The vBow: Development of a Virtual Violin Bow Haptic Human-Computer Interface

Charles Nichols
Center for Computer Research in Music and Acoustics, Stanford University
660 Lomita, Stanford, CA 94305-8180, USA
cnichols@ccrma.stanford.edu

Abstract

This paper describes the development of a virtual violin bow haptic human-computer interface, which senses bow position with encoders, to drive bowed-string physical model synthesis, while engaging servomotors, to simulate the haptic feedback of a violin bow on a string. Construction of the hardware and programming of the software are discussed, as well as the motivation for building the instrument, and its planned uses.

Keywords

Violin, Bow, Controller, Haptic, HCI, Interface.

MOTIVATION

As a performer and composer, I found the MIDI violin a poor musical controller for interactive computer music. Due to the errors produced by its translation of the vibration of the violin strings into MIDI pitch and velocity data, it was difficult to get an accurate or expressive translation from performance gesture to synthesized sound. Also, because of the latency of the system, there was a considerable delay between the performer bowing the string and the MIDI device producing a sound.

As I began to develop ideas about a design for an alternative expressive musical controller, based on the paradigm of the violin, I learned of the work underway, to improve the expressiveness or playability of the bowed-string physical model [6]. An instrument with an array of sensors, that mapped physical bowing gesture to parameters of this improved bowed-string physical model, seemed like an excellent design for an expressive violin bow controller.

At the same time, I learned of the research underway, studying the affect of haptic feedback on the response of a musician to an instrument [4, 5]. According to the findings, not only does a performer respond to kinesthetic response when learning an instrument [5], but also the addition of haptic feedback to a performance system improves playing accuracy [4]. These studies suggested the importance of incorporating haptic feedback into the design of an alternative violin bow controller.

HARDWARE

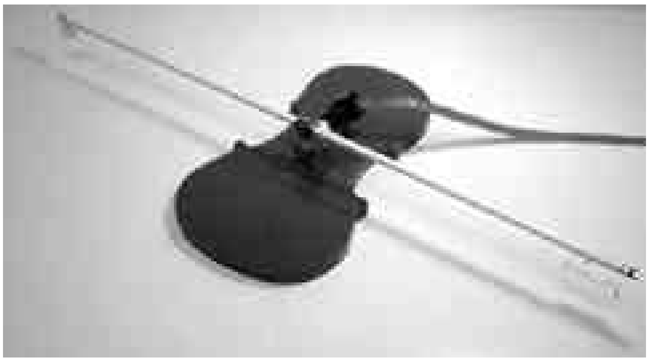

Figure 1. The vBow, version 1.

The first version of the vBow was designed around a single servomotor with a digital encoder. The servomotor is secured to a housing machined from acrylic, that is screwed to a violin-shaped wood base. A capstan screwed to the shaft of the servomotor fits through a hole drilled through the depth of the acrylic housing. Attached to the top of the housing with a strip of metal is a linear bearing, through which a fiberglass rod passes. Attached to either end of the rod are an acrylic tip and frog. A cable stretches from the tip to the frog, through a hole drilled through the width of the housing, and wraps around the capstan attached to the shaft of the servomotor. As the violinist draws the rod of the vBow back and forth, the cable spins the capstan attached to the shaft of the motor, and the encoder reads the counts per revolution of the shaft [2].

Figure 2. The vBow, version 2.

The second version of the vBow is constructed from pieces that were laser cut from acrylic and additionally machined with a mill. The capstans were turned from an aluminum rod with a lathe. The cable used in the first version, stretched from tip to frog across a fiberglass rod, is also used in the second. In addition to using the single servomotor with encoder of the first version, the second version also uses three more servomotors with encoders to sense rotational motion across, vertical motion above, and longitudinal motion along the virtual strings [3].

Figure 3. The vBow, version 2.1.

