Haptic Sound: Expressive Control of Song Playback Using Haptics

Alex Baumann, Georges Grinstein

University of Massachusetts Lowell
Institute for Visualization and Perception Research
Olsen Hall Rm 301, Lowell MA 01854
abaumann@cs.uml.edu
grinstein@cs.uml.edu

ABSTRACT

Haptics devices are an excellent tool to add user feel and expression to applications. For music software they can provide realistic simulations of real world instruments, or heighten the experience of interacting with musical objects. This paper focuses on an application that uses haptics as a means of interacting with pre-recorded pieces of music allowing the user to navigate and add personal expression. Simplicity and ease of use were keys to the design so that people without any prior musical knowledge were able to create and interact with music right away.

1. INTRODUCTION

As a collaborative effort between our lab and SensAble Technologies Inc we created an application that allowed users to control the playback of pieces of music using a haptics device. The software was written and tested for the SensAble Omni haptics device and became one of several demo applications shown at SIGGRAPH 2004 as it highlighted an uncommon use of haptics.

Rather than creating an application that simulated an expressive real world instrument using haptics, we wanted to allow users to instantly create music that always sounded good, was easy to play and could be interpreted in various ways. The software was implemented with a music playback theme where the user is able to control the progression through the songs and variations within using the haptics device. This allowed users who had no musical background to use and understand the software right away, as well as more advanced users to experiment within the musical piece.

1.1. Background

HapticSound combined 3D graphics using OpenGL, MIDI controlled sound produced by DirectX, and haptic control and force feedback. Users were shown a colorful two dimensional surface with a winding path from left to right that represented the timeline of a pre-programmed song. Users were able to follow the path while constrained haptically to the timeline, or move freely across the plane without constraints. A selected piece of music was played back \as it was recorded when moving at a constant speed from left to right along the timeline, but users could easily play the song faster, slower, or in reverse by simply moving the haptic device in that manner along the path.

The timeline cut a smooth groove across the plane, resulting in hills adjacent to the timeline that contained controlled interpolations between notes in the song. Additional haptic device input such as force against the plane and twisting of the stylus were mapped to musical parameters, allowing further interpretation of the piece of music.

Colors on the surface represented musical notes of a certain instrument in the piece of music. Matching colors were mapped on an image of a musical keyboard in the bottom left of the screen. When the stylus probed a given color on the surface, the note corresponding to that color was highlighted on the keyboard and could be heard. This provided a visual link between color, pitch and user interaction, which we felt would be useful in teaching basic music theory, although we found that most users did not notice this link without us pointing it out. Figure 1 shows the surface created for Beethoven's "Für Elise". The colors represent the notes played by the main piano in that piece.

A small team of students created the software during a very short period of time so the feature set had to be interesting enough to be compelling, yet simple enough for users to understand the concept right away. Though the entire set of originally intended features did not end up in the final application, we were able to create a demo that greatly engaged the crowds and highlighted the original intention of the project.

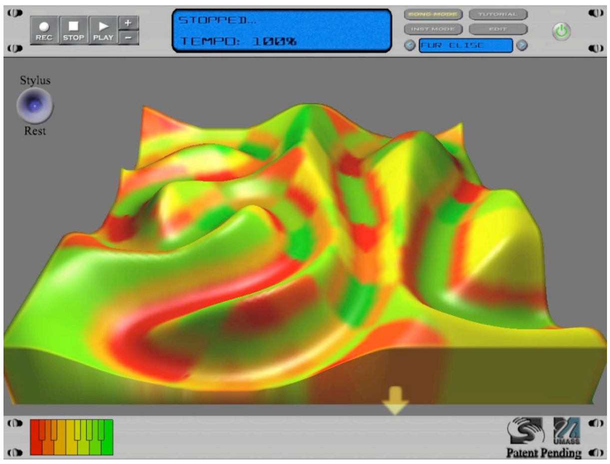

Figure 1: *HapticSound application with Beethoven's "Für Elise" loaded.*

1.2. No Music Skill Required

The main goal of this application was to investigate how haptics could be used as an expressive tool for musical exploration. Though there were many directions this project could have taken, we decided to focus on the concept that making or playing music should be easy. The need for simplicity was a driving force behind some of the features in the application, such as haptic constraint to the song path, a simple abstract representation of a song, and the smooth, colorful, and visually appealing layout of the 3D surface.

