# MODELING INTERACTIONS BETWEEN RUBBED DRY SURFACES USING AN ELASTO-PLASTIC FRICTION MODEL

*Federico Avanzini* — Dip. Ingegneria dell'Informazione, University of Padova, Via Gradenigo 6/A, 35131 - Padova, Italy, avanzini@dei.unipd.it

*Stefania Serafin* — CCRMA, Dept. of Music, Stanford University, The Knoll, 660 Lomita, Stanford, CA, 94309, serafin@ccrma.stanford.edu

*Davide Rocchesso* — Dip. Informatica, University of Verona, Strada Le Grazie, 15, 37134 - Verona, Italy, rocchesso@sci.univr.it

**ABSTRACT**

A physically based model of the frictional interaction between dry surfaces is presented. The paper reviews a number of static and dynamic friction models, and discusses numerical techniques for the accurate and efficient numerical implementation of a dynamic elasto-plastic model. An application to the bowed string is provided, and the resulting simulations are compared to recent results from the literature.

## 1. INTRODUCTION

Friction, the tangential force between two surfaces in contact, needs to be minimized in most engineering applications, since its highly nonlinear behavior affects the performances of the systems. The effectiveness of model-based control strategies for friction compensation is ultimately limited by the ability of the models to accurately describe the dynamics of the physical system.

In an attempt to better predict friction phenomena at low velocities (i.e., stick slip, pre-sliding, frictional memory, etc.) researchers have recently developed dynamic models for friction. These dynamic models describe the dependence of friction on the relative velocity between the two sliding bodies through a differential equation. Dynamic models are able to take into account presliding behavior for very small displacements, where the friction force increases gradually with the displacement.

The first dynamic model was proposed by Dahl (see [9] for a review): it accounts for presliding displacement, but not for the Stribeck effect, i.e. the dip of the force at low relative velocities. The LuGre model [5] extends Dahl's work in order to include the Stribeck effect. However, this model exhibits drift for arbitrarily small external forces, which is not physically consistent. This effect has been explained in [6] by observing that LuGre does not allow purely elastic regime: therefore, a class of *elasto-plastic* models has been proposed in [6], where the drawbacks of LuGre are overcome.

To our knowledge, the attention on frictional phenomena in sound synthesis applications has been focused on the most common family of musical instruments whose main excitation mechanism is friction, i.e. the family of bowed string instruments. Theoretical models of the motion of a bowed string assume that the frictional force due to the bow rosin is only dependent on the relative sliding velocity. Smith and Woodhouse [12] have observed that the rosin exhibits plastic deformation at the bow contact point. For this reason they proposed a *plastic* model in which friction has a dependence on the temperature variations in the interfacial rosin layer. This model exhibits a hysteresis loop in the velocity versus force plane.

The sound synthesis algorithm described in this paper is based on the elasto-plastic model, already used in [7] for haptic rendering applications. This model has already been applied to a bowed string in [11]. However, in that application the authors chose the same numerical implementation used in [7], which introduces one sample delay to solve computational problems.

The model proposed in this paper is intended to provide a general description of the non-linear friction between two resonating objects. The numerical implementation provides efficient and accurate computation of the non-linear equation, and follows an object-oriented approach in which the resonating objects and their coupling are modeled in a modular way. As an application, a simulation of a string excited by a bow is developed. We show that the elasto-plastic models predict the same qualitative behavior as the plastic model [12].

Section 2 describes the continuous-time model, and Sec. 3 its discretization. Section 4 discusses two real-time `pd`[1] modules that implement of the model, and Sec. 5 presents applications to the bowed string and to more general friction based sound effects.

## 2. THE MODEL

Section 2.1 reviews the elasto-plastic friction model used in the remainder of the paper. In Sec. 2.2, this model is applied to an exciter-resonator system.

