AUDIO-HAPTIC INTERACTION WITH THE DIGITAL WAVEGUIDE BOWED STRING

Stephen Sinclair, Gary Scavone, Marcelo M. Wanderley

McGill University
CIRMMT, Montreal, Canada
{sinclair,gary}@music.mcgill.ca, marcelo.wanderley@mcgill.ca

ABSTRACT

In this work we investigate an adaptation of digital waveguide physical modeling of the bowed string for use with force-feedback haptic interaction. A method to interface the waveguide with a friction algorithm designed for haptic devices is presented, and execution on sample-accurate and multi-rate hardware architectures is described.

1. INTRODUCTION

Simulating the bowed string in real-time has been a popular topic in sound synthesis. There exist several works where haptic interaction has been used to provide a more intimate sense of control over the simulation. Some of these [9, 8] have used a stick-slip variation of Hayward and Armstrong's friction model (H-A) [5]. This model is a reformulation of friction to depend on position instead of velocity for use in sampled systems like haptic devices. To avoid differentiating the signal with respect to time, it models static friction as a restorative force between an anchor position, w_k, established on first contact, and the sampled end effector position, x_k. The anchor is incrementally "dragged" in the direction of movement according to a parameter α.

$$z_k = x_k - w_k \tag{1}$$

$$w_k = w_{k-1} + \alpha(z_k)\,|x_k - x_{k-1}|\,z_k \tag{2}$$

The simplest choice of α, a discontinuous function based on a small distance $Z_{\max}$, ensures that the position is never differentiated when the velocity is close to zero:

$$\alpha(z) = \begin{cases} \frac{1}{Z_{\max}}, & |z| > Z_{\max} \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

This produces a stick-slide behaviour. Hayward and Armstrong [5] also describe other choices of α, which can be used to introduce a stick-slip duty cycle regime. Both aforementioned bowing simulations [9, 8] exploited this stick-slip behaviour to model the haptics of bowing. This haptic model was accompanied by a digital waveguide sound synthesis running in parallel, which took velocity and pressure information from the haptic interaction.

However, O'Modhrain [9] found that, surprisingly, players performed worse when presented with the haptic friction model, as opposed to feeling only the normal forces. She hypothesized that this was due to the uncanny valley: that the friction was not realistic enough to promote skill transfer from the real bowing experience to the simulated, but that it was close enough to be confusing.

Perhaps one way to explain this lack of realism is to say that there was not a high correlation between the feel of friction and the sound of the simulation. There would certainly be some perceived connection between the two modalities, since the volume and timbral characterics of the sound are affected in real-time by the haptic interaction, and some oscillatory behaviour may also leak through from the haptic forces to the velocity parameter of the waveguide model. However, after implementing this configuration, we found that the feeling was of somehow *enveloping* the parameters of the sound model through gesture, rather than being truly integrated into the dynamics of the system. It is important to notice that the stick-slip friction used to excite the sound synthesis model, which was based on a bow table, was independent of and different from the haptic algorithm, which used the H-A technique.

An alternative approach was presented by Florens [4]. Instead of having separate threads of computation for haptics and sound, Florens made use of technology developed at ACROE, INPG, to produce both sound and haptics on the same processor in a single computational loop. Originally the haptic portion was synchronously updated at a lower speed (3 kHz), but later, speeds of 44 kHz were achieved using this technology [7]. The physical model in this case was built using the CORDIS-ANIMA formalism, a language for defining mass-spring networks that allows non-linear links [2]. The "bow" mass was connected to the "string" masses through a velocity look-up table based on a classical Stribeck friction curve. This caused a bowing sound to emerge from the resulting stick-slip friction.

Since this showed that high-frequency, synchronous computation could provide an enhanced audio-haptic interactive experience [7], we decided that similar success should be achievable with waveguide models as long as the audio and

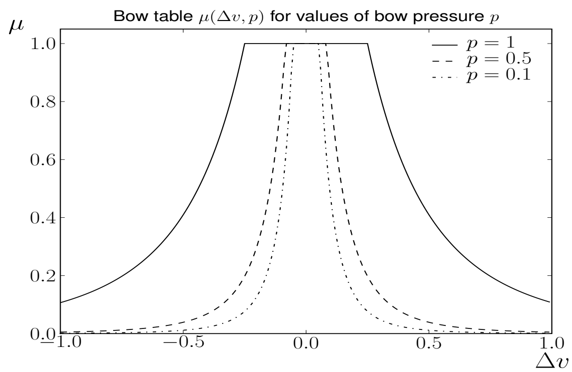

Figure 1. The STK bow table.

haptic components are consistent and synchronized. One the one hand, this might be achieved by using the H-A stick-slip model to excite the waveguide, bypassing the bow table, or on the other, by using the bow table to produce haptic friction. In this paper we describe the latter approach, albeit using the H-A stick-*slide* technique to smooth the interface between end effector and bow table.

