



**QUEEN'S  
UNIVERSITY  
BELFAST**

## **Quadratic spline approximation of the contact potential for real-time simulation of lumped collisions in musical instruments**

Bhanuprakash, A., Van Walstijn, M., & Chatziioannou, V. (2024). Quadratic spline approximation of the contact potential for real-time simulation of lumped collisions in musical instruments. In *27th International Conference on Digital Audio Effects (DAFx24): Proceedings* (pp. 155-162). (Proceedings of the International Conference on Digital Audio Effects, DAFx). [https://www.dafx.de/paper-archive/search?years\[\]=2024&p=3](https://www.dafx.de/paper-archive/search?years[]=2024&p=3)

### **Published in:**

27th International Conference on Digital Audio Effects (DAFx24): Proceedings

### **Document Version:**

Publisher's PDF, also known as Version of record

### **Queen's University Belfast - Research Portal:**

[Link to publication record in Queen's University Belfast Research Portal](#)

### **Publisher rights**

Copyright 2024 the authors.

This is an open access article published under a Creative Commons Attribution License (<https://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution and reproduction in any medium, provided the author and source are cited.

### **General rights**

Copyright for the publications made accessible via the Queen's University Belfast Research Portal is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

### **Take down policy**

The Research Portal is Queen's institutional repository that provides access to Queen's research output. Every effort has been made to ensure that content in the Research Portal does not infringe any person's rights, or applicable UK laws. If you discover content in the Research Portal that you believe breaches copyright or violates any law, please contact [openaccess@qub.ac.uk](mailto:openaccess@qub.ac.uk).

### **Open Access**

This research has been made openly available by Queen's academics and its Open Research team. We would love to hear how access to this research benefits you. – Share your feedback with us: <http://go.qub.ac.uk/oa-feedback>

# QUADRATIC SPLINE APPROXIMATION OF THE CONTACT POTENTIAL FOR REAL-TIME SIMULATION OF LUMPED COLLISIONS IN MUSICAL INSTRUMENTS

Abhiram Bhanuprakash\* and Maarten van Walstijn

Sonic Arts Research Centre (SARC)  
Queen's University Belfast  
Belfast, UK  
{abhanuprakash01 | m.vanwalstijn}@qub.ac.uk

Vasileios Chatziioannou

Department of Music Acoustics (IWK)  
University of Music and Performing Arts  
Vienna, Austria  
chatziioannou@mdw.ac.at

## ABSTRACT

Collisions are an integral part of the sound production mechanism in a wide variety of musical instruments. In physics-based real-time simulation of such nonlinear phenomena, challenges centred around efficient and accurate root-finding arise. Nonlinearly implicit schemes are normally ill-suited for real-time simulation as they rely on iterative solvers for root-solving. Explicit schemes overcome this issue at the cost of a slightly larger error for a given sample rate. In this paper, for the case of lumped collisions, an alternate approach is proposed by approximating the contact potential curve. The approximation is described, and is shown to lead to a non-iterative update for an energy-stable nonlinearly implicit scheme. The method is first tested on single mass-barrier collision simulations, and then employed in conjunction with a modal string model to simulate hammer-string and slide-string interaction. Results are discussed in comparison with existing approaches, and real-time feasibility is demonstrated.

## 1. INTRODUCTION

Inter-object collisions are among the nonlinear acoustic phenomena that significantly influence the sonic character of various musical instruments. Certain collisions where contact occurs over a narrow region could be treated as *lumped*, e.g: piano hammer-string [1], xylophone mallet-bar [2] interaction, while others are generally regarded as *distributed*, e.g: sitar string-bridge [3], snare drum wire-membrane [4] interaction. These phenomena are often modelled by means of a Hertzian contact power law [5]. Partly owing to the one-sided nature of the nonlinearity, simulating them numerically in real-time - while preserving system properties critical in musical instrument simulation like passivity and stability - is a challenging task [6].

A range of energy-stable time-stepping schemes exists in the collision modelling literature. At one end of the spectrum we have *explicit* schemes [7, 8], where an auxiliary variable is introduced in order to quadratise the contact potential, which thereby allows the current system variable and the auxiliary variable to be explicitly expressed as functions of their previous values. Although this feature makes these schemes suited for real-time simulation, recent

This work was supported by the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 812719.

Copyright: © 2024 Abhiram Bhanuprakash et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, adaptation, and reproduction in any medium, provided the original author and source are credited.

analysis [8] has revealed an inherent issue of the energy quadratisation approach: due to the dependency of the numerical energy on the auxiliary variable, these schemes do not exhibit a unique trajectory in the contact region of their phase-space, which can lead to increased errors. For example, perceptually significant wideband noise-like artefacts are produced in the case of a sitar-like configuration, suppressing which requires at least 4x oversampling [8].

At the other end of the spectrum lie *nonlinearly implicit* (NLI) schemes [6], where the current value of the system variable appears within a nonlinear implicit function. This leads to a transcendental equation to be solved at each time step, and for this purpose iterative solvers such as Newton's method have been traditionally employed. However, due to the uncertainty in the number of iterations, the computational cost of such schemes varies at each time step, and thus approaches employing iterative solvers are not well-suited for real-time simulation [7, 8].

Somewhere in between these two extremities lies another approach: to fix the power law exponent to unity and thereby obtain a specific non-iterative branched-form update for an NLI scheme [9]. This has since been adopted in a few case studies involving lumped collisions [10, 11]. However, such a model does not cover the full range of compliances involved in musical instruments - e.g., felt hammers in pianos have a large stiffness range for a given force range [12], and the measured power law exponent is generally around 2.5 [1]. In this paper, by approximating the contact potential curve, this approach is extended to the more general case of lumped collisions with any valid power law exponent.

## 2. LUMPED COLLISION MODEL

Throughout the paper,  $d_v$  and  $\partial_v$  denote the total and partial derivatives with respect to the variable  $v$ , respectively. Consider contact between two objects at positions  $u_1$  and  $u_2$  at time  $t$ , and let  $y = y(t) := u_1(t) - u_2(t)$ . If the contact occurs over a narrow region, the contact force  $F$  can be modelled in the form of a power law [5] as:

$$F(y) = -\kappa [y]_+^\alpha = -\partial_y V(y), \quad (1)$$

where the power law parameters  $\kappa > 0$  and  $\alpha \geq 1$  depend on the geometries and materials of the objects that come into contact. From Hertzian contact theory [5], the exponent  $\alpha$  is typically 1 or 1.5. If one includes cases of strongly nonlinear compression of felts (e.g. kettledrum mallets [13] or piano hammer [14]), then  $1 \leq \alpha \leq 3$  can be considered as a musically relevant range. With  $y$  now interpreted as a compression variable,  $[y]_+ := \max(0, y)$  represents the inter-object compression, and

$$V(y) = \frac{\kappa}{\alpha + 1} [y]_+^{\alpha+1} \quad (2)$$

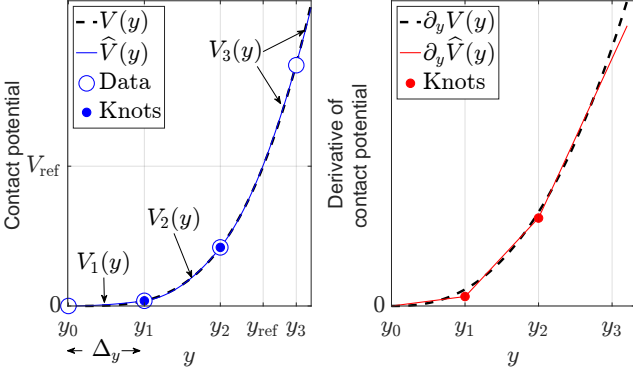


Figure 1: Quadratic spline approximation of the contact potential (left, blue); its derivative (right, red), for  $\alpha = 2.5$  and  $\Delta_y = 0.39 y_{\text{ref}}$  ( $N = 3$ ). In both plots, the black dashed line shows the underlying power law.

is the contact potential.

