

Controlling a Physical Model with a 2D Force Matrix

Randy Jones, Andrew Schloss
University of Victoria, Canada
Music Intelligence & Sound Technology Interdisciplinary Centre
(MISTIC)
rj@csc.uvic.ca, aschloss@finearts.uvic.ca

ABSTRACT

Physical modeling has proven to be a successful method of synthesizing highly expressive sounds. However, providing deep methods of real time musical control remains a major challenge. In this paper we describe our work towards an instrument for percussion synthesis, in which a waveguide mesh is both excited and damped by a 2D matrix of forces from a sensor. By emulating a drum skin both as controller and sound generator, our instrument has reproduced some of the expressive qualities of hand drumming. Details of our implementation are discussed, as well as qualitative results and experience gleaned from live performances.

Keywords

Physical modeling, instrument design, expressive control, multi-touch, performance

1. INTRODUCTION

Thanks to continuing advances in processor speed, the possibilities for real time physical modeling on commodity computers continue to multiply. In particular, the 2D waveguide mesh, first described by Van Duyne and Smith [24], and refined by others ([5] [6] [1]), can now be implemented in real time at sufficient sizes to make rich sounds.

In live performance, richness also depends on the control hardware and its link to the synthesis method used. Intimate control over a sounding object is central to musical expression. Julius O. Smith has expressed this elegantly by stating “A musical instrument should be ‘alive’ in the hands of the performer.” [21] We would like to make instruments that explore the potential of physical modeling while maintaining the liveness associated with acoustic instruments, thereby rewarding practice and virtuosity with a consistent and deep expressive potential.

The challenge of making deep instruments for computer music performance is addressed by Settel and Lippe [20]. They note that while you can bang on an empty oil drum in a large number of different ways, and get just as large a

variety of sounds, playing a synthesizer will always generate sounds bounded by its programming. Though it’s true that any musical system will be bounded by its programming, an issue that Settel and Lippe address by getting “out of the box” with electroacoustic instruments, we hope that our fine-grained approach to sensor/synthesis connection will let us make synthesizers that are more like the oil barrel.

One of the controllers we are using, the Tactex MTC Express, samples pressure at an array of 72 points on a surface. The difficulty of using the many degrees of freedom available from the MTC Express in the context of mappings is mentioned in [26]. Likewise, the 2D waveguide mesh model we have been exploring for percussion synthesis has a huge number of possible parameters: at each of hundreds of points on the mesh, a signal can be added and local properties of the mesh such as damping and tension can be changed. In previous work, waveguide meshes have been excited at single points (deinterpolated as discussed by Välimäki [23]), and live control over damping, so far as we know, has only been applied to parameters that affect the model globally.

Our approach, suggested by the capabilities of the MTC Express but by no means restricted to it, is to create a 2D force matrix from our controller data. This matrix is considered to be a continuous 2D field, sampled in time at the audio signal rate and in space at the dimensions of the physical model. We apply it applied to both excitation and damping of the model at each point. It is made from the raw MTC Express data by spatial interpolation, and from stick positions on another controller we are using, the Radio Drum [15], by rasterizing the points of surface contact. This control flow is shown in Figure 1. The force matrix allows all of the data from each controller to affect our sound model in a meaningful way.

As noted by Tindale et. al. [22], some percussion techniques require the player to hit the drum in a number of places simultaneously. In hand drumming, a great variety of expressive techniques arise from a trained performer’s control over the timing and area of hand contact. Distinct kinds of strokes create different musical effects by emphasizing audible variations in the drum’s vibration, suppressing certain resonant modes of the system and emphasizing others. A survey of strokes and damping mechanisms in Tabla drumming is given in Kapur et. al.[11]. The complex interactions between hand, drum head and drum allow for a handful of very distinct sounding types of strokes, and a vast expressive space between them. Rather than trying to design a particular set of mappings which affords such a vast degree of expression, we are working to add a sufficient control

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

NIME07, New York City

Copyright 2007 Copyright remains with the author(s).

