PULSE-IT: LIGHTWEIGHT AND EXPRESSIVE SYNTHESIS OF WIND INSTRUMENT PLAYING

Christian Dittmar[1] **Johannes Zeitler**[2] **Stefan Balke**[2]
Simon Schwär[2] **Meinard Müller**[2]
[1] Fraunhofer Institute for Integrated Circuits IIS, Erlangen, Germany
[2] International Audio Laboratories, Erlangen, Germany
christian.dittmar@iis.fraunhofer.de

Late-Breaking / Demo Session Extended Abstract, ISMIR 2025 Conference

ABSTRACT

Wind instruments enable highly expressive performances through playing techniques such as vibrato, slurs, and growl. In this demo, we introduce PULSE-IT, a lightweight synthesis method that combines simple signal processing components with data-driven control signals. Based on a pulsetable oscillator and time-variant filtering, PULSE-IT supports both re-synthesis and cross-synthesis (timbre transfer) of expressive wind instrument playing at low computational cost. Despite its simplicity, the method yields convincing results and provides a practical alternative to more complex neural approaches. Audio examples and interactive visualizations are available at our accompanying website [1].

1. INTRODUCTION AND RELATED WORK

Synthesizing expressive wind instrument performances remains a challenging task due to the continuous, nuanced control wind instrument players exert over pitch, dynamics, and timbre. In recent years, large neural models have achieved breakthrough results in the synthesis of music signals from textual descriptions [1], symbolic music representations [2, 3], or audio feature sequences [4, 5]. However, these neural audio synthesis models are often heavy in terms of computational cost and required training data.

In contrast, interpretable and efficient alternatives are found in traditional methods such as sample-concatenative synthesis [6] and physical modeling [7–10]. While these are frequently dismissed as insufficiently expressive, commercial products like SampleModeling [2], AudioModeling [3], and AcousticSamples [4] demonstrate that meticulously engineered instrument models and control schemes can yield musically convincing results. Although technical details behind these tools are usually proprietary, the wind instrument models published by Joel Blanco Berg [5] are open patches whose inner workings can be reproduced.

Our late-breaking demo revisits a lineage of pulse-based synthesis techniques [11–13] and combines them with ideas from early data-driven approaches [14–17]. In particular, we focus on pulse forming synthesis [13], as used in the electronic wind instruments Martinetta [6] and Variophon [7]. In our variant of pulse forming, we use arrays of single-period pulses that carry the spectral envelope of wind instruments played at different pitches (henceforth called pulsetable oscillator). Drawing inspiration from the simplicity of analog synthesis [7], we implement a lightweight and modular DSP pipeline whose core components are driven by control signals derived from symbolic inputs or real-world instrument recordings.

Our contributions are: First, a lightweight synthesis pipeline feeding a pulsetable oscillator through a time-variant low-pass filter. Second, a demonstration how to realize expressive playing techniques such as vibrato, growl, and slurs via micro-modulations. Third, a proof-of-concept for cross-synthesis (timbre transfer) of real instrument recordings without any neural training procedures.

2. METHOD

2.1 Overview

As depicted in Fig. 1, the PULSE-IT system consists of four main blocks: **(1)** Control signal generators, **(2)** Pulsetable oscillator, **(3)** Time-variant low-pass filter, and **(4)** Reverb post-processing. Essential to our method is the pulsetable oscillator that generates a monophonic pulse train. Besides the target instrument selection, its only parameter is the fundamental frequency (F_0), reciprocal to the spacing between the pulses (period duration T_0). The main idea is that single-period pulses (see Fig. 2) for each chromatic note playable on the target instrument are stored in the pulsetable, indexed by their reference F_0^r. For any input target frequency F_0^t, the entry with the smallest difference $|F_0^t - F_0^r|$ is retrieved and used for the pulse train

[1] https://www.audiolabs-erlangen.de/resources/MIR/2025_DittmarZBSM_WindInstrumentSynth_ISMIR-LBD
[2] https://www.samplemodeling.com
[3] https://audiomodeling.com/swam-engine
[4] https://www.acousticsamples.net
[5] https://lydoel.gumroad.com
[6] https://muwiserver.univie.ac.at/martinetta
[7] http://www.variophon.de/vario_e.htm

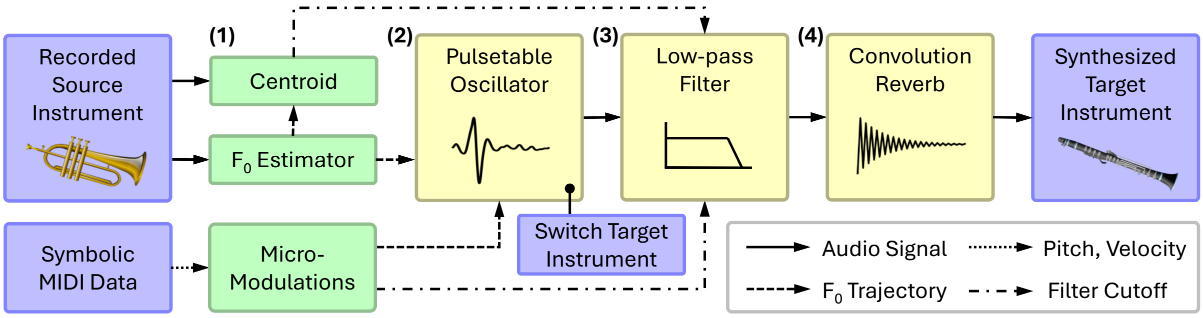

Figure 1. PULSE-IT system overview: purple boxes represent data input and output, green are control signal generators and yellow are DSP blocks.