Since mastering a real musical instrument takes years of practice, we felt that haptics would be an excellent tool to allow us to create an environment in which users could instantly produce good sounding music. Rather than playing a single instrument, it became apparent that using a piece of music and the familiar notion of a timeline through the song would allow users to instantly grasp the concept and start interacting with the software right away. We allow the haptic stylus to be constrained to the timeline so that the user who is trying the software for the first time can easily move forward and backward in the song without getting lost. This constraint can be removed, at which time the user is free to play other areas of the surface that contain interpolated song data.

Another mode accessible from our application allowed users to create their own music rather than control the playback of a selected piece of music. In this mode, parameters of the surface were mapped to musical scales across different instruments. This mode had a surface with 4 quadrants, the top left of which had jazz oriented instruments all tuned to a common jazz scale so that it appeared that many musicians were all playing at once. This also allowed users without any musical background to create music that always sounded "good". Different musical styles were laid out across the surface, and were defined in a file that was defined by us. Figure 2 shows the HapticSound application in this mode with its four colored quadrants. This paper does not go into detail about this mode, since its main focus is the control and interpretation of existing musical scores.

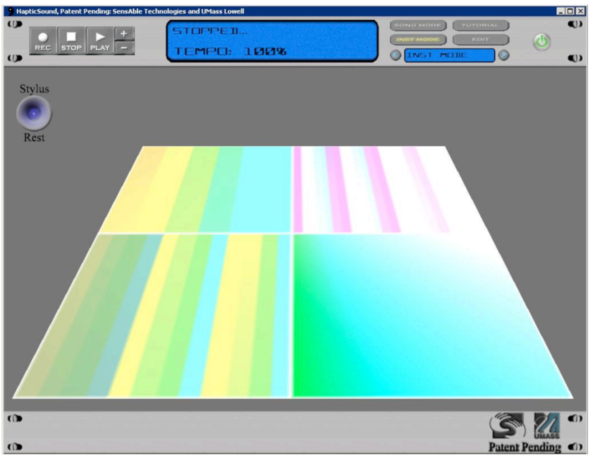

Figure 2: *HapticSound in "Instrument" mode*

1.3. Timeline

One of the most difficult aspects of the design of this application was determining how to take a piece of music and create both a haptic and visual representation of it that made immediate sense to users. Once we had decided upon using a musical piece with a timeline that the user could navigate through, we had to determine how to represent this in an appealing and straightforward manner. A three dimensional object such as a sphere may have been too difficult for a user to grasp and would lead to problems of occlusion, so we decided to place the timeline on a nearly planar surface.

Rather than having a straight line progressing the song from beginning to end we wanted to create some sort of space-filling curve that covered the surface and had a clear start and end to it. The original intention had been to generate this curve programmatically, and ideally link the pattern shown on the screen to some aspects about the song, such as simple versus complex regions of the track. Since time did not permit us to do this, we simplified the problem by parsing an image of a curve that we drew and placed on the surface. This meant that we had the same curve for each song that the software could load, but this was acceptable for the purposes of our project and was easy to remove later.

The timeline curve was used to "cut" a virtual river through the 3D landscape, and several smoothing passes were used to give us a very soft and natural surface to interact with. This was important not only for the visual aesthetic of the application, but also for the haptic feedback that the surface provided. The grooves cut out on the surface allowed a user to feel when they had moved away from the timeline onto the interpolated portions of the musical score, though it was much easier to follow the song when the haptic constraint to the timeline was used. Further work on the haptic feedback parameters would have allowed us to very easily remain on the timeline without using a constraint to it, making it easier to shift from the original progression of the song to the interpolated "hills" surrounding it. Moving along the timeline allowed the user to control aspects such as tempo and moving in reverse, providing a simplified "conductor" feel.