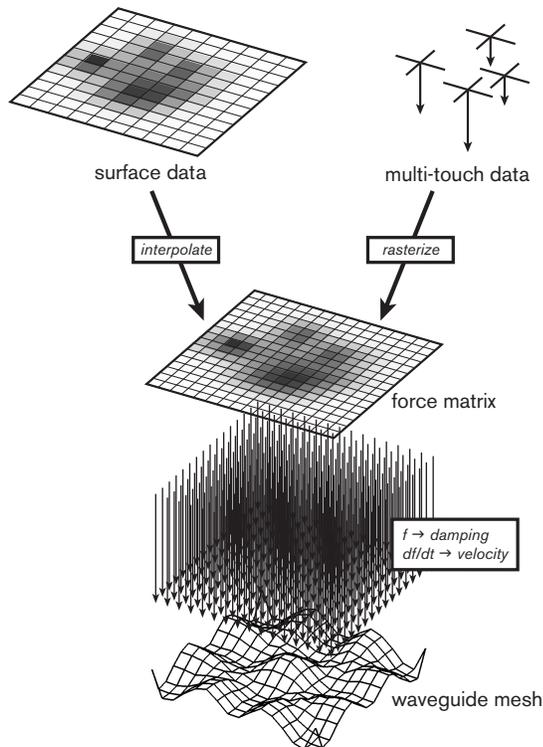


Figure 1: Control flow through 2D force matrix.

bandwidth between sensor and physical model to allow the expressive qualities of drumming to arise as emergent phenomena.

2. RELATED WORK

The use of banded waveguides to synthesize Tabla sounds is presented in Kapur et. al.[11]. In this work, a Tabla was augmented with force sensing resistors to recognize around six different stroke types. MIDI was used to trigger the synthesis engine with different parameters according to the type of stroke detected. This approach is complementary to ours; though it lacks the fine-grained control we are focusing on, a set of model parameters is hand-tuned for each stroke type, which results in realistic renditions of acoustic instruments. Similar work has been presented by Marshall et. al. [14], who use a 3D position sensor to control excitation and damping of modal synthesis.

The Korg Wavedrum is an interesting example of a hybrid acoustic/modeling instrument which provides for intimate control [18]. The Wavedrum has three contact microphones underneath an actual drumhead, which excite a synthesis engine. Because of the physical resonant properties of the drumhead, the sound changes in an organic way as one drums on it. For an electronic instrument, the Wavedrum feels very alive, though the range of sounds made is restricted in part because they all share the same physical exciter.

Hoskinson et. al. [7] present another approach to creating a large bandwidth between controller and synthesizer: image-based control for modal synthesis. Based on neural networks, their work also addresses the problem of preserv-

ing a very large yet playable control space.

2.1 Sensors

Our approach lends itself to use with a wide range of controllers currently available. Of the sensor strategies for capturing percussive gestures compared in [22], the MTC Express lends itself most capably to producing a dynamic 2D force matrix. However, any pressure sensor that can be mapped to a 2D surface, such as the Buchla Thunder or Continuum Fingerboard, could be used. Capacitive sensors such as the Smartskin [17], as well as the optical technique used in [4], can roughly determine pressure within a useful range from skin contact area, as long as a bare hand is used.

One drawback of current surface force sensors is their general rarity and associated high cost. Promising techniques for experimenting with low-cost custom force sensors are given in [13].

2.2 Physical Modeling

Real time implementations of finite difference schemes, including the 2D waveguide mesh, have been actively explored in recent years. Though digital waveguides are more efficient than finite difference schemes for a given system, they do not offer control over the state of the whole simulation, which is our focus. Bilbao and Fitch [3] also use the control afforded by the waveguide mesh in their work on prepared piano synthesis.

Our focus so far has been on intimate control rather than on refining sound quality. A combination of models as described by Aird et.al. [1], will be an important route to more natural sounds. Karjalainen [12] gives a useful overview of discrete-time modeling paradigms, which shows how to create a hybrid system incorporating one or more instances of a 2D waveguide mesh.