2. DIGITAL WAVEGUIDE BOWING

Digital waveguide modeling (DWM) is a computationally efficient method for the physical modeling of 1-D systems that has been used extensively for real-time sound synthesis. It has been mathematically formalized by Smith [12] and others and has been shown to be a viable real-time technique for approximating non-linear interactions with linear resonating structures.

In the digital waveguide bowed string, wave propagation on the string is modeled using bidirectional delay lines. A time-varying scattering junction represents the non-linear bow-string interaction, dividing the string into two sections, and is implemented with an efficient look-up table [11]. In the approach of Smith [11], the "bow table" is a static, memoryless map between an input differential velocity (bow - string velocity) and a reflection coefficient. An example is shown in Fig. 1. The reflection coefficient defines the portion of string wave energy reflected and transmitted at the bow-string junction. A basic implementation of this model is represented by the block diagram of Fig. 2 and is available in the Synthesis Toolkit in C++ (STK) [3].

In the flat "sticking region", the string velocity is equated with the bow velocity, since they are stuck together. When the velocity difference exceeds the flat region, the string becomes unstuck and kinetic friction takes over. As this process repeats, a sawtooth movement naturally arises due to this non-linearity.

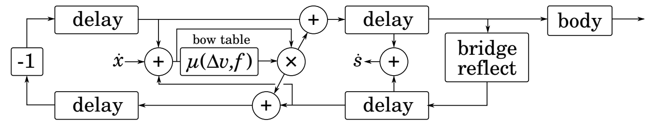

Figure 2. Waveguide for bowed string, from Smith [11].

3. HAPTIC AND SOUND EQUIPMENT

We used a TGR device from Ergos Technologies[1], which is controlled by the Toro-16 DSP board from Innovative Integration. This hosts a Texas Instruments TMS320C6711 digital signal processor, which can perform real-time single-sample floating-point computation with analog input and output. The STK Bowed model executed at 16 kHz.[2] Since this was within the audio range and well above the usual 1000 Hz used with most haptic devices, we proceeded with this result. Sound was produced by using an extra analog channel from the DSP board connected to a mixer and BM-15 Dynaudio speaker. We used a 2-degree-of-freedom manipulator attachment for the TGR.

As mentioned below, we also tested the algorithm on some commercial devices which lacked a dedicated DSP, in order to see how performance would compare in these circumstances.

4. INTERFACING WITH THE BOW TABLE

The bow table output can be considered as an absorption coefficient for the bow, where a value of $\mu = 0$ allows total transmission between the two halves of the waveguide, and a value of $\mu = 1$ absorbs all wave energy into the bow. In the sound model, this is used to determine how the string velocity propagates, but we can also consider how it determines the resistance experienced by the bow. During stick, where $\mu = 1$, the bow should be restricted from moving, and otherwise it should experience some friction force in opposition to the direction of movement. Thus, the value $(1-\mu)$ can be seen as a transmission coefficient for the bow, used with velocity to calculate friction force.

However, we know that using the velocity of the haptic device will give rise to problems. Calculating velocity from a sampled position will introduce a prohibitive amount of noise [5]. Furthermore, no matter what forces we apply, we must consider that we cannot actually stop the end effector from moving during stick since the user's hand can always push through the device's resistance.

A solution is to introduce a virtual bowing point b which we can control completely, and use this to calculate friction

[1] This is the same hardware as used by e.g. [7], and several other works from ACROE.

[2] We believe this could be improved significantly by applying processor-specific code optimizations.

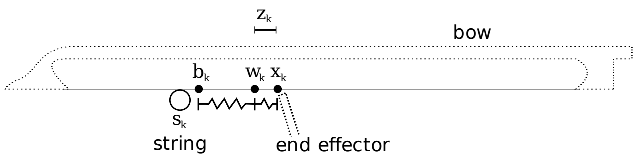

Figure 3. Location of each variable in a physical context.