### 2.1. The friction model

The friction model we use is based on the bristle interpretation of friction as shown in figure 1. Let's consider two facing surfaces

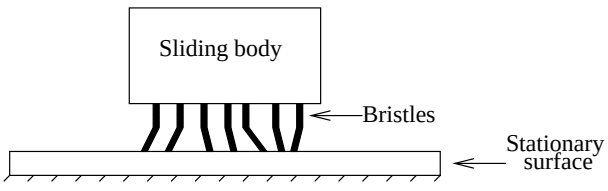


Figure 1: *Bristle model.*

[1] http://www.pure-data.org

with bristles extending from each, as shown in figure 1. The friction between the two surfaces is assumed to be caused by a large number of bristles, each contributing a fraction of the total friction load. The load contributed by each bristle is proportional to the strain of the bristle, i.e. the bristles act as linear springs. When the strain exceeds a certain level the bond is broken. In LuGre model [5], friction is modeled as the average deflection of the bristles. When a tangential force is applied, the bristles deflect and, if the deflection is large enough, the bristles start to slip. Denoting by $z$ the average bristle deflection, and by $v$ the relative velocity between the two surfaces, the model is given by:

$$\dot{z} = v - \frac{|v|}{g(v)} z \ , \tag{1}$$

$$f_{fr} = \sigma_0 z + \sigma_1 \dot{z} + f(v) \ , \tag{2}$$

where $\sigma_0$ is the stiffness of the bristles, and $\sigma_1$ is the bristle damping (see figure 2). The additional term $f(v)$ accounts for the viscous friction. As an example, a special case of the LuGre model (sometimes referred to as *standard parametrization* [9]) assumes linear viscous friction, and the term $f(v)$ takes the form $\sigma_2 v$. Equation (1) is such that in steady state the deflection $z$ approaches the value

$$z_{ss}(v) = g(v)\text{sgn}(v) \ . \tag{3}$$

The function $z_{ss}(v)$ has the typical shape used for interaction force in classic bowed string modeling [8], accounting for the so-called Stribeck effect.

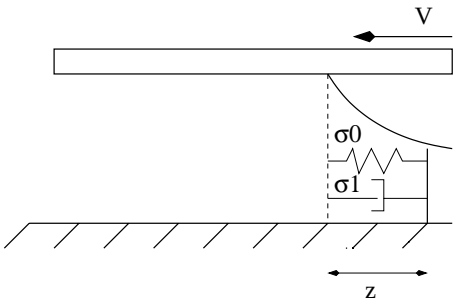


Figure 2: *The LuGre model.*

The physical interpretation of this model is as follows. Contact surfaces are very irregular at microscopic level. We visualize this as two rigid bodies that make contact through elastic bristles. When a tangential force is applied, the bristles will deflect like springs and dampers which give rise to the friction force. The average deflection of the bristles corresponds to the internal state of the dynamic friction model $z$.

It has been shown that, when using the LuGre model, a drift is always produced when arbitrarily small forces or torques are applied. This behavior is not physically consistent, since in reality an object remains situated in the presence of small loads. In order to overcome this drawback, a class of *elasto-plastic* models has been proposed in [6], where the following formulation is used for the bristle displacement:

$$\dot{z} = f_{NL}(v,z) = v\left[1 - \alpha(z,v)\frac{z}{z_{ss}(v)}\right] \ , \tag{4}$$

while the friction force $f_{fr}$ is modeled as in Eq. (2). The main difference with the LuGre model is represented by the function $\alpha(z,v)$: this is an adhesion map which controls the rate of change of $z$; specifically, when the bristle displacement $z$ is smaller than a given breakaway displacement $z_{ba}$, then $\alpha$ is set to zero and consequently equation 4 reduces to $\dot{z} = v$ (see figure 3). This means that in the range $|z| \leq z_{ba}$ the model shows purely elastic presliding regime, and therefore does not exhibit drift. Note that LuGre is a special case of Eq. 4 where $\alpha(z,v) \equiv 1$.

### 2.2. The exciter-resonator model

In what follows, we name "bow" one of the modal objects and we use the subscript $b$ when referring to its variables. Similarly, the other "resonator" is indicated by the subscript "r". The resonator and the bow are modeled as a set of $N_r$ and $N_b$ second order mechanical oscillators respectively. The exciter and the resonator can be made interact via the friction model of equations (4) and (2), where linear viscous friction is assumed. The resulting system is described by the following set of equations:

