Frequency Dependent Locally Reacting Surfaces

Introduction

Modelling boundary surfaces in the digital waveguide mesh (DWM) is a problem that is yet to be solved in a way that is flexible, accurate, and realistic. The 'ideal' boundary model for room acoustics applications will reflect incident sound waves with some frequency-dependent (and tunable) absorption of wave energy. Methods from the literature each have unique drawbacks, meaning none are particularly satisfactory for applications where realism is required. This essay will review some methods for boundary modelling, explain which method was chosen to implement, and present an analysis of the implementation.

KW-Pipe Technique

This method is described by Murphy & Beeson (2007) and Kelloniemi (2006).

There are two main technically-equivalent formulations of digital waveguides meshes, known as *W-models* and *K-models*. W-models are based on travelling wave variables, which allow for straightforward interaction with a variety of termination types, such as rimguides, fractional delays, or wave digital filters. Wave digital filters, in particular, could be used to model frequency-dependent boundaries and air absorption. However, W-models have great memory requirements, making them impractical for large multi-dimensional simulations. K-models are based on Kirchhoff variables, and depend on physical quantities rather than travelling wave components. Under certain conditions, K-models and finite-difference time-domain (FDTD) simulations are equivalent. FDTD models have much smaller memory requirements than W-models, at the cost of decreased flexibility of filtering, as these models can't directly interact with wave digital filters.

The KW-pipe is a 'converter' between wave- and Kirchhoff- variables, which is designed to allow the majority of a model (that is, the air-filled space inside it) to be constructed as a K-model waveguide mesh. At the boundaries of the model, the KW-pipe is used to connect K-model nodes to W-model nodes. These W-model nodes can then be connected to wave digital filters to simulate frequency-dependent absorption of wave energy. The complete model retains both the memory-efficiency of the K-model and the termination flexibility of the W-model, with the drawback of additional implementation complexity at the interface between the two model types.

This sounds extremely promising, but has a major drawback, as described by (**???**): while the inside of the mesh will be 2- or 3-dimensional, the boundary termination afforded by the wave-variable boundary is 1-dimensional - each boundary node connects to just the closest interior node. As a result, the edges and corners are not considered to be part of the model, as these nodes do not have a directly adjacent interior node. Additionally, the 1D boundary termination equation implies a smaller internodal distance than that of the 2D or 3D mesh interior. This means that when updating an interior node next to a boundary, the internodal distance is greater than when updating the boundary node itself. For these reasons, the 1D termination is unphysical and can lead to large errors in the phase and amplitude of reflections.

Locally Reacting Surfaces Technique

This method, described by (**???**), aims to create physically correct higher-dimensional boundaries by combining a boundary condition, defined by a boundary impedance, with the multidimensional wave equation. This leads to a model for a 'locally reacting surface' (LRS), in which boundary impedance is represented by an infinite-impulse-response (IIR) filter.

A surface is 'locally reacting' if the normal component of the particle velocity on the boundary surface is dependent solely upon the sound pressure in front of the boundary. In most physical surfaces, the velocity at the surface boundary will also be influenced by the velocity of adjacent elements on the boundary, so LRS is not a realistic physical model in the vast majority of cases.

However, despite that it is not a realistic physical model, the implementation of the LRS modelling technique is both stable and accurate, as opposed to the 1D KW-pipe termination, which does not accurately model even locally-reacting surfaces.

The LRS model leads to an implementation that is efficient (as it is based completely on the K-model/FDTD formulation) and tunable (boundaries are defined by arbitrary IIR filters).

Choice of Technique

The LRS technique was chosen, as it represented the best compromise between memory efficiency, customization and tuning, and realism. The particular strengths of this model are its performance and tunability, though as mentioned previously it is not physically accurate in many cases. That being said, none of the boundary models considered are particularly realistic, so even for applications where realism is the most important consideration, the LRS model seems to be the most appropriate.