3. IMPLEMENTATION

A hypothetical ideal percussion sensor would have a spatial resolution on the order of 1mm in order to capture the nuances of hand or stick position, especially in damping. It would also be capable of audio sampling rates, which could allow the synthesis model to be excited directly by tiny bounces off of the drum head and by friction from rubbing across it. While the spatial resolutions of our available sensors meet these goals, their sampling rates do not. We describe here a hybrid approach which has let us create a successful musical connection between our current hardware and our synthesis model.

3.1 Sensors

Both of our sensors have been described in detail elsewhere. We refer interested readers to [15] and [10] for background information on the Radio Drum and MTC Express, respectively. We have added a calibration layer to the Radio Drum that allows us to receive accurate pressure data as a player presses the sticks down into its surface. Other recent work on the Radio Drum [16] has moved the “brain” of the controller into software, allowing us to use its data to excite the mesh at near audio signal rates (10kHz). The MTC Express is limited to a sampling rate of 120Hz when providing raw sensor data, which led us to develop our hybrid matrix/centroids excitation discussed below.

3.2 Physical Model

Our 2D waveguide mesh was implemented in the Max/MSP/Jitter environment. A scalar value at each mesh junction represents the velocity of an idealized membrane at that point. We use an 8-connected interpolated rectangular mesh, as described by Savioja and Välimäki[23]. While Fontana and Rocchesso [5] have shown that a triangular mesh geometry has desirable properties including better wave dispersion characteristics, we chose a rectangular mesh geometry because its calculation can be approached as a standard 3x3 convolution. This made our implementation easier, and may open the door to interesting methods of acceleration such as the use of graphics coprocessors. A simple 4-connected mesh was also implemented, which would offer more opportunities for optimization, but its decay characteristics sounded far less natural to us.

We can continuously tune the tension of the mesh by varying the self-loop impedances of the junctions, generating new coefficients for the convolution kernels. Interested readers are directed to [2] for a thorough discussion of coefficient generation and stability bounds for this and other waveguide mesh schemes. We add exponential damping per junction by simply multiplying the mesh with a damping matrix at each sample.

The mesh can be excited with an audio signal at any one position, which is deinterpolated to the four neighboring junctions as described in [6]. Excitation of all the mesh points simultaneously is also possible. We have implemented this using Jitter matrices at around 100Hz, so significant aliasing is potentially present in our incoming signal. Karjalainen [12] discusses the spurious oscillations that can result from inherent feedback in the finite difference time domain (FDTD) structure we use when high frequencies are present. We counteract these oscillations by bandlimiting the input with a matrix of FIR filters.

Our initial scalar implementation of a 16 by 16 mesh at single-precision floating point required approximately 40% of the CPU time of a 1.67 GHz G4 PowerPC processor. Optimizing the mesh with the AltiVec vector processing instruction set, we were able to achieve a 400% speed increase. The scalar code compiled on a 1.83GHz Intel Core Duo takes about 40% of one processor core; SSE optimization has not yet been done.

3.3 Connection

We use the 2D force matrix as a layer of abstraction between our sensors and our model, as shown in Figure 1. Data from the MTC Express are interpolated spatially to match the size of the mesh. The Radio Drum is treated as a multi-touch interface. When present, its two touches are rasterized to the force matrix as circles which grow in size according to pressure exerted on the drum's foam surface. The force matrix affects two parameters of the mesh at each junction: damping and membrane velocity.

Though we think of our force matrices as bundles of signals, a fully synchronous signal processing approach may be overkill given the rates of sensor data we have available. Also, Jitter is a useful tool for working with matrices, making implementation straightforward. In order to use Jitter while keeping the benefits of the signal data we get from the Radio Drum, we adopted a hybrid approach to control: excitations are sent to the mesh both as audio signals and as Jitter matrices. The signals are moved to specific positions on the mesh surface, depending on the positions of the

Radio Drum sticks, or centroids extracted from the MTC Express data. This allows the raw gesture data from the Radio Drum or even sound samples to be used to excite the model. Damping, on the other hand, is always done with the force matrix alone.

