Noise-Free Haptic Interaction with a Bowed-String Acoustic Model

Stephen Sinclair*
Input Devices and Music Interaction Laboratory, CIRMMT, McGill University

Marcelo M. Wanderley†
Input Devices and Music Interaction Laboratory, CIRMMT, McGill University

Vincent Hayward‡
UPMC Univ Paris 06, UMR 7222, Institut des Systèmes Intelligents et de Robotique, F-75005, Paris, France

Gary Scavone§
Computational Acoustics Modeling Laboratory, CIRMMT, McGill University

ABSTRACT

Force-feedback interaction with a bowed string model can suffer critically from noise in the velocity signal derived from differentiating position measurements. To address this problem, we present a model for bowed string interaction based on a position-constraint friction. We validate the proposed model by comparing to previous work using off-line simulations, and show measurements of interaction on haptic hardware. This noise-free excitation signal leads to cleaner string motion than previous models, thereby improving the quality of force and audio synthesis in the presence of noise.

1 INTRODUCTION

There are several choices available to combine haptic force-feedback interaction with audio feedback, each affording different levels of immersive experience.

A common approach, particularly in interaction with rigid-body virtual environments, is to use collision events to trigger sounds. These sounds may simply be pre-recorded sound files, or they may be parameterised on the colliding objects' physical parameters using techniques such as, e.g., modal [6] or granular synthesis [1]. For continuous sounds, stochastic or other continuous signals can be generated to trigger events and modulate sound parameters [17].

In an example of musical interaction, friction has been simulated in the haptic medium, and the device velocity used to control a bowed string synthesis model [11]. In these approaches, the haptic and audio interaction are decoupled. Information flows exclusively from the haptic simulation to the audio synthesizer, and these simulations are not synchronous and do not execute at the same rate. The vibrations of simulated objects, responsible for sound generation, are not felt.

In contrast, Florens describes a synchronous technique where haptic forces and sound feedback are generated from the same model [8]. We shall call this approach *acoustic interaction.* Although humans do not have motor control in the acoustic range, which goes up to 20 kHz or more, non-linear excitation, giving rise to higher harmonics, does operate in this range. Therefore, simulating the system at the acoustic rate allows a complete synchronization and coupling of the two modalities. However, this coupling comes at a price: the haptic device's own mechanical and electrical properties are in the loop that controls sound synthesis, meaning that non-ideal characteristics such as inertia, friction, and noise will affect the behaviour of the physical model.

In this article, we focus on the issue of noise which is highly significant in the acoustic and in the haptic domains alike. While noise does not tend to be a problem with sound-generating striking gestures that depend on position, rubbing and bowing gestures involve forces that depend on velocity. Velocity estimation, especially at high frequency, tends to be noisy; any position noise from sensors or encoders, as found in most force feedback devices on the market, is accentuated by differentiation and this noise enters the synthesis loop. While higher-order estimation techniques can improve the situation [3, 10], noise is nonetheless emphasized when a high sampling frequency is employed and is particularly problematic if it is present in the signal path to the loudspeaker.

One example where this issue becomes critical is in the bowed string. In this situation, the bow and the string are coupled by a non-linear viscous relationship which determines their relative sticking and slipping behaviour [15]. In a standard model of bow-string friction, a function—the *friction curve*—relates differential velocity between the string and the bow to friction force, see Fig. 1. This function slopes downward after an initial maximum, such that a higher velocity difference produces a decrease in friction force. This allows the string to slip away from the bow, and remain detached until their relative velocities are low enough that friction causes sticking, allowing the bow to pull on the string once more.

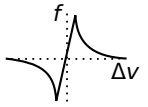

Figure 1: Standard friction curve relating bow velocity to friction force.

The friction curve models the effect of a substance on the bow called rosin—we call this the 'standard' model, because, as explained in [15], it is now known that hysteretic behaviour arises from complex thermal dynamics induced by the presence of rosin. We do not address this hysteresis here, but it has been shown that the friction model used in the current work [9], described in Sec. 2.2, can also give rise to a similar hysteresis [12].

Since end-effector velocity is needed to control the coupling between string and bow, any noise present in the velocity signal can potentially be found in the string motion. The velocity noise margin also has implications for the stability of viscous rendering, forcing the use of a diagonal portion of the friction curve in the region close to $\Delta v = 0$, Fig. 1, which leads to unrealistic sliding at low velocity.

Even if such noise were not felt in the force signal, much of it occuring above haptic frequencies, in musical applications such as this the string velocity is played through a loudspeaker, making it much more apparent. This effect is sufficiently debilitating that even with a high quality system with 16-bit analog conversion, spontaneous generation of audible noise can occur without the end effector being touched. Counterintuitively, the situation worsens as temporal and spatial resolution are increased, since lower resolution in both domains is effectively equivalent to averaging noise.

