19th INTERNATIONAL CONGRESS ON ACOUSTICS
MADRID, 2-7 SEPTEMBER 2007

MODELING VIRTUAL PERCUSSIONS AND THEIR VIRTUAL ACTUATORS

PACS: 43.55.Cs

Rovetta, Diego[1]; Sarti, Augusto[2]: Scarparo, Gabriele[3]; Tubaro, Stefano[4]
[1]Politecnico di Milano; Piazza Leonardo da Vinci 32, 20133 Milano, Italy; rovetta@elet.polimi.it
[2]Politecnico di Milano; Piazza Leonardo da Vinci 32, 20133 Milano, Italy; sarti@elet.polimi.it
[3]Politecnico di Milano; Piazza Leonardo da Vinci 32, 20133 Milano, Italy; scarparo@elet.polimi.it
[4]Politecnico di Milano; Piazza Leonardo da Vinci 32, 20133 Milano, Italy; tubaro@elet.polimi.it

ABSTRACT

One of the advantages of using a physical modeling method for the generation of percussions sounds is that the quality of the generated sounds depend on the point of interaction and how interaction takes place. Unfortunately, commercial actuators are not ready to accommodate such controls, therefore we need to resort to using alternate solutions. We propose a method for not just designing a percussion sets but also for interacting with it, based on touch estimation and tracking in solid surfaces. Contact positions are estimated from sounds acquired by contact sensors on the surface, the interaction point is localized and tracked, the interaction sound at the point of interaction is reconstructed and used as an input of the physical model of the percussion.

INTRODUCTION

Modeling sounds in a physical fashion is widely considered an extremely attractive solution to a variety of problems that other sound synthesis solutions are affected by. In fact, it gives access to novel timbral dynamics, mapping of algorithmic parameters onto manageable and playable controllers, minimal and meaningful set of control parameters, responsiveness, playability, access to prior experience, etc. Physical models, however, require the user to produce physical control parameters, and conventional controllers usually lack such ability. In this paper we assess this particular problem for the specific case of virtual percussions. We will see that, while numerous physical synthesis models are available in the literature, not so much can be found on the controller's side. We thus propose a simple, yet effective, solution to this problem, based on a tangible acoustic interface that can be built with any thin surface and a set of vibrational sensors.

PHYSICAL MODELING

The physical synthesis of sounds consists of modeling the vibrational phenomena that occur in a complex resonating structure, which can be made of a number of simpler resonators connected together. The vibrational phenomena are normally caused and, possibly, sustained by the interaction with other structures. Such structures and their interaction are usually modeled with a global approach (as a single system), but flexibility and scalability issues suggest us that it would be highly desirable to adopt a block-wise approach, based on the individual synthesis and discretization of the interacting parts. When dealing with the interaction between physical models and the related stability issues, it is important to define the signal flow between blocks in terms of a pair of dual variables (a through variable, such as velocity, flow, and current; and an across variable, such as force, pressure and voltage) for each connection port. This provides us with a characterization of the energy exchanged between blocks: so we see physical systems as an "interconnection" of functional blocks that are modeled as black boxes. This lumped-parameter description, however, does not force any assumption on the internal description of the individual blocks, which could be distributed parameter systems, as long as they communicate with the rest of the structure through a finite set of "ports".

To guarantee that the structure behaves in a physical fashion, blocks have to interact with each other in compliance with continuity laws (for example laws of dynamics, Kirchoff laws) and of preservation laws (for example energy, momentum): as we are working in a block wise fashion, we are basically thinking of lumped mechanical circuits, therefore the global continuity laws specify the topology of interconnection between blocks. If the continuity laws are satisfied, then several preservation laws are automatically satisfied as well.

PERCUSSION MODELING

A simple exciter for membrane interaction is the hammer model of Fig. 1, which is made of a mass connected to a nonlinear spring that models the limited compressibility of the felt as well as the contact condition. This corresponds to an equivalent electrical circuit that includes an inductor (mass) in series with a nonlinear capacitor (spring). Typically the nonlinear curve is a function that is identically zero when the compression is negative (contact condition) and returns a force that is proportional to a power of the spring's compression (typically the fourth power).

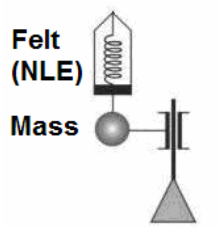

Figure 1.-Mechanical model of a hammer.

