CHORDSNEST: DEVELOPING A MULTIPHONICS PALETTE FOR MAXSCORE

Aigerim Seilova
Hochschule für Musik und Theater Hamburg
aigerim.seilova@hfmt-hamburg.de

James Tsz-Him Cheung
Hochschule für Musik und Theater Hamburg
tsz.him.cheung@hfmt-hamburg.de

Georg Hajdu
Hochschule für Musik und Theater Hamburg
georg.hajdu@hfmt-hamburg.de

ABSTRACT

ChordsNest is a new extension for MaxScore (a notation environment for Max) that provides an integrated multiphonics workspace. A multiphonic is an extended technique where multiple notes are produced simultaneously by a typically monophonic instrument. This tool addresses the difficulty of notating and accessing such by uniting chord-like notation, fingering diagