DMRN+20: DIGITAL MUSIC RESEARCH NETWORK
ONE-DAY WORKSHOP 2025

KING'S COLLEGE LONDON
TUE 16 DECEMBER 2025

Differentiable Karplus-Strong

Pablo Tablas de Paula, David Marttila and Joshua D. Reiss

Centre for Digital Music, Queen Mary University of London, United Kingdom, ptabladsdepaula@qmul.ac.uk

***Abstract*— We present a differentiable, time-domain implementation of an extended Karplus–Strong algorithm (KSA) that can be optimised end-to-end within a Differentiable DSP (DDSP) paradigm. We demonstrate its capability for instrument modelling by reconstructing an acoustic guitar dataset, where our proposed Gradient Descent (GD) baseline outperforms a Genetic Algorithm (GA). Lastly, we identify limitations tied to unmodelled inharmonicity and nonlinear behaviour.**

I. DIFFERENTIABLE KARPLUS-STRONG

Since Engel *et al.* introduced the DDSP paradigm, many have extended the approach to specific synthesis techniques [1, 2]. We propose to use DDSP to optimise KSA for instrument modelling. KSA discretises the 1D wave equation using a delay line with an *averaging loop filter*:

$$y[n] = \tfrac{1}{2}\big(y[n-L] + y[n-L-1]\big), \tag{1}$$

where L is the length of the delay line in samples and defines pitch, obtained by $L = f_s/f_0$, where f_s and f_0 are the sampling and desired fundamental frequencies [3].

Previously, *decay*, in the form of a gain stage multiplied by this average filter, was optimised through GD by minimising Multi-Scale Spectral Loss ($\mathcal{L}_{\text{MSS}}$) [2]. We extend this with *Lagrange Interpolation* for fractional L and replace the averaging filter with a 1st-order-IIR filter for richer frequency-dependent decay [4]. Lastly, we add *distance* between pluck and listener, *pluck position* in the string and playing *dynamics* with a gain stage, comb and low-pass filters respectively preceding the KSA [5]. Our implementation is fully in the time-domain, both at optimisation and inference thanks to new developments in IIR filters for DDSP [6], allowing naturally for time-varying f_0 and filter coefficients. In practice, the excitation parameters are averaged to be piecewise constant between plucks, while the string parameters vary per sample. A diagram of the complete system is shown in Figure 1.

II. EVALUATION

We optimise our architecture with GD and a GA computationally about one order of magnitude higher, both minimising $\mathcal{L}_{\text{MSS}}$, as configured in [7], of an acoustic guitar subset with no reverberation and pitches within the range E2-E6 (N=240) [8]. We obtain f_0 with *Yin* and place bursts of length L samples at indices flagged by an *onset detector*[1] [9]. We find our proposed GD optimisation outperforms GA across $\mathcal{L}_{\text{MSS}}$ (the same as used for optimisation), Mel-frequency cepstral coefficient (MFCC) and two perceptually-aligned embedding distances [10], as shown in Table 1.

Table 1: Baseline Comparison. Lower values, in bold, are better.

Baseline	$\mathcal{L}_{\text{MSS}}$	MFCC	Encodec	MS-CLAP
Gradient Descent	**0.79 ± 0.11**	**2.32 ± 1.50**	**19.89**	**111.30**
Genetic Algorithm	1.43 ± 0.31	4.10 ± 1.75	216.93	346.93

Secondly, we assess independent pitch and dynamics ranges using oracle f_0. Table 2 shows our system underperforms in the lower register and at higher dynamics, precisely where inharmonicity and nonlinear behaviour—unaccounted for in our model—are more prominent in acoustic guitars [11]. Additionally, we do not model the guitar body, which is especially relevant at 100 and 200 Hz, where the Helmholtz and plate modes reside [4]. Future work should address this: body resonances can be efficiently captured with *warped parallel filters*, and inharmonic behaviour can be introduced with all-pass filters in the string loop [11, 12]. Predicting time-varying inharmonicity with neural networks to model nonlinearities could also be an interesting direction for further research.