## 2.1. Quadratisation of the Contact Potential

Introducing an auxiliary variable  $\psi$ , the contact potential can be re-written [7] as

$$V = \frac{1}{2} \psi^2. \quad (3)$$

From (1) and (3), the contact force can then be written in terms of  $\psi$  and a new gradient variable  $g := \partial_y \psi = \sqrt{\frac{1}{2} \kappa (\alpha + 1)} [y]_+^{\alpha-1}$  as

$$F = -\psi g, \quad d_t \psi = g d_t y. \quad (4)$$

## 2.2. Quadratic Spline Approximation of the Contact Potential

In musical instrument modelling, one would typically have an estimate of the collision energy range which can be considered as a reference, say  $[0, V_{\text{ref}}]$ . From (2), the corresponding reference compression range is  $[0, y_{\text{ref}}]$ , where

$$y_{\text{ref}} = \left[ \frac{(\alpha + 1) V_{\text{ref}}}{\kappa} \right]^{\frac{1}{\alpha+1}}. \quad (5)$$

As shown in Figure 1, we can sample the contact potential curve at  $(N + 1)$  regularly spaced compression values  $y_j = j \Delta_y$ ,  $j = 0, 1, 2, \dots, N$ ,  $\Delta_y > 0$ , where  $N$  is such that the last compression value  $y_N \in [y_{\text{ref}}, y_{\text{ref}} + \Delta_y]$ , i.e.,  $N = \left\lceil \frac{y_{\text{ref}}}{\Delta_y} \right\rceil$ , where  $\lceil \cdot \rceil$  represents the ceiling operator.  $V$  can now be approximated in a piecewise-quadratic fashion as

$$\widehat{V}(y) = \begin{cases} V_j(y) & : y \in [y_{j-1}, y_j], j = 1, 2, \dots, N-1, \\ V_N(y) & : y \geq y_{N-1}, \\ 0 & : \text{otherwise,} \end{cases} \quad (6)$$

where

$$V_j(y) = a_j y^2 + b_j y + c_j, \quad j = 1, 2, \dots, N \quad (7)$$

is the  $j^{\text{th}}$  quadratic segment, with  $V_j(y_j) = V(y_j)$  being the sampled data. For  $\widehat{V}$  and its derivative  $\partial_y \widehat{V}$  to represent a one-sided nonlinearity, we require both the first quadratic segment  $V_1(y)$  and its derivative  $\partial_y V_1(y)$  to pass through the origin. Further, for  $\widehat{V}$  to be a continuous function, successive quadratic segments need

to meet at their ‘knots’  $\{(y_j, V_j(y_j))\}$ ,  $j = 1, 2, \dots, N-1$ , and for  $\widehat{V}$  to be at least  $C^1$  smooth, the first derivatives of successive segments need to be equal at their knots. These conditions can be written as

$$b_1 = c_1 = 0, \quad (8)$$

$$V_j(y_j) = V_{j+1}(y_j), \quad \text{and} \quad (9)$$

$$\partial_y V_j(y_j) = \partial_y V_{j+1}(y_j), \quad j = 1, 2, \dots, N-1. \quad (10)$$

The design parameter  $\Delta_y$  controls the accuracy of the approximation - the accuracy is generally expected to increase as  $\Delta_y$  is reduced (or equivalently, the number of quadratic segments  $N$  is increased). Now, with conditions (8) - (10) imposed, the piecewise-quadratic approximation  $\widehat{V}$  in (6) becomes a quadratic spline approximation (QSA) of the contact potential. Here we compute the spline co-efficients  $\{a_j, b_j, c_j, j = 1, 2, \dots, N\}$  by adapting the recipe described for cubic splines in [15] for the case of quadratic splines, and solving a system of linear equations involving a bi-diagonal matrix (see companion website<sup>1</sup> for more detail).

One such  $\widehat{V}$  and its derivative  $\partial_y \widehat{V}$  (which is the corresponding linear spline approximation of  $-F$ ) are shown in Figure 1. Notice that, with only 3 segments, the QSA seems to be quite close to the underlying contact potential curve, while the approximation is more visually apparent in the case of the first derivative.

Observations over a large range of  $\kappa$ ,  $\alpha$  and  $\Delta_y$  suggest that  $\widehat{V}$  is strictly convex. Eqs. (6) and (8) together with convexity imply that  $\widehat{V}(y) \geq 0 \forall y$ . From the above, it is clear that  $\widehat{V}$  has the same key features as  $V$ . Hence we can generally conclude that, for any system, approximating  $V$  by  $\widehat{V}$  does not change the form of the energy balance, and energy-stable behaviour is preserved. Lastly, it is not difficult to see that  $\widehat{V} = V$  for  $\alpha = 1$ , since in this case the underlying contact potential curve is quadratic for  $y \geq 0$ .

## 3. NUMERICAL PRELIMINARIES

Let  $\Delta_t = 1/f_s$  denote the temporal step with sampling frequency  $f_s$ . Further, as per standard practice, let  $u^n$  be the numerical approximation of a variable  $u$  at time  $t = n\Delta_t$ . Let us now define elemental temporal difference and averaging operators:

$$\delta_t u^n := \frac{u^{n+\frac{1}{2}} - u^{n-\frac{1}{2}}}{\Delta_t}, \quad \mu_t u^n := \frac{u^{n+\frac{1}{2}} + u^{n-\frac{1}{2}}}{2}. \quad (11)$$

Derivatives of  $u$  can now be approximated by the following centred discrete operators:

$$\delta_t u^n := \delta_t \mu_t u^n = \frac{u^{n+1} - u^{n-1}}{2\Delta_t} \approx \partial_t u \Big|_{t=n\Delta_t}, \quad (12)$$

$$\delta_t^2 u^n := \delta_t \delta_t u^n = \frac{u^{n+1} - 2u^n + u^{n-1}}{\Delta_t^2} \approx \partial_t^2 u \Big|_{t=n\Delta_t}. \quad (13)$$

Similarly, the following centred discrete averaging operators can be defined:

$$\mu_t u^n := \frac{u^{n+1} + u^{n-1}}{2} \approx u \Big|_{t=n\Delta_t}, \quad (14)$$

$$\mu_t^2 u^n := \mu_t \mu_t u^n = \frac{u^{n+1} + 2u^n + u^{n-1}}{4} \approx u \Big|_{t=n\Delta_t}. \quad (15)$$

<sup>1</sup> Companion website: <https://github.com/abhirambhanuprakash/qs-LumpedCollisions>

#### 4. CASE STUDY I : MASS-BARRIER SINGLE COLLISIONS

A good starting point to investigate numerical schemes for the simulation of lumped collisions is the simple problem of a mass  $m$  colliding with a rigid barrier placed at  $u_b$ . Let  $u = u(t)$  be the displacement of the mass at time  $t$ , and let  $u_0$  and  $v_0$  be the initial displacement and velocity of the mass, respectively. The equation of motion of the mass-barrier system can be written as

$$m \, d_t^2 u = F(y) = -\partial_y V(y), \quad (16)$$

where the compression variable  $y(t) := u(t) - u_b$ , and  $F(y)$  and  $V(y)$  are the contact force and potential as defined in (1) and (2), respectively.