In this paper we propose a model in which stick-slip state is determined by proximity of the integrated string position to the bow instead of requiring a sampled velocity signal. We first describe previous bow-string models used in haptics and sound synthesis. We then describe a proposed model based on a position-constraint implementation of friction which is more robust for force-feedback

*e-mail: sinclair@music.mcgill.ca
†e-mail: marcelo.wanderley@mcgill.ca
‡e-mail: hayward@isir.upmc.fr
§e-mail: gary.scavone@mcgill.ca

interaction. Finally, we discuss offline simulations comparing the current model to previous work, and show an example of virtual, online bowing with a force-feedback device.

2 Previous work

2.1 String models

To our knowledge, the first implementation of acoustic-haptic interaction with a simulated bowed string employed a force-driven modal mass-spring model [8]. The string is a set of tuned masses, which interact bidirectionally through a force-velocity function with a mass, which is in turn coupled viscoelastically to the haptic end effector position. Although this link filters much of the noise present in the velocity, this comes at the expense of bandwidth. When the link is stiffened, the velocity noise becomes evident.

In the sound synthesis literature, a common approach for real-time bowed string synthesis involves the use of digital waveguide (DWG) techniques [16]. In the DWG model, the string resonator is an efficient time-domain model (a delay line + filters to model loss) that propagates a bidirectional velocity wave. The differential velocity, $\Delta v = v_b - v_s$, where v_b is the bow velocity and v_s is the string velocity, is used to index the bow-string reflection coefficient μ, usually called the "bow table", which determines velocity propagation at the bow-string scattering junction. The reflection coefficient μ equals one for a region around $\Delta v = 0$, where the string and the bow are assumed to be stuck together. In this region, the string takes on the bow's velocity and internal energy in the string is reflected; outside this region, μ drops exponentially toward zero so that energy can propagate through the junction.

In this paper, we refer to offline simulations on a DWG model based on an implementation available in the *Synthesis ToolKit* [4], and described in Fig. 2. This model can be connected to a haptic device, with incoming velocity calculated from force f_s on the string and the string's wave impedance [2], as shown in Fig. 3. However, this leaves noise removal up to the velocity estimator; velocity along with any residual noise is transmitted by a viscous link, and suffers from similar issues to [8] when the bandwidth is increased.

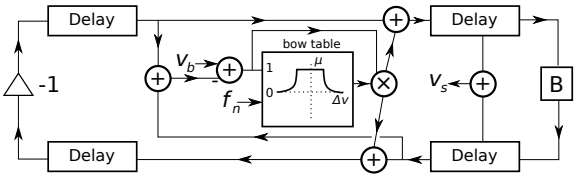

Figure 2: The digital waveguide implementation of a bowed string, from [16]. B is a filter modeling loss in the resonator at the bridge. Energy is perfectly reflected at the nut, indicated by a gain of -1.

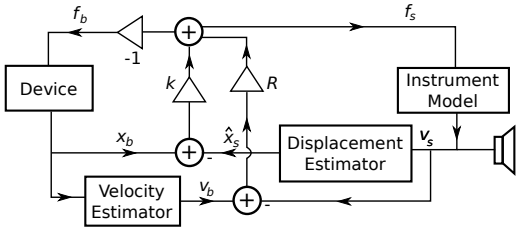

Figure 3: Interfacing an instrument model to a haptic device, from [2].

Please note that comparisons in this paper, which focus on the DWG technique, do not include the Florens model [8] because it uses a different approach for both resonator and excitation. We prefer to keep the resonator consistent and thus compare only different excitation models. A qualitative comparison of the Florens model with the model described in Sec. 2.3 is available in [13].

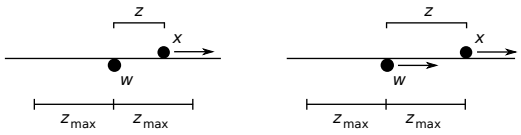

Figure 4: Left: Object position x moving away from adhesion point w and towards the z_{max} boundary. Right: x has reached the boundary, so w also moves with the same velocity, providing a sliding friction regime.

2.2 Position-constraint friction

In [9], a general friction model which we refer to as H-A friction was proposed that is robust to velocity noise, and therefore performs well for haptic rendering. It is based on an expression of Dahl friction in terms of displacement instead of a time derivative. Briefly, Dahl's model [5], expresses friction in terms of two points: x, the position of a moving object, and w, the point of adhesion to the surface. The difference between these points, $z = x - w$, represents strain or micro-movement that models pre-sliding during static friction. This small amount of energy is stored due to elastic properties of the materials. When z exceeds a threshold z_{max}, excess energy is released as the contact point changes, representing a dynamic friction regime, see Fig. 4.

