.* (1993b).

4.4.1 FIR-Type Fractional-Delay Finger-Hole Model

We return to study the finger-hole junction discussed in Section 2.5.3. The volume velocity signals that describe the three-port junction can be expressed in the (discrete) time domain as

$$u_{\mathrm{a}}^{-}(n,m+D) = u_{\mathrm{b}}^{-}(n,m+D) + u_{\mathrm{s}}^{-}(n) + w(n) \tag{4.29a}$$

$$u_{\mathrm{b}}^{+}(n,m+D) = u_{\mathrm{a}}^{+}(n,m+D) + u_{\mathrm{s}}^{-}(n) + w(n) \tag{4.29b}$$

$$u_{\mathrm{s}}^{+}(n) = -u_{\mathrm{s}}^{-}(n) - 2w(n) \tag{4.29c}$$

where D is the real-valued spatial index corresponding to the position of the center of the finger hole, and signal $w(n)$ is the output of the reflection filter $R(z)$, i.e.,

$$w(n) = r(n) * w_1(n) \tag{4.29d}$$

where $r(n)$ is the impulse response of the reflection filter $R(z)$, $*$ denotes the convolution sum, and the input signal $w_1(n)$ is

$$w_1(n) = u_{\mathrm{a}}^{+}(n,m+D) + u_{\mathrm{b}}^{-}(n,m+D) + u_{\mathrm{s}}^{-}(n) \tag{4.29e}$$

The signal flow graph for a finger-hole model has been depicted in Chapter 2 in Fig. 2.20.

The input signals of the tone-hole model, $u_{\mathrm{a}}^{+}(n,m+D)$ and $u_{\mathrm{b}}^{-}(n,m+D)$, can be computed at an arbitrary point on the digital waveguide using interpolation. When FIR interpolation is employed, the approximations to these signals can be written in vector form as

$$u_{\mathrm{a}}^{+}(n,m+D) = \mathbf{h}^T \mathbf{u}^{+}(n,m) \tag{4.30a}$$

$$u_{\mathrm{b}}^{-}(n,m+D) = \mathbf{h}^T \mathbf{u}^{-}(n,m) \tag{4.30b}$$

where the vectors are

$$\mathbf{h} = \begin{bmatrix} h(0) & h(1) & h(2) & \cdots & h(N) \end{bmatrix}^T \tag{4.30c}$$

$$\mathbf{u}^{+}(n,m) = \begin{bmatrix} u^{+}(n,m) & u^{+}(n,m+1) & u^{+}(n,m+2) & \cdots & u^{+}(n,m+N) \end{bmatrix}^T \tag{4.30d}$$

$$\mathbf{u}^{-}(n,m) = \begin{bmatrix} u^{-}(n,m) & u^{-}(n,m+1) & u^{-}(n,m+2) & \cdots & u^{-}(n,m+N) \end{bmatrix}^T \tag{4.30e}$$

and N is the order of the FIR interpolator. The coefficients $h(n)$ can be, e.g., the Lagrange interpolation coefficients (see Section 3.3).

The results $u_{\mathrm{a}}^{-}(n,m+D)$ and $u_{\mathrm{b}}^{+}(n,m+D)$ of the computation in Eqs. (4.29a) and (4.29b) also have fractional spatial indices $m + D$ and should thus be stored "between the samples" of the delay line. In practice this is not necessary, since the signal $u_{\mathrm{b}}^{-}(n,m+D)$ really is the same as $u_{\mathrm{a}}^{-}(n,m+D)$—both are part of the lower delay line of the waveguide. Also $u_{\mathrm{a}}^{+}(n,m+D)$ is the same as $u_{\mathrm{b}}^{+}(n,m+D)$. The subscripts 'a' and 'b' essentially refer to the order of computation. This situation is exactly the same

as in the case of the FD KL junction (see Section 4.3.2) where the direct signal was not processed at all. Thus the computations of the FD finger-hole model can be expressed in vector form as

$$\mathbf{u}^-(n,m) = \mathbf{u}_n^-(n,m) + \mathbf{h}\Big[u_s^-(n) + w(n)\Big] \tag{4.31a}$$

$$\mathbf{u}^+(n,m) = \mathbf{u}^+(n,m) + \mathbf{h}\Big[u_s^-(n) + w(n)\Big] \tag{4.31b}$$