Because of the yaw caused by the leverage of drawing the vBow perpendicular to the long arm of the second version, two improvements were made to the second version. The arm was shortened, and the acrylic pieces were secured together more tightly. Lock nuts screwed at either end of threaded pins, rather than the original e-clips which fit into slots cut into lengths of stainless steel rod, were used to secure the acrylic pieces together.

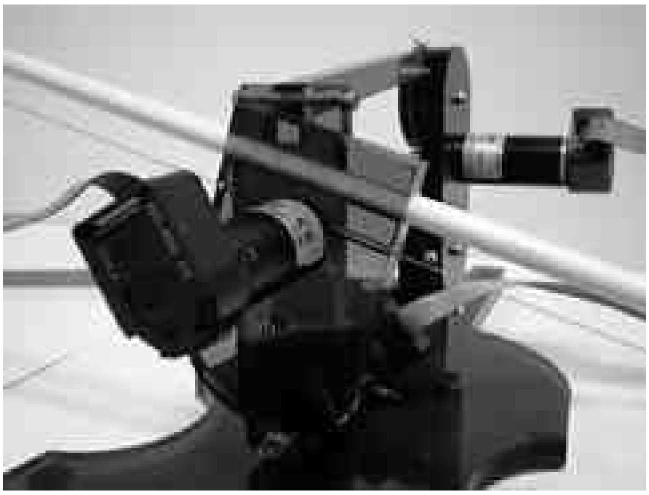

Figure 4. Rotational motion.

One of the additional servomotors with encoders of the second version senses bow rotation across multiple virtual strings, providing for string crossings in the sound synthesis software, and will produce the haptic feedback of additional resistance in the control software, when the vBow reaches the encoder count for each virtual string. As the performer rotates the vBow across multiple virtual strings, the synthesis software will produce double and triple stops, and the control software will produce the feeling of coming into contact with each virtual string.

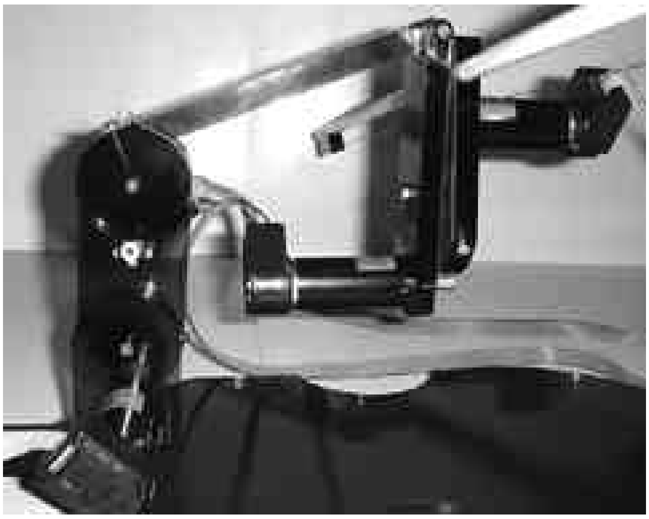

Figure 5. Vertical motion.

Another of the additional servomotors with encoders senses bow position above and into the virtual strings, allowing for various attack velocities in the sound synthesis software, and will produce the haptic feedback of string elasticity, when the vBow reaches the encoder count for the surface of the virtual strings. As the performer lowers the vBow into the virtual strings, the synthesis software will produce the attack of a violin timbre, and the control software will produce the feeling of pushing into the virtual strings.

Figure 6. Longitudinal motion.

The last of the additional servomotors with encoders senses longitudinal position of the bow along the length of the virtual strings, supplying bow position in relation to the virtual bridge to the sound synthesis software, and will produce the haptic feedback of longitudinal friction, proportional to vertical position, when the vBow moves towards or away from the encoder limit for the virtual bridge. As the performer moves the vBow along the length of the virtual strings, the synthesis software will change from a sul ponticello to a sul tasto timbre, and the control software will produce the feeling of longitudinal friction along the length of the virtual strings.

