

The Electronic Tabla Controller

Ajay Kapur

akapur@princeton.edu

Georg Essl

gessler@cs.princeton.edu

Philip Davidson

philipd@princeton.edu

Perry R. Cook

prc@cs.princeton.edu

Department of Computer Science (also Music)
Princeton University, Princeton, New Jersey

Abstract

This paper describes the design of an electronic Tabla controller. The E-Tabla controls both sound and graphics simultaneously. It allows for a variety of traditional Tabla strokes and new performance techniques. Graphical feedback allows for artistic display and pedagogical feedback.

Keywords

Electronic Tabla, Indian Drum Controller, Physical Models, Graphical Feedback



Figure 1. Picture showing North Indian Tabla. The Bayan is the silver drum on the left. The Dahina is the wooden drum on the right.

INTRODUCTION

Tabla are a pair of hand drums traditionally used to accompany North Indian vocal and instrumental music. The silver, larger drum (shown in Figure 1) is known as the *Bayan*. The smaller wooden drum is known as the *Dahina*. [1] The pitch can be tuned by manipulating the tension on the *pudi* (drumhead). The Bayan is tuned by adjusting the tightness of the top rim. The Dahina can be tuned similarly, as well as by adjusting the position of the cylindrical wooden pieces on the body of the drum. Tabla are unique because the drumheads have weights at the center made of a paste of iron oxide, charcoal, starch, and gum (round, black spots shown in the Figures). [2] Also, the Tabla makes a myriad of different sounds by the many different ways it is stroked. These strokes follow a tradition which has been passed on from generation to generation, from *guru* (teacher, master) to *shikshak* (student) in the country of India. The combination of the “weighting” of the drum-

head, and the variety of strokes by which the Tabla can be played, gives the drum a complexity that makes it a challenging controller to create, as well as a challenging sound to simulate. In the remainder of this paper, we will present:

- The evolution of the technology of the Tabla from its origins until the present day.
- The traditional playing style of the Tabla, on which the controller is modeled.
- The creation of a real-time Tabla controller, using force-sensors.
- The physical modeling of the sound of the Tabla using banded waveguides.
- The creation of a real-time graphics feedback system that reacts to the Tabla controller.

EVOLUTION OF THE TABLA

There are a few accounts for the origin of the Tabla. A mythological account reads: “Once, a long time ago, during the transitional period between two Ages... people took to uncivilized ways ... ruled by lust and greed [as they] behaved in angry and jealous ways, [while] demons, [and] evil spirits... swarmed the earth. Seeing this plight, Indra (The Hindu God of thunder and storms) and other Gods approached God Brahma (God of creation) and requested him to give the people a *Krindaniyaka* (toy) ... which could not only be seen, but heard, ... [to create] a diversion, so that people would give up their bad ways.” [1] One of the *Krindaniyakas*, which Brahma gave to humans was the Tabla. Other legends state that the Tabla was created in the 18th Century by Sidhar Khan Dhari, a famous *Pakhawaj* player.



Figure 2. Picture showing a Mridangam, a drum of the Pakhawaj family.

Pakhawaj is a genre of Indian drum defined by a barrel with drum-heads on either side. The *Mrindangam*, shown in Figure 2, is one drum in this family of drums. It was said that Sidhar Khan provoked an angry dispute after losing a music contest and his Pakhawaj was chopped in half by a sword. Thus, the first Tabla was created accidentally [3]. Some Tablas were created out of clay, others out of wood. As technology for producing metal alloys evolved, the Bayan started to be molded out of brass and steel [4].

As the popularity of the Tabla spread to the western hemisphere, nearly coincident with emergence of the personal computer, scientists began to combine the Tabla with computers. In 1992, James Kippen created software which allowed a user to input a traditional Tabla rhythmic pattern, which the computer would then use to synthesize an improvised pattern that followed traditional rules for variation [5]. In 1998, Mathew Wright and David Wessel of University of California Berkeley, aimed to achieve a similar goal, with a real time interface and unique data structure. They successfully created software that generated “free and unconstrained” music material, which could fit into a given traditional rhythmic structure. [6] Jae Hun Roh and Lynn Wilcox created two pressure sensitive pads to input rhythms. These patterns are then used to generate new phrases based on traditional Tabla patterns. [7] Meanwhile, Talvin Singh created a direct input from his Tabla to computer effects, achieving sound manipulations in an invention he calls “Tablatronics” [8][9].