$$\begin{cases} m_{bi}\ddot{x}_{bi} + r_{bi}\dot{x}_{bi} + k_{bi}x_{bi} = f_{be} - f_{fr}, \\ \qquad\qquad (i = 1 \ldots N_b) \\ m_{rj}\ddot{x}_{rj} + r_{rj}\dot{x}_{rj} + k_{rj}x_{rj} = f_{re} + f_{fr}, \\ \qquad\qquad (j = 1 \ldots N_r) \\ v = \sum_{i=1}^{N_b} \dot{x}_{bi} - \sum_{j=1}^{N_r} \dot{x}_{rj}, \qquad \text{(relative velocity)} \\ \dot{z} = f_{NL}(v,z) = v\left[1 - \alpha(z,v)\frac{z}{z_{ss}(v)}\right] , \\ f_{fr} = \sigma_0 z + \sigma_1 \dot{z} + \sigma_2 v, \qquad \text{(friction force)} \end{cases} \tag{5}$$

where the $x$'s represent the modal displacements, while $z$ is the mean bristle displacement (see [6, 9]). The terms $f_{be}$ and $f_{re}$ represent external forces. Explicit expressions for the functions $\alpha$ and $z_{ss}$ are available from the literature. Here we follow [5] by defining $z_{ss}$ as

$$z_{ss}(v) = \frac{\text{sgn}(v)}{\sigma_0}\left[f_c + (f_s - f_c)e^{-(v/v_s)^2}\right], \tag{6}$$

where $f_c, f_s$ are the Coulomb force and the stiction force respectively, while $v_s$ is the Stribeck velocity. Following [6], the function $\alpha(z,v)$ is parametrized as:

$$\alpha(v,z) = \begin{cases} 0 & |z| < z_{ba}, \text{sgn}(v) = \text{sgn}(z) \\ \alpha_m & z_{ba} < |z| < z_{ss}(v), \text{sgn}(v) = \text{sgn}(z) \\ 1 & |z| > z_{ss}(v), \text{sgn}(v) = \text{sgn}(z) \\ 0 & \text{sgn}(v) \neq \text{sgn}(z) \ , \end{cases} \tag{7}$$

where $z_{ba}$ is the breakaway displacement below which the presliding is purely elastic. $\alpha_m$ is indeed a function of $v$ and $z$, which describes the transition between elastic and plastic behavior, as

$$\alpha_m(v,z) = \frac{1}{2}\left[1 + \sin\left(\pi\frac{z - \frac{1}{2}(z_{ss}(v) + z_{ba})}{z_{ss}(v) - z_{ba}}\right)\right]. \tag{8}$$

Therefore, for sgn$(v)$ =sgn$(z)$ the general shape of the function $\alpha$ is the one given in Fig. 3.

## 3. DISCRETE-TIME EQUATIONS

System (5) is structurally identical to the impact model presented in [3], therefore similar numerical techniques can be applied in

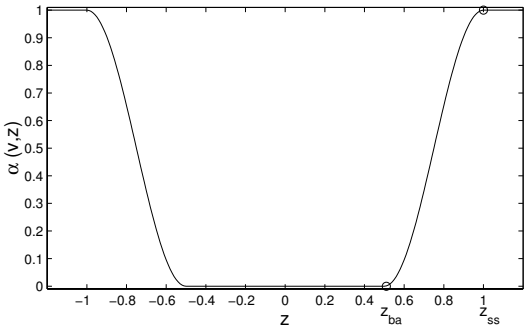


Figure 3: *Plot of the function* $\alpha(v,z)$ *for sgn*$(v) =$ *sgn*$(z)$.

order to discretize the continuous-time model. The equations for the modal objects are discretized with the bilinear transformation, and in the discrete-time domain the two objects appear as two filter banks of second-order bandpass filters, each one accounting for one specific mode of the objects.

By applying the bilinear transformation to the first equation in system (5), the resonator is represented by the discrete-time equations

$$\begin{cases} \boldsymbol{x_{rl}}(n) = \boldsymbol{A_{rl}}\boldsymbol{x_{rl}}(n-1) + \boldsymbol{b_{rl}}[f(n)+f(n-1)], \\ \qquad\qquad (\text{for} \quad l = 1 \dots N_r), \\ \boldsymbol{x}_r(n) \triangleq \sum_{l=1}^{N} \boldsymbol{x}_{rl}(n), \\ f(n) = f_{re}(n) + f_{fr}(n), \end{cases} \tag{9}$$

where the vectors $\boldsymbol{x_{rl}}$ are defined as $\boldsymbol{x_{rl}} = \begin{bmatrix} x_{rl} \\ \dot{x}_{rl} \end{bmatrix}$, and $f$ is the total force acting on the resonator. Similar equations, with matrices $\boldsymbol{A_{bl}}$ and vectors $\boldsymbol{b_{bl}}$ (for $l = 1 \dots N_b$) are found for the bow. We refer to [2] for detailed definitions of these matrices and vectors.