A new Max/MSP/Jitter object was written to get the centroids of clusters of *taxels*, gridwise pressure measurements on the surface. A previous approach to getting centroids from the MTC Express [25] used a k-means clustering algorithm. While robust, this solution required significant CPU. Our object, *2up.jit.centroids*, works more efficiently by exploiting spatial coherence of the taxels using a greedy algorithm seeded by maximum values. The object has also been used by others, including Bob Huott [8]. It is available with source code on the Web [9].

4. QUALITATIVE RESULTS

Our synthesis system has been used by each of us in concert performances. Andy Schloss has worked with the system both solo and in a trio combining computer music with Latin Jazz. The trio, with pianist Hilario Durán and violinist Irene Mitri, performed in February 2007 at Real Art Ways in Hartford, CT and at an Electronic Music Foundation show in New York. Jovino Santos Neto was the pianist for a show at CCRMA in April 2006. Randy Jones played a solo show of visual music incorporating the system at NIME 2005 in Vancouver.

We find that the malleability of the sound and its response to control make the system compellingly playable and, one hopes, listenable. Though a single sound from the system heard in isolation does not match the richness of an acoustic instrument, over time the depth of control available becomes clear. In particular, some sonic phenomena associated with physical drumming are captured well.

A full-hand slap in the middle of the pressure sensor sounds very different from a sharp tap on the edge. While a slap in the center creates lower frequencies and heavily damps all of the membrane's modes except the fundamental, the edge tap excites the mesh with high frequencies that go relatively undamped. As one moves between these positions and playing styles, the sound changes smoothly.

The pitch of the ringing drum can be altered with a stroke that moves across the drum head, as is possible on the Tabla. When a hand moves on the pressure sensor surface toward the center, it creates a smaller effective area over which resonance occurs, and the pitch is raised as a result.

Another acoustic phenomenon we reproduce is ringing from pulling quickly off the sensor. Since we excite the mesh with the derivative of force from the sensor at each node, energy is added to the mesh (but in the up direction) when a hand or stick is quickly pulled away. This can add a subtle ringing to a performance which mimics that of a physical drum: another detail which adds to the sense of realism.

Examples of these sounds are available on the internet [19].

5. FUTURE DIRECTIONS

We plan to make a force sensor with a faster sample rate, using new reconfigurable single-point pressure sensors from Tactex. The idea of exciting the model with audio from the sensor, as in the Wavedrum, is also appealing and would work well with our hybrid approach.

There are a number of areas in which we could improve the sound of our model. Adding a shell to the drum is probably the most effective first step forward. The addition of nonlinear excitation and filtering is also a very promising area, though proving the stability of nonlinear filters in discrete-time systems is still a difficult problem [3].

6. CONCLUSION

By creating a fine-grained connection between sensors and a physical model, we have made a very responsive performance system in which phenomena associated with physical instruments emerge naturally. This is satisfying to us, as a kind of aesthetic confirmation of a theoretical proposition. Our goal is to reproduce not the specific sounds of acoustic instruments, but the intimate control over sounds that enables complex and engaging musical expression. We hope that our approach will aid in the construction of deep instruments for computer music performance.

7. ACKNOWLEDGEMENTS

Financial support for this work was provided by NSERC and SSHRC Canada.