There are a number of commercially available hand-drum controllers such as Buchla’s Thunder [10], Korg’s WaveDrum [11], and Roland’s HandSonic [12]. This project, however, uses a new physical model of Tabla acoustics, and our main goal is to preserve the traditional appearance, feel, and performance characteristics of North Indian classical Tabla drumming.

TABLA STROKES

It is important to understand the traditional playing style of the Tabla to see how our controller models its hand movement. Below is a picture explaining the names of the different parts of the Tabla *pudi* (drum head).



Figure 3. Picture showing parts of the Tabla *pudi*.

Bayan Strokes

There are two strokes played on the Bayan. The *Ka* stroke is executed by slapping the flat left hand down on the Bayan as shown in Figure 4 (a). Notice the tips of the fingers extend from the *maidan* through to the *chat* and over the edge of the drum. The slapping hand remains on the drum after it is struck to kill all resonance, before it is released away. The *Ga* stroke, shown in Figure 4 (b), is executed by striking the *maidan* directly above the *syahi* with the middle and index fingers of the left hand. When the fingers strike, they immediately release away from the drum, to let the Bayan resonate with sound. The heel of the left hand controls the pitch of the *Ga* stroke, as shown in Figure 4 (c). It controls the pitch at the attack of the stroke, and can also bend the pitch while the drum is resonating. Pitch is controlled by two variables of the heel of the hand: force on to the *pudi*, and the position on the *pudi* from the edge of the *maidan* and *syahi* to the center of the *syahi*. The greater the force on the *pudi*, the higher the pitch. The closer to the center of the *syahi*, the higher the pitch. [1]

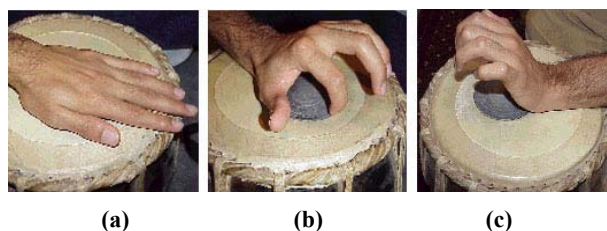


Figure 4. Pictures showing traditional strokes played on Bayan

Dahina Strokes

There are six main strokes played on the Dahina. The *Na* stroke, shown in Figure 5 (a), is executed by lightly pressing the pinky finger, of the right hand down between the *chat* and the *maidan*, and lightly pressing the ring finger down between the *syahi* and the *maidan*, in order to mute the sound of the drum. Then one strikes the *chat* with the index finger and quickly releases it so the sound of the drum resonates. The *Ta* stroke is executed by striking the middle finger of the right hand at the center of the *syahi*, as shown in Figure 5 (b). The finger is held there before release so there is no resonance, creating a damped sound. The *Ti* stroke, shown in Figure 5 (c), is similar to *Ta* except the middle and ring finger of the right hand strike the center of the *syahi*. This stroke does not resonate and creates a damped sound. The *Tu* stroke is executed by striking the *maidan* with the index finger of the right hand and quickly releasing, as shown in Figure 5 (d). This stroke resonates the most because the pinky and ring fingers are not muting the *pudi*. [1] The *Tit* stroke, shown in Figure 5 (e), is executed similar to *Na*, by lightly pressing the pinky finger, of the right hand down between the *chat* and the *maidan*, and lightly pressing the ring

down between the *syahi* and the *maidan*. The index finger now strikes the *chat*, quickly releasing to let it resonate. The index finger strike on the *chat* is further away from the pinky and ring finger, than it is on the *Na* stroke. *Tira* is a combination of two strokes on the Dahina, which explains the two syllables of the stroke. It is executed by shifting the entire right hand from one side of the drum to the other. It creates a damped sound at each strike. This stroke is shown in Figure 5 (f) and Figure 5 (g).