The bristle equation of dynamics $\dot{z} = f_{NL}(v,z)$ is also discretized using the bilinear transformation, with time quantum $T_s$. Since this is a first order equation, discretization by the trapezoid rule is straightforward:

$$\begin{aligned} z(n) &\approx \underbrace{z(n-1) + \frac{T_s}{2}y(n-1)}_{\tilde{z}(n)} + \frac{T_s}{2}y(n) \\ &\triangleq \qquad\qquad\quad \tilde{z}(n) \qquad\quad + \boldsymbol{k}(2)y(n) \end{aligned} \tag{10}$$

where $y \triangleq f_{NL}(v,z) = v\left[1 - \alpha(z,v)\frac{z}{z_{ss}(v)}\right]$ and $\boldsymbol{k}(2) \triangleq \frac{T_s}{2}$. While $\tilde{z}(n)$ is a computable quantity (i.e., it is a linear combinations of known variables), the coefficient $\boldsymbol{k}(2)$ isolates the dependence of $z(n)$ upon the force $y$ at the current time instant $n$.

We still need to express the relative velocity $v$ as the sum of its computable part and the contribution of the term $y(n)$ (similarly to (10)). This is less straightforward, since the friction force itself depends explicitly upon $v$. From the third of (5), exploiting (9), recalling (10), and defining $b \triangleq \left[\sum_{i=1}^{N_b} \boldsymbol{b}_{bi}(2) + \sum_{j=1}^{N_r} \boldsymbol{b}_{rj}(2)\right]$, we obtain

$$\begin{aligned} v(n) = \quad & \frac{1}{1+\sigma_2 b}\left\{\sum_{i=1}^{N_b}\left\{\dot{\tilde{x}}_{bi}(n) + \boldsymbol{b}_{bi}(2)[f_{be}(n) - \sigma_0\tilde{z}(n)]\right\} - \right. \\ & \left. - \sum_{j=1}^{N_r}\left\{\dot{\tilde{x}}_{rj}(n) + \boldsymbol{b}_{rj}(2)[f_{re}(n) + \sigma_0\tilde{z}(n)]\right\}\right\} - \\ & - \frac{b}{1+\sigma_2 b}\left(\sigma_0\frac{T_s}{2} + \sigma_1\right)y(n) = \\ \triangleq \quad & \tilde{v}(n) + \boldsymbol{k}(1)y(n). \end{aligned} \tag{11}$$

### 3.1. The Newton-Raphson algorithm

Equations (10) and (11) show that when the two modal objects are coupled through the non-linear friction interaction, the resulting system exhibits computational problems in that a delay-free path is generated in the computation: namely, the relative velocity $v(n)$ and the bristle displacement $z(n)$ depend on the term $y(n)$, which in turn depends on the pair $(v(n), z(n))$.

The last equation of the discrete-time system provides the non-linear term $y$ as

$$y(n) = f_{NL}\left(\tilde{v}(n) + \boldsymbol{k}(1)y(n), \tilde{z}(n) + \boldsymbol{k}(2)y(n)\right). \tag{12}$$

In order to find the current value $y(n)$, the implicit equation (12) must be solved. As shown by Borin *et al.* [4], an efficient solution is provided by the K-method. As in [3], the nonlinear function $f_{NL}(n)$ is computed iteratively using the Newton-Raphson algorithm: early results suggest that the number of iterations remains smaller than seven for typical parameter configurations. Thus, the final algorithm can be written as:

```
for    n = 1 ... samplelength
   [ x̃_bi(n) ; ẋ̃_bi(n) ] (∀i), and [ x̃_rj(n) ; ẋ̃_rj(n) ] (∀j)
   z̃(n) and ṽ(n)
   y(n) with Newton-Raphson
   v(n) as v(n) = ṽ(n) + k(1)y(n)
   z(n) as z(n) = z̃(n) + k(2)y(n)
   f_fr(n) as f_fr(n) = σ0 z(n) + σ1 y(n) + σ2 v(n)
   [ x_bi(n) ; ẋ_bi(n) ] = [ x̃_bi(n) ; ẋ̃_bi(n) ] + b_bi[f_be(n) − f_fr(n)] (∀i),
   [ x_rj(n) ; ẋ_rj(n) ] = [ x̃_rj(n) ; ẋ̃_rj(n) ] + b_rj[f_re(n) + f_fr(n)] (∀j)
end
```