##### 4.1. Discretisation

Eq. (16) can be discretised as

$$m \, \delta_t^2 u^n = F^n. \quad (17)$$

Expanding (17) results in a nonlinear equation in  $s$  of the form

$$s + z - qF^n = 0, \quad (18)$$

where  $s = y^{n+1} - y^{n-1}$  is the update step,  $z = 2(u^{n-1} - u^n)$  is a history variable, and  $q = \frac{\Delta t^2}{m}$  is a constant. Based on how the contact force  $F$  is discretised, various numerical schemes are proposed in the literature. Here, bearing in mind that energy conservation and stability are critical in musical instrument simulation, we select two schemes having these key properties, described below.

###### 4.1.1. Explicit scheme

Based on quadratising the contact potential as in (3), (4) can be discretised as

$$F^n = -(\mu_t \psi^n) g^n, \quad \delta_t \psi^n = g^n \delta_t y^n. \quad (19)$$

The second equation in (19) results in a separate update of  $\psi$  as

$$\psi^{n+\frac{1}{2}} = \psi^{n-\frac{1}{2}} + \frac{1}{2} g^n s, \quad (20)$$

which when substituted in (18) yields the solution

$$s = -\frac{z + q\psi^{n-\frac{1}{2}} g^n}{1 + \frac{1}{4} q(g^n)^2}. \quad (21)$$

A variant of this scheme bounds  $\psi^{n+\frac{1}{2}} \geq 0$  and partly in order to respect this constraint, calculates  $g^n$  in a branched form (see section 3.3.6 in [8]). Here this variant (denoted EXP) is selected, as it has desirable properties such as a provably non-adhesive contact force [8].

###### 4.1.2. Nonlinearly implicit scheme

Another choice of discretising  $F$  [6] is to use the discrete gradient:

$$F^n = -\frac{\delta_t V(y^n)}{\delta_t y^n} = -\frac{V(s + y^{n-1}) - V(y^{n-1})}{s}. \quad (22)$$

Substituting (22) in (18) gives the nonlinear equation

$$s + z + q \frac{V(s + y^{n-1}) - V(y^{n-1})}{s} = 0. \quad (23)$$

Because  $V$  is described by a power law, eq. (23) is a transcendental equation with the unknown  $s$  appearing implicitly within

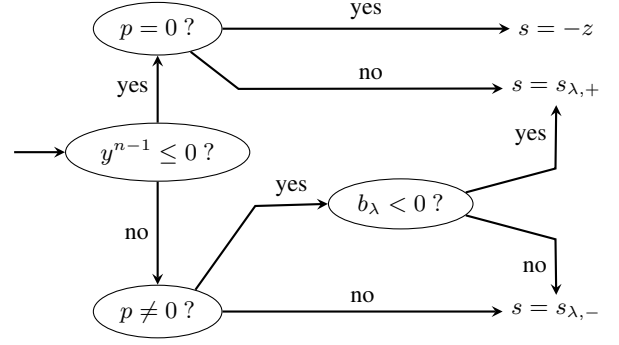


Figure 2: Branched solution with the QSA method.

the nonlinear term. As mentioned in the introduction, traditionally the solution for  $s$  is obtained using iterative methods. Here, this NLI scheme updated by employing a Newton-Raphson solver is denoted IMP.

With either scheme, once  $s$  is found, one can update the compression variable  $y^{n+1} = s + y^{n-1}$  and then update the displacement as  $u^{n+1} = y^{n+1} + u_b$ . The reader is referred to [8] for the energy balance equations of the system for both schemes.

##### 4.2. A Non-Iterative Update for a Nonlinearly Implicit Scheme: The QSA Method

Starting from the same discretisation of  $F$  as the NLI scheme (eq. (22)) but with  $V$  replaced by  $\hat{V}$ , we have the nonlinear equation

$$s + z + q \underbrace{\frac{\hat{V}(s + y^{n-1}) - \hat{V}(y^{n-1})}{s}}_{G(s)} = 0. \quad (24)$$

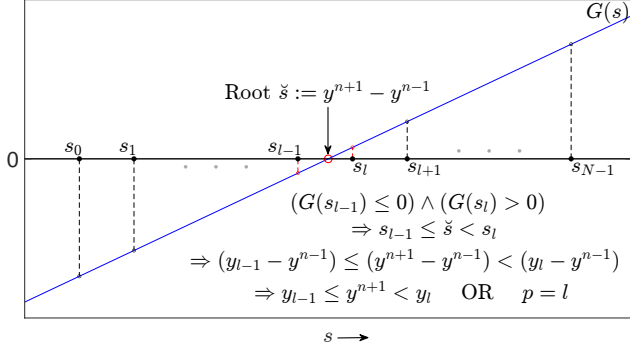
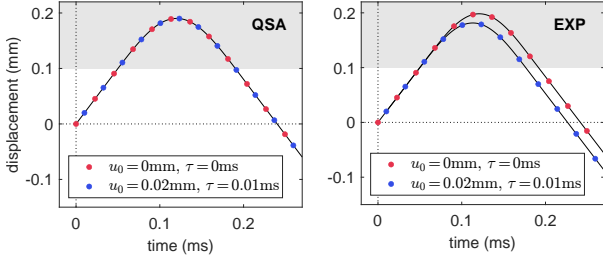
Following the argument in section 2.2 in [16], here using the convexity of  $\hat{V}$  we can show that  $d_s G(s) \geq 1$ , and thereby  $G(s)$  is not only monotonically increasing, but also has a unique root. Further exploiting the convexity of  $\hat{V}$ , the solution to (24) can be expressed in branched-form, as illustrated by the flowchart in Figure 2, where

$$s_{\lambda,\pm} = \frac{-b_{\lambda} \pm \sqrt{b_{\lambda}^2 - 4a_{\lambda}c_{\lambda}}}{2a_{\lambda}}, \quad (25)$$

and where  $\{a_{\lambda}, b_{\lambda}, c_{\lambda}\}$  are the coefficients of the quadratic  $\lambda(s) := s G(s)$  (see companion website <sup>1</sup> for a derivation). Because the conditions of a particular branch might result in one or both of  $\hat{V}(s + y^{n-1})$  or  $\hat{V}(y^{n-1})$  collapsing to zero,  $\lambda(s)$  takes different forms in different branches. The coefficients  $\{a_{\lambda}, b_{\lambda}, c_{\lambda}\}$  for each branch are given in Table 1.

Table 1: Coefficients of the quadratic  $\lambda(s)$  for the branches in Figure 2.

Branch	$a_{\lambda}$	$b_{\lambda}$	$c_{\lambda}$
$y^{n-1} \leq 0, p = 0$	1	$z$	0
$y^{n-1} \leq 0, p \neq 0$	$1 + q a_p$	$z + q(2a_p y^{n-1} + b_p)$	$q V_p(y^{n-1})$
$y^{n-1} > 0, p \neq 0$	$1 + q a_p$	$z + q(2a_p y^{n-1} + b_p)$	$q \{V_p(y^{n-1}) - \hat{V}(y^{n-1})\}$
$y^{n-1} > 0, p = 0$	1	$z$	$-q \hat{V}(y^{n-1})$


 Figure 3: Logic to find the QSA segment index  $p$ .