In [9] it is observed that in one version of Dahl's model, written,

$$v_z = v_x(1 - \alpha \operatorname{sgn} v_x\, z) = v_x - v_w,$$

the role of velocities v_x, v_w, and v_z are only to express that during sliding, the x and w points move at the same rate, so $v_z = 0$. The same idea can be expressed in terms of position by stating that during sliding, $x = w \pm z_{max}$ and $|z| = z_{max}$.

Thus, w is a point established on initial contact ($w_0 = x_0$), and $z = x - w$. As x moves away from w, $|z|$ approaches z_{max}. When it crosses this boundary, w is adjusted to be on the border of $x \pm z_{max}$. Establishing a friction force $f = -kz$ provides a smooth sensation of static and sliding friction. More generally,

$$\Delta w = \alpha(z) z \Delta x,$$

where α is a function specifying friction charateristics. A sliding friction as described can be chosen by setting $\alpha = 1/z_{max}$ when $|z| > z_{max}$, and zero otherwise.

One assumption in this friction model is that w is a point on a static surface. In this work, we effectively attach w to a dynamic model, a string, which causes modulation of model forces based on z. This energy is delayed and reflected, eventually affecting later motion of w, which results in periodic oscillation.

2.3 Previous use of H-A friction for bow haptics

In [14], an alternative excitation mechanism is proposed for the DWG model which inserts a virtual point x_w, similar to an H-A anchor point, between the end effector position x_b and the bow-string contact position. This model will be referred to as TWOPOINT.

Like in Sec. 2.2, the x_w point only moves when the distance $x_b - x_w$ exceeds a small region z_{max} around the initial contact position. Additionally, x_w is further connected by an elastic link to a point whose movement is controlled by the reflection coefficient μ, providing a grounded link during sticking and free movement during slip. This μ-controlled point will not be further discussed here since it only affects force feedback, which is not considered in our use of this model for offline comparisons.

Since x_w does not move until x_b exceeds a certain proximity, taking the resonator input velocity as v_w instead of v_b allows it to be exactly zero when the end effector is not moving sufficiently, which alleviates the problem of spontaneous noise when the operator is

not moving or touching the handle. However, during movement, the velocity of v_x is perfectly transmitted to v_w; although it is somewhat masked by the sound of the string, noise is nonetheless still clearly audible (and felt!) when the velocity is non-zero, giving the sound and feel a dry, chalky quality.

3 DISTANCE-BASED BOW-STRING FRICTION

In the current work, bow-string friction is again synthesized using an H-A friction model, however the anchor point is chosen to be related to the string position, tracked by a constrained integrator, and sliding is used to determine energy release into the string model. We refer to this model as DISTPLUCK, since the distinguising characteristic is the use of relative distance to trigger impulses.

We call the anchor point x_s, representing the string position constrained to bow movement. The end effector is considered as the bow position x_b. Very small bow movements lead only to pre-sliding, and do not allow the string to slip, but if the distance $z = x_b - x_s$ exceeds a small region $z_{\max}$, then the positions should be adjusted to ensure that $|z| < z_{\max}$. This proximity constraint models the fact that the bow contact position changes during slip.

Either x_s or x_b could be displaced to constrain z; we choose to adjust x_s since this can be used to simultaneously track bow slip and string motion, minimizing the number of state variables. When slip occurs, the bow experiences a sliding friction in opposite direction to bow movement. The sliding friction force is related to the normal force component by:

$$f_b = -f_n z,$$

where f_n is the normal force component. We obtain f_n by applying a virtual wall model $f_n = kx_n$ when $x_n < 0$, $f_n = 0$ otherwise, on the vertical axis orthogonal to the bowing direction, such that the device displays the penalty force $-f_n$ in the same axis.

Additionally, the string is affected by restoring force f_r. Corresponding velocity $v_r = (f_r/m)\Delta t$ is added to the string, where m is the linear mass of the string. This force has the same sign as bow friction, opposite to bow motion. When sliding occurs, v_r must describe the release velocity of the string, which, as will be shown, depends on its deflection ΔL.

Note that due to the constraint on x_s, we are not tracking string position at the bowing point, but rather the constraint ensures that z is proportional to bow-string tension. Therefore the full deflection of the string is not known. However, release occurs at maximum tension, when $|z| = z_{\max}$. We now assume that maximum tension corresponds to a maximum deflection of the string $\Delta L_{\max}$ which is proportional to $z_{\max}$. We define a proportion λ, such that $\Delta L_{\max} = \lambda z_{\max} f_n$. The string's maximum deflection is related by λ to maximum tension and normal force. We can essentially choose λ such that it scales impulses of f_r as desired, since this is similar to choosing the friction coefficient of the bow-string contact.