For all of these additional servomotors with encoders, a short cable is fastened with screws to the curved acrylic piece above, so that when the acrylic piece is moved, the curved part and the attached cable glide over the surface of the capstan, spinning the shaft of the servomotor. Conversely, as the control software engages the servomotor, the capstan moves the cable, along with the acrylic piece it is attached to.

The author manufactured the acrylic pieces, using a laser-cam, mill, and tap, the aluminum capstans, using a lathe, and the threaded pins, using a die.

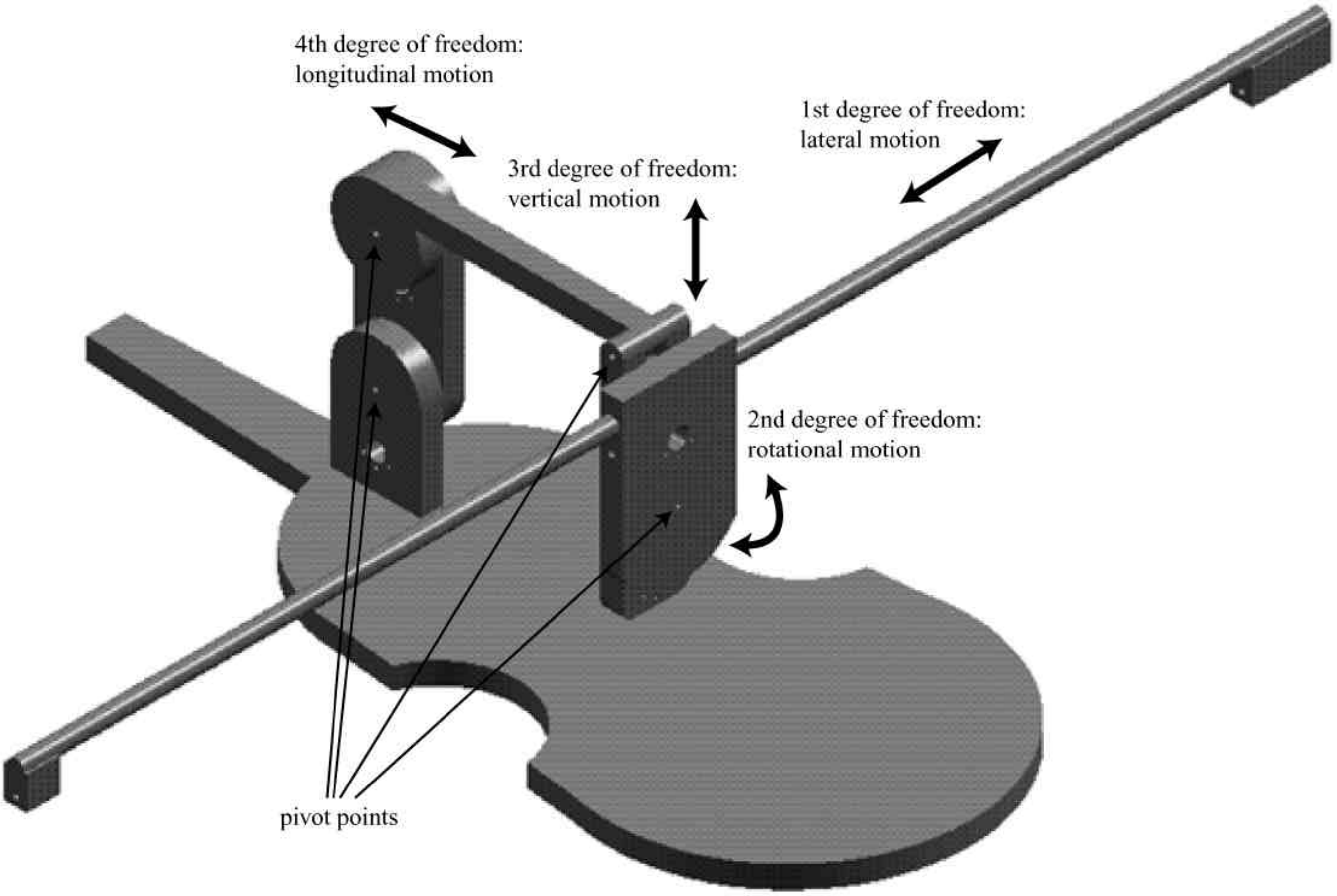

Figure 7. Degrees of freedom.