 Figure 4: Collisions of a mass with a stiff barrier for two initial displacements: with the QSA method with  $N = 5$  (left), and with EXP (right). In both subplots, one of the simulations (blue) is time-shifted by  $\tau$  such that the trajectories obtained by the two simulations overlap until the collision instant. In all simulations,  $\kappa = 10^8$ ,  $\alpha = 1.3$ ,  $m = 10g$ ,  $v_0 = 2\text{ms}^{-1}$  and  $\Delta t = 1/44100$  s

#### 4.2.1. Segment finding procedure

In the above,  $p$  represents the index of the quadratic spline segment that corresponds to the interval in which  $y^{n+1}$  lies. In other words,

$$p := \begin{cases} \left\lceil \frac{y^{n+1}}{\Delta y} \right\rceil & : y^{n+1} > 0, \\ 0 & : \text{otherwise.} \end{cases} \quad (26)$$

Because  $y^{n+1} = s + y^{n-1}$  is a shifted version of  $s$ , finding the  $y$ -interval in which  $y^{n+1}$  lies is equivalent to finding the  $s$ -interval in which the root  $\tilde{s}$  of  $G(s)$  lies. Therefore, we take  $y_j$ ,  $j = 0, 1, 2, \dots, N-1$  as test values for  $y^{n+1}$ , or equivalently,  $s_j = y_j - y^{n-1}$  as test roots of  $G(s)$ , and evaluate  $\{G(s_j)\} \forall j$ . The index  $p$  is then determined as the one at which the sequence  $\{G(s_j)\}$  changes sign from negative to positive. See Figure 3 for a pictorial sketch of this logic. If  $G(s_0) > 0$ , then it means that  $\tilde{s} < s_0$ , which implies  $y^{n+1} < y_0 = 0$ , and hence  $p = 0$ . By a similar argument, for the other edge case where  $G(s_{N-1}) < 0$ , one can derive that  $p = N$ . This procedure involves at most  $N$  quadratic function evaluations at each time step, and therefore is not expected to be computationally intensive for a reasonably small  $N$ .

### 4.3. Numerical Simulation

Figure 4 depicts the displacement of a mass colliding with a stiff barrier as a function of time, simulated with the QSA method and EXP. For different initial displacements, the time-shifted trajec-

ries obtained with the QSA method overlap, similar to IMP (not shown in figure). In contrast, in the case of EXP the trajectories diverge after the collision instant, showing an instance of larger error of the scheme due to a non-unique phase-space trajectory (see introduction).

## 5. CASE STUDY II : HAMMER-STRING INTERACTION

For a given number of segments, the QSA is expected to be less accurate for a larger  $\alpha$ . Hence, piano hammer-string interaction, where  $\alpha \approx 2.5$  [1] (see introduction), serves as a musically relevant edge case to investigate at most how many segments ( $N$ ) are required for reasonably accurate simulations with the QSA method. To this end, we choose a stiff string model with simplified damping, omitting complex phenomena such as longitudinal modes, body coupling, etc., as in [8, 11]. The equations governing the motion of the hammer-string system [8] are given by

$$\rho A \partial_x^2 u = T \partial_x^2 u - EI \partial_x^4 u + 2\rho A (\eta_2 \partial_x^2 - \eta_0) \partial_t u + \mathcal{F}, \quad (27)$$

$$m_h d_t^2 w_h = -F, \quad (28)$$

where  $u = u(x, t)$  is the displacement of the stiff string at time  $t$  and position  $x$  along its length  $L$ .  $\rho$ ,  $A$ ,  $T$ ,  $E$ ,  $I$ ,  $\eta_2$  and  $\eta_0$  represent the string's mass density, area of cross section, tension, Young's modulus, moment of inertia, frequency-dependent and frequency-independent damping constants, while  $m_h$  and  $w_h = w_h(t)$  denote the mass and displacement of the hammer, respectively. Given that the hammer comes into contact with the string over a narrow region around a point, the contact force density  $\mathcal{F} = \mathcal{F}(x, t)$  can be modelled as

$$\mathcal{F} = \delta(x - x_h) F, \quad (29)$$

where the dirac delta function  $\delta(x - x_h)$  represents the lumped interaction of the hammer at a fixed position  $x_h$  along the length of the string, and  $F$  is the hammer-string contact force as defined in eq. (1), with the hammer-string compression variable  $y(t) := u(x_h, t) - w_h(t)$ . Further, we assume simply-supported string boundary conditions

$$u(0, t) = 0, \quad u(L, t) = 0, \quad (30)$$

$$\partial_x^2 u(0, t) = 0, \quad \partial_x^2 u(L, t) = 0. \quad (31)$$

### 5.1. Modal Expansion

To ensure accurate string resonance frequencies, following [8], the solution to (27) is expanded as

$$u(x, t) = \sum_{i=1}^M v_i(x) \tilde{u}_i(t), \quad (32)$$

where  $v_i(x) = \sin(\beta_i x)$  and  $\beta_i = i\pi/L$  are the mode shape and wave number of the  $i^{\text{th}}$  mode,  $\tilde{u}_i(t)$  is the  $i^{\text{th}}$  modal displacement, and  $M$  is the number of modes ( $M \rightarrow \infty$  for an exact solution). Eq. (32) can be written concisely in vector-matrix form as

$$u = \mathbf{v}^T \tilde{\mathbf{u}}, \quad (33)$$

where  $\mathbf{v} = \mathbf{v}(x)$  and  $\tilde{\mathbf{u}} = \tilde{\mathbf{u}}(t)$  are  $M \times 1$  vectors whose  $i^{\text{th}}$  elements are  $v_i(x)$  and  $\tilde{u}_i(t)$ , respectively. As done in [8, 10], substituting (32) in (27), pre-multiplying by  $\mathbf{v}^T$  and spatially integrating over  $[0, L]$  yields a set of modal ODEs

$$\tilde{m} d_t^2 \tilde{\mathbf{u}} + \mathbf{R} d_t \tilde{\mathbf{u}} + \mathbf{K} \tilde{\mathbf{u}} = F \mathbf{h}, \quad (34)$$

where  $\tilde{m} = \rho AL/2$  is the modal mass, and  $\mathbf{R}$  and  $\mathbf{K}$  are  $M \times M$  positive definite diagonal matrices, and

$$\mathbf{h} = \mathbf{v}(x_h) \quad (35)$$

is the  $M \times 1$  vector whose elements are the samples of the mode shapes at  $x_h$ . From (33) and (35), we can re-write  $y$  as

$$y = \mathbf{h}^T \tilde{\mathbf{u}} - w_h. \quad (36)$$

## 5.2. Discretisation

Following [8],  $M$  is chosen such that the highest mode frequency is lesser than the Nyquist frequency  $f_s/2$ , and the modal string ODEs (34) are discretised as

$$\tilde{m} \delta_t^2 \tilde{\mathbf{u}}^n + \hat{\mathbf{R}} \delta_t \tilde{\mathbf{u}}^n + \hat{\mathbf{K}} \mu_t^2 \tilde{\mathbf{u}}^n = F^n \mathbf{h}, \quad (37)$$

where  $F^n$  is the discretised contact force as given by eq. (22) with

$$y^n = \mathbf{h}^T \tilde{\mathbf{u}}^n - w_h^n \quad (38)$$

being the discrete version of the compression variable (36).  $\hat{\mathbf{R}}$  and  $\hat{\mathbf{K}}$  are  $M \times M$  diagonal matrices with diagonal entries representing the damping and stiffness constants that are adjusted to compensate for numerical dispersion [8], respectively. Further, the hammer dynamics (28) are discretised as

$$m_h \delta_t^2 w_h^n = -F^n. \quad (39)$$

From eqs. (37) and (39), the update equations are derived as

$$\tilde{\mathbf{u}}^{n+1} = \tilde{\mathbf{u}}^{n-1} - \tilde{\mathbf{z}} + \mathbf{C} \mathbf{h} F^n, \quad (40)$$