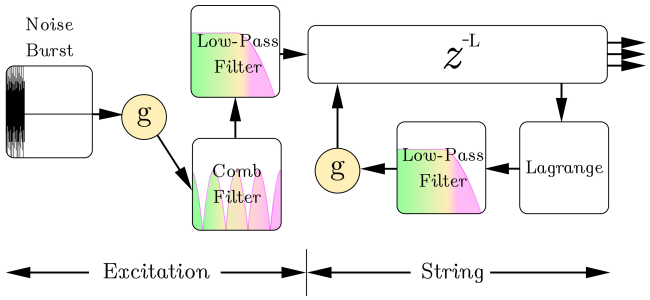

Figure 1: Proposed architecture, where coloured blocks are optimised by gradient descent. *Lagrange* is short for Lagrange interpolation, while *g* stand for gain stages.

Table 2: $\mathcal{L}_{\text{MSS}}$ mean difference to overall mean (0.62) across MIDI velocities and octaves for our GD baseline. Negative values are better.

↓ Oct. Vel.→	25	50	75	100	127	mean
E2–E3	+0.27	+0.23	+0.33	+0.30	+0.36	+0.30
E3–E4	-0.02	+0.07	-0.02	+0.07	+0.01	+0.02
E4–E5	-0.18	-0.10	-0.05	-0.13	-0.08	-0.11
E5–E6	-0.19	-0.26	-0.27	-0.18	-0.11	-0.20
mean	-0.03	-0.01	+0.00	+0.02	+0.05	0.00

III. REFERENCES

[1] J. Engel *et al.*, "DDSP: Differentiable digital signal processing," in *Proc. ICLR*, Jan. 2020.

[2] B. Hayes *et al.*, "A review of differentiable digital signal processing for music and speech synthesis," *Front. Signal Process.*, vol. 3, Jan. 2024.

[3] K. Karplus and A. Strong, "Digital synthesis of plucked-string and drum timbres," *Computer Music J.*, vol. 7, no. 2, pp. 43–55, 1983.

[4] V. Välimäki *et al.*, "Physical modeling of plucked string instruments with application to real-time sound synthesis," *J. Audio Eng. Soc.*, vol. 44, no. 5, p. 331–353, May 1996.

[5] D. A. Jaffe and J. O. Smith, "Extensions of the Karplus-Strong plucked-string algorithm," *Computer Music J.*, vol. 7, no. 2, pp. 56–69, 1983.

[6] C.-y. Yu *et al.*, "Differentiable all-pole filters for time-varying audio systems," in *Proc. of DAFx*, 2024.

[7] S. Schwär *et al.*, "Multi-scale spectral loss revisited," *IEEE Signal Process. Lett.*, vol. 30, pp. 1712–1716, 2023.

[8] J. Engel *et al.*, "Neural audio synthesis of musical notes with wavenet autoencoders," in *Proc. ICML*, 2017, pp. 1068–1077.

[9] B. McFee *et al.*, "librosa: Audio and music signal analysis in python." *SciPy*, vol. 2015, pp. 18–24, 2015.

[10] M. Tailleur *et al.*, "Correlation of fréchet audio distance with human perception of environmental audio is embedding dependent," in *32nd European Signal Process. Conf. (EUSIPCO)*, 2024, pp. 56–60.

[11] J. S. Abel *et al.*, "Robust, efficient design of allpass filters for dispersive string sound synthesis," vol. 17, no. 4, pp. 406–409, 2010.

[12] B. Bank, "Direct design of parallel second-order filters for instrument body modeling," in *Proc. of ICMC*, 2007.

This work was supported by UKRI grant EP/S022694.

[1] Implementation, examples and experiments can be found at `https://ptablasdpaula.github.io/DiffKarplusStrong_DMRN20/`.