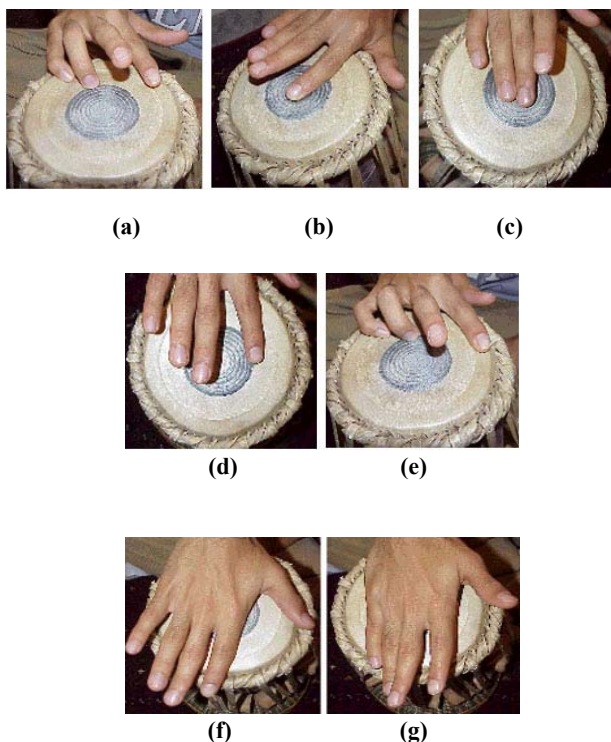


Figure 5. Picture showing traditional strokes played on the Dahina.

THE MIDI TABLA CONTROLLER

We modeled our controller based on the hand positioning and movements of the strokes discussed. We used square force sensing resistors (square FSR) to input force of different finger strikes, and long force sensing resistors (long FSR) to obtain the position of finger strikes, as well as force. [13] All events are converted to MIDI signals and sent out via a MIDI output.

The Bayan Controller

The Bayan Controller was created using two square FSRs, and one long FSR. Figure 6 shows a layout of these FSRs. The top square FSR is used to capture *Ka* stroke events, when a player slaps down with their left hand. If it receives a signal, then the other two FSRs are ignored. The square FSR in the middle, captures *Ga* stroke events, when struck by the middle and index finger of the left hand. The long FSR controls the pitch of the *Ga* stroke events, using two variable: force and posi-

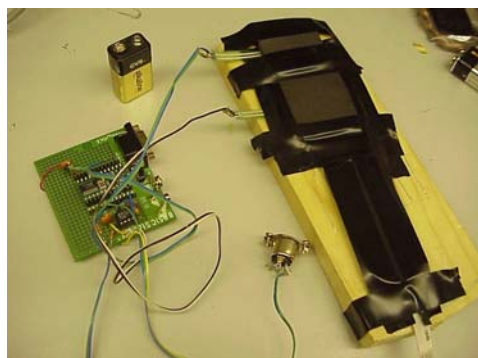


Figure 6. Picture showing Bayan controller FSR layout



Figure 7. Picture showing Dahina controller FSR layout

tion. The greater the force exerted by the heel of the left hand, the higher the pitch. The closer the heel of the hand gets to the *Ga* FSR, the higher the pitch. The pitch can be bent after a *Ga* stroke is triggered.

The Dahina Controller

To implement the Dahina Controller, we used four FSRs: two long FSRs, one square FSR, and one small FSR (which acts like a square FSR). Figure 7 shows a layout of these FSRs. The small FSR triggers a *Tit* stroke event. It measures the velocity of the index finger's strike. The square FSR triggers a *Tira* stroke event. It measures the velocity of the hand slapping the top of the drum. If the *Tira* FSR is struck, all other FSRs



Figure 8. The Electronic Tabla Controller

are ignored. If the *Tit* FSR is struck, both long FSRs are ignored. The long FSR on the right in Figure 7, is the ring finger FSR, and the long FSR on the left is the index finger FSR. If there is a little force on the ring finger FSR (modeling a mute), and the index finger FSR is struck at the edge of circle, a *Na* stroke is triggered. If the index finger FSR is struck near the center of the circle, a *Ta* stroke is triggered. If there is no force on the ring finger FSR, and the index finger FSR is struck, then a *Tu* stroke is triggered. When the ring finger FSR is struck with enough force, and not held down, then a *Ti* stroke is triggered. Thus we have modeled every stroke that we discussed above. Figure 8 shows a picture of both controllers in their constructed Tabla encasements.

SOUND SIMULATION

The electronic Tabla controller signals can be used with any standard MIDI device to produce sound. However, the typical synthesis methods do not properly mimic the dynamics of the Tabla drums and hence the performance sound in relation to strokes is not well captured. Physical modeling is known to allow for direct physical interactions and hence the control values produced by the Tabla controller can be directly used as inputs rather than first finding a mapping that relates controller-output to synthesis-relevant parameters. We use the “banded waveguides” which were originally introduced for one-dimensional structures like bar percussion instruments [14] but has recently been generalized to higher-dimensional structures [15].