As for the membrane model, there are several alternative for modeling a 2D vibrating structure. What we adopted are two methods: the Functional Transformation Method [1] and the Digital Waveguides mesh [2][3][4].

The Functional Transformation Method (FTM) starts from the partial differential equation (PDE) that describes the vibration of a membrane:

$$S^4\nabla^4 z - c^2\nabla^2 z + \frac{\partial^2}{\partial t^2} z + d_1 \frac{\partial}{\partial t} z + d_3 \frac{\partial}{\partial t}\nabla^2 z = v \qquad \text{(Eq. 1)}$$

where z(t, x, y) is the membrane's deflection, v(t, x, y) is the excitation function and ∇ denotes the derivative with respect to the space coordinates (x,y).

The FTM uses the Laplace transform and the Sturm-Liouville transform to eliminate the space derivatives and the time derivatives, respectively, while accounting for boundary conditions. With this method the PDE with boundary condition is turned into an algebraic problem.

Another efficient method to model the wave propagation within a membrane rises from a multidimensional extension of the Digital Waveguides (DWG) [5].

A Digital Waveguide results from the digitalization of the one dimensional wave equation solution, which has the form of a superposition of a forward-traveling wave and a backward-traveling wave: digital filters can be added to the structure between one unit delays in order to account for frequency-dependent losses and wave dispersion due to media regidity.

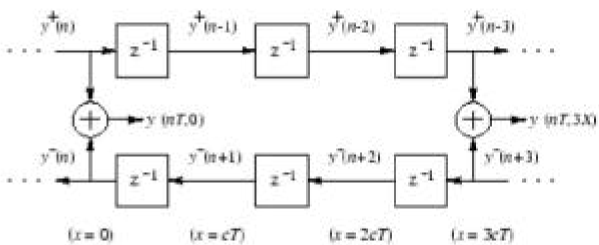

Figure 2. – A lossless Digital WaveGuide.

A 2-D Digital Waveguide Mesh is a regular array of discretely spaced 1-D digital waveguides, interconnected at their intersections trough “junctions”. Such nodes can also perform filtering operations in order to account for waves that should follow diagonal paths and need, therefore, delay compensation for path length differences.
In figure 2 a numerical simulation of a rectangular membrane hit near the border is shown: solid line peaks match very well black dots, that show the theoretical resonance frequencies.

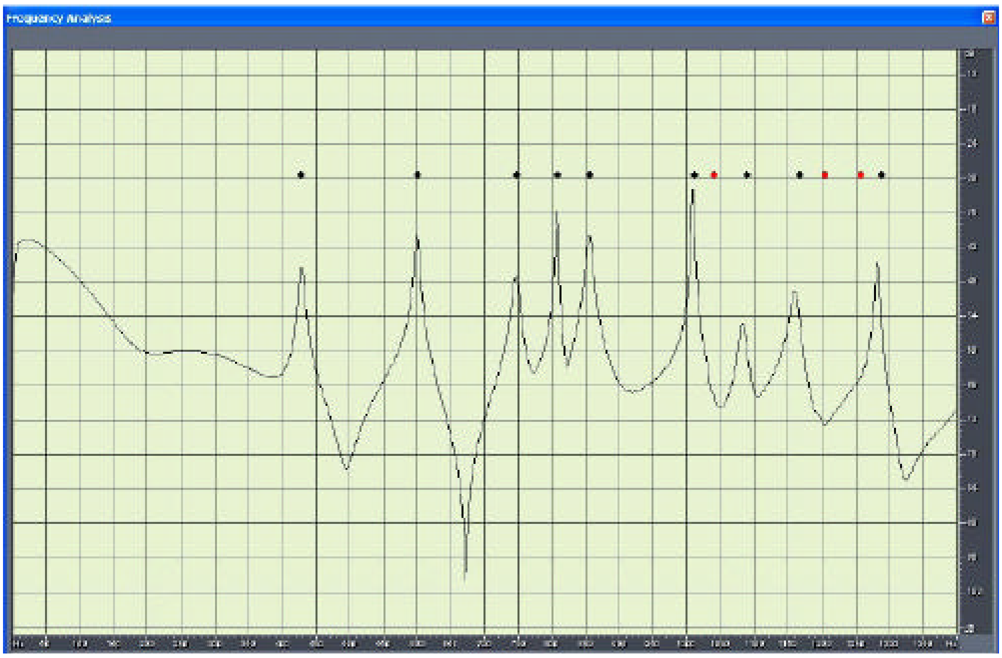

Figure 2. – Frequency response of a rectangular membrane simulation. The dots denote the theoretically determined resonance frequencies.