As far as the Newton-Raphson method is concerned, at each time step $n$ the value of $y(n)$ is computed by finding a local zero of the function $g(y) = [f_{NL}(\tilde{v}(n) + \boldsymbol{k}(1)y(n), \tilde{z}(n) + \boldsymbol{k}(2)y(n)) - y]$ in this way:

$y_0 = y(n-1)$
$k = 1$
while (err $<$ Errmax)
$\quad g(y_k) = f_{NL}\left(\tilde{v}(n) + \boldsymbol{k}(1)y_k, \tilde{z}(n) + \boldsymbol{k}(2)y_k\right) - y_k$
$\quad y_{k+1} = y_k - \frac{g(y_k)}{g'(y_k)}$
$\quad$ err = abs$(y_{k+1} - y_k)$
$\quad k = k + 1$
end
$y(n) = y_k$

Thus, the derivative $g'(y)$ has to be computed. This is done in successive steps as a composite derivative [2].

## 4. IMPLEMENTATION

### 4.1. Modal friction

We have implemented a `pd` plugin, structurally based on the model of impact interaction between modal objects described in [10]. Indeed, the friction interaction plugin is distributed together with the impact interaction models as part of the `interaction_modal` package, downloadable from the Sounding Object web site [2].

When opening the `interaction_modal` folder, one finds a few subdirectories that reflect the modular structure of the plugins:

`resonators`: contains the implementation of resonators described as a bank of modal mechanical oscillators, each discretized with the bilinear transformation. External forces can be applied at specified interaction points, each point being described by a set of numbers that weight each mode at that point. Displacement or velocity are returned as outputs from the modal object.

`interactors`: for friction interaction, a function computes the forces to be applied to two interacting resonators. To do that, it uses the "free-developed" resonator velocities $\tilde{v}$ and bristle displacement $\tilde{z}$, and it computes the $\boldsymbol{k}$ matrix using parameters of the modal objects.

`sound_modules`: contains a subdirectory for each plugin implemented, where the structures and functions required by `pd` are provided. Here, the external appearance (default parameter values, inlets and outlets) of the plugin is also defined.

Figure 4 displays the `friction_2modalb` ("2" stands for two modal objects, and "b" stands for bilinear transformation) plugin inserted in a `pd` patch. In the top-right corner, sliders control the parameters of the second modal object (resonator). Here, the sliders labeled `lev*` control the effectiveness of interaction of the exciter with each of the resonator modes. In fact, in the implementation chosen for modal resonators there is no explicit control on vibrating masses, but these "gains" allow to adjust the relative weight between exciter and resonator, thus affecting the $\boldsymbol{k}$ matrix in a predictable way.

### 4.2. Waveguide friction

It is interesting to examine how the elasto-plastic friction model behaves in the well-studied context of bow-string interaction. To

[2] http://www.soundobject.org

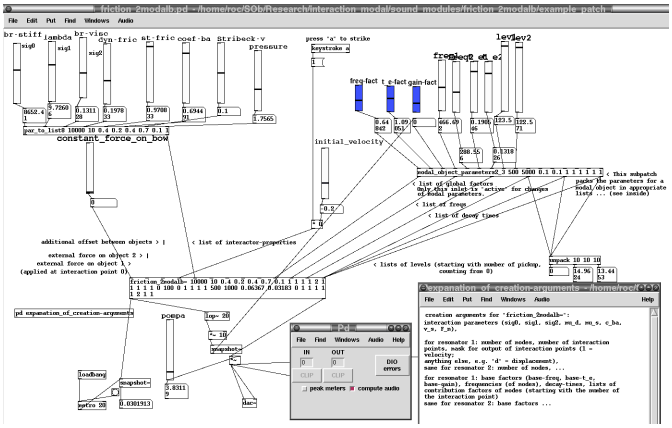

Figure 4: `pd` *patch for elasto-plastic friction interaction between two modal objects.*

experiment with that, we developed a `pd` plugin called `kviolin`, where a waveguide string model is substituted to the modal resonator. It can be easily seen that the structure of the discrete-time equations remains the same when using a waveguide resonator, only the explicit expression for the K-matrix needs to be changed.