Banded waveguides are a generalization of digital waveguide filters [smith] which accommodate complex material behavior and higher dimensions by modeling the traveling waves for each model frequency separately

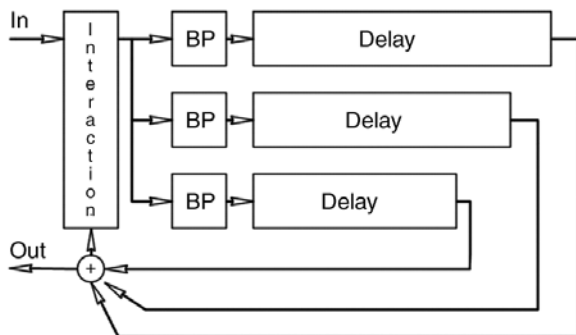


Figure 9. Figure showing banded waveguide schematic.

as is depicted in Figure 9.

These collection of banded wavepaths which build the full system have, however a geometric correspondence which allows to find the interactions points. Modes come about as standing waves, which is equivalent to the condition that traveling waves close onto themselves. Hence the task of finding geometric positions from modes corresponds to finding paths that close onto themselves and finding the matching mode for that path.

This problem has been studied by Keller and Rubinow [16] and the construction of finding these paths on a circular membrane is depicted in figure 10.

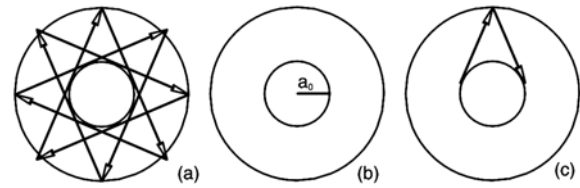


Figure 10. Figures showing construction of paths that close onto themselves

Tabla strokes correspond to feeding strike-velocities at the right positions into the delay-lines. A particularly interesting performance stroke is the *Ga* stroke performed on the Bayan depicted in figure 4 (c). It includes a pitch bend that is achieved by modifying the vibrating area due to pushing forward. This can be viewed as a moving boundary, which in case of banded waveguide corresponds to a shortening of the closed wavepaths, in turn corresponding to a shortening of the delay-lines of the model. A comparison of a recorded and a simulated *Ga* stroke can be seen in figure 11.

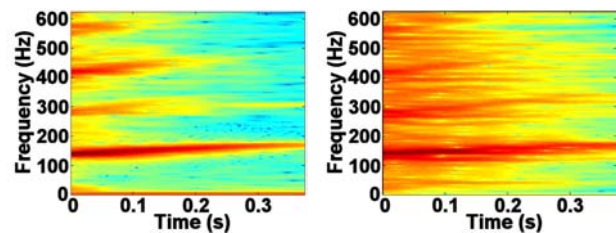


Figure 11. Sonograms comparing recorded (left) and simulated (right) *Ga* strike.

GRAPHIC FEEDBACK

The visual system for the electronic Tabla is designed to augment the experience of the electronic Tabla by providing player and audience with a visual display that dynamically responds to the drums in parallel to the audio response. Since the audio synthesis requires most of processing power of the audio machine, graphics processing occurs on a second machine, with controller messages routed to both systems. As the Dahina drum is still in development, we will describe the response of the system to Bayan strikes.

Our concept for the graphics system began as a combination of geometric form with fluid motion. To respond to the percussive energy of Tabla music, the visualization we developed is based on a particle system in which strikes made by the player appear as patterns composed of small shapes which are the basic visual elements of the display. As the player makes *Ka* and *Ga* strikes on the Bayan controller, particles are rearranged into lines, circles, cardioids, and other shapes depending on the type and quality of each strike. The velocity and pitch

are mapped to the size, color, complexity and physical characteristics of the patterns we create.