Another example is shown in figure 3, where a circular membrane is simulated with a triangular mesh.

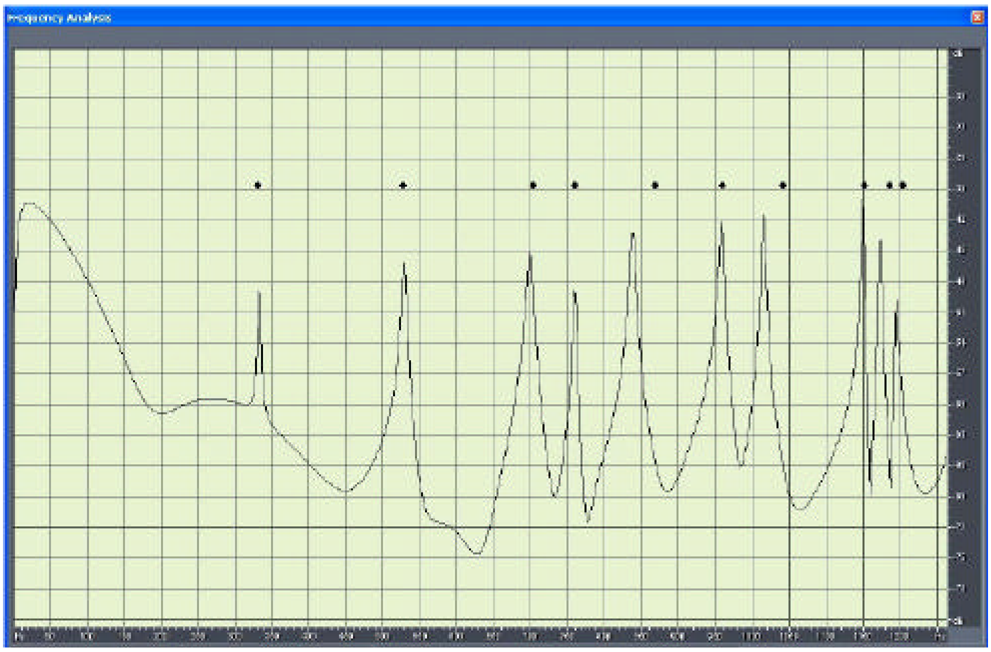

Figure 3. – Frequency response of a circular membrane simulation. The dots denote the theoretically determined resonance frequencies.

One more alternative to the above-mentioned methods consists of modeling free propagation in solid surfaces in such a way to account for not just wavefront propagation but also its dispersion due to the inherent rigidity of the media. Considering that the vibrating surface is thin, we can safely assume that the modes of propagation are only limited to surface (Lamb) waves, which undergo dispersion as they travel through the media. Wave propagation can thus be modeled through ray tracing and every time the ray bounces on the border, we can account for the dispersion that the wave has undergone in the last path after the previous reflection. This dispersion can be lumped into a single all-pass filter of higher order. There are several methods that are able to trace paths, some of which work in real time. In particular, a method proposed in [7] is based on beam tracing and is able to build a beam tree (tree of possible reflections and branchings of acoustic beams as they encounter reflectors on their way) on the fly. This last aspect is quite important because the excitation point (source location) changes every time, generating a new beam tree. Notice, however that this approach is suitable only for short sound and complex geometries.

VIRTUAL ACTUATORS

The first virtual drum actuator, a percussion controller called “Daton”, was developed in the 80’s by Max Mathews: it is a sensitive plate that responds to the location and force of a strike, measuring the different forces at four pressure sensors applied on the plate corners. A refinement of this system, called “Radio Baton”, was presented by Mathews and Boie.
Later the majority of virtual drum actuators were based on virtual reality tracking systems: “Lightning”, for example, was an interface designed by Donald Buchla, where an optical tracker measures the horizontal and vertical positions of a pair of wireless wands. Other systems are instead based on 2D motion tracking with cameras.

Localization of tactile interactions

All the systems described above use active techniques to localize the point of interaction: this approach restricts the mobility of the users, constraining them to be in certain locations during the interaction with the computer.
The other group of techniques, called passive, rely rather on the analysis of the acoustic vibrations generated at the points of contact: these methods are more promising if you want to develop new touch-based interfaces, that have to be scalable in dimensions, cheap and built with materials and devices that allow them to be suitable for any condition and environment.