Implementation and Testing

Implementation

Filter Design

The first step as to design appropriate reflectance and impedance filters. The design process suggested by (**???**) is to design a reflectance filter based on octave-band amplitudes, and then to transform this into and impedance filter. However, there are two caveats to this process.

Firstly, each boundary node will have at least one unique filter, so the order of the reflectance filter must be low to keep processing times short. This means that the most accurate filter-design method of producing a high-order finite-impulse-response (FIR) filter will be impractical. Additionally, the reflectance filter must be a *single* filter - that is, it cannot be made up of multiple filters in parallel, so the order must be kept low to ensure numerical stability.

Secondly, the transformation process (which transforms the reflectance filter into a boundary-impedance filter) places restrictions upon which reflectance filters are valid. The transformation process is as follows:

$$\xi_W(z) = \frac{1 + R_0(z)}{1 - R_0(z)} \tag{1}$$

where $\xi_W(z)$ is the boundary impedance filter, and $R_0(z)$ is the reflectance filter. For a filter with unity gain (i.e. $R_0(z) = 1$) the transformed filter has a denominator of 0, and therefore infinite gain. This method is therefore incapable of modelling perfectly reflective surfaces, and for very reflective surfaces, some smaller-than-unity value must be chosen for the surface gain.

The final filter design process was put together with these caveats in mind, and with the additional goal of interoperating with pre-existing code for manipulating 3D models with surfaces. The raytracer that preceded the waveguide used surfaces specified as amplitudes in eight octave-bands. It was decided that the waveguide would use the same format, but just use a certain number of the lowest bands - for the tests here, the three lowest bands were used.

For each band, a biquad notch filter was designed with centre frequency and amplitude dependent on the centre frequency and specified amplitude of that octave band. The numerator and denominator coefficients of these filters were combined using polynomial multiplication, to produce coefficients for a single 6th-order filter with the same response as the three notch filters placed in series. This was the reflectance filter. These coefficients were then transformed using equation 1 to give coefficients for a boundary impedance filter.

Boundary Formulation

The heart of the model is a set of three equations, which are found by combining the discrete 3D wave equation with the discrete boundary condition for locally-reacting surfaces.

The first equation describes the update procedure for a one-dimensional boundary, which in this case is a boundary which is adjacent to a single inner-node, a single outer node, and is surrounded by four other boundary nodes. Such boundaries are found within the outer surfaces of a 3D model.

The second equation defines how two-dimensional boundaries should be updated. 2D boundaries are found at edges, where two one-dimensional boundaries with different orientations meet.

The third equation is for 3D boundaries, where three 2D boundaries meet at a corner.

These boundary types are shown in figure 1.

The simulation itself proceeds in the same way as a 'normal' FDTD simulation, but simply uses a different update equation depending on whether the node is an inner or a kind of boundary node.

Testing Procedure

Simulation Parameters

- cubic mesh of $250 \times 250 \times 250$ nodes was used
- sampling frequency of the simulation was set to 8KHz
  - giving an upper stability bound of 2KHz
- simulation was set to run for 350 steps
- source and receiver nodes were place 54 node-spacings away from the centre of the boundary being tested (see figure 2).

Method

A simulation was run using the parameters above, and the reflected signal at the receiver was recorded. This first recording, r_f, contained a direct and a reflected response. Then, the room was doubled in size along the plane of the wall being tested, essentially removing this boundary from the model. The simulation was run again, recording just the direct response at the receiver (r_d). Finally, the receiver position was moved to its reflected position 'through' the tested wall, and the simulation was run once more, producing a free-field response (r_i).

The reflected response was isolated by subtracting r_d from r_f (cancelling the direct response). This isolated reflection is r_r. To find the effect of the frequency-dependent boundary, the frequency content of the reflected response could be compared to the free-field response r_i. This was achieved by windowing r_r and r_i with the right half of a Hanning window, then taking FFTs of each. The experimentally determined numerical reflectance was determined by dividing the absolute values of the two FFTs.

