

MUMT 618: Week #12

1 Multi-Dimensional Modeling

Traditional finite-difference approaches for the simulation of wave motion in multi-dimensional systems have been used for many decades. Digital waveguide techniques for multi-dimensional acoustic modeling were first published by Van Duyne and Smith [1993] and have been explored in various contexts over the past 20 years. In fact, the digital waveguide approach represents an alternative formulation of a finite difference scheme for multi-dimensional modeling. In general, these techniques involve significant computational demands and suffer from dispersion errors.

1.1 A Rectilinear Two-Dimensional Plate Model

- A two-dimensional grid of intersecting vertical and horizontal digital waveguides is shown in Fig. 1. The junctions, labeled **S**, are 4-port scattering junctions. The double-arrow lines represent bi-directional unit delay elements.

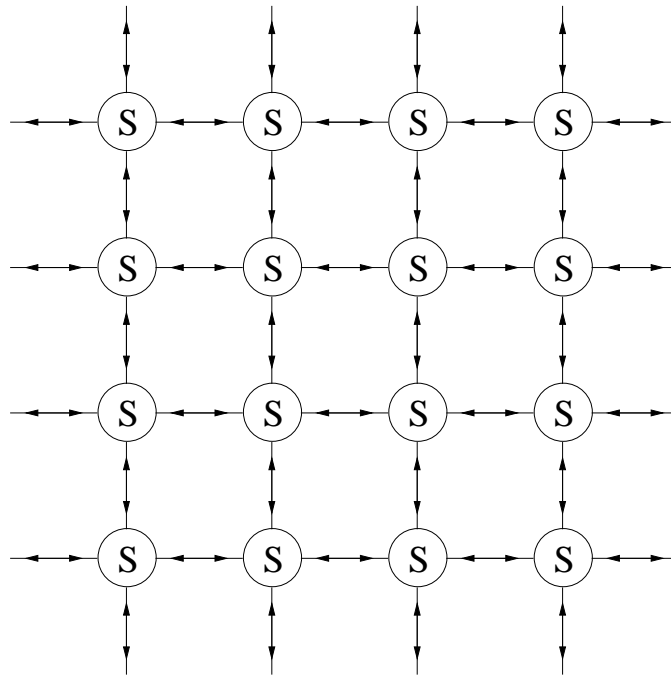


Figure 1: A two-dimensional waveguide grid pattern.

- We can view the grid of Fig. 1 as a lattice of interconnected vibrating strings, in which case the displacements or velocities at the four ports of each junction must be equal and the forces at each junction must sum to zero.

- Alternately, we can view the grid of Fig. 1 as a lattice of interconnected acoustic elements, in which case the pressures at the four ports of each junction must be equal and the flows at each junction must sum to zero.

1.2 Lossless Scattering

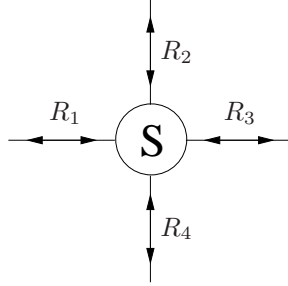


Figure 2: A four-port scattering junction.

- A four-port scattering junction is shown in Fig. 2.
- When viewed as a mechanical system, each junction is bounded by the following conditions:
 1. The port velocities at a given junction must be equal because the medium is continuous at that point: $v_1 = v_2 = \dots = v_N$
 2. The forces exerted by all the connections must balance or sum to zero at the junction: $f_1 + f_2 + \dots + f_N = 0$
- The traveling-wave components at the junction, $v_i = v_i^+ + v_i^-$ and $f_i = f_i^+ + f_i^-$, are related by their respective port wave impedances: $f_i^+ = R_i v_i^+$ and $f_i^- = -R_i v_i^-$.
- When these relationships are combined, the following lossless scattering equations for a generalized N -port junction can be derived:

$$v_J = \frac{2 \sum_i R_i v_i^+}{\sum_i R_i} \quad \text{and} \quad v_i^- = v_J - v_i^+$$

