A 3D MULTI-PLATE ENVIRONMENT FOR SOUND SYNTHESIS

Alberto Torin
Acoustics and Audio Group
University of Edinburgh
Edinburgh, UK
A.Torin@sms.ed.ac.uk

Stefan Bilbao
Acoustics and Audio Group
University of Edinburgh
Edinburgh, UK
sbilbao@staffmail.ed.ac.uk

ABSTRACT

In this paper, a physics-based sound synthesis environment is presented which is composed of several plates, under nonlinear conditions, coupled with the surrounding acoustic field. Equations governing the behaviour of the system are implemented numerically using finite difference time domain methods. The number of plates, their position relative to a 3D computational enclosure and their physical properties can all be specified by the user; simple control parameters allow the musician/composer to play the virtual instrument. Spatialised sound outputs may be sampled from the simulated acoustic field using several channels simultaneously.

Implementation details and control strategies for this instrument will be discussed; simulations results and sound examples will be presented.

1. INTRODUCTION

Non-linear vibrations of thin plates have been the object of intense study [1]. Typical phenomena, like crashes and pitch glide effects, cannot be captured by a linear model, and these constitute, probably, some of the most interesting features of these objects from a perceptual perspective. The use of these elements in a sound synthesis environment becomes, therefore, a very attractive possibility. Amongst the first attempts at simulating gongs sounds we can cite the work of Van Duyne et al. (see, e.g., [2] for a complete summary.) More recently, sound synthesis of non-linear plates using a modal approach has been performed by Ducceschi et al. with convincing results [3].

When it comes to sound synthesis of percussion instruments, modularity seems to be the key word, from the well-known CORDIS-ANIMA [4] and Mosaic/Modalys systems [5, 6], to more recent works by Bilbao [7] and Avanzini et al. [8]. Several basic elements can be combined at the user’s discretion in order to create a complex system that can serve as a virtual instrument. Following this approach, in this paper we present a sound synthesis environment composed of several non-linear thin plates. The main novelty of this work is the introduction of an explicit coupling between the plates and the surrounding air.

The underlying physical model of this system will be described in Section 2 while a numerical implementation based on finite difference time domain methods will be discussed in Section 3. In Section 4, a brief outline of instrument design and control issues will be presented. Finally, simulation results and sound examples can be found in Section 5.

2. DESCRIPTION OF THE MODEL

The system under analysis is composed of several plates housed in a 3D enclosure \( V \) of air, with which they are coupled (see Figure 1). Simulations of instruments embedded in 3D have already been performed in the past [9, 10], and this work adopts the same approach.

2.1. Plates

The main components of this model are \( N \) thin plates defined over rectangular regions \( P_i \), with \( i = 1, \ldots, N \), all parallel to the \( xy \) plane, and with centres at coordinates \( x^{(i)} = (x^{(i)}_x, y^{(i)}_y, z^{(i)}_z) \). As usual, the main physical variable is the transverse displacement \( w(x,y,t) \) of the plate, at position \( (x,y) \) and time \( t \). The equations of motion for the \( i \)-th plate are those for stiff objects with viscoelastic loss [11] and geometric non-linearities [12] (generally referred to as von Kármán equations, in the literature), and can be written as:

\[
\begin{align*}
\partial_t w^{(i)} - \kappa_i^2 \Delta_{2D}^2 w^{(i)} + \sigma_i \Delta_{2D} \partial_t w^{(i)} + \frac{1}{\rho_i H_i} \mathcal{L}(w^{(i)}, \Phi^{(i)}) \\
+ \frac{1}{\rho_i H_i} (f^+_i + f^-_i) + \frac{1}{\rho_i H_i} \delta(x_i - x_{exc}, y_i - y_{exc}) f_{exc}.
\end{align*}
\]

(1)