Restoring force for small string deflections is the tension T times the curvature of the string $\sin(\theta) = \Delta L/L\beta$, where β represents the position of the bow-string junction as a fractional of the string length. The total restoring force f_r is the sum of both sides,

$$f_r = \frac{\Delta LT}{L\beta} + \frac{\Delta LT}{L(1-\beta)}.$$

Substituting the string tension $T = mL(2F_0)^2$, and $\Delta L = \lambda z f_n$,

$$f_r = \frac{m(2F_0)^2}{\beta - \beta^2},$$

$$v_r = -\lambda z \frac{(2F_0)^2 \Delta t}{\beta - \beta^2} f_n, \text{when } |z| \geq z_{\max}.$$

The simulation parameters are F_0, Δt, λ, and β, which can be lumped together as $\gamma = 4\lambda {F_0}^2/(\beta - \beta^2)$, giving

$$v_r = -\gamma z f_n, \text{when } |z| \geq z_{\max}.$$

This added velocity, whose amplitude during impulses will be effectively equal to γ, is propagated in the wave, and eventually pulls the x_s integrator in and out of the $z_{\max}$ region when it reflects, creating stick-slip motion. Fig. 5 shows the resonator input signal generated by friction force developing into a series of periodic plucks. The shape of impulses in $v_r(t)$ are square; therefore, no sampling noise is transmitted to the resonator; only some aliasing noise may be present.

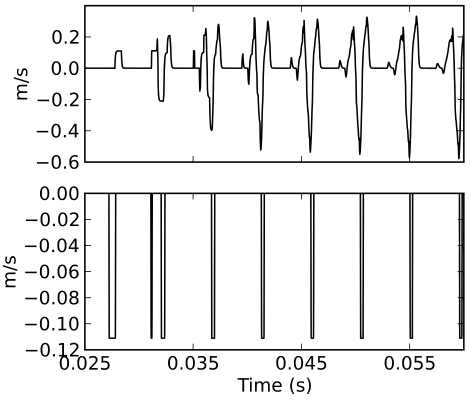

Figure 5: String velocity v_s (top), resonator input signal v_r (bottom).

As with H-A friction, the $|z| \geq z_{\max}$ condition can be more elegantly taken into account using α. A difference from H-A friction is that after integrating string motion, $|z|$ can temporarily end up some variable distance beyond $z_{\max}$. Therefore instead of integrating v_x as in the H-A model, it is necessary to adjust x_s by $z - z_{\max}\operatorname{sgn} z$ whenever $|z| \geq z_{\max}$. This implies the following update to x_s:

$$\Delta x_s = \alpha(z) z_{\max}(z - z_{\max}\operatorname{sgn} z) + v_s \Delta t.$$

That is, x_s is moved closer to $x_b \pm z_{\max}$ by a proportion determined by α, plus the string's change in position.

Finally, v_r is also scaled by $\alpha(z) z_{\max}$ to indicate that the string responded to a ratio of the full restoring force, making the $|z| \geq z_{\max}$ condition implicit:

$$v_r(t) = \alpha(z)\, z_{\max} \gamma z f_n.$$

Note that since $v_r(t)$ does not include a term to subtract propagated velocity between the two parts of the waveguide, string energy is always transmitted. Therefore DISTPLUCK does not model reflection during the sticking state as in DWG. A block diagram of the described algorithm can be found in Fig. 6.

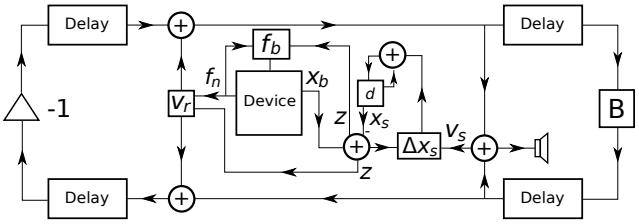

Figure 6: The DISTPLUCK model with digital waveguide resonator. The formulas for Δx_s, v_r and f_b are shown as unexpanded functions here, and are defined in Sec. 3. The d block prior to x_s is a unit delay. The device provides position x_b and normal force f_n; the latter is determined by a simple virtual wall model in the vertical axis, not shown here.