- For homogeneous, isotropic media, the various wave impedances are equal, allowing the junction velocity calculation above to be simplified to:

$$v_J = \frac{2}{N} \sum_i v_i^+$$

- For the case of a four-port ($N = 4$) junction, the scale factor $2/N$ reduces to $1/2$, which can be implemented in fixed-point arithmetic with a bit shift.
- The junctions are separated from each other by unit delays, so an input wave variable of one port of one junction is equal to the output at the adjacent junction port from the previous time step:

$$v_{x,y}^+(n) = v_{x-1,y}^-(n-1)$$

- The various junction equations can be manipulated to show that the 2D digital waveguide mesh structure implements the standard second-order finite difference scheme applied to the partial differential wave equation of the ideal membrane (see this link for details),

$$u_{tt}(t, x, y) = c^2 [u_{xx}(t, x, y) + u_{yy}(t, x, y)],$$

with wave propagation speed $c = \sqrt{2}/2$ and the time and spatial sampling intervals ($X = \Delta x$, $Y = \Delta y$, $T = \Delta t$) all equal to one another.

- The 2D rectilinear mesh implements a wave propagation speed equivalent to one-half unit diagonal distance per time sample. Intuitively, this can be understood as occurring because of a doubling of waveguide mass density.

1.3 Dispersion

- In an ideal membrane, the speed of wave travel is independent of direction and spatial frequency (*i.e.*, there is no wave *dispersion*). In a time interval T , all wave components will advance in position by cT .
- The propagation distances between junctions on a 2D rectilinear grid, however, are not spatially uniform in all directions. As a result, wave propagation on the grid varies with direction as well as spatial frequency.

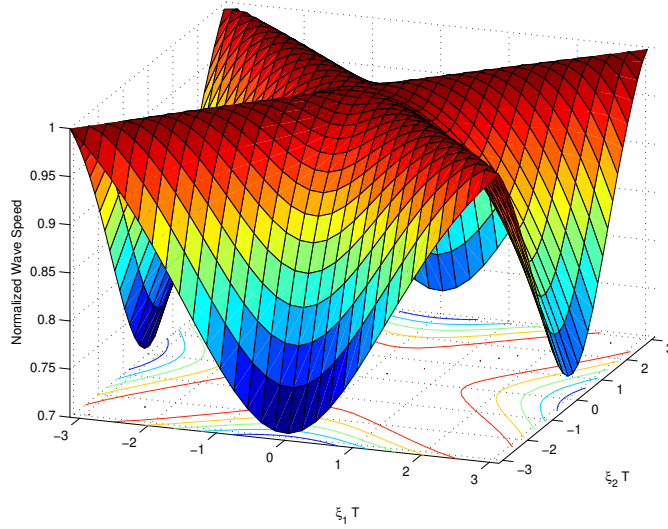


Figure 3: Mesh wave speed vs. spatial frequency.

- Figure 3 is a plot of normalized wave speed versus spatial frequency and direction. The center region corresponds to low spatial frequencies and outer regions correspond to higher spatial frequencies. Angular direction, as seen from the frequency-plane origin, corresponds exactly to the direction of travel on the mesh.
- No dispersion occurs for those frequencies and directions where the normalized wave speed equals 1 in Fig. 3. Thus, wave motion at low frequencies and along diagonal directions to the mesh coordinate system travel without dispersion error.
- Waves traveling along the coordinate axes of the mesh, however, have wave speeds that fall off in proportion to frequency.
- In a bounded mesh, dispersion errors result in mistunings of resonant modes.

1.4 Simulations

- The Matlab script `mesh2d.m` implements a 2D rectilinear digital waveguide mesh structure, allowing visualization of wave propagation over time in the structure. This same algorithm is also implemented for realtime synthesis in the `Mesh2D` class of the Synthesis ToolKit in C++ (STK).

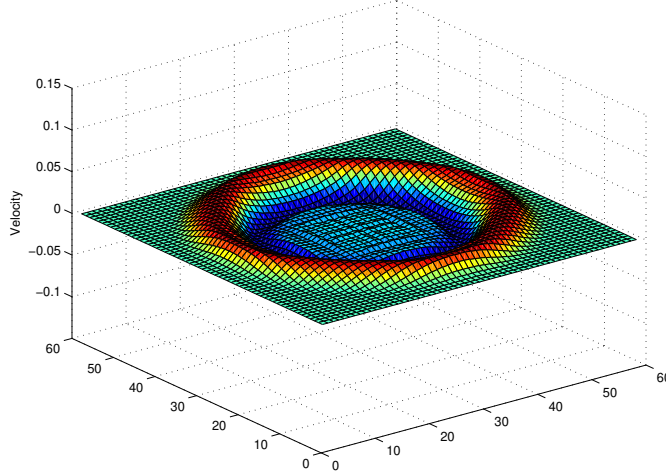


Figure 4: Wave visualization on a 2D waveguide mesh.