$$w_h^{n+1} = w_h^{n-1} - z_h - \xi_h F^n, \quad (41)$$

where  $\tilde{\mathbf{z}} = \mathbf{B} \tilde{\mathbf{u}}^{n-1} - \mathbf{A} \tilde{\mathbf{u}}^n$ ,  $z_h = 2(w_h^{n-1} - w_h^n)$ , and  $\xi_h = \Delta_t^2/m_h$ .  $\mathbf{A}$ ,  $\mathbf{B}$  and  $\mathbf{C}$  are  $M \times M$  diagonal matrices with the  $i^{\text{th}}$  diagonal element depending on the natural frequency and decay rate of the  $i^{\text{th}}$  mode (see [8] for more detail). Pre-multiplying (40) by  $\mathbf{h}^T$  and subtracting (41), we have the scalar update equation

$$\underbrace{y^{n+1} - y^{n-1}}_s + \underbrace{(\mathbf{h}^T \tilde{\mathbf{z}} - z_h)}_z - \underbrace{(\mathbf{h}^T \mathbf{C} \mathbf{h} + \xi_h)}_q F^n = 0, \quad (42)$$

which is of the same form as (18), and therefore we can solve for  $s$  non-iteratively by the branched-form solution of the QSA method (see Figure 2). Once we obtain  $s$ , we can update  $F^n$  from (42) as  $F^n = (s + z)/q$ , and then substitute the value of  $F^n$  in the string and hammer update equations (40) and (41) to update  $\tilde{\mathbf{u}}^{n+1}$  and  $w_h^{n+1}$ , respectively.

For the continuous and discrete energy balance equations of the hammer-string system, the reader is referred to [8].

## 5.3. Numerical Simulation

Single hammer-string collisions were simulated with string and hammer parameters set to the measurements reported in [1]. As an inverse measure of the QSA method's 'accuracy' (with respect to the IMP result) we define a log normalised difference  $\log_{10} (||\mathbf{F}_{\text{QSA}} - \mathbf{F}_{\text{IMP}}|| / ||\mathbf{F}_{\text{IMP}}||)$ , where  $\mathbf{F}_{\text{QSA}}$  and  $\mathbf{F}_{\text{IMP}}$  are the hammer forces as functions of time obtained by the QSA method and IMP, respectively. The hammer force evolutions in Figure 5 (left) suggest that generally the accuracy of the QSA method can be increased by reducing  $\Delta_y$ , and the plot over a range of  $\Delta_y$  in Figure 5 (right) confirms this trend.

Sounds of single hammer strikes were synthesised with low

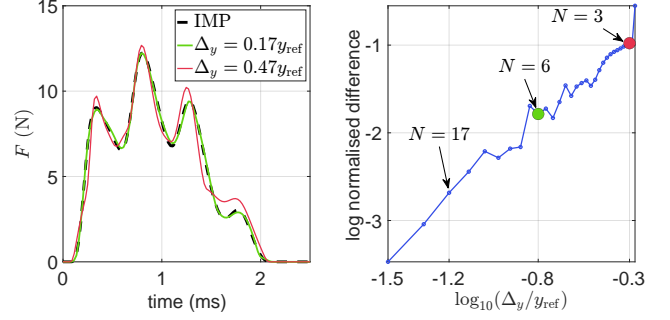


Figure 5: Left: Hammer force evolutions for a single hammer strike for two  $\Delta_y$  values (corresponding to  $N = 3$  (red) and  $N = 6$  (green)) compared with IMP (dashed black); Right: Log of the normalised difference between the hammer force signals obtained with the QSA method and IMP as a function of  $\log_{10}(\Delta_y/y_{\text{ref}})$ . Red and green circles correspond to the  $\Delta_y$  values in the left plot. The initial hammer velocity is  $3.5\text{ms}^{-1}$ ,  $\kappa = 4.5 \times 10^9$ ,  $\alpha = 2.5$  and  $\Delta_t = 1/44100$  s for all cases.

initial hammer velocity ( $1\text{ms}^{-1}$ ) and high string damping ( $\eta_0 = 8\text{s}^{-1}$ ), with both the IMP and QSA methods, and with  $N$  varied from 2 to 30 in the QSA case (see companion website<sup>1</sup>). Listening to these sounds indicates that differences are barely noticeable for  $N$  larger than 13. In less extreme cases than the hammer-string interaction, a smaller  $N$  is expected to suffice for reasonably accurate simulation. The optimal value of  $N$  needs to be determined on a case-by-case basis.

## 6. CASE STUDY III : SLIDE-STRING INTERACTION

Slide-string instrumentalists slide a cylindrical object (generally with their left hand) over a string to produce continuously-varying pitch patterns such as glissando and vibrato. In addition to this, they use the right-hand fingers to pluck/damp the string, and the left-hand finger trails the slide object to damp the non-speaking length of the string. The reason for choosing this case study is twofold: (a) from Hertzian contact theory,  $\alpha$  is around 1.5 in this case [5], therefore the QSA method can be tested with  $\alpha$  closer to unity, and (b) real-time performance can be tested with complex time-varying control inputs. To account for the time-varying slide contact and finger forces, we modify the model in (27) as [11]

$$\rho A \partial_t^2 u = T \partial_x^2 u - EI \partial_x^4 u + 2\rho A [\eta_2 \partial_x^2 - \eta_0] \partial_t u + \mathcal{F}_o + \mathcal{F}_l + \mathcal{F}_r, \quad (43)$$

where the string parameters are as before, and  $\mathcal{F}_o = \mathcal{F}_o(x, t)$ ,  $\mathcal{F}_l = \mathcal{F}_l(x, t)$  and  $\mathcal{F}_r = \mathcal{F}_r(x, t)$  are the slide contact, left-hand finger and right-hand finger force densities, respectively. Because the slide is cylindrical in shape, it comes into contact with the string over a narrow region and therefore, similar to the hammer-string case,  $\mathcal{F}_o$  can be modelled in a lumped fashion as

$$\mathcal{F}_o = \delta(x - x_o) F, \quad (44)$$

where  $x_o = x_o(t)$  is the slide contact position, and  $F$  is the slide-string contact force as defined in (1). As in [11], combining the excitation and damping action of the finger forces,  $\mathcal{F}_l$  and  $\mathcal{F}_r$  are phenomenologically modelled as

$$\mathcal{F}_\phi = \chi_\phi [F_{\phi,e} + F_{\phi,r}], \quad \phi = l, r \quad (45)$$

where  $F_{\phi,e} = F_{\phi,e}(t)$  and  $F_{\phi,R} = F_{\phi,R}(t)$  are the excitation and damping forces, and  $\chi_\phi = \chi_\phi(x, t)$  is the spatial distribution of finger  $\phi$ , respectively. The finger damping force is modelled as being proportional to the magnitude of the finger excitation force, i.e.,

$$F_{\phi,R}(t) = -r |F_{\phi,e}(t)| d_t u_\phi, \quad (46)$$

where  $r$  is a finger damping parameter, and

$$u_\phi = u_\phi(t) = \int_{x_\phi - W_\phi/2}^{x_\phi + W_\phi/2} \chi_\phi(x, t) u(x, t) dx \quad (47)$$

represents the spatially averaged string displacement at  $x_\phi$ . Motivated by [10], here the spatial distributions are modelled as

$$\chi_\phi(x, t) = \begin{cases} \frac{\pi}{2W_\phi} \cos\left[\frac{\pi}{W_\phi}(x - x_\phi)\right] & : |x - x_\phi| \leq \frac{W_\phi}{2}, \\ 0 & : \text{otherwise,} \end{cases} \quad (48)$$

where  $x_\phi = x_\phi(t)$  and  $W_\phi$  represent the centre and width of  $\chi_\phi(x, t)$ , respectively. In order to incorporate hand compliance in the model, the same string-slide-hand physical configuration as in [11] is chosen, featuring a slide-hand oscillator (SHO) formed by a slide mass  $m_o$ , a hand stiffness  $k_H$  and a hand damping parameter  $r_H$ , whose dynamics can be described by

$$m_o d_t^2 w_o + r_H d_t(w_o - w_H) + k_H(w_o - w_H) = -F, \quad (49)$$

where  $w_o = w_o(t)$  and  $w_H = w_H(t)$  are the displacements of the slide and hand, respectively. Note that  $w_H, x_o, x_\phi$  and  $F_{\phi,e}$  are control inputs. Further, simply supported conditions (eqs. (30) and (31)) are assumed for this case study as well, as in [11].