The model runs in real-time with moderate system requirements, and it allows to dynamically reach several oscillation regimes. As any violin player knows, a joint balanced control of bow pressure and velocity is crucial to achieve the desired oscillation quality, and the temporal development of these control curves is also crucial. It is almost impossible to achieve a satisfactory sound evolution using only sliders to control the parameters. Therefore, we used Damien Henry's `xgui` [3] to design a control panel (see figure 5), where a 2D pad allows simultaneous control of bow pressure and velocity. The plugin and example patches are available from the SOb project web site [4].

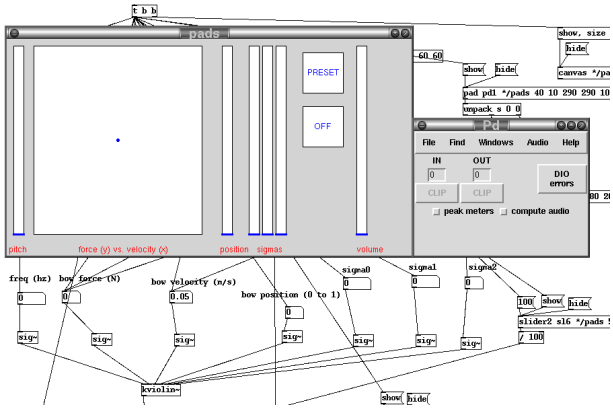


Figure 5: *Fragment of a* `pd` *patch for elasto-plastic string bowing. The main parameters are accessible via a control panel designed using* `xgui`.

[3] http://dh7.free.fr
[4] http://www.soundobject.org

## 5. APPLICATIONS

### 5.1. The bowed string

In the past, theoretical modeling of the motion of a bowed string has assumed that the frictional force due to the rosin on the bow was determined only by the relative speed between the bow and the string.

However, recent results [12] have demonstrated that this is not the case. Experiments show that the relative velocity versus friction force curve exhibits a hysteresis loop. This loop suggests the need for a different friction model. In [12] a model is proposed where the friction force depends on variations of temperature at the rosin layer between the two bodies in contact. The authors called this model *plastic*, since it accounts for plastic deformations of the rosin layer according to the variations of temperature. In this model, friction is given by:

$$f_{fr} = \mu(T) F_N \cdot \mathrm{sgn}(v) \tag{13}$$

where $F_N$ is the normal bow force,

$\mu(T)$ is a temperature dependent friction coefficient as described in [12], and $v$ is as before the relative velocity between the bow and the string.

Figure 6 shows the results of applying this model to the simulation of a cello $D$ string, tuned to $147$ Hz, with a $Q$ factor of $500$ and a stiffness coefficient $B = 0.0003$ N m$^2$. The simulations of the string were performed using a waveguide model. As in typical implementations, losses at the extremities were lumped into low-pass filters, in order to obtain the desired $Q$ factor. The top of Fig. 6 shows the snapshot of the time-domain velocity waveform at the bow point obtained during the steady state portion of the motion, for $V_b = 0.1$ m/s and $F_N = 1.1$ N. Note how the Helmholtz motion, i.e. the ideal motion of a bowed string, is achieved. The bottom of Fig. 7 shows the plot of the coefficient of friction versus velocity. For comparison, the exponentially decaying steady-state curve is overlapped.

In order to obtain a comparison with the model proposed in [12], we applied our model to the simulation of a string excited by a bow, using the same waveguide cello $D$ string as before.

In order to relate the parameters of the elasto-plastic model to the parameters that drive a bowed string instrument, the relationship between these parameters must be found.

The parameters $(f_c, f_s)$ are related to the normal force $F_N$ through the static and dynamic friction coefficients as$f_s = \mu_s F_N$ and $f_c = \mu_d F_N$, where in violin bows the static and dynamic coefficients take values $\mu_s \approx 0.4 \ldots 0.5$ and $\mu_d \approx 0.2$, respectively (see [1]).