- The rectilinear 2D (and 3D) grid is convenient because it can be implemented with relatively simple data structures (typically a double, alternating matrix scheme) and it is relatively easy to calculate the grid geometry for a given 2D or 3D space. Further, the 2D mesh can be implemented without multiplies in fixed-point arithmetic.
- Though not apparent in the simulation, the response of the rectilinear waveguide mesh aliases at a normalized temporal frequency of 0.25 (where $0.5 = 1/2$ the digital sample rate).
- The 2D rectilinear grid can be easily extended to three dimensions. In this case, however, each grid junction has 6 ports and the resulting scattering equations are not multiply-free for fixed-point implementations.

1.5 Alternative Mesh Geometries

- 2D and 3D simulations need not be limited to rectilinear grid patterns. A variety of alternative topologies have been proposed and explored for improved dispersion characteristics, as well as better efficiency.
- A two-dimensional, three-port hexagonal grid pattern is shown in Fig. 5.
- A two-dimensional, six-port triangular grid pattern is shown in Fig. 6.
- The dispersion error for the triangular waveguide mesh is shown in Fig. 7.
- When the rectilinear, hexagonal, and triangular meshes are compared with respect to a given space aliasing error, the triangular waveguide mesh uses the least number of junctions per unit area (the hexagonal mesh requires the most junctions per unit area) [Fontana and Rocchesso, 1998].
- When the rectilinear, hexagonal, and triangular meshes are compared with respect to a minimum necessary sample rate, the triangular waveguide mesh again does the best job (and again, the hexagonal mesh requires the highest sample rate) [Fontana and Rocchesso, 1998].
- The rectilinear mesh requires the least number of operations and memory per unit time and unit area [Fontana and Rocchesso, 1998].
- Van Duyne and Smith [1996] proposed a 4-port tetrahedral grid pattern for modeling 3D objects. The resulting dispersion error pattern varies with dimension. Memory and sample rate comparisons (compared to a 3D rectilinear mesh) are provided in the cited paper.

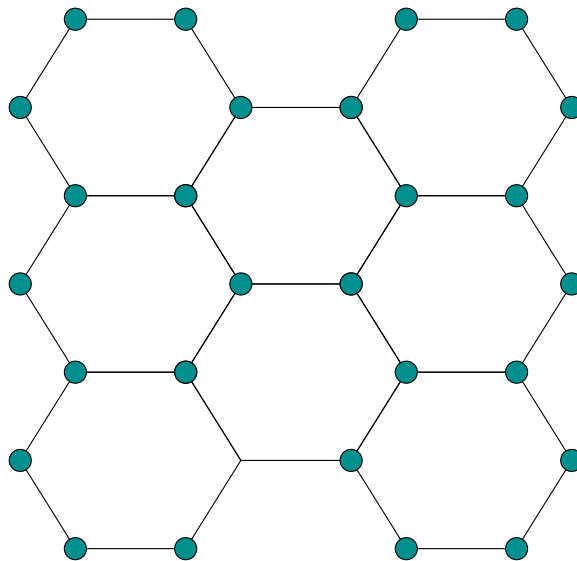


Figure 5: A two-dimensional 3-port hexagonal mesh pattern.

1.6 Methods for Reducing Dispersion Error

- Savioja and Välimäki [2000] propose both deinterpolation and frequency warping techniques to reduce dispersion error.
- Deinterpolation or point-spreading is used to implement equal unit-delay lengths for propagation in all directions. In this way, the dispersion error is made relatively uniform in all directions.
- Once the dispersion is made relatively uniform in all direction, Savioja and Välimäki [2000] use an FIR warping filter with unit delays replaced by first-order allpass sections to either pre- or post-process the mesh signal and further correct dispersion errors. This technique is only applicable in non-realtime situations.

1.7 Further Online Links

- S. Van Duyne and J.O. Smith (1993). Physical modeling with the 2-D digital waveguide mesh, In *Proc. 1993 Int. Computer Music Conf.*, pp. 40-47, Tokyo, Japan.
- Bilbao, S. (2001). Wave and Scattering Methods for the Numerical Integration of Partial Differential Equations, Ph.D. Thesis, Department of Electrical Engineering, Stanford University.
- Savioja, L. (1999). Modeling Techniques for Virtual Acoustics, Doctoral Thesis, Helsinki University of Technology.