8. REFERENCES

- [1] M. Aird, J. Laird, and J. Ffitch. Modeling a drum by interfacing 2D and 3D waveguide meshes. In *Proc. Int. Computer Music Conf. (ICMC00) Berlin Germany, 2000*.
- [2] S. Bilbao. *Wave and Scattering Methods for the Numerical Integration of Partial Differential Equations*. PhD thesis, Stanford University, 2001.
- [3] S. Bilbao. Prepared Piano Sound Synthesis. In *Proc. of the 9th Int. Conference on Digital Audio Effects*, pages 77–82, 2006.
- [4] P. Davidson and J. Han. Synthesis and control on large scale multi-touch sensing displays. In *New Interfaces for Musical Expression*, pages 216–219. IRCAMCentre Pompidou Paris, France, France, 2006.
- [5] F. Fontana and D. Rocchesso. A new formulation of the 2D-waveguide mesh for percussion instruments. In *Proceedings of the XI Colloquium on Musical Informatics, (Bologna, Italy)*, pages 27–30, 1995.
- [6] F. Fontana, L. Savioja, and V. Välimäki. A modified rectangular waveguide mesh structure with interpolated input and output points. *Proc. Int. Computer Music Conf., (La Habana, Cuba)*, pages 87–90, 2001.
- [7] R. Hoskinson, K. van den Doel, and S. Fels. Real-time adaptive control of modal synthesis. In *New Interfaces for Musical Expression*, pages 99–103. National University of Singapore, Singapore, 2003.
- [8] R. Huott. Precise control on compound curves. In *New Interfaces for Musical Expression*, pages 244–245. National University of Singapore, Singapore, 2005.
- [9] R. Jones. Retrieved March 1, 2007. [Online] 2up.jit.centroids object, at <http://2uptech.com/objects.html>.
- [10] R. Jones. MTC Express Multi-touch Controller(review). *Computer Music Journal*, 25(1):97–99, 2001.
- [11] A. Kapur, G. Essl, P. Davidson, and P. Cook. The Electronic Tabla Controller. *Journal of New Music Research*, 32(4):351–359, 2003.
- [12] M. Karjalainen. Discrete-Time Modeling and Synthesis of Musical Instruments. In *Proc. Workshop on Applications of Signal Processing to Acoustics and Audio (WASPAA)*, 2004.
- [13] R. Koehly, D. Curtil, and M. Wanderley. Paper FSRs and latex/fabric traction sensors: methods for the development of home-made touch sensors. In *New Interfaces for Musical Expression*, pages 230–233. IRCAMCentre Pompidou Paris, France, France, 2006.
- [14] M. Marshall, M. Rath, and B. Moynihan. The virtual Bodhran: the Vodhran. In *New Interfaces for Musical Expression*, pages 169–170. National University of Singapore, Singapore, 2002.
- [15] M. Mathews and W. Schloss. The radio drum as a synthesizer controller. In *Proceedings of the International Computer Music Conference*, 1989.
- [16] B. Nevile, P. Driessen, and W. Schloss. A new control paradigm: software-based gesture analysis for music. *Communications, Computers and signal Processing, 2003. PACRIM. 2003 IEEE Pacific Rim Conference on*, 1, 2003.
- [17] J. Rekimoto. SmartSkin: an infrastructure for freehand manipulation on interactive surfaces. *Proceedings of the SIGCHI conference on Human factors in computing systems: Changing our world, changing ourselves*, pages 113–120, 2002.
- [18] G. Rule. Keyboard Reports: Korg Wavedrum. *Keyboard*, 21(3):72–78, 1995.
- [19] A. Schloss and R. Jones. Retrieved April 16, 2007. [Online] Sound examples at <http://people.finearts.uvic.ca/~aschloss/publications/nime2007.htm>.
- [20] Z. Settel and C. Lippe. Convolution brother’s instrument design. In *New Interfaces for Musical Expression*, pages 197–200. National University of Singapore, Singapore, 2003.
- [21] J. Smith. Physical modeling synthesis update. *Computer Music Journal*, 20(2):44–56, 1996.
- [22] A. Tindale, A. Kapur, G. Tzanetakis, P. Driessen, and A. Schloss. A Comparison of Sensor Strategies for Capturing Percussive Gestures . In *New Interfaces for Musical Expression*, pages 200–203, 2005.
- [23] V. Välimäki and L. Savioja. Interpolated and warped 2-D digital waveguide mesh algorithms. In *COST G-6 Conference on Digital Audio Effects*, pages 7–9, 2000.
- [24] S. Van Duyne and J. Smith. Physical modeling with the 2-D digital waveguide mesh. In *Proc. Int. Computer Music Conf*, pages 40–47, 1993.
- [25] C. van Wrede, P. Laskov, and G. Rättsch. Using classification to determine the number of finger strokes on a multi-touch tactile device. In *European Symposium on Artificial Neural Networks*, pages 549–554, 2004.
- [26] D. Wessel, M. Wright, and J. Schott. Intimate musical control of computers with a variety of controllers and gesture mapping metaphors. In *New Interfaces for Musical Expression*, pages 1–3. National University of Singapore, Singapore, 2002.