The differential operators \( \Delta_{2D} \) and \( \Delta_{2D}^2 \) are the Laplacian and biharmonic operators, respectively, with \( \Delta_{2D} = \partial_{xx} + \partial_{yy} \). The stiffness parameter \( \kappa_i \) is defined as:

\[
\kappa_i = \sqrt{E_i H_i^2 / 12 \rho_i (1 - \nu_i^2)},
\]

(2)

where (dropping the subscripts) \( \rho \) is the plate density, in kg/m\(^3\), \( H \) is the thickness, in m, \( E \) is Young’s modulus, in kg/s\(^2\)m, and \( \nu \) is the dimensionless Poisson’s ratio. \( \sigma \) is the coefficient governing viscoelastic losses, in m\(^2\)/s. All these parameters can in principle be distinct for the various plates.

The last term in the first line of (1) is responsible for non-linear effects and can be obtained from a fuller model when in-plane inertia is neglected [12]. When acting on two test functions \( \xi \) and \( \chi \), the operator \( \mathcal{L} \) gives:

\[
\mathcal{L}(\xi, \chi) = \partial_{xx} \xi \partial_{xx} \chi + \partial_{yy} \xi \partial_{yy} \chi - 2 \partial_{xy} \xi \partial_{xy} \chi.
\]

(3)

\( \Phi(x,y,t) \) is the so-called Airy’s stress function, and must satisfy the following constraint:

\[
\Delta_{2D}^3 \Phi^{(i)} = -\frac{E_i H_i}{2} \mathcal{L}(w^{(i)}, w^{(i)}).
\]

(4)
(1) and (4) must be considered, then, as a set of two coupled equations for each plate.

The second line of (1) includes the external forces acting on the plate. $f_i^+$ and $f_i^-$ represent the pressure of air acting above and below the surface, while $f_i,_{exc}$ is the excitation force. Their explicit expressions will be given below.

Appropriate conditions must be supplied at the boundary of $P_i$ for both $w^{(i)}$ and $\Phi^{(i)}$. For $w^{(i)}$, there are three interesting options from a sound synthesis point of view: clamped, simply supported and free conditions. For $\Phi^{(i)}$, instead, the usual choice is free condition. (See [13] for details.)

2.2. Air

The air surrounding the plates is described by a velocity potential $\Psi(x, y, z, t)$ which satisfies the wave equation:

$$\partial_t^2 \Psi = c_a^2 \Delta_3 \Psi,$$

where $c_a$ is the speed of sound in air (here, 340 m/s), and $\Delta_3$ is the 3D Laplacian. $\Psi$ is related to the more familiar quantities $p$ and $\nu$ (pressure and particle velocity) by:

$$p = \rho_a \partial_t \Psi, \quad \nu = -\grad \Psi,$$

where $\rho_a$ is the density of air (1.21 kg/m$^3$) and $\grad$ is the 3D gradient.

At the boundary $\partial V$ of the computational box it is necessary to implement absorbing conditions. One possible choice, which is convenient for reducing the computational complexity of the scheme, is a first-order Engquist Majda condition [14], defined as:

$$\partial_t \Psi + c_a n \cdot \grad \Psi = 0,$$

where $n$ denotes the unit vector normal to the wall and pointing outwards.

2.3. Coupling Conditions

Coupling conditions between the plate and the air can be obtained by enforcing continuity of pressure $p$ and particle velocity $\nu$ at the interface. When considering the acoustic field $\Psi$, they can be written as

$$f_i^+ = -\rho_a \lim_{z \to z_i^+} \partial_z \Psi |_{\tau_i}, \quad f_i^- = \rho_a \lim_{z \to z_i^-} \partial_z \Psi |_{\tau_i},$$

and

$$\partial_t w_i = \lim_{z \to z_i^+} \partial_z \Psi |_{\tau_i} = \lim_{z \to z_i^-} \partial_z \Psi |_{\tau_i}.$$  \hspace{1cm} (9)

These conditions hold over the plate regions $P_i$.