To find the accuracy of the boundary model, the numerical reflectance was compared to the theoretical reflection of the digital impedance filter being tested, which is defined as:

$$R_{\theta,\phi}(z) = \frac{\xi_W(z)cos\theta cos\phi - 1}{\xi_W(z)cos\theta cos\phi + 1} \tag{2}$$

where θ and ϕ are the reflection azimuth and elevation respectively.

This test was run several times, for various filter configurations and azimuth/elevation combinations.

The results are shown below:

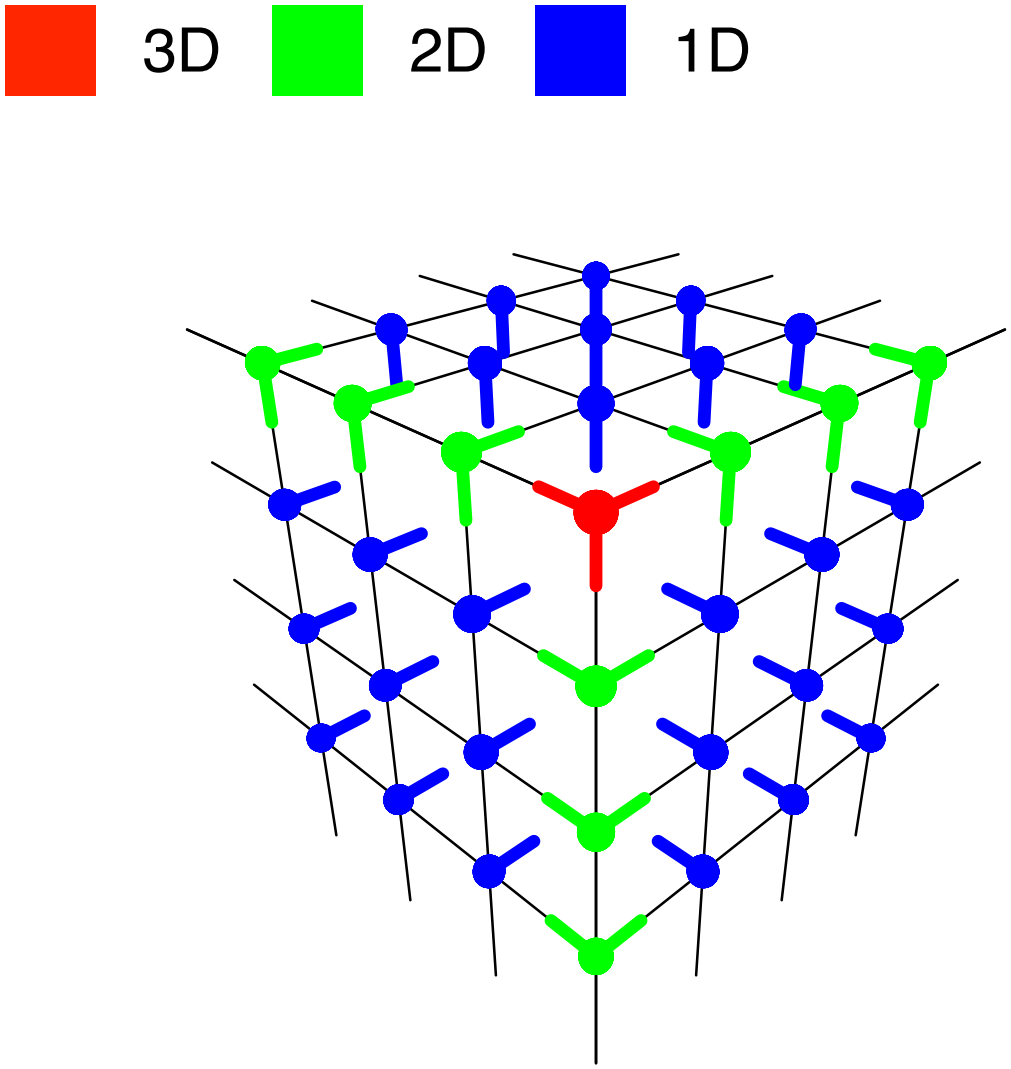

Figure 1: The three different types of boundary in the 3D rectilinear mesh. The lines from each node show the directions of connected nodes.