1.4. Song Data

We had hoped to allow loading of MIDI files that are easily attainable on the Internet as the song data. Time did not permit us to do this, and further research would be required to determine how to take any type of song in any musical genre and best lay it out on a surface so that a user can easily interact with it and interpret in a musically pleasing manner. Since we were not able to do this, we created our own file format that defined what MIDI notes would be played at what time for each instrument used in the song. Songs were transcribed from MIDI files hand and tweaked until they sounded correct when played across the surface. We were only able to do this for a few songs, but this method still allowed us to create a successful application prototype. This file structure and method of defining music was limited and we only implemented very simple songs, but one could easily see how this could be extended to very complex and engaging pieces of music that a user could control and interpret.

The original intention was to use the distance the user traveled along the timeline as the position within the musical piece. However this did not allow us to easily access interpolated musical notes that were located adjacent to the timeline. We instead created a two-dimensional array corresponding to the coordinates of the surface filled with musical notes for each MIDI instrument used in the piece. This way we were able to use the X and Y coordinate of the user on the surface as an index into the table of musical notes.

In the first pass of creating this table, we simply filled in the values corresponding to the path along the timeline curve. The values within the rest of the surface corresponded to the height of the surface perpendicular to each point along the timeline. In other words, as a user moved up a hill (in the Z direction) to the right of each point along the curve the corresponding pitch along this path increases, whereas to the left it decreases. This procedure stops at the top of each hill, filling notes across the entire surface without passing over any area more than once.

This timeline was at the bottom of these hills and one could progress through the song by following the bottom of these valleys. It was difficult, however, to do this accurately and quickly, so the constraint to the timeline helped. Moving along the constrained timeline also had no friction, allowing quick and smooth movements, whereas the surface had a small amount of friction to avoid slipping across the surface in undesired directions. We also added limitations as to how quickly one could move through a song on the constrained timeline, as the haptics API had to calculate and maintain these constraints as the movement occurred. In order to greatly increase the tempo of a song, one would either have to move very rapidly, which caused problems with the constraint, or the song would have had to have been created with less space between notes, meaning a smaller distance traveled on the surface would cover more notes in the song. The problem with the later approach would have been the difficulty to easily and accurately control movement through the song, as the precision of the input would decrease.

Notes that were filled in as the height increased were not simply the next musical note on a keyboard (i.e. if the note was a C, the next note was not necessarily C#), but rather the next note in a sequence that was defined in the song data file. This sequence was determined by us for each song, and corresponded to some common scale in the song. This meant that at each point along the timeline one could play this same scale by moving perpendicularly to the curve, but that each scale would start from the note on the current point along the timeline. Users were then able to play a scale in the key of the current song position, thus allowing interpretation of the song within scales appropriate to the song itself. This also meant that the same song could be played in different keys by moving along the side of a hill parallel to the timeline.

We had planned to interpolate the song data across the surface using linear or exponential interpolation. This proved to be unsatisfying musically as the notes that filled the rest of the surface usually had very little to do with the song. For this reason, we used the aforementioned scales so that users could move freely across the surface and experiment while maintaining themes that are appropriate to the song. In order for these scales to be generated programmatically we would need to determine how to calculate scales that are appropriate for each song loaded. It is likely that a good method of doing this would produce many different scales that vary throughout the song, as the chords and scales used in the original song vary. The hope was that, much like in jazz music, one could play a song and then go off in different directions musically, while being able to easily come back to the original theme.

The colors on the surface corresponded to color scales that we also stored within the song data files that we created. Each song had its own programmable color scale, and each note within an octave of a musical keyboard corresponded to a different color. Different octaves were darker (lower octave) or lighter (higher octave). When the user touched the haptic stylus to any color on the screen, the corresponding MIDI note that was played was highlighted on the keyboard in the bottom left of the screen. This provided both an aesthetically pleasing surface as well as a simple to follow correspondence between color and musical note. Though the user does not need to know anything about the actual notes that are being played on a musical keyboard, this method of coupling color to musical note in an interactive manner would offer a simple method of teaching the basics of music theory. This requires more experimentation and analysis as we found that most users did not even look at the keyboard.