3. FINITE DIFFERENCE IMPLEMENTATION

The numerical implementation of the model has been performed using finite difference time domain methods [15]. Possible schemes for the non-linear plate are given in [16], while in [17] the coupling between the air and thin structures (membranes, in this case) is discussed. Therefore, implementation details will be omitted in the present work; the notation used here is drawn from [18].

The physical variables $w^{(i)}$ and $\Psi$ are approximated over regular Cartesian grids, with spacings $h_i$ and $h_a$, respectively. The discrete time step is $k = 1/F_s$, with sample rate $F_s$ chosen a priori.
The choice for the discretization of the non-linear term in (1) and (4) is discussed in detail in [16], where the explicit expression for the operator $l$ is given, together with a finite difference version of the boundary conditions.

Coupling conditions will be discretized in Section 3.3 while the discrete excitation will be discussed in Section 4.2.

### 3.2. Scheme for the Acoustic Field

The acoustic field $\Psi(x,y,z,t)$ can be approximated as $\Psi^n_{l,m,p}$. The finite difference version of (5) is:

$$\Delta_t \Psi = c_0^2 \Delta \Psi,$$

where the 3D Laplacian $\Delta$ is simply an extension of the 2D operator $\Delta$. A possible implementation of absorbing conditions is given, together with a finite difference version of (5) over the boundary $\partial V$ of the box can be found in [17].

### 3.3. Discrete Coupling Conditions

A detailed discussion of the coupling mechanism can be found in [17], therefore only a schematic outline will be given here. The vertical coordinate $z$ of each plate is slightly modified to the nearest value $z_i$ that lies half way between two neighbouring slices $\Psi^+_i$ and $\Psi^-_i$ of the acoustic field. Let $p^+_i = z_i + h_z/2$ and $p^-_i = z_i - h_z/2$ be the coordinates of such slices. Moreover, two interpolants $I^+_i$ and $J^+_i$ must be defined between the two grids and (4) become:

$$f^+_i = -\rho_0 I^+_i \delta_t \Psi^+_i, \quad f^-_i = \rho_0 I^+_i \delta_t \Psi^-_i,$$

and

$$J^-_i \delta_t u^{(i)} = -\delta_x \Psi^+_i - \delta_x \Psi^-_i.$$  

### 3.4. Stability conditions

Stability conditions for the above schemes can be easily obtained via energy analysis techniques (see [18]). For the plates’ grids, one obtains:

$$h_i^2 \geq 4k \sigma_i + 4k \sqrt{\sigma_i^2 + \kappa_i^2},$$

while, for the acoustic field,

$$h^2 \geq 3c_0^2 k^2.$$  

### 4. INSTRUMENT DESIGN, CONTROL AND OUTPUT

While designing a sound synthesis environment, one has to consider the parameters that a musician or composer will need to specify in order to create and play his or her own instrument. It is obvious that the more parameters there are, the more cumbersome the implementation will be. Furthermore, it has been shown that some physical quantities have more importance than others from a perceptual point of view.

In the present case, the various parameters can be grouped into three classes: instrument design, control and output.

#### 4.1. Instrument Design

At the beginning of the code, the user has to specify the geometric and physical description of the system. The former includes number, position and dimensions of the plates, together with the size of the computational box; the latter refers to all the constants appearing in (1), as well as to the boundary conditions for the various plates. In this second case, rather than exploring the entire space of physical parameters, it is useful to lump some of them under different “labels” (like steel, copper, etc.) according to perceptual considerations.

#### 4.2. Control

Control parameters define how the virtual instrument is played. In order to reduce an already heavy computational load, a single strike is modeled as a raised cosine over time [17]. The external force $f_{exc}$ in (1) corresponding to a strike of duration $\tau$ starting at $t = 0$ can be written as:

$$f_{exc}(t) = \begin{cases} \frac{F_{max}}{2}(1 - \cos(2\pi t/\tau)) & \text{for } 0 \leq t \leq \tau, \\ 0 & \text{else} \end{cases}$$

where $F_{max}$ is the maximum value of $f_{exc}$. A finite difference implementation of (18) is straightforward. Starting from this basic element, it is possible to create a series of strikes that can emulate complex gestures; the main difficulty, especially when the number of strikes becomes large, is in specifying for each of them $F_{max}$, $\tau$, the starting instant of excitation $T_{exc}$ and the striking position $x_{exc}, y_{exc}$. One possibility for avoiding the declaration of all these quantities is to randomize them (within predefined limits).