SOFTWARE

The software for the first version of the vBow uses Synthesis ToolKit (STK) objects and methods written in C++ [1], for the sound synthesis of the system, and the C/C++ library for the Windows 95/98 driver bundled with the ServoToGo data acquisition and servomotor control card, for the servomotor control.

The software reads the encoder count, as the cable attached to the fiberglass rod spins the capstan, and the servomotor shaft it is attached to. The sampled encoder count is used as the velocity parameter for the bowed-string physical model sound synthesis.

The software also generates a randomly varied vibration with the servomotor, in the direction of the vBow stroke, producing an additional frictional drag on the virtual bow.

The software for the second version of the vBow, which the author is in the process of writing, still uses the STK objects and methods for the sound synthesis of the system, but now uses the C/C++ libraries written for the ServoToGo card by Quality Real-Time Systems (QRTS), for the servomotor control.

Whereas the libraries for the ServoToGo card, used for the first version of the vBow, were written for a Windows 95/98 driver, the QRTS libraries, used for the second version, are for use in software running under RTLinux. Programming the control software for the vBow to run under RTLinux, a real-time version of the popular Unix-based operating system, will eliminate latency caused by the Windows operating system.

APPLICATION

The first version of the vBow has been a useful instrument for testing the expressiveness of the bowed-string physical model and the efficacy of adding haptic feedback to a musical controller system.

As the user draws the vBow slowly, the synthesis software produces a scratchy timbre. As the user draws the vBow quickly, the synthesis software produces a flautando timbre. And, as the user draws the vBow at an optimal speed, the synthesis software produces a clear bowed-string timbre.

The second version of the vBow will be a versatile and expressive electronic musical instrument for interactive computer music performance. Because the encoder data can be mapped to the parameters of any synthesis software, the vBow can be used to perform any synthesis instrument. Also, because the haptic feedback of the controller can be varied continually during performance, the kinesthetic feedback of the vBow can be used as a compositional element.

ACKNOWLEDGMENTS

Thanks to the Center for Computer Research in Music and Acoustics, the Department of Music, and the School of Humanities and Sciences, at Stanford University, for financial and facilities support for this project, and to Advanced Motion Controls, Maxon Precision Motors, The Olander Company, and Sava Industries for the donation of equipment and supplies. Additional thanks to the Product Realization Lab at Stanford University, for the use of their equipment.

REFERENCES

[1] Cook, P. R. and G. Scavone, "The Synthesis ToolKit (STK)", *Proceedings of the International Computer Music Conference*, Beijing (1999).

[2] Nichols, C, "The vBow: A Haptic Musical Controller Human-Computer Interface", *Proceedings of the International Computer Music Conference*, Berlin (2000).

[3] Nichols, C, "The vBow: Two Versions of a Virtual Violin Bow Controller", *Proceedings of the International Symposium of Musical Acoustics*, Perugia (2001).

[4] O'Modhrain, S. and C. Chafe, "Incorporating Haptic Feedback into Interfaces for Music Applications", *Proceedings of the International Symposium on Robotics with Applications, World Automation Conference*, Maui (2000).

[5] O'Modhrain, S. and C. Chafe, "The Performer-Instrument Interaction: A Sensory Motor Perspective", *Proceedings of the International Computer Music Conference*, Berlin (2000).

[6] Serafin, S., J. O. Smith, III, and J. Woodhouse, "An Investigation of the Impact of Torsion Waves and Friction Characteristics on the Playability of Virtual Bowed Strings", *Proceedings of the IEEE Workshop on Signal Processing to Audio and Acoustics*, New Paltz (1999).