### 6.1. Modal Expansion

Following the same recipe as in section 5.1, we substitute (32) in (43), pre-multiply by  $\mathbf{v}^T$  and spatially integrate over  $[0, L]$  to obtain the modal ODEs

$$\tilde{m} d_t^2 \tilde{\mathbf{u}} + \mathbf{R} d_t \tilde{\mathbf{u}} + \mathbf{K} \tilde{\mathbf{u}} = F \mathbf{h}_o + \sum_{\phi=1,r} (F_{\phi,e} + F_{\phi,R}) \mathbf{h}_\phi, \quad (50)$$

where  $\tilde{m}$ ,  $\mathbf{R}$  and  $\mathbf{K}$  are as defined in section 5.1, and  $\mathbf{h}_o$  and  $\mathbf{h}_\phi$  are time-varying  $M \times 1$  vectors, defined by

$$\mathbf{h}_o = \mathbf{v}(x_o(t)), \quad \text{and} \quad (51)$$

$$\mathbf{h}_\phi = \int_{x_\phi - W_\phi/2}^{x_\phi + W_\phi/2} \mathbf{v}(x) \chi_\phi(x, t) dx, \quad (52)$$

with  $h_{\phi,i} = \frac{\pi^2 \cos(\beta_i W_\phi/2)}{\pi^2 - \beta_i^2 W_\phi^2} \sin(\beta_i x_\phi(t))$  being the  $i^{\text{th}}$  element of  $\mathbf{h}_\phi$ ,  $i = 1, 2, \dots, M$ . Further, substituting (33) in (47),  $u_\phi$  can be re-written as

$$u_\phi = \mathbf{h}_\phi^T \tilde{\mathbf{u}}. \quad (53)$$

As in the hammer-string case, from (33) and (51), we can write the slide-string compression variable  $y := u(x_o, t) - w_o$  as

$$y = \mathbf{h}_o^T \tilde{\mathbf{u}} - w_o. \quad (54)$$

### 6.2. Energy Balance

Here, pre-multiplying (50) by  $(d_t \tilde{\mathbf{u}})^T$  and (49) by  $d_t w_o$ , the following energy balance equation for the modal slide-string formulation is derived:

$$d_t H = (P_o + \sum_{\phi=1,r} P_\phi) - (Q_{\text{str}} + Q_o + \sum_{\phi=1,r} Q_\phi), \quad (55)$$

where

$$H = \frac{1}{2} \tilde{m} \|d_t \tilde{\mathbf{u}}\|^2 + \frac{1}{2} \tilde{\mathbf{u}}^T \mathbf{K} \tilde{\mathbf{u}} + \frac{1}{2} m_o (d_t w_o)^2 + \frac{1}{2} k_H (w_o - w_H)^2 + V \quad (56)$$

is the Hamiltonian, now with the extra potential energy term due to the hand stiffness  $k_H$ .  $P$  and  $Q$  are driving and dissipated powers, with subscripts  $o$ ,  $\text{str}$  and  $\phi$  representing the slide, the string and finger  $\phi$  respectively, and are given by

$$P_o = [r_H d_t w_o - k_H (w_o - w_H)] d_t w_H - F \partial_x u|_{x=x_o} d_t x_o, \quad (57)$$

$$P_\phi = (d_t \tilde{\mathbf{u}})^T \mathbf{h}_\phi [F_{\phi,e} - r |F_{\phi,e}| (d_t \mathbf{h}_\phi)^T \tilde{\mathbf{u}}], \quad (58)$$

$$Q_{\text{str}} = (d_t \tilde{\mathbf{u}})^T \mathbf{R} (d_t \tilde{\mathbf{u}}), \quad Q_o = r_H (d_t w_o)^2, \quad \text{and} \quad (59)$$

$$Q_\phi = r |F_{\phi,e}| \|(d_t \tilde{\mathbf{u}})^T \mathbf{h}_\phi\|^2. \quad (60)$$

Note that  $P_o$  involves terms proportional to the vertical and horizontal control input velocities  $d_t w_H$  and  $d_t x_o$ , and  $P_\phi$  contains a term due to the changing excitation position  $(d_t \mathbf{h}_\phi)$ . Such terms are expected when there is time-variance [11, 17].

### 6.3. Discretisation

The modal ODEs in (50) are discretised as

$$\tilde{m} \delta_t^2 \tilde{\mathbf{u}}^n + \hat{\mathbf{R}} \delta_t \tilde{\mathbf{u}}^n + \hat{\mathbf{K}} \mu_t^2 \tilde{\mathbf{u}}^n = F^n \mu_t \mathbf{h}_o^n + \sum_{\phi=1,r} (F_{\phi,e}^n + F_{\phi,R}^n) \mu_t \mathbf{h}_\phi^n, \quad (61)$$

where  $\hat{\mathbf{R}}$  and  $\hat{\mathbf{K}}$  are the damping and stiffness diagonal matrices which compensate for numerical dispersion as before. The finger damping force (eq. (46)) is discretised as

$$F_{\phi,R}^n = -r |F_{\phi,e}^n| \delta_t u_\phi^n, \quad (62)$$

where from (53) we have

$$u_\phi^n = (\mathbf{h}_\phi^n)^T \tilde{\mathbf{u}}^n. \quad (63)$$

The SHO dynamics (eq. (49)) can be discretised as

$$m_o \delta_t^2 w_o^n + r_H \delta_t (w_o^n - w_H^n) + k_H \mu_t^2 (w_o^n - w_H^n) = -F^n. \quad (64)$$

From (61) and (64), the following string and slide update equations can be derived:

$$\tilde{\mathbf{u}}^{n+1} = \tilde{\mathbf{u}}^{n-1} - (\tilde{\mathbf{z}} - \tilde{\mathbf{e}}) + \mathbf{C} \left\{ F^n \mu_t \mathbf{h}_o^n + \sum_{\phi=1,r} (F_{\phi,e}^n + F_{\phi,R}^n) \mu_t \mathbf{h}_\phi^n \right\}, \quad (65)$$

$$w_o^{n+1} = w_o^{n-1} - z_{\text{sho}} - q_{\text{sho}} F^n, \quad (66)$$

where

$$\tilde{\mathbf{z}} = \mathbf{B} \tilde{\mathbf{u}}^{n-1} - \mathbf{A} \tilde{\mathbf{u}}^n, \quad (67)$$

$$\tilde{\mathbf{e}} = \mathbf{C} \sum_{\phi=1,r} F_{\phi,e}^n \mu_t \mathbf{h}_\phi^n, \quad (68)$$

$$z_{\text{sho}} = \frac{2(\mathcal{K}_H + 1) w_o^{n-1} + (\mathcal{K}_H - 1) w_o^n - \xi_o \mathcal{T}_H^n}{1 + \mathcal{K}_H + \mathcal{R}_H}, \quad \text{and} \quad (69)$$

$$q_{\text{sho}} = \frac{\xi_o}{1 + \mathcal{K}_H + \mathcal{R}_H}, \quad (70)$$

with  $\mathbf{A}$ ,  $\mathbf{B}$  and  $\mathbf{C}$  as defined in section 5.2.  $\mathcal{K}_H = \frac{k_H \Delta_t^2}{4m_o}$ ,  $\mathcal{R}_H = \frac{r_H \Delta_t}{2m_o}$ , and  $\xi_o = \frac{\Delta_t^2}{m_o}$  are constants, and

$$\mathcal{T}_H^n = k_H \mu_t^2 w_H^n + r_H \delta_t \cdot w_H^n \quad (71)$$

is a ‘hand term’ fully determined by the control input  $w_H^n$  and the hand compliance parameters  $k_H$  and  $r_H$ .