This simplified approach to strike generation, though perhaps slightly primitive in that it lacks the ability to capture more subtle features of the mallet interaction, such as contact-recontact phenomena, has been successfully used in the past in connection to the modular environment described in [7] to create several musical works. That being said, a mallet-plate interaction model would probably allow the composer/musician a more precise control over the instrument. An efficient and stable implementation of this non-linear contact force with the simultaneous presence of the plate non-linearity is currently under study.

#### 4.3. Output

As already mentioned, the entire acoustic field is modeled explicitly here. This allows the musician to draw outputs from any position within the box, by sampling air pressure variations generated by strikes on the plates. Multi-channel sounds are an interesting possibility, and they present virtually no additional computation cost, as output writing involves only a few multiplications and additions. In this case, one has to specify only the coordinates for each output location. As this is a time domain simulation, a moving output position can also be easily implemented.

#### 5. RESULTS

##### 5.1. Interaction through Air

In this model, acoustic pressure generated by a strike on a single plate will propagate within the box and excite the other plates, as well. Figure 2 shows the behavior of the system after a raised
5.2. Sound example

Sound examples obtained with this virtual instrument can be found in the author’s website:

http://www2.ph.ed.ac.uk/~s1164558

$\text{t} = 0.35\text{ ms}$

$\text{t} = 1.00\text{ ms}$

$\text{t} = 1.50\text{ ms}$

$\text{t} = 2.60\text{ ms}$

$\text{t} = 4.00\text{ ms}$

$\text{t} = 6.00\text{ ms}$

$\text{t} = 8.60\text{ ms}$

$\text{t} = 11.0\text{ ms}$

Figure 2: Acoustic pressure propagation generated by a strike on the upper plate, at times as indicated. Central cross sections of the acoustic field along the $xz$ and $yz$ planes are plotted, as well. The projections of the plates on these planes are marked with white lines.

6. CONCLUDING REMARKS

In this paper, a 3D environment based on a physical model of nonlinear plate vibration has been presented. It has been shown how a finite difference implementation of such a system offers enough flexibility to be used as a sound synthesis tool. Clearly, though, more work needs to be done in order to transform this into a mature musical instrument.

First of all, the usability of the system needs to be improved. On the one hand, it is necessary to define a “map” of perceptually meaningful physical parameters for the plates. This would spare the musician the daunting task of exploring a vast but sometimes perceptually redundant parameter space. On the other hand, the control strategy is still rather crude. When the number of strikes increases, the definition of hundreds of numbers could become a lengthy process. To this end, randomization of strikes offers some advantages, but may limit the creativity of the composer. A viable option could be the use of breakpoint functions over time to describe the global behaviour of relevant variables.

Secondly, many additional features could be added in order to obtain more interesting sounds. Some possibilities are spring-damper connections between the plates [7], bowing gestures [18], binaural sound output location [21]. By working with musicians and composers at Edinburgh University we look forward to exploring the strengths and the limits of the current model, and make improvements to it accordingly.

The computational complexity of this model has not been discussed in this paper. As is often the case for finite difference simulations, the algorithm presented above could require even several hours of computation in MATLAB for few seconds of output, depending on the number and sizes of the plates! This sometimes discouraging issue can be overcome with the use of parallel hardware, such as graphical processing units (GPGPUs). Possible speed-ups could be as far as tens of times [10, 22]. Algorithms for a fast parallel implementation of the present model are currently under study at Edinburgh University.

7. ACKNOWLEDGMENTS

Thanks to the anonymous reviewers for the useful comments and suggestions on how to improve this work.

8. REFERENCES