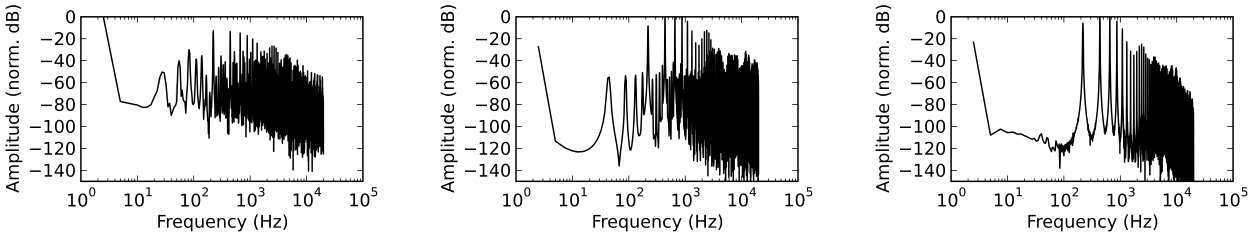

Figure 8: Spectra for steady state signals using the TWOPOINT model and two configurations of the DISTPLUCK model, F_0=220 Hz, simulated in clean conditions. It can be seen that the rounded α for DISTPLUCK provides an anti-aliasing effect.

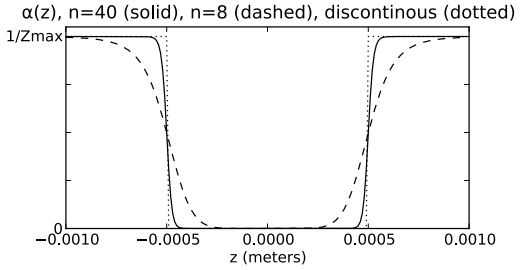

Figure 7: Curves denoted by different versions of $\alpha(z)$.

3.1 Partial slip provides anti-aliasing

Describing dynamic behaviour in terms of proportions based on $\alpha(z)$ allows consideration of other choices for friction, enabling modeling of partial slip.

For instance, [9] considers a *stick-creep-slip-slide* configuration, where $\alpha(z)$ slowly approaches $1/z_{\max}$ instead of being discontinuous:

$$\alpha(z) = \frac{1}{z_{\max}} \frac{z^8}{{z_{\text{stick}}}^8 + z^8},$$

where z_{stick} is a region just inside $z_{\max}$ defining the boundary between stick and slip. Here, we are not considering the contact presliding stick-slip motion considered in [9], since net displacement is, in the case of bow-string contact, dominated by the string behaviour. However, the *creep* region represents a valuable physical effect because it establishes a dissipative regime where the bow and string may be partially slipping, comparable to the slope on the sides of the DWG reflection coefficient.

For the present purpose of simulating rosin-like friction the slope is not sufficient, creating a small velocity bias that fails to create a stick-slip behavior. The function can be generalized to tighten the slope, dropping z_{stick}:

$$\alpha(z) = \frac{1}{z_{\max}} \frac{z^n}{{z_{\max}}^n + z^n},$$

taking n to high values such as 40 or more, see Fig. 7. This function more closely approaches a discontinuous α, but gives an initial velocity bias that builds up to a series of impulses more gradually, increasing the attack time. The pulses themselves are more rounded, providing significantly more roll-off towards the Nyquist frequency which helps avoid spectral fold-over, see Fig. 8. To allow for high-rate real-time execution, a piece-wise linear approximation of the rounded α is used in both the offline and online simulations, with the curved regions within positive and negative $z_{\max} \pm 0.01$ mm composed of 100 linear segments, and the rest set to 0 and $1/z_{\max}$ respectively.

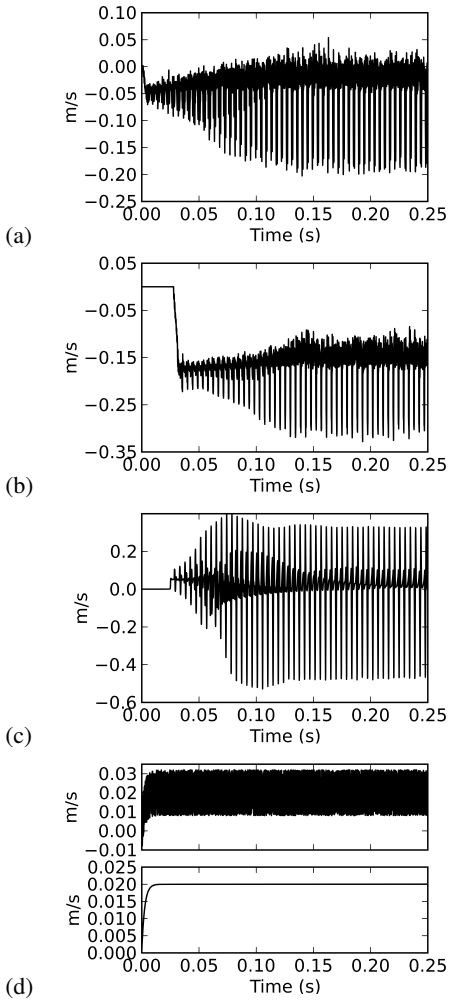

Figure 9: String velocity, v_s, in noisy condition for models (a) BOWED, (b) TWOPOINT, (c) DISTPLUCK. (d) Simulation of bow velocity, v_s, from backward difference, noisy condition above, no noise below.