forces. This allows to model friction as a spring between b and the end effector position x, similar to H-A friction, and to restrict the motion of b during stick:

$$\begin{aligned} b_{k+1} &= b_k + (x_k - b_k)(1-\mu_x) \\ &= x_k - (x_k - b_k)\mu_x \end{aligned} \tag{4}$$

where $\mu_x = \mu(v_x - v_s, p)$, p is the downward bow pressure, and v_x and v_s are the end effector and string velocities, respectively. We can get the desired friction behaviour by modulating friction force F_f by μ_x and the bow pressure as

$$F_f = (x_k - b_k)p\mu_x. \tag{5}$$

Increased bow pressure scales the bow table vertically, but the maximum remains at 1, effectively elongating the sticking region. Specifically, the STK bow table implementation (Fig. 1) provides:

$$\mu(\Delta v, p) = \min(\left[|\Delta v(5-4p)| + 0.75\right]^{-4}, 1) \tag{6}$$

Physically, the presence of this spring is not unnatural: when the hair is stuck to the string, there is some compliance, as the string itself acts as a spring. Additionally, the hair is not rigid and can compress in the longitudinal direction. For example, Adrien [1] suggested modeling the bow dynamics by a mass-spring damper system.

However, the problem remains that we are using v_x to calculate μ_x, which we stated is undesirable due to noise. We could use v_b instead, but unfortunately, this creates a circular dependency since then the motion of b would depend on its own velocity. To solve this, we introduce another virtual coupling to replace the end effector velocity with a virtual velocity. The point w will sit within a very short maximum distance $Z_{\max}$ from x and only move when this distance is exceeded. Thus, w acts as the H-A anchor point for very small velocities, but is quickly superceded by the bow friction as velocity increases. A diagram depicting each point is shown in Fig. 3.

Replacing μ_x with $\mu_w = \mu(v_w - v_s, p)$, we get a final set of equations.

$$z_k = x_k - w_k \tag{7}$$

$$w_{k+1} = \begin{cases} x_k - Z_{\max}\frac{z_k}{|z_k|} & |z_k| > Z_{\max} \\ w_k & |z_k| \le Z_{\max} \end{cases} \tag{8}$$

$$b_{k+1} = w_k - (w_k - b_k)\mu_w \tag{9}$$

$$F_f = (x_k - w_k)p\mu_w + (w_k - b_k)p\mu_w \tag{10}$$

$$= (x_k - b_k)p\mu_w \tag{11}$$

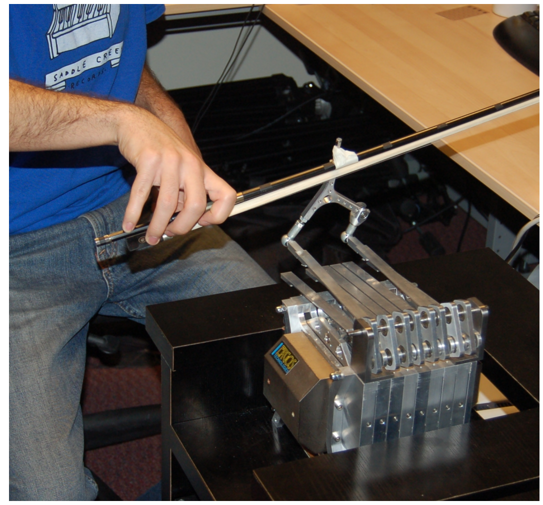

Figure 4. Using a real bow as end effector attached to the Ergos TGR.

The constant $Z_{\max}$ should be set to a value just outside the position signal's noise level, so that it is easily overcome by the sticking region of μ_w as p increases. Equations 10 and 11 show that the H-A friction and bow friction sum together as a combined spring to create the friction force.

Note that while the use of a virtual point helps to reduce noise during low velocity gestures, its velocity is estimated when it moves, and thus this algorithm requires a good velocity estimator. We used an adaptive windowing technique described by Janabi-Sharifi et al. [6].

5. PLAYING IMPRESSIONS

While simple, the feeling of bowing exhibited by the proposed model does informally bear a convincing resemblance to the actual experience. Throughout development we have compared with a real violin and cello by applying various bow forces, and we feel that there are definite similarities between the sticking and oscillatory behaviours of the real vs. simulated bowing experience. Bowing gently creates a familiar vibration, while pressing harder can create a crunchy sound and feel.

However, it is clear that several physical phenomena are missing from the current implementation and the effect of these is more apparent in haptics than in the sound. For instance, the model feels somewhat "dry". We attribute this to the lack of an advanced friction model, taking into account elasto-plastic dynamics and the effect of rosin, such as described by Serafin [10]. Other phenomena such as torsion waves, string stiffness, sympathetic vibrations in the other strings, and a better body response may help to increase the perception of realism.