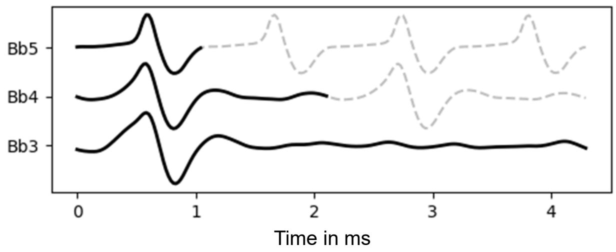

Figure 2. Example for selected single-period pulses of a Bb trumpet stored in the pulsetable oscillator.

generation. In practice, the selected pulse waveform is just read out element by element from the pulsetable and sent to the output of the oscillator, without any modification. Only in case $F_0^t \ll F_0^r$, zero padding is applied since there are not enough elements in the array. Conversely, if $F_0^t \gg F_0^r$ the pulse waveform is reset to its first element before having been read out completely. This can of course lead to undesirable phase jumps and should be avoided.

Simply put, those single-period pulses are short waveform snippets taken from the stable part of single note recordings. The snippet length is determined by the period duration of the played tone. Similar to earlier works [16], we source them from anechoic recordings of the University of Iowa Musical Instrument Samples (MIS) [8] database [18]. In Fig. 2, we show three pulses extracted from Bb trumpet tones played at highest intensity (fortissimo). The bold curves show the pulse shape, the semi-transparent dashed curves show how the original recording continues beyond the snippet of interest. For visualization purposes, we picked notes that are spaced three octaves apart. Relative to the pulse of Bb3 ($F_0^r \approx 233$ Hz, $T_0^r \approx 4.3$ ms), Bb4 is only half as long, and Bb5 only a quarter.

As discussed in [11–13], a single-period pulse encodes the instantaneous spectral envelope, which is strongly correlated to the timbre of the resulting tone. Since wind instruments commonly exhibit increasing amplitude of upper harmonics (brightness) with increasing intensity [9,14,19], we store only pulses corresponding to the highest intensity of the target instrument inside the pulsetable. The subsequent low-pass filter (sixth-order Butterworth with a slope of 36 dB per octave) is then used to attenuate higher harmonics, emulating tones with lower intensity. The final reverb block after the low-pass filter is then used to give a more believable impression of the instrument being played within a real acoustic environment (e.g., a concert hall).

2.2 Control Signals for Micro-Modulations

Wind instrument players exert continuous control over pitch and brightness, with transitions between notes strongly influenced by their context. In order to mimic expressive articulations and to achieve believable synthesis results, subtle fluctuations have to be applied both to the fundamental frequency F_0^t and the cutoff F_c of the low-pass filter. The term micro-modulations has been used for this principle in [13]. Due to space-limitations, we refer to our accompanying website [1] for examples on how to realize plausible micro-modulations.

2.3 Timbre Transfer (Cross-Synthesis)

Instead of generating artificial control signals for F_0^t and F_c, they can also be extracted from a source instrument recording (e.g., trumpet) and applied to a different target instrument (e.g., clarinet). To this end, we extract control signals (F_0 trajectory, spectral centroid) from monophonic instrument recordings using standard music processing tools (e.g., YIN, spectral centroid tracking). The resulting cross-synthesis timbre transfer is similar to what DDSP [4] achieves, but without neural audio model training. This simple structure allows us to isolate and control expressive parameters more transparently than in black-box neural systems and even allows for combining extracted and generated control signals flexibly.

2.4 Results, Examples, and Future Work

We present various audio examples and interactive visualizations on our accompanying website [1]. Key examples include re-synthesis and cross-synthesis of expressive phrases from corpora like ChoraleBricks [20], URMP [21], and Bach10 [22]. Furthermore, we include examples of generated articulations based on symbolic inputs, that are surprisingly expressive and plausible. Future research directions include perceptual listening tests, integration into interactive applications (e.g., music education), extension to other instrument families, and combination with machine-learning based performance modeling for increased expressivity.

[8] https://theremin.music.uiowa.edu/MIS.html

3. ACKNOWLEDGMENTS

The International Audio Laboratories Erlangen are a joint institution of the Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU) and Fraunhofer Institute for Integrated Circuits IIS. This work was partly funded by the Deutsche Forschungsgemeinschaft (DFG; German Research Foundation) under grant numbers 500643750 (MU 2686/15-1) and 555525568 (MU 2686/18-1).