4 OFFLINE SIMULATIONS

The *Synthesis ToolKit*'s BOWED class, which is a straight-forward implementation of a DWG model, was compared to TWOPOINT, and DISTPLUCK models using offline simulations at 40 kHz under noise-free and noisy conditions. The input position signal is intended to mimic a bow stroke starting at zero and reaching a constant velocity of 0.02 m/s (Fig. 9d). In the noisy condition, sim-

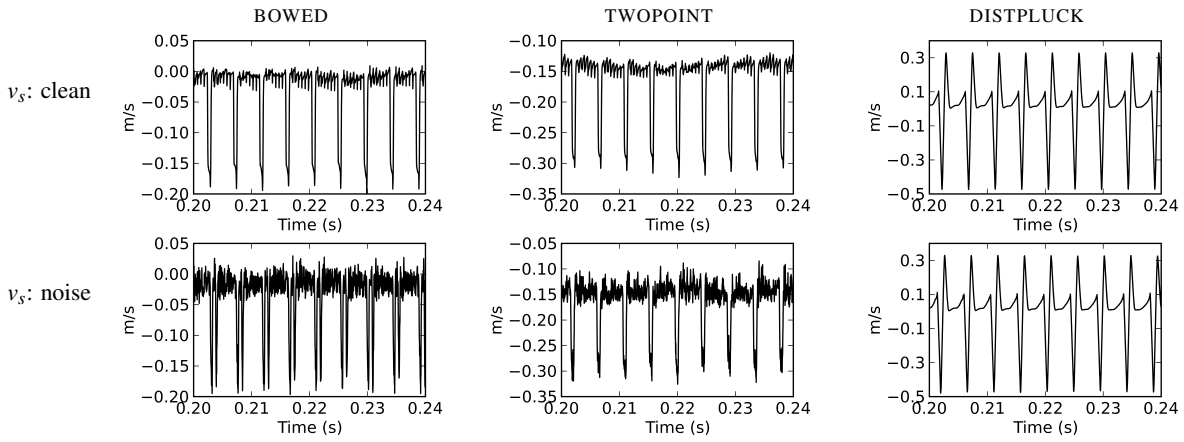

Figure 10: String velocity v_s during the steady-state of each model for clean and noise conditions. The input velocity is the same as that given in Fig. 9d.

ulated noise is added to the position signal with an amplitude of 0.6 μm. Velocity is derived by backward difference, leading to a velocity noise amplitude of 0.024 m/s.

Using a real haptic device, of course, the input would be affected by the displayed bow friction force. We have opted to simulate in open loop since this work is primarily affected by input signal quality: since the control signal is slow-moving, inserting a second-order device model would only low-pass filter the noise.

Normal force was held constant at 1.0 N, the fundamental frequency F_0 set to 220 Hz (open A3 cello string), and the value of $z_{\max}$ was 0.5 mm for all cases. The DISTPLUCK simulations used a smooth α with $n = 40$. For parameters affecting tension restoring force, $\beta = 0.87$, and $\lambda = 6 \times 10^{-8}$, giving $\gamma = 0.1$. Good results were achieved with $\gamma \approx 0.07$ to 1.0 or so—the exact value does not seem to be critical, though small values produce sliding friction instead of oscillation, and high values cause the simulation to diverge.

Figure 9 shows that the presence of the H-A model in both the TWOPOINT and DISTPLUCK cases has the effect of delaying the attack onset. This is because with an initial $x_s = x_b = 0$, then x_b must reach $z_{\max}$ before anything significant happens. As mentioned, this helps avoid spontaneous noise due to small changes in the end effector position signal when the handle is not visibly moving. Therefore DISTPLUCK shares this desired characteristic with the previous TWOPOINT model. The $z_{\max}$ region is kept small so that the user has the impression that it responds quickly to direction reversal.

Another property is that the DISTPLUCK model does not exhibit the same degree of DC bias. This is due to the fact that the string position is constrained to the bow position, and the harder the constraint, the less offset is imposed—the discontinuous choice for α creates zero bias. For smooth α, the bias disappears after the attack reaches steady state: the resonator input between plucks is zero, whereas the other algorithms center the oscillation around v_b. In general, DC-bias is not considered to be a problem because it would be filtered out by the instrument body, which is not modeled here.