#### 6.4. Solving for the Contact and Hand Damping Forces

To solve for the forces  $F^n$  and  $F_{\phi,R}^n$ , a similar approach as in [10] is taken. Pre-multiplying (65) by  $(\mathbf{h}_o^n)^T$  and  $(\mathbf{h}_\phi^n)^T$ ,  $\phi = 1, r$ , a set of three scalar equations is obtained as

$$s_\zeta = e_\zeta - z_\zeta + \theta_{\zeta o} F^n + \sum_{\phi=1,r} \theta_{\zeta \phi} F_{\phi,R}^n, \quad \zeta = o, l, r \quad (72)$$

where  $s_\zeta = (\mathbf{h}_\zeta^n)^T (\tilde{\mathbf{u}}^{n+1} - \tilde{\mathbf{u}}^{n-1})$ ,  $e_\zeta = (\mathbf{h}_\zeta^n)^T \tilde{\mathbf{e}}$ ,  $z_\zeta = (\mathbf{h}_\zeta^n)^T \tilde{\mathbf{z}}$ ,  $\theta_{\zeta o} = (\mathbf{h}_\zeta^n)^T \mathbf{C} \mu_t \mathbf{h}_o^n$  and  $\theta_{\zeta \phi} = (\mathbf{h}_\zeta^n)^T \mathbf{C} \mu_t \mathbf{h}_\phi^n$ . Now we use (62) to eliminate  $s_l$  and  $s_r$ . This allows expressing each of  $F_{1,R}^n$  and  $F_{r,R}^n$  as linear functions of  $F^n$  alone, which when substituted in the equation for  $s_o$  give us a single scalar string update equation

$$(\mathbf{h}_o^n)^T \tilde{\mathbf{u}}^{n+1} = (\mathbf{h}_o^n)^T \tilde{\mathbf{u}}^{n-1} - z_{\text{str}} + q_{\text{str}} F^n, \quad (73)$$

where  $z_{\text{str}}$  is a combined ‘history and excitation variable’ and  $q_{\text{str}}$  is a linear combination of  $\theta_{oo}$ ,  $\theta_{ol}$  and  $\theta_{or}$ . Finally, subtracting (66) from (73) yields

$$\underbrace{y^{n+1} - y^{n-1}}_s + \underbrace{(z_{\text{str}} - z_{\text{sho}})}_z - \underbrace{(q_{\text{str}} + q_{\text{sho}})}_q F^n = 0, \quad (74)$$

which is of the form in (18) and thus can be non-iteratively solved with the QSA method (see section 4.2). Once  $s$  is found,  $F^n = (s + z)/q$  can be computed, and this can then be used to update  $F_{\phi,R}^n$ ,  $\phi = 1, r$  from the linear relations obtained previously. The contact forces  $F^n$  and  $F_{\phi,R}^n$  are then substituted in (65) to update the modal displacements  $\tilde{\mathbf{u}}^{n+1}$ .

#### 6.5. Numerical Energy Balance

Pre-multiplying (61) by  $(\delta_t \cdot \tilde{\mathbf{u}}^n)^T$  and (64) by  $\delta_t \cdot w_o^n$ , a numerical energy balance that is analogous to the continuous domain balance (eq. (55)) can be derived as

$$\delta_t H^n = (P_o^n + \sum_{\phi=1,r} P_\phi^n) - (Q_{\text{str}}^n + Q_o^n + \sum_{\phi=1,r} Q_\phi^n), \quad (75)$$

where the numerical Hamiltonian

$$\begin{aligned} H^{n+\frac{1}{2}} = & \frac{1}{2} \left\{ \tilde{m}_l \|\delta_t \tilde{\mathbf{u}}^{n+\frac{1}{2}}\|^2 + (\mu_t \tilde{\mathbf{u}}^{n+\frac{1}{2}})^T \widehat{\mathbf{K}} (\mu_t \tilde{\mathbf{u}}^{n+\frac{1}{2}}) \right. \\ & \left. + m_o (\delta_t w_o^{n+\frac{1}{2}})^2 + k_H (\mu_t^2 (w_o^{n+\frac{1}{2}} - w_H^{n+\frac{1}{2}}))^2 \right\} \\ & + \mu_t V(y^{n+\frac{1}{2}}). \end{aligned} \quad (76)$$

The numerical driving and dissipation powers are

$$\begin{aligned} P_o^n = & \{ r_H \delta_t \cdot w_o^n - k_H \mu_t^2 (w_o^n - w_H^n) \} \delta_t \cdot w_H^n \\ & - F^n \left\{ \frac{\delta_t \cdot (\mu_t \cdot u(x_o^n, n \Delta_t))}{\delta_t \cdot x_o^n} \right\} \delta_t \cdot x_o^n, \end{aligned} \quad (77)$$

$$P_\phi^n = \delta_t \cdot \tilde{\mathbf{u}}^n \mu_t \mathbf{h}_\phi^n \left\{ F_{\phi,e}^n - r |F_{\phi,e}^n| (\delta_t \cdot \mathbf{h}_\phi^n)^T \mu_t \cdot \tilde{\mathbf{u}}^n \right\}, \quad (78)$$

$$Q_{\text{str}}^n = (\delta_t \cdot \tilde{\mathbf{u}}^n)^T \widehat{\mathbf{R}} (\delta_t \cdot \tilde{\mathbf{u}}^n), \quad Q_o^n = r_H (\delta_t \cdot w_o^n)^2, \quad \text{and} \quad (79)$$

$$Q_\phi^n = r |F_{\phi,e}^n| \left\{ (\delta_t \cdot \tilde{\mathbf{u}}^n)^T \mu_t \cdot \mathbf{h}_\phi^n \right\}^2. \quad (80)$$

#### 6.6. Numerical Simulation

Slide gestures were simulated with the same control inputs and parameters as in Figure 3 in [11] and a three-slide musical phrase comprising slide attachment/detachment, glissando and vibrato was synthesised with both IMP and the QSA method (with  $N = 5$ ). With both methods, the slide-string rattle could be heard, particularly just after the pluck releases in the sound examples [see companion website], and the time-variation of the harmonics is similar to the finite-difference (FD) model in [11] (which uses the branched-form solution restricted to unity  $\alpha$ ), with the excitation sustained via the energy injected by the time-variance of the control inputs.

For  $\alpha = 1$  there is no audible difference between the sounds from the QSA method and IMP - this is expected because the QSA is exact in this case. With  $\alpha = 1.5$ , the collisions are slightly softer as expected with both methods. However, a subtle high frequency ‘glitter’-like artefact can be heard in the background with the QSA method. This effect can be subdued by reducing  $N$ , which indicates that this is likely caused by jumps in the derivatives of the contact force introduced by segment switching.