4. REFERENCES

[1] J. Copet, F. Kreuk, I. Gat, T. Remez, D. Kant, G. Synnaeve, Y. Adi, and A. Défossez, "Simple and controllable music generation," in *Proceedings of the Annual Conference on Neural Information Processing Systems (NeurIPS)*, New Orleans, LA, USA, December 2023.

[2] E. Cooper, X. Wang, and J. Yamagishi, "Text-to-speech synthesis techniques for midi-to-audio synthesis," in *Proceedings of the ISCA Speech Synthesis Workshop (SSW)*, Budapest, Hungary, August 2021, pp. 130–135.

[3] B. Maman, J. Zeitler, M. Müller, and A. H. Bermano, "Multi-aspect conditioning for diffusion-based music synthesis: Enhancing realism and acoustic control," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2024.

[4] J. H. Engel, L. H. Hantrakul, C. Gu, and A. Roberts, "DDSP: Differentiable digital signal processing," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020.

[5] Y. Wu, E. Manilow, Y. Deng, R. Swavely, K. Kastner, T. Cooijmans, A. C. Courville, C. A. Huang, and J. H. Engel, "MIDI-DDSP: detailed control of musical performance via hierarchical modeling," in *Proceedings of the International Conference on Learning Representations (ICLR)*, April 2022.

[6] E. Maestre, R. Ramírez, S. Kersten, and X. Serra, "Expressive concatenative synthesis by reusing samples from real performance recordings," *Computer Music Journal*, vol. 33, no. 4, pp. 23–42, 2009.

[7] R. W. Berry, "Experiments in computer controlled acoustic modelling (A step backwards??)," in *Proceedings of the International Computer Music Conference, (ICMC)*. Cologne, Germany: Michigan Publishing, September 1988.

[8] J. O. Smith, "Physical modeling using digital waveguides," *Computer Music Journal*, vol. 16, no. 4, pp. 74–91, 1992.

[9] D. H. Keefe, "Physical modeling of wind instruments," *Computer Music Journal*, vol. 16, no. 4, pp. 57–73, 1992.

[10] W. D'haes and X. Rodet, "A new estimation technique for determining the control parameters of a physical model of a trumpet," in *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, London, UK, 2003.

[11] X. Rodet, "Time-domain formant-wave-function synthesis," *Computer Music Journal*, vol. 8, no. 3, pp. 9–14, 1984.

[12] C. Roads, "Introduction to pulsar synthesis," *The Journal of the Acoustical Society of America*, vol. 109, no. 5_Supplement, pp. 2401–2401, 2001.

[13] M. Oehler, "Die digitale Impulsformung als Werkzeug für die Analyse und Synthese von Blasinstrumentenklängen," Doctoral Dissertation, Hochschule für Musik und Theater Hannover, 2007.

[14] A. Horner and J. Beauchamp, "Synthesis of trumpet tones using a wavetable and a dynamic filter," *Journal of the Audio Engineering Society*, vol. 43, no. 10, pp. 799–812, 1995.

[15] I. Derenyi and R. B. Dannenberg, "Synthesizing trumpet performances," in *Proceedings of the International Computer Music Conference (ICMC)*. Ann Arbor, Michigan, USA: Michigan Publishing, October 1998.

[16] M. Rocamora, E. López, and L. Jure, "Wind instruments synthesis toolbox for generation of music audio signals with labeled partials," in *Proceedings of the Brazilian Symposium on Computer Music (BSCM)*, Recife, Pernambuco, Brasil, September 2009, pp. 1–12.

[17] N. Hu, "Automatic construction of synthetic musical instruments and performers," Doctoral Dissertation, Carnegie Mellon University, 2018.

[18] E. F. Feichtner and B. Edler, "Description of the single note database SNDB," in *Proceedings of the Audio Engineering Society Convention*. New York City, USA: Audio Engineering Society, October 2018.

[19] M. Campbell, J. Gilbert, and A. Myers, *The science of brass instruments*. Springer, 2021, vol. 436.

[20] S. Balke, A. Berndt, and M. Müller, "ChoraleBricks: A modular multitrack dataset for wind music research," *Transaction of the International Society for Music Information Retrieval (TISMIR)*, vol. 8, no. 1, pp. 39–54, 2025.

[21] B. Li, X. Liu, K. Dinesh, Z. Duan, and G. Sharma, "Creating a multitrack classical music performance dataset for multimodal music analysis: Challenges, insights, and applications," *IEEE Transactions on Multimedia*, vol. 21, no. 2, pp. 522–535, 2019.

[22] Z. Duan, B. Pardo, and C. Zhang, "Multiple fundamental frequency estimation by modeling spectral peaks and non-peak regions," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 8, pp. 2121–2133, 2010.