In Fig. 10, notice the presence of noise in the steady state of BOWED and TWOPOINT as evidenced by the clear differences between clean and noisy conditions. In comparison, the string velocity for DISTPLUCK is almost identical in both conditions. In BOWED and TWOPOINT there is also some amount of high-frequency vibration between impulses even in the clean condition that is not present in DISTPLUCK. This is circulation in the shorter of the two delay line portions due to reflection at the bow-string junction during stick, which is not modeled in DISTPLUCK.

5 Online simulations

Fig. 11 shows a recording of a bow stroke using DISTPLUCK on a haptic device. The device used was the ERGOS hardware developed at ACROE [7], controlled by a Toro-16 board (Innovating Integration) which features a TI 320C6711 floating point DSP with 16-bit ADC/DAC converters triggered by a variable-rate internal interrupt clock. The device has linear actuators coupled by a parallel mechanism and position is sensed by linear variable differential transformers (LVDT). The algorithm was executed at 30 kHz, and all parameters were the same as described in Sec. 4. The fourth and fifth rows of Fig. 11 show the substantial difference in signal-to-noise ratio between the detected position and a backward-difference velocity estimate.

We can see that the commanded friction force oscillates at similar frequencies to the string. The haptic device's inertia inevitably filters much of this; we are simulating acoustic vibration at 30 kHz, but the device's frequency response is linear to no more than 350 Hz [13]. Therefore we could safely sample the device less frequently than the audio computations without going below the device's maximum responsiveness, limited mostly by stability concerns. This would allow to widen the velocity estimator's window to decrease noise. However, some noise would still be present, which would be heard and felt in the BOWED and TWOPOINT models, and this would vary between different devices that feature more or less inertia and force capabilities. Fig. 11 demonstrates that the string velocity in DISTPLUCK is clean in these worst-case conditions where velocity information is essentially absent.

A phenomenon that became evident in online simulations which cannot be easily seen graphically is that there is a slight increase in pitch when the bow is lifted and the string resonates on its own. This is caused by discretization error: impulses occur after an exceeded distance, and therefore they are always determined late by one time step, causing the period to increase slightly as the bow is lifted from the string and the resonator is no longer driven. The change is approximately 0.5 to 1.5 Hz. Correcting for this is a topic still under investigation.

6 Conclusion

A model for haptic interaction with a bowed string was described that makes uses of forces derived from a position-constraint friction algorithm. This avoids the necessity of deriving a velocity signal from measured position. Clean impulses are generated based on relative displacement, leading to stick-slip behaviour. Off-line simulations and measured results confirm that the model exhibits the desired properties.

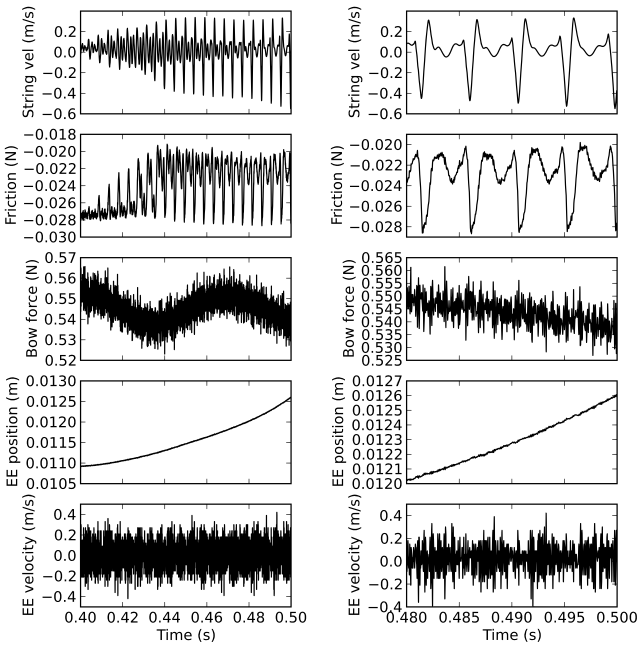

Figure 11: Two zoom levels of a recording of DISTPLUCK running at 30 kHz on a haptic device. From top to bottom, the plots are: string velocity v_s, friction force, normal force, horizontal position of end effector, and velocity estimated by backward difference.

Some limitations are the lack of reflection at the bow-string junction, and a small but audible change in pitch when the bow is lifted off the string. Both of these issues are currently under investigation and will be the subject of future work.