The inverse problem formulation of the localization problem

Consider the situation depicted in Figure 4: the source localization problem can be formulated as follows [6]:

- Parameters (experiment configuration): sensors positions $S_i(x_i, y_i)$, i = 0, 1, 2, 3;
- Unknown quantities (model): source position $m = [x,y]^T$;
- Observed data: let di be the distance between source and i-th sensor and consider R0 as reference sensor; the observed data vector is $d_{obs} = [?d_1, ?d_2, ?d_3]^T$, where $?d_i = d_i - d_0$;

The link between model and observed data leads to a Jacobian matrix G, linearized around reference model $m_0 = [x_0, y_0]^T$

$$G = \begin{bmatrix} \frac{x-x_0}{d_0} - \frac{x-x_1}{d_1} & \frac{y-y_0}{d_0} - \frac{y-y_1}{d_1} \\ \frac{x-x_0}{d_0} - \frac{x-x_2}{d_2} & \frac{y-y_0}{d_0} - \frac{y-y_2}{d_2} \\ \frac{x-x_0}{d_0} - \frac{x-x_3}{d_3} & \frac{y-y_0}{d_0} - \frac{y-y_3}{d_3} \end{bmatrix} \qquad \text{(Eq. 2)}$$

and to the linear system

$$\begin{bmatrix} \Delta d_1 \\ \Delta d_2 \\ \Delta d_3 \end{bmatrix} = G \begin{bmatrix} x \\ y \end{bmatrix} \qquad \text{(Eq. 3)}$$

The following iterative fix-point algorithm, based on the Tarantola technique for non-linear inverse problems, has been proposed

$$m_{k+1} = m_{pr} - [G_k{}^T C_d{}^{-1} G_k + C_m{}^{-1}]^{-1} G_k{}^T C_d{}^{-1} [(g(m_k) - d_{obs}) - G_k(m_k) - m_{pr}] \qquad \text{(Eq. 4)}$$

where m_{pr} and C_m are respectively the mean and the covariance matrix of the a priori model, d_{obs} is the observed data vector, C_d is the covariance matrix of the measured uncertainties and of the modelling errors, G_k is the linearized Jacobian matrix at iteration k, m_k and m_{k+1} are the model vectors at k_{th} and $(k+1)_{th}$ iterations.

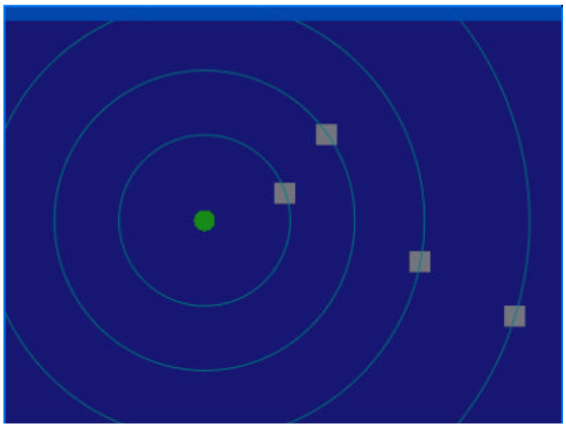

Figure 4. – Source position (green circle) with four sensors (squares)

APPLICATION EXAMPLE

In this section we present a “virtual drum” system, used during a “Territori Diagonali” live performance recently at the opera theatre of Como, Italy.
The setup consists of a plexiglas board, of size 150x100 cm., with four accelerometers (Knowles BU-1771) glued near its corners and connected to a Presonus “Firepod” audio card. The board is placed vertically in front of the user and a Graphical User Interface is projected directly on it, so that the user is able to “draw” his drum set (Figure 5), placing the virtual membranes on the screen, with his finger or with a drum stick, and setting their physical properties (such as shape, size and elastic parameters).

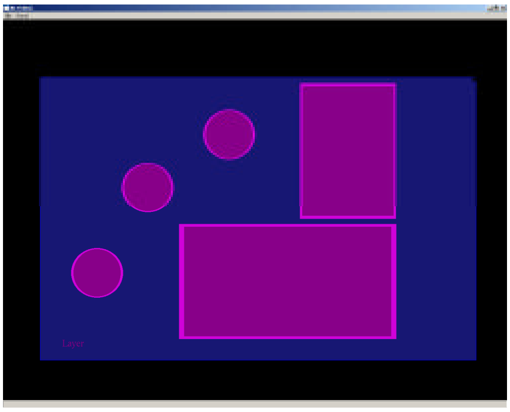

Figure 5.-The Graphical User Interface projected on the board.