In the earliest prototype of this application, we loaded in black and white images as bump map data. Just as the brightness component was used to affect height of the planar surface, this component was also used to affect the pitch of an instrument. This did not prove to be very musically appealing and was often too random. Later in the development we created the ability to map color to pitch and experimented with different color patterns. Using this we were able to load in a color image and listen to it on a surface. Though this did not always produce musically appealing results, it was an interesting experiment that we would like to explore further.

1.5. User Control and Expression

The surface we generated allowed users to control a piece of music by either moving along the timeline or across the surface as they saw fit. Additionally, haptic input such as rotation of the stylus and pressure against the surface were used to add expression to the musical piece. Each of these parameters was assigned to some MIDI parameter and could affect multiple instruments and instrument parameters at once. In *Beethoven*'s "*Für Elise*" we mapped rotation of the stylus to vibrato on a string sound and pressure to the volume of these strings. This allowed the user to play through the song and adjust their pressure or rotation to affect string sounds that followed the main piano melody and provided an intensified effect to the song that was quite pleasing.

Tweaking the mappings between haptic control and feel and mapped MIDI parameters was extremely important to the success of this prototype. We had to experiment significantly with different combinations and ranges until settings were found that allowed users to easily interact with the music. It was very important that the right amount of user movement mapped to a musically appropriate and controllable amount of sound mapping. A user should not have to strain and twist the device in excessive amounts just to affect certain parameters. For this reason, we had mappings that took the haptic input and scaled it from some minimum to some maximum MIDI parameter amount. This mapping could also be offset by constant in order to set the middle or start of some parameter to a place that a user would typically feel is a comfortable starting point on the device.

Though it was not defined in the song data format, we realized that certain parameters required different types of scaling. Some had to be scaled exponentially or by some n-degree until they felt musically correct. Instrument volume, for instance, worked best with an exponential scaling because humans do not notice steps in volume as much as a parameter such as pitch. It felt more appropriate to be able to greatly increase the volume of an instrument with relatively small movements. In this manner, we lost more fine grain control of this parameter, but in the case of volume it was not needed.

1.6. Similar Work

Though there are many research documents related to the use of haptics in a musical context, most of these dealt with simulating the natural feel and control of musical instruments. The "Haptic Music" application was shown at ICAD 2000 by Gandy and Quay from Georgia Institute of Technology [1]. This system focused on collaboration between individuals and sets of jazz styled instruments that could be combined. The HapticSound application we created also had a mode that allowed a user to create new music from groups of instruments tuned in different styles of music, but the main focus of our application was interpreting existing music. The simplification of musical creation to some basic haptic input is similar, and we had envisioned the ability to share and collaborate music with individuals across the Internet.

One application was presented at NIME05 [2] and allowed interpretation of a composed piece of music, as well as other more advanced gestural and control actions. The main focus of both applications seems to be the simplification of the music process to an interactive and enjoyable activity for musical and non-musical individuals alike. The user interaction involved in this application appears more focused on either playing back a static piece of music or creating new music using the haptics device. The main differentiating factor between the two appears to be that we focused on allowing a user to add expression to existing pieces of music while they are controlling playback through the piece, rather than doing this through different modes.

An interesting paper that was also presented at NIME05 [3] by the CCRMA at Stanford University involved common physical musical interactions and how they could be used to provide haptic input to musical applications. They further investigated the use of different haptic feedback in musical control, as well as a method akin to scratching a turntable to control playback of a piece of music.