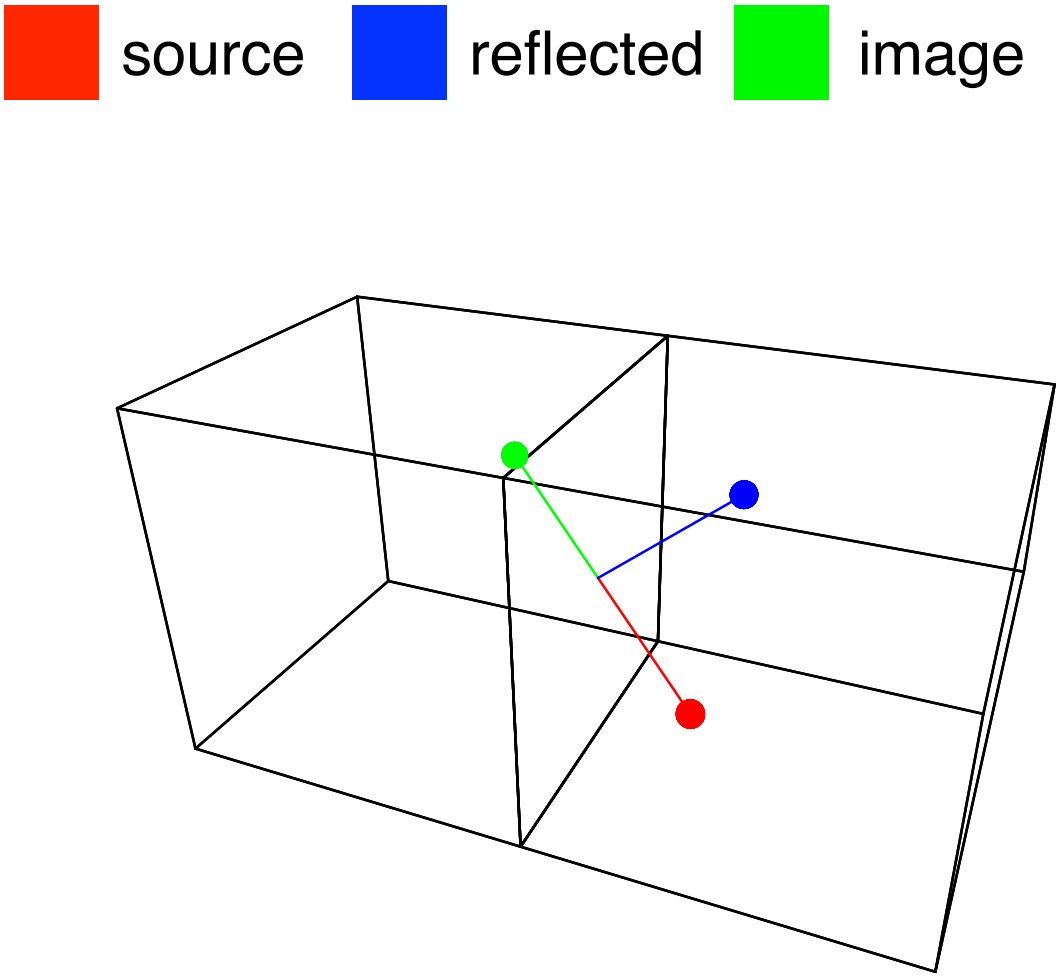

Figure 2: Model showing the testing setup.

Bibliography

Kelloniemi, A. (2006, September). Frequency-dependent boundary condition for the 3D digital waveguide mesh. Proceedings of the 9th International Conference on Digital Audio Effects.

Murphy, D. T., & Beeson, M. (2007). The kw-boundary hybrid digital waveguide mesh for room acoustics applications. *IEEE Transactions on Audio, Speech, and Language Processing*, *15*(2).