To more completely characterise this work, it is necessary to further examine the behaviour under known conditions in bowed string physics. There are several techniques that have been used to evaluate numerical models of bowed strings [18]: checking the model against Shelleng's playability diagrams, which characterise oscillation regimes in terms of force and β-ratio parameters; verifying the emergence of a "flattening effect" at certain parameters, described in [18] (a sudden drop in frequency due to hysteresis); and the characterisation of raucous motion at hard normal force, to name a few.

Since the quality of a sounding model can be hard to judge objectively, blind evaluations with professional string players should be performed to establish preferences under various conditions and fundamental frequencies. For instance, as mentioned, the human frequency discrepancy between modalities provides room to make use of velocity filtering in order to improve the estimate; since the bowed string naturally exhibits some noise due to the physical characteristics of the bow's horse hair, it is not known where the threshold of acceptability lies for the existence of noise in the resulting sound. This noise-free algorithm could provide a baseline for comparison. Such a threshold would provide a target error margin for velocity measurement that could be taken into consideration for future device development, simplifying the implementation of velocity-based effects.

7 Acknowledgements

This work was 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.

References

[1] S. Barrass and M. Adcock. Interactive granular synthesis of haptic contact sounds. In *Proceedings of the AES 22nd International Conference on Virtual, Synthetic and Entertainment Audio*, pages 270–277, Helsinki University of Technology, Espoo, Finland, June 2002.

[2] E. Berdahl, G. Niemeyer, and J. Smith, III. Using haptic devices to interface directly with digital waveguide-based musical instruments. In *Proceedings of the Ninth International Conference on New Interfaces for Musical Expression, Pittsburgh, PA*, pages 183–186, 2009.

[3] R. H. Brown, S. C. Schneider, and M. G. Mulligan. Analysis of algorithms for velocity estimation from discrete position versus time data. *IEEE Trans. on Indus. Elec.*, 39(1):11–19, Feb. 1992.

[4] P. R. Cook and G. Scavone. The synthesis toolkit (STK). In *Proceedings of the International Computer Music Conference*, pages 164–166, 1999.

[5] P. R. Dahl. Solid friction damping of mechanical vibrations. *AIAA Journal*, 14(12):1675–1682, 1976.

[6] D. DiFilippo and D. K. Pai. The AHI: An audio and haptic interface for contact interactions. In *Proceedings of the 13th Annual ACM Symposium on User Interface Software and Technology*, pages 149–158, 2000.

[7] J. Florens, A. Luciani, C. Cadoz, and N. Castagné. ERGOS: Multi-degrees of freedom and versatile force-feedback panoply. In *Proceedings of Eurohaptics*, pages 356–360, 2004.

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

[9] V. Hayward and B. Armstrong. A new computational model of friction applied to haptic rendering. In *Proceedings of The Sixth International Symposium on Experimental Robotics VI*, pages 403–412, London, UK, March 2000. Springer Verlag.

[10] F. Janabi-Sharifi, V. Hayward, and C.-S. J. Chen. Discrete-time adaptive windowing for velocity estimation. *IEEE Trans. on Cont. Sys. Tech.*, 8(6):1003–1009, Nov. 2000.

[11] C. Nichols. The vBow: development of a virtual violin bow haptic human-computer interface. In *Proceedings of the Conference on New Interfaces for Musical Expression*, pages 29–32, Dublin, Ireland, 2002.

[12] S. Serafin, C. Vergez, and X. Rodet. Friction and application to real-time physical modeling of a violin. In *Proceedings of the International Computer Music Conference*, pages 375–377, Beijing, China, 1999. ICMA.

[13] S. Sinclair, J.-L. Florens, and M. M. Wanderley. A haptic simulator for gestural interaction with the bowed string. In *10ème Congrès Français d'Acoustique*, Lyon, France, Apr. 2010. SFA.

[14] S. Sinclair, G. Scavone, and M. M. Wanderley. Audio-haptic interaction with the digital waveguide bowed string. In *Proc. of the International Computer Music Conference*, pages 275–278, Montreal, Canada, 2009.

[15] J. Smith and J. Woodhouse. The tribology of rosin. *Journal of the Mechanics and Physics of Solids*, 48(8):1633–1682, 2000.

[16] J. O. Smith. Synthesis of bowed strings. *The Journal of the Acoustical Society of America*, 71(S1):S101, April 1982.

[17] K. van den Doel, P. G. Kry, and D. K. Pai. FoleyAutomatic: physically-based sound effects for interactive simulation and animation. In *Proceedings of the 28th annual conference on computer graphics and interactive techniques*, pages 537–544, 2001.

[18] J. Woodhouse. Bowed string simulation using a thermal friction model. *Acta Acustica united with Acustica*, 89:355–368, 2003.