The breakaway displacement $z_{ba}$ is also influenced by the normal force: note that in order for $\alpha(v, z)$ to be well defined, the inequality $z_{ba} < z_{ss}(v) \forall v$ must hold (this remark is made also in [6]). Since $\min_v z_{ss}(v) = f_c/\sigma_0$, a suitable choice for $z_{ba}$ is

$$z_{ba} = c f_c/\sigma_0 = c\mu_d F_N/\sigma_0, \qquad \text{with} \qquad c < 1, \tag{14}$$

which says that the breakaway displacement increases with the normal force, as one would expect.

The remaining parameters are taken from the literature. Note that $\sigma_0$ defines the compliance of the dynamic model (for $\sigma_0 \to \infty$ the bristles do not move anymore) or, equivalently, the magnitude of the allowed presliding displacement, while $\sigma_1$ describes the internal dissipation of vibrating bristles.

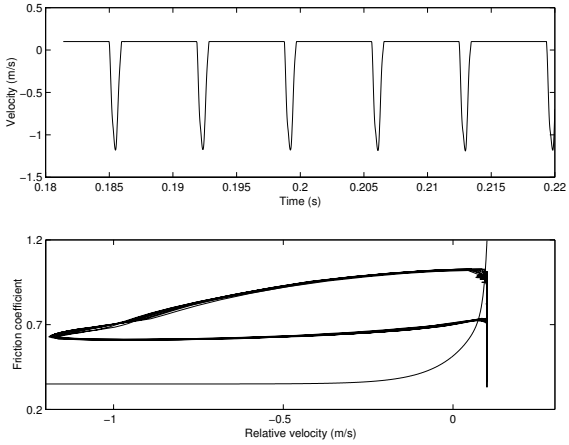


Figure 6: *Top: velocity at the bow point for a cello D string (*147 *Hz) bowed with $V_b = 0.1$ m/s and $F_N = 1.1$ N. Bottom: coefficient of friction versus velocity. Both plots are generated using the model proposed in [12]. Courtesy of Jim Woodhouse. For comparison, the exponentially decaying curve displayed in the plot shows the steady-state friction curve used in [12]. Note how, as the authors show, no part of the hysteretical curve follows the steady state curve.*

Analogously to Fig. 6, Fig. 7 shows the results of applying the elasto-plastic model to the simulation of a cello $D$ string. The top plot provides a snapshot of the time-domain velocity waveform at the bow point obtained during the steady state portion of the motion: it can be seen that Helmholtz motion is achieved. The bottom plot shows the coefficient of friction versus velocity.

Note how the behavior of the two models is qualitatively similar. Both models show a friction curve with an hysteretical behavior. The differences on the shape of the friction curve are due to the fact that the approach of the two models is completely different. Indeed, while the first model is purely based on the thermodynamical properties of rosin, the model that we use is purely mechanical.

One advantage of using a purely mechanical model is that it can be easily generalized to other rubbed surfaces in contact, as described in section 5.2. From a perceptual point of view, however, the behavior of the two models is strongly similar.

### 5.2. Friction based sound effects

The bowed string examined in the previous example is the most common instrument driven by friction. However, friction driven oscillations appear in all systems in which rubbed dry surfaces are in contact, such as squeaking doors and chalks. It is interesting to notice that the same friction model used for the bowed string can be applied, with adjustments to the parameters both for the exciter and the resonator, to all these other situations.

This allows the possibility of creating interesting sonorities both from a musical and a sound-effects producer viewpoint. Once the parameters are tuned, in fact, the models can be used to reproduce many sonic effects that appear in our world.

Our modal synthesis framework is general enough to allow a cartoon reproduction of many friction-based sonorities. However, when looking for accurate reproduction of friction phenomena one

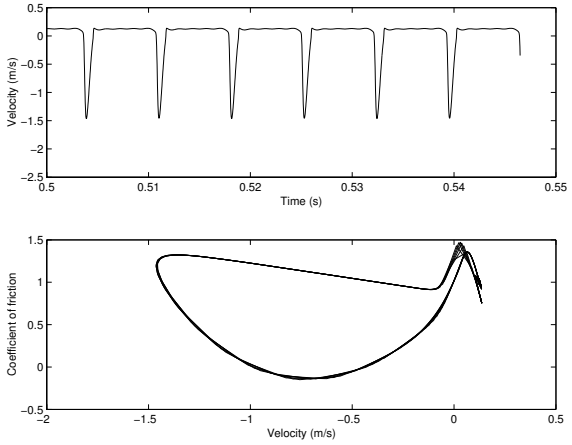


Figure 7: *Top: velocity at the bow point for a cello D string (*147 *Hz) bowed with $V_b = 0.1$ m/s and $F_{eb} = 1.1$ N. Bottom: velocity versus coefficient of friction.*

should bear in mind that "There are many different mechanisms. To construct a general friction model from physical first principles is simply not possible" [9].