A notable feature of the simulation with the model proposed here is the disappearance of the noise-like components that were mistaken for ‘slide noise’ in [11]. However, this has little to do with QSA, and can be attributed to the spatially exact mode shape sampling at the contact points in the modal formulation, which avoids the non-smooth (de)-interpolation used in [11]. For sound examples, the reader is referred to the companion website<sup>1</sup>.

#### 6.7. Computational Efficiency

Following [8, 17], real-time factor (RTF) is defined as the amount of time that passes with the computation of one second of audio output. With the assumption that the control parameters vary relatively slowly over time, in a Matlab simulation these were updated blockwise for every  $N_b$  samples, employing linear interpolation as in [17]. Average RTFs of a few updates that are part of the simulation loop are listed in Table 2. Note that the average RTF of the QSA solver is about 0.045, which shows that the proposed method does not contribute much to the runtime. The RTF of updates of the time-varying parameters (such as the mode shape vectors) reduces as  $N_b$  is increased, as some of the computationally expensive operations are performed only once per block. Including the time taken to memorise values, the RTF of the whole simulation is about 0.2 for  $N_b = 64$ .

### 7. CONCLUDING REMARKS

By means of approximating the contact potential curve with a quadratic spline, a non-iterative technique for the update of an NLI scheme



Table 2: Average RTFs of QSA solver and updates within the simulation loop for various values of  $N_b$

$N_b$	8	16	32	64
QSA solver	0.050	0.047	0.041	0.040
Parameter updates	0.104	0.088	0.074	0.071
Linear string update	0.025	0.024	0.022	0.022

is proposed for the specific case of lumped collisions. The approximation retains key properties of the underlying Hertzian power law such as convexity and non-negativity, and thereby preserves salient features including: (a) the energy-stable behaviour of the continuous-domain system, and (b) the uniqueness of solution of the NLI scheme. Initial value problem simulations indicate that the uniqueness of the phase-space trajectories of the NLI scheme is also kept intact. Results on string-object interactions show that accuracy can be controlled by varying the number of spline segments  $N$ , and also suggest that a small  $N$  may be sufficient for musical instrument simulations. Further, a modal slide-string model is introduced, and continuous and numerical energy balances are derived. Simulations employing the proposed method demonstrate real-time feasibility under time-varying control inputs. Thus, without compromising on efficiency, the proposed method extends the branched-form solution to the case of non-unity  $\alpha$ , side-stepping the issues associated with existing collision modelling techniques.

Nevertheless, as the adage goes - "there is no such thing as a free lunch", and the above benefits come with a few drawbacks. Firstly, the proposed method is not readily extendable to distributed collisions. This is because application of the method in a distributed scenario generally results in a system of coupled quadratic equations, and solving such a system is non-trivial. An exception to this is the specific case of a barrier with aligned grid [9], which is of little practical significance as this is often not the case in musical applications, for example, in guitar string/fretboard collisions, fret positions may not be aligned with the (regularly spaced) string grid. Further, due to the non-smooth nature of the contact force approximation, the latent higher-order derivative structure is broken and this likely causes the production of (mildly audible) high-frequency artefacts in sustained contact simulations. Although this issue is seemingly alleviated by deploying cubic and higher-order splines instead of quadratic splines, doing so would raise non-trivial questions around ensuring convexity of the approximation. Moreover, the authors' attempts at computing robust numerical solutions of higher-order polynomial equations indicate that it (a) is more computationally expensive than solving quadratic equations, and (b) is non-trivial, as finite precision could lead to large errors. The inquisitive reader is encouraged to refer to [18] in order to appreciate the intricacies of cubic equation solving.

## 8. REFERENCES

[1] A. Chaigne and A. Askenfelt, "Numerical simulations of piano strings. I. A physical model for a struck string using finite difference methods," *J. Acoust. Soc. Am.*, vol. 95, pp. 1112–1118, 1994.

[2] A. Chaigne and V. Doutaut, "Numerical simulations of xylophones. I. Time domain modeling of vibrating bars," *J. Acoust. Soc. Am.*, vol. 101, pp. 539–557, 1997.

[3] C. Vyasrayani, S. Birkett, and J. McPhee, "Modeling the

dynamics of a vibrating string with a finite distributed unilateral constraint: Application to the sitar," *J. Acoust. Soc. Am.*, vol. 125, no. 6, pp. 3673–3682, 2009.

[4] S. Bilbao, "Time domain simulation and sound synthesis for the snare drum," *J. Acoust. Soc. Am.*, vol. 131, no. 1, pp. 914–925, 2012.

[5] V. L. Popov, *Contact Mechanics and Friction: Physical Principles and Applications*, Springer, Heidelberg, New York, 2010.

[6] S. Bilbao, A. Torin, and V. Chatziioannou, "Numerical modeling of collisions in musical instruments," *Acta Acustica united with Acustica*, vol. 101, no. 1, pp. 155–173, 2015.

[7] M. Ducceschi and S. Bilbao, "Non-iterative solvers for non-linear problems: the case of collisions," in *Proc. 22nd Int. Conf. on Digital Audio Effects (DAFx-19)*, 2019, pp. 17–24.

[8] M. van Walstijn, V. Chatziioannou, and A. Bhanuprakash, "Implicit and explicit schemes for energy-stable simulation of string vibrations with collisions: Refinement, analysis, and comparison," *J. Sound and Vibration*, vol. 569, pp. 117–1968, 2024.

[9] S. Bilbao, "Numerical modeling of string barrier collisions," in *Proc. Int. Symp. Musical Acoustics (ISMA 2014)*, 2014.

[10] M. van Walstijn, J. Bridges, and S. Mehes, "A real-time synthesis oriented tanpura model," in *Proc. Int. Conf. Digital Audio Effects (DAFx-16)*, 2016, pp. 175–182.

[11] A. Bhanuprakash, M. van Walstijn, and P. Stapleton, "A finite difference model for articulated slide-string simulation," in *Proc. Int. Conf. Digital Audio Effects (DAFx-20)*, 2020, pp. 195–202.

[12] D. Russell and T. Rossing, "Testing the nonlinearity of piano hammers using residual shock spectra," *Acustica*, vol. 84, no. 5, pp. 967–975, Sept. 1998.

[13] L. Rhaouti, A. Chaigne, and P. Joly, "Time-domain modeling and numerical simulation of a kettledrum," *J. Acoust. Soc. Am.*, vol. 105, no. 6, pp. 3545–3562, June 1999.

[14] A. Stulov, "Hysteretic model of the grand piano hammer felt," *J. Acoust. Soc. Am.*, vol. 97, no. 4, pp. 2577–2585, April 1995.

[15] R. H. Bartels, J. C. Beatty, and B. A. Barsky, *An introduction to splines for use in computer graphics & geometric modeling*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1987.

[16] V. Chatziioannou and M. van Walstijn, "Energy conserving schemes for the simulation of musical instrument contact dynamics," *J. Sound and Vibration*, vol. 339, pp. 262–279, 2015.

[17] M. van Walstijn, V. Chatziioannou, and A. Bhanuprakash, "Tunable collisions: Hammer-string simulation with time-variant parameters," in *Proc. Int. Conf. Digital Audio Effects (DAFx-23)*, 2023.

[18] J. F. Blinn, "How to solve a cubic equation, part 5: Back to numerics," *IEEE Computer Graphics and Applications*, vol. 27, no. 3, pp. 78–89, 2007.