1.7. Lessons Learned

This prototype application gave us a good sense of how to create more advanced haptically driven music applications. One important aspect of such applications is the mappings between haptics and sound. These must be easy for a user to use, as well as musically appropriate. Certain mappings we tried required twisting the device in difficult ways, especially when trying to add expression using two or more degrees of freedom. For this reason, it is important to provide mappings that require reasonably small amounts of motion that will not strain the user. It also means that two musical expression parameters that will be used in conjunction should be placed on some set of input parameters that are easily controllable at the same time so as to limit awkward movements.

Haptic to musical parameter mappings must also be chosen so that the amount of human precision possible on that input matches the required precision of the musical parameter. Controlling volume, for instance, does not require the same level of control as for pitch since humans do not perceive changes in volume as well as changes in pitch. Trying to find the appropriate combination of controllability, comfort and precision requires much experimentation.

As this application has been demonstrated many times to varying groups of individuals we have gotten a significant source of feedback for improvements that could be made to the software. One major concern that many (including ourselves) noticed was that the hills surrounding the timeline were too high, and therefore difficult to navigate with the haptics device. Though these hills could stay as high visually, we need to make the transitions across them easier to control.

1.8. Future Considerations

There are many improvements that could be made to the software, the most important of which being the automation of surface creation from standard MIDI files. Right now the process of turning a song into an interactive haptic surface is not entirely manual but still very time consuming. This process should be done algorithmically,. Since each song can be so different musically, the musical aspects contained in various songs would have to be analyzed and determined how they pertain to various visual and haptic parameters. Generalizing this may prove difficult, but simplified models that create surfaces that are of varying quality depending on the song would be a first step. Simplifying the actual song data to reduce complexity may also aid this process.

The improvements and additional implementation that would be incorporated into later versions of this software depends on its intended application. As it stands currently, the software was more of a proof of concept and experimentation tool. We foresaw two major directions that this software could take. One is as a tool that allowed people to create, interact with and interpret music. The other is a tool for learning music at various levels. Both of these could be incorporated into a single application that offers varying levels of difficulty and learning. For true beginners without any musical background, controls similar to those provided in our current application could be provided. This would allow a user to simply move around and hear recognizable music, and start to get a sense of how their interaction relates to that music. As users became more advanced, different modes could be used to offer further musical expression.

We had also intended that users would want to record and share their interpretations with others. For this reason we incorporated a recorder within the application that sampled the haptic input as the user controlled the device. This allowed us to play back the interpreted musical piece by creating virtual haptic input to the musical and visual system. This could be easily extended to use the standard MIDI file format to store both haptic input and the resulting musical output into a single file that could be shared over the Internet. Users would be able to play back the musical interpretation from the player to hear and see what the user did with their haptics device, or use this interpreted piece as a song file that they could place on the surface and interact with further.

2. CONCLUSIONS

We chose to implement a haptics application that allowed users to instantly control recognizable music in order to simplify the process of making music. By doing this we allowed any level of user to play with a piece of music through haptic interaction. The combination of haptic control, colors mapped to musical notes and simple musical ideas such as tempo led us to believe this could become an excellent tool to teach basic musical theory. This concept could be extended and adapted to grow with the user as they learn more about music and want to explore more musical expression.

3. REFERENCES

[1] M. Gandy and A. Quay, "Haptic Music: Non-Musicians Collaboratively Creating Music," *ICAD2000*, Atlanta, Georgia, April 2000, Poster/Demo Session 2

[2] X. Rodet, J.P. Lambert, R. Cahen, T. Gaudy, F. Guedy, F. Gosselin and P. Mobuchon, "Study of haptic and visual interaction for sound and music control in the Phase project" *NIME2005,* Vancouver, British Columbia, May 2005, pp. 109 - 115.

[3] W. Verplank, "Haptic Music Exercises" *NIME2005,* Vancouver, British Columbia, May 2005, pp. 256 – 258

4. ACKNOWLEDGEMENTS

Team members included Jon Victorine, Abdul Bezrati, John Sharko and Vivek Gupta. Funding support came from SensAble Technologies Inc. Many thanks to those involved and the support from SensAble Technologies Inc.