2 Physically Informed Sonic Modeling (PhISM)

The multidimensional modeling techniques presented in the previous section would seem to be an obvious choice for synthesizing a variety of percussion instrument sounds. For example, a circular membrane can be modeled with a two-dimensional waveguide system and then coupled to a drum cavity model. However, there are several drawbacks to this approach: 1. It requires significant computational power; and 2. The resulting sounds are often not significantly better than ones produced using efficient modal synthesis techniques. Cook [1997] proposed a series of simplified approaches to the modeling of percussion sounds that he generally refers to as “physically informed sonic modeling (PhISM)” Drawing on techniques from physical modeling, Fourier analysis/synthesis, and granular synthesis, a sonically convincing and computationally efficient synthesis

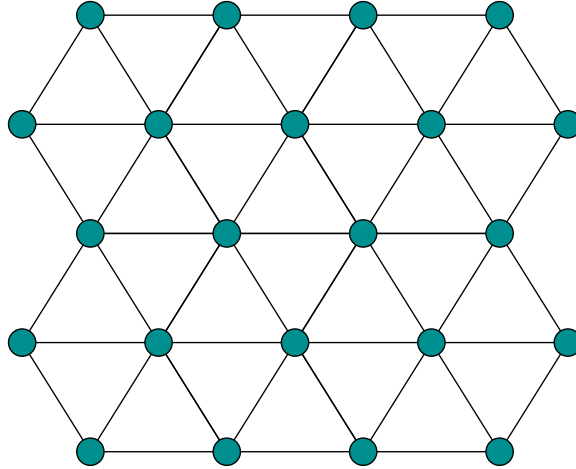


Figure 6: A two-dimensional 6-port triangular mesh pattern.

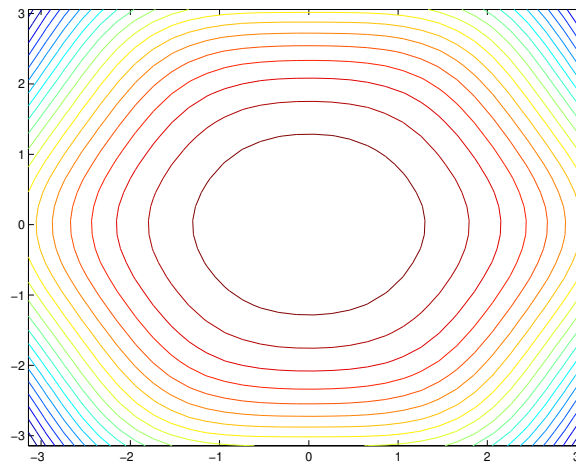


Figure 7: A contour plot of wave dispersion in the 6-port triangular mesh pattern.

method is developed that is founded on an understanding of the physical behavior of systems without actually attempting to accurately simulate their vibrational patterns and interactions.

Cook [1997] offers two PhISM approaches: Physically Informed Control of Modal Synthesis (PhISAM) and Physically Informed Stochastic Event Modeling (PhISEM). PhISAM provides augmented parametric control to modal synthesis models. The PhISEM algorithm is based on pseudorandom overlapping and adding of small grains of sound or pseudorandom modification of the parameters of a parametric synthesis model, according to rules and parameters derived from off-line physical simulations and heuristics. The PhISEM approach is suitable for synthesizing sounds characterized by random interactions of sound-producing component objects, such as a maraca, sleigh bell, bamboo wind chimes, and water drops.

2.1 Physically Informed Control of Modal Synthesis (PhISAM)

- The “physically informed spectral additive modeling” (PhISAM) technique is based on modal synthesis with the addition of “physical” parametric controls such as “stick hardness” and “strike position”.
- The approach is suitable for resonant percussion instruments characterized by impulsive excitation of a relatively few exponentially decaying, weakly coupled sinusoidal modes (*i.e.*, marimba, vibraphone,

cowbell, ...).