## 6. CONCLUSIONS

In this paper we have proposed a sound synthesis algorithm for audio rendering of frictional interactions between rubbed surfaces, based on elasto-plastic friction models. The numerical implementation has been discussed, and it has been shown that the discretization strategy here proposed allows real-time implementation.

An application to the specific case of the bowed string has been given. In this situation, the model exhibits hysteretical behavior, which has been measured on real instruments by Smith and Woodhouse. However, further studies are needed in order to develop a more rigorous comparison between the elasto-plastic model and the plastic model developed by Smith and Woodhouse.

The flexibility of our model allows us to apply it to different situations in which frictional interactions exist. Due to the large number of control parameters, controlling the sound module is not a trivial task. Therefore, future work shall concentrate on finding effective control strategies and effective mappings between the parameters that are available from typical human-computer interfaces (e.g., pointing devices or pen-based input devices) and the physical parameters of the model.

## 7. ACKNOWLEDGMENT

We are thankful to Matthias Rath for providing the software infrastructure described in section 4.1, and to Gianpaolo Borin for his help in software debugging.

This work was partially supported by the European Commission's Future & Emergent Technologies collaborative R&D programme (SOb – the Sounding Object project, IST-2000-25287[5]).

[5]http://www.soundobject.org

## 8. REFERENCES

[1] A. Askenfelt. Measurement of the bowing parameters in violin playing. II: Bow-bridge distance, dynamic range, and limits of bow force. *J. Acoust. Soc. Am.*, 86(2):503–516, Aug. 1989.

[2] F. Avanzini, M. Rath, and D. Rocchesso. Modal resonators. In *Models and Algorithms for Sounding Objects*, pages 6–9. The SOb Project, 2002. Available from http://www.soundobject.org.

[3] F. Avanzini and D. Rocchesso. Modeling Collision Sounds: Non-linear Contact Force. In *Proc. COST-G6 Conf. Digital Audio Effects (DAFx-01)*, pages 61–66, Limerick, Dec. 2001.

[4] G. Borin, G. De Poli, and D. Rocchesso. Elimination of Delay-free Loops in Discrete-Time Models of Nonlinear Acoustic Systems. *IEEE Trans. Speech Audio Process.*, 8(5):597–606, 2000.

[5] C. Canudas de Wit, H. Olsson, K. J. Åström, and P. Lischinsky. A new model for control of systems with friction. *IEEE Trans. Autom. Control*, 40(3):419–425, 1995.

[6] P. Dupont, V. Hayward, B. Armstrong, and F. Altpeter. Single State Elasto-Plastic Friction Models. *IEEE Trans. Autom. Control*, 47(5):787–792, may 2002.

[7] V. Hayward and B. Armstrong. A new computational model of friction applied to haptic rendering. In P. Corke and J. Trevelyan, editors, *Experimental Robotics VI*, pages 403–412. Springer-Verlag, 2000.

[8] M. E. McIntyre, R. T. Schumacher, and J. Woodhouse. On the oscillations of musical instruments. *J. Acoust. Soc. Am.*, 74(5):1325–1345, 1983.

[9] H. Olsson, K. J. Åström, C. Canudas de Wit, M. Gäfwert, and P. Lischinsky. Friction models and friction compensation. *European J. of Control*, 4(3):176–195, 1998.

[10] M. Rath. Sound design around a real-time impact model. In *Models and Algorithms for Sounding Objects*, pages 32–45. The SOb Project, 2002. Available from http://www.soundobject.org.

[11] S. Serafin, C. Vergez, and X. Rodet. Friction and Application to Real Time Physical Model of a Violin. In *Proc. Int. Computer Music Conf. (ICMC'99)*, Beijing, Oct. 1999.

[12] J. H. Smith and J. Woodhouse. The tribology of rosin. *J. Mech. Phys.Solids*, 48:1633–1681, 1999.