Once this configuration phase is concluded, the virtual drum is ready to be played: when the user hits the board, the interaction point is localized by the system, and the hit membrane is detected. This information, together with a measure of interaction strength, is sent to the sound synthesis engine that computes the sound. A picture of the system is shown in Figure 6.

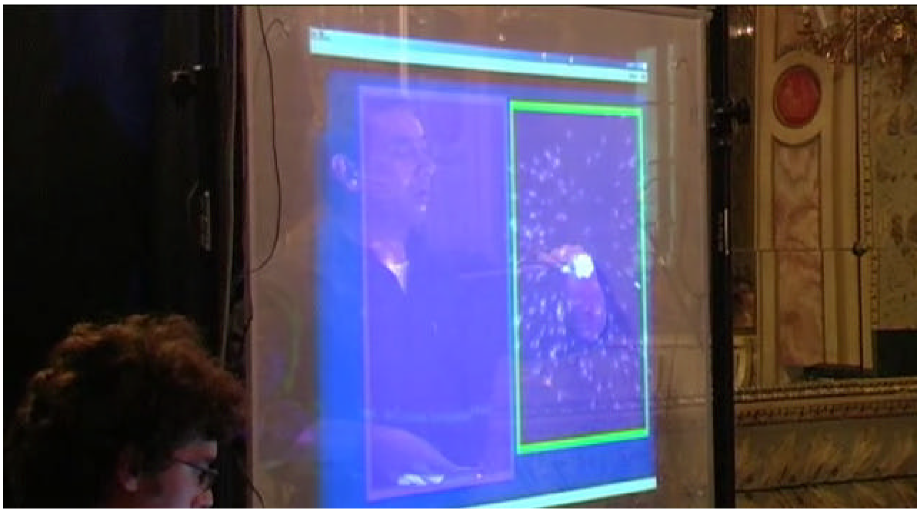

Figure 6.- Live performance at the opera theatre of Como, Italy.

CONCLUSIONS

The virtual drum controller used for the performance in Como was able to accommodate percussive interactions but was unable to deal with dragging interactions. We recently developed, however, a novel solution for the real-time tracking of continuous interactions on relatively thin surfaces, based on the backtracking of dispersion phenomena that occur in travelling waves. The method is very fast and accurate and is able to reconstruct even small movements of the fingerpoint such as writing. We are currently developing a novel controller for virtual percussions that exploits such tracking features for modelling brushes or other interesting interaction modalities that are often used by drummers. We are also developing solutions for the automatic classification of interactions in order to identify the tool used for interacting with the surface (a brush, a drumstick, etc.). This will allow the drummer to change tool without having to inform the system. One important issue is latency, which is quite important to drummers. Our system provides reduced time lags (around 30ms) but still needs improvements. Finally, one important unaddressed issue concerns the possibility of multiple simultaneous interactions, which are currently not allowed. With this goal in mind we are already experimenting with blind channel identification method.

References:

[1] L. Trautmann, S. Petrausch, R. Rabenstein, “Physical Modeling of Drums by Transfer Function Methods”, University of Erlangen-Nuremberg.

[2] S. A. Van Duyne, J. O. Smith III, “Physical Modeling with the 2-D Digital Waveguide Mesh”, Stanford University.

[3] S. A. VanDuyne, J.O. Smith III, “The 2-D Digital Waveguide Mesh”, Stanford University.

[4] F. Fontana, D. Rocchesso, “Signal-Theoretic Characterization of Waveguide Mesh Geometries for Models of Two-dimensional Wave Propagation in Elastic Media”, IEEE trans. on Speech and Audio, 2000.

[5] J. O. Smith III, “Physical Modeling using Digital Waveguides”, Stanford University, Computer Music Journal, Part I, Volume 16, no. 4, pp.74-91, Winter 1992.

[6] G. De Sanctis, D. Rovetta, A. Sarti, G. Scarparo, S. Tubaro, "Localization Of Tactile Interactions Through Tdoa Analysis: Geometric Vs. Inversion-based Method", European Signal Processing Conference, September, Florence (Italy), 2006

[7] M. Foco, A. Sarti, S. Tubaro, "beam tracing", DAFx…