Once particles have been placed, their continuing motion is controlled by a vector field which imposes forces on each particle. After a strike places a form on the screen, the form will break apart and returns to the background motion. The vector field can also be configured to respond to the movement and positions of particles. The behavior of the field is governed by a distribution of 'cells' which determine the forces that will be exerted in their local area, based on the number and distribution of particles in their domain. Through this feedback of cell-particle dynamics, we obtain behaviors which can mimic real-world systems. By altering both the physical characteristics of particles and the specifics of cell-response behaviors, we can use the same system to produce a variety of effects (see Figure 12)

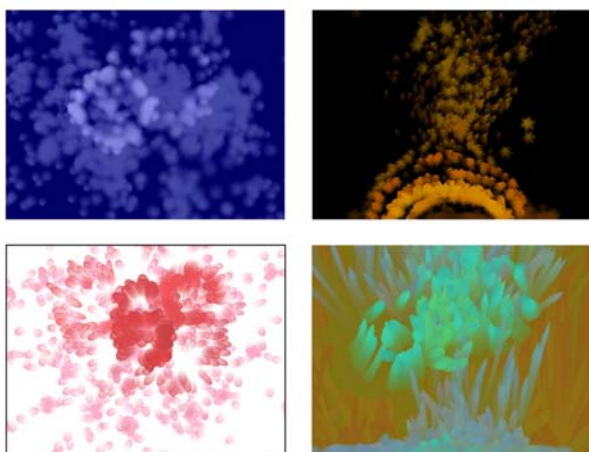


Figure 12. Different modes of visual feedback.

In addition to effects produced by the different strikes, it is also helpful to provide a response to the state of the drum itself. Since the Bayan is responsive to palm pressure on the head of the drum, we visually impart a sense of increase or decrease in tension on the drum head through corresponding compressive or decompressive forces to the particle systems [17].

CONCLUSION

We presented a real-time device for Tabla performance. The Tabla controller augments the traditional interactions in various ways. The performer can now choose the sound production, independent of the physical interaction. Automated teaching feedback becomes possible. We illustrate this ability by providing performance-dependent visual feedback.

ACKNOWLEDGMENTS

Our thanks to Manjul Bhargava for his Tabla playing and recorded sounds. Special thanks to David Hittson for photographing Tabla strokes.

REFERENCES

- [1] Courtney, David R. *Fundamentals of Tabla: Complete Reference for Tabla*, vol. 1, pp. 1-15 (Sur Sangeet Services, 1995).
- [2] Rossing, Thomas D. *The Science of Sound*, pp. 373-374 (Addison-Wesley Publishing Company, 1999).
- [3] See <http://www.Tabla.com/articles/part1a.html>
- [4] Courtney, David R. "Repair and Maintenance of Tabla", *Percussive Notes*, vol. 31, no. 7, pp 29-36, (October 1993).
- [5] Kippen, James. "Tabla Drumming and the Human-Computer Interaction", *The World of Music*, vol. 34, no. 3, pp 72-98, (1992).
- [6] Wright, Mattew & David Wessel, "An Improvisation Environment for Generating Rhythmic Structures Based on North Indian 'Tal' Patterns". Available at: <http://cnmat.cnmat.berkeley.edu/ICMC98/papers/html/wright-wessel-tal-demo.html>
- [7] Roh, Jea Hun & Lynn Wilcox, "Exploring Tabla Using Rhythmic Input," in Proceeding of the CHI '95, Denver, CO, May 7-11, pp. 310-311, (1995).
- [8] Kaul, Vatsala. "Talvin Singh", Available at: <http://www.india-today.com/today/121998/boom.html>
- [9] Tsering, Lisa. "Talvin Singh Has Seen the 21st Century, and It's "O.K.", Available at: <http://members.tripod.com/~LisaTsering/talvin.html>
- [10] Available at: <http://www.buchla.com/historical/thunder/>
- [11] Available at: <http://www.korg.com>
- [12] Available at: <http://www.rolandus.com/pdf/roland/HPD-15.pdf>
- [13] Available at: <http://www-ccrma.stanford.edu/CCRMA/Courses/252/sensors/sensors.html>
- [14] Essl, Georg & Perry R. Cook, "Banded Waveguides: Towards Physical Modeling of Bar Percussion Instruments," In Proc. Int. Computer Music Conf. (ICMC), Beijing, 22-28 October, pp 321-324, (1999).
- [15] Essl, Georg & Perry R. Cook, "Sound Propagation Modeling in Solid Objects," submitted to IEEE Computer Graphics and Applications.
- [16] Keller, J. B. & S. I. Rubinow, "Asymptotic Solution of Eigenvalue Problems," *Annals of Physics* 9, pp 24-75, (1960).
- [17] Stam, J., and Fiume, E., "Turbulent Wind Fields for Gaseous Phenomena", SIGGRAPH '93, 369-376, (1993).