Note that these forms only involve *deinterpolation.*

Since the last term of the above equations is the same, it is economical to compute it separately and then use it in the two equations. This is expressed as

$$\mathbf{u}^-(n,m) = \mathbf{u}^-(n,m) + \mathbf{v}(n) \tag{4.32a}$$

$$\mathbf{u}^+(n,m) = \mathbf{u}^+(n,m) + \mathbf{v}(n) \tag{4.32b}$$

where we have defined a vector

$$\mathbf{v}(n) = \mathbf{h}\Big[u_s^-(n) + w(n)\Big] \tag{4.32c}$$

Here the signal $w(n)$ is the output of the reflection filter $R(z)$. Its input signal $w_1(n)$ defined by Eq. (4.29e) also involves fractional indices, and thus it has to be computed using interpolation, that is

$$\begin{aligned} w_1(n) &= \mathbf{h}^T\mathbf{u}^+(n,m) + \mathbf{h}^T\mathbf{u}^-(n,m) + u_s^-(n) \\ &= \mathbf{h}^T[\mathbf{u}^+(n,m) + \mathbf{u}^-(n,m)] + u_s^-(n) \end{aligned} \tag{4.33}$$

The last form suggests an efficient way to calculate the input for the reflection filter $R(z)$: first add the values in the two delay lines of the digital waveguide and only after that interpolate†. Note that as the signals $u^+(n)$ and $u^-(n)$ propagate in opposite directions the coefficients of the interpolating filter appear in time-reversed order with respect to one of them. In other words, in the upper delay line the FIR interpolating filter approximates the fractional delay d, while in the lower line it approximates the complementary fractional delay δ.

Note that altogether two FD filterings are involved in the interpolated finger-hole model: interpolation is needed when Eq. (4.33) is evaluated and deinterpolation when Eq. (4.32c) is evaluated. In addition, there is a simple digital filter $R(z)$ is the system, and two additions are needed. It is seen that there is considerably more computation in this model than, e.g., in the FD KL model discussed in Section 4.3.

4.4.2 Approximation Error Due to FD Filters

As in the case of the FD KL junction, the approximation error due to the fractional delay filters is in effect applied *twice* to the FD three-port: first in the interpolation of the delay-line value and then in the deinterpolation of the result back to the waveguide.

† This trick in the interpolated finger-hole model can be compared with the interpolated tube model discussed above in Section 4.3. Figure 4.19 illustrates the idea that both interpolation and deinterpolation only need to be applied once in an FD scattering junction.

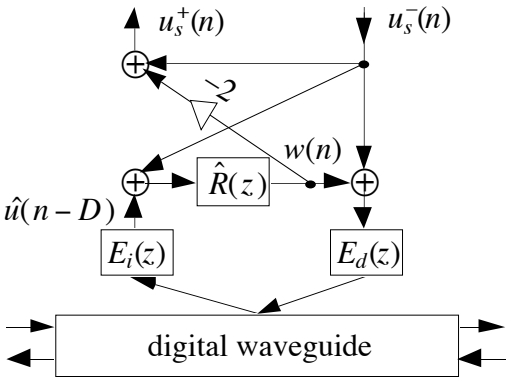

Fig. 4.28 A signal flow diagram of a finger-hole model showing the interpolation and deinterpolation errors.

It is worth pointing out that the interpolated signal values are only used in the computation of the effect of the junction and deinterpolation is only applied to the result of this calculation. Thus, the interpolation error does not affect the signal that would anyway propagate in the waveguide if the junction were not there.

Figure 4.28 shows a high-level block diagram of the finger-hole model. The transfer functions $E_\mathrm{i}(z)$ and $E_\mathrm{d}(z)$ present the errors of the interpolating and deinterpolating filter, respectively. In this diagram the waveguide is considered ideal, i.e., no errors due to digital implementation are present.

The effect of the FD approximation in the finger-hole model derived above can be studied by computing the transfer function of a simple bore model. In this experiment one end of the tube is assumed to be ideally closed (i.e., reflection coefficient is 1) and the other end is assumed to be matched (or equivalently an infinite-length tube is considered). A finger hole is placed at a distance $K = 50.5$ unit samples from the closed end. The transfer function of this system is

$$H(e^{j\omega}) = \frac{-2j\omega E_\mathrm{i}(e^{j\omega})R(e^{j\omega})e^{-j\omega K}}{1 - E_\mathrm{i}(e^{j\omega})R(e^{j\omega})E_\mathrm{d}(e^{j\omega})e^{-j\omega K}} \tag{4.34}$$

where the transfer functions presenting the interpolation and deinterpolation error are

$$E_\mathrm{i}(e^{j\omega}) = e^{j\omega TD}H_\mathrm{i}(e^{j\omega}) \tag{4.35a}$$

$$E_\mathrm{d}(e^{j\omega}) = e^{j\omega TD}H_\mathrm{d}(e^{j\omega}) \tag{4.35b}$$

Figure 4.29 presents the magnitude responses for the ideal analog system, a system with the reflection filter $R(z)$ but with ideal fractional delays, and a realistic digital system with the digital reflection filter and an FD interpolator and deinterpolator implemented using first-order Lagrange interpolation (i.e., linear interpolation). The sampling rate used in this example is 44 kHz. It is seen that in this *worst case* (i.e., the fractional delay is $d = 0.5$) the resonances are only slightly disturbed in the frequency band of interest, i.e., below 10 kHz. Thus it seems to be sufficient to use simple linear interpolators in the implementation of finger-hole models.