- Frequency tracking, or mode “picking”, can be accomplished using Fourier analysis, linear predictive coding, or other all-pole filtering techniques.
- Prototype excitations can be determined by recording a stick strike while damping the resonant modes of an instrument or by extracting a residual signal from an LPC or sinusoidal analysis.
- Physical parametric controls, such as strike position and stick hardness, can be simulated using simple rules to control modal gains.
- The analysis/resynthesis approach can be outlined as:
 1. Take a recorded sound and perform high-order LPC, ARMA, or SMS sinusoidal + noise analysis:
 - Determine the N highest-Q resonances or highest-amplitude FFT peaks;
 - Extract LPC residual or SMS noise envelope for resynthesis excitation or record a dry strike.
 2. Derive resynthesis parametric control rules:
 - For strike position, analyze modal behavior for various strike position excitations;
 - For stick hardness and strike vigor, first choose resynthesis method (resonant filters or sinusoidal functions) and then derive rules.
 3. Perform resynthesis with derived rules and include dynamics of player/performer.
- A PhISAM system block diagram is shown in Fig. 8 below.

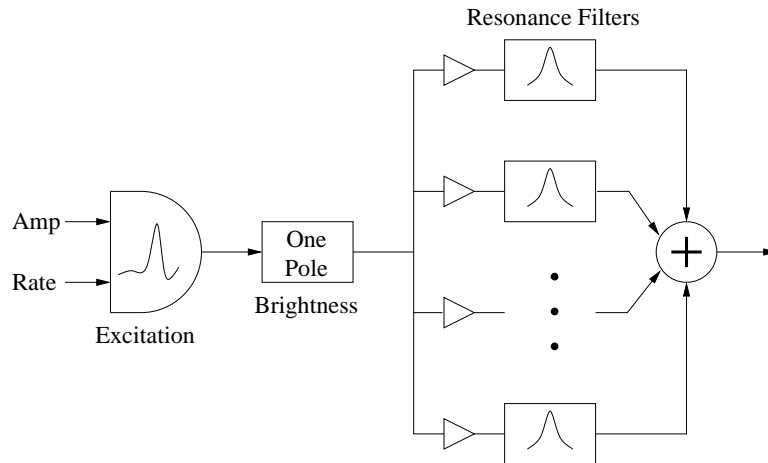


Figure 8: A PhISAM system block diagram.

- The “stick hardness” can be simulated by varying the playback rate of the strike signal. That is, a softer strike can be simulated using a recording hard strike by reading through the sound at a slower rate. This effectively lowers the frequency content of the original sound.

2.2 Particle Modeling

- A variety of sounds are produced by the interaction of one or more moving particles enclosed within a possibly resonant object.
- Using basic physical laws governing the motion and interaction of point masses, it is possible to create a numerical simulation of such a system.
- In general, one must keep track of the position and velocity for each particle in the system.

- For example, a whistle produces sound via the interaction of an enclosed “pea” with a recorder-like air jet mechanism.
 - In the absence of the “pea”, the whistle would produce a relatively stable pitch that is slightly influenced by air jet speed.
 - When the circulating “pea” interacts with the air jet system, there is a decrease in pitch (about 7%), an increase in amplitude (about 6 dB), and a small increase in the noise component (about 2 dB)[Cook, 2002].
 - The motion and position of the “pea” can be reasonably modeled with a two-dimensional circular shell system (the STK Sphere class).
 - An efficient synthesis approach for the whistle is to model the result of the jet mechanism with a simple sinusoidal oscillator and to use the output of the spatial “pea” model to control both amplitude and frequency modulation of the oscillator. This approach is implemented in the STK Whistle class.
- For multi-particle, “shaker” percussive instruments, only collisions with the enclosing structure are considered to generate an audible event (inter-particle collisions need to be accounted for in the equations of motion but are not considered to be sonically relevant).
- When particles interact or collide with the enclosing structure, boundary conditions based on momentum and center of mass can be applied to determine subsequent motion.
- While this approach provides an accurate simulation of such a system, it tends to be computationally intensive.

2.3 PhISEM Particle Models

- The PhISEM approach uses the results of “off-line” particle interaction simulations to derive statistical collision distributions.
- From a numerical simulation of a maraca, statistics were collected with respect to frequency, waiting time, and sound intensity of bean collisions with the outer gourd wall.
- In general, a short-term energy input produced an exponential decay in intensity of collisions.
- The likelihood of a significant sound-producing collision was found to be roughly constant, until all beans were nearly at rest.
- A process with constant event probability at any given time has an exponential distribution in time (*i.e.*, the time duration between events decays away exponentially over time).
- The collected statistics could then be used to control a model in which events trigger distinct overlapping, decaying signals.
- However, a more efficient approach is found by noting the noise-like nature of the resulting sound from a maraca. This suggests the use of decaying noise to represent collision events and a single second-order digital filter to model the gourd resonance.
- Because the sum of exponentially decaying random noise is equal to a single noise source multiplied by a decaying value, only a single exponential decay and a single noise source are required to compute the total sound.
- The maraca synthesis algorithm requires only two random-number calculations (for collision-event likelihood and noise sound source), two exponential decay multiplications (for net system energy and sound level), and one biquad filter, as shown in Fig. 9.
- The Matlab script `maraca.m` implements the Maraca PhISEM model of Cook [2002].
- More complex resonant structures can be incorporated with additional second-order filters.