Another consideration is the frequency response of the bow itself. Not all frequencies travel all the way from the hair-string interaction point to the player's right hand without considerable damping. However, applying a basic filter to the haptic output might degrade the overall system feedback. We have tried simply attaching a bow to the device's end effector, pictured in Fig. 4, which considerably improved the user experience in the opinion of at least two experienced string players.

6. MULTIRATE CONSIDERATIONS

For full fidelity, it is necessary to execute the sounding model and the haptic servo-loop as different outputs of a single, synchronous loop, as we have done with the Toro and TGR hardware. However, it is possible to gracefully degrade haptic output for execution on lower-frequency systems, while keeping high quality sound synthesis.

For example, we have used the MPB Technologies Freedom 6S device in one instance, and the Novint Falcon device in another. In these cases the sound model was executed at 48 kHz on the CPU while the haptic device was updated synchronously at 1000 Hz. For down-sampling, μ is simply averaged over the sample period. (While p and v_x remain constant, v_s changes throughout the period.) It is also possible to use maximum or minimum μ, which feels harsher or softer, respectively. This gives an impression of bowing, with synchronized haptic osillations, though the haptic experience feels like a lower frequency than the sound. The use of a spring for friction works well to guarantee stability even at lower frequencies. Note that in this configuration, the audio may be slightly delayed due to the computer sound card's output buffer, so a low-latency operating system should be used.

7. CONCLUSION

We have discussed a technique for applying H-A friction to the excitation portions of a waveguide bowing model for interfacing with a haptic device. In future work, we plan to include several more realistic physical phenomena, such as elasto-plastic friction, torsion waves, and stiffness. We would also like to apply this technique to other waveguide models.

8. ACKNOWLEDGEMENTS

Special thanks to Avrum Hollinger and Erika Donald, violin and cello players, for testing early versions of the work and providing extensive comments and ideas. This work has been funded by a grant from the Natural Sciences and Research Engineering Council of Canada (NSERC) and MPB Technologies Inc. to the first author, as well as by a grant from the NSERC "Enactive" special research opportunity.

9. REFERENCES

[1] J. M. Adrien, "Étude de structures complexes vibrantes, applications à la synthèse par modèle physique." Ph.D. dissertation, Université Paris 6, 1988.

[2] C. Cadoz, A. Luciani, and J. L. Florens, "CORDIS-ANIMA: a modeling and simulation system for sound and image synthesis—the general formalism," *Comp. Mus. J.*, vol. 17, no. 1, pp. 19–29, 1993, MIT Press.

[3] P. R. Cook and G. Scavone, "The synthesis toolkit (STK)," in *Proc. of the Int. Comp. Music Conf.*, 1999, pp. 164–166.

[4] J.-L. Florens, "Expressive bowing on a virtual string instrument." in *Lecture Notes in Computer Science: Gesture-Based Communication in Human-Computer Interaction, 5th International Gesture Workshop*, A. Camurri and G. Volpe, Eds. Genova, Italy: Springer Berlin / Heidelberg, April 2003, vol. 2915, pp. 487–496.

[5] V. Hayward and B. Armstrong, "A new computational model of friction applied to haptic rendering," in *Proc. of The Sixth Int. Symp. on Exp. Robotics VI*. London, UK: Springer Verlag, Mar. 2000, pp. 403–412.

[6] F. Janabi-Sharifi, V. Hayward, and C.-S. J. Chen, "Discrete-time adaptive windowing for velocity estimation," *IEEE Trans. on Cont. Sys. Tech.*, vol. 8, no. 6, pp. 1003–1009, Nov. 2000.

[7] A. Luciani, J.-L. Florens, D. Couroussé, and C. Cadoz, "Ergotic sounds: A new way to improve playability, believability and presence of digital musical instruments," in *Proc. of Enactive '07*, Grenoble, France, Nov. 2007, pp. 373–376.

[8] C. Nichols, "The vBow: a virtual violin bow controller for mapping gesture to synthesis with haptic feedback," *Org. Sound*, vol. 7, no. 2, pp. 215–220, 2002.

[9] S. M. O'Modhrain, "Playing by feel: Incorporating haptic feedback into computer-based musical instruments," Ph.D. dissertation, Stanford University, Nov. 2000.

[10] S. Serafin, "The sound of friction: real-time models, playability and musical applications." Ph.D. dissertation, Stanford University, CCRMA, June 2004.

[11] J. O. Smith, "Efficient simulation of the reed-bore and bow-string mechanisms," in *Proc. of the Int. Comp. Music Conf.* The Hague: CMA, 1986, pp. 275–280.

[12] ——, "Physical modeling using digital waveguides," *Comp. Mus. J.*, vol. 16, no. 4, pp. 74–91, 1992.