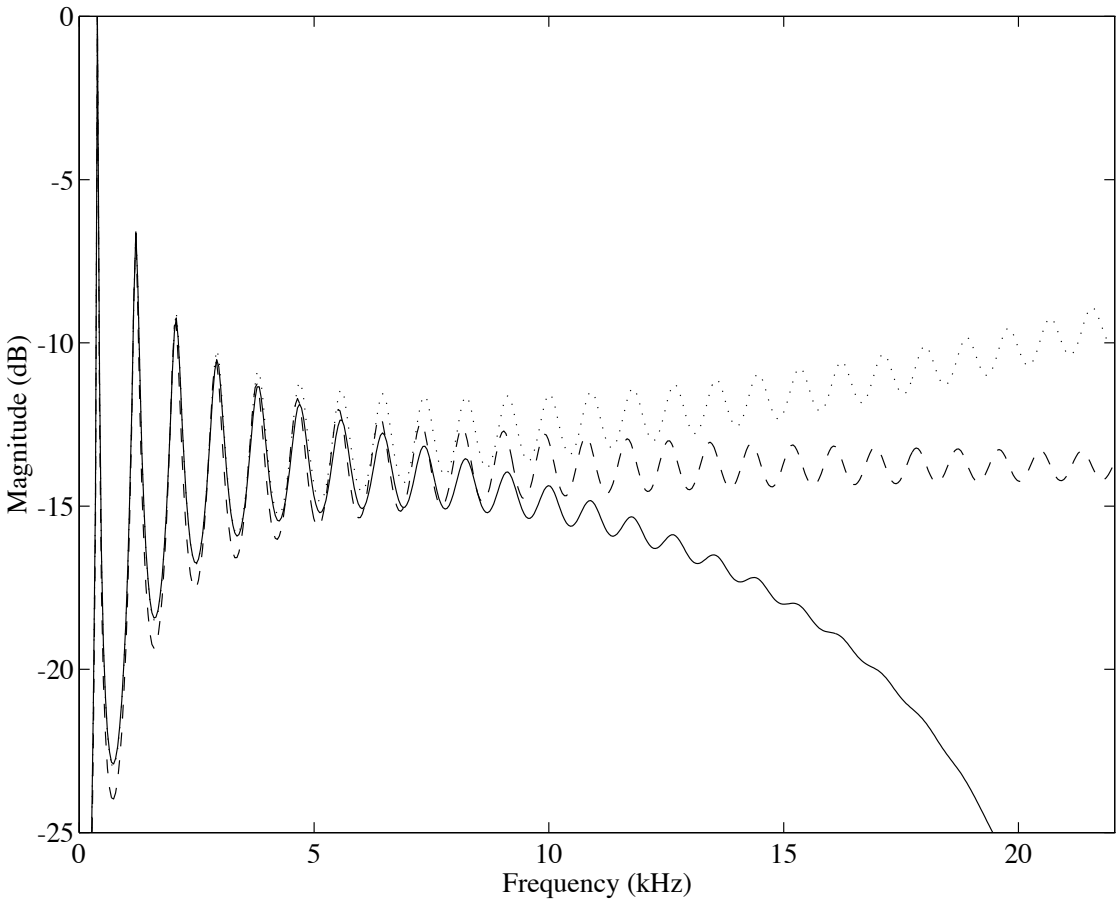

Fig. 4.29 The magnitude responses of the ideal bore model (dashed line), a model with a digital reflection function but ideal interpolators (dotted line), and its realistic implementation with first-order Lagrange interpolation and deinterpolation (solid line).

4.4.3 Implementation Issues

A single model for an open finger hole is computationally relatively efficient. However, woodwind instruments have several, up to approximately twenty finger holes, and if all of them were simulated by the technique introduced above, the computational load would be extremely large. Fortunately, it is not necessary to implement all the open holes since only the two or three first open holes determine the fundamental pitch of the instrument (Benade, 1960).

The closed holes have only a slight influence on the effective length of the bore and they also act as lowpass filters. These effects can be taken into account by far simpler methods than using a fractional delay three-port junction. Thus, it is suggested that a woodwind bore model with finger holes be implemented based on *two or three FD junctions* that model the first open holes. When a hole is closed the junction is removed and it is "grown" at another location in the waveguide. Using this strategy, the typical playing techniques of woodwind instruments, including the key claps, can be simulated. More tone holes may be needed for producing multiphonics.