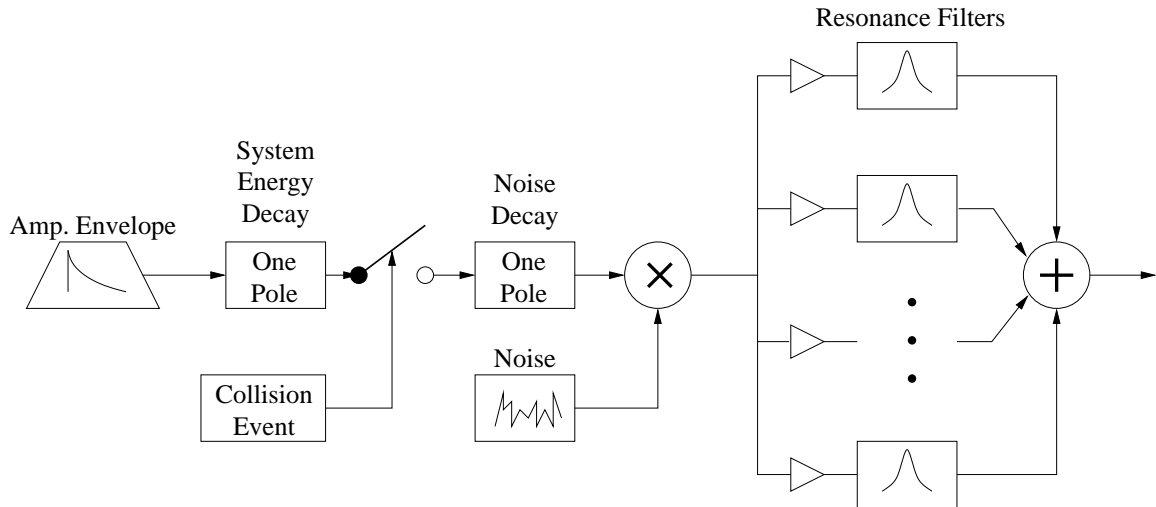


Figure 9: A PhISEM system block diagram.

2.4 Stochastic Resonances in PhISEM

- The PhISEM approach can also be applied to systems, such as the tambourine and sleighbells, that produce a somewhat random distribution of resonances around a set of center frequencies.
- The same event and system energy decay behaviors apply, though the implemented resonance frequencies are randomly varied with each collision event.
- The tambourine model uses two resonances for the cymbals and one weak resonance for the wooden frame.
- The sleighbell model uses five resonant filters.
- Other objects that can be simulated in this way include bamboo wind chimes and coin interactions with a mug.
- Water drop sounds can also be synthesized with this model by modifying the resonant frequency during each drop event (upward sweeping trajectories).

References

- P. R. Cook. Physically informed sonic modeling (phism): Synthesis of percussive sounds. *Computer Music Journal*, 21(3):38–49, 1997.
- P. R. Cook. *Real Sound Synthesis for Interactive Applications*. A.K. Peters, Ltd., Natick, MA, 2002.
- F. Fontana and D. Rocchesso. Signal-theoretic characterization of waveguide mesh geometries for membrane simulation. In *Proceedings of the 1998 International Computer Music Conference*, Ann Arbor, MI, 1998. Computer Music Association.
- L. Savioja and V. Välimäki. Reducing the dispersion error in the digital waveguide mesh using interpolation and frequency-warping techniques. *IEEE Transactions on Speech and Audio Processing*, 8(2):184–194, 2000.
- S. Van Duyne and J. O. Smith. Physical modeling with the 2-D digital waveguide mesh. In *Proceedings of the 1993 International Computer Music Conference*, pages 40–47, Tokyo, Japan, 1993. Computer Music Association.

S. Van Duyne and J. O. Smith. The 3D tetrahedral digital waveguide mesh with musical applications. In *Proceedings of the 1996 International Computer Music Conference*, pages 411–418, Hong Kong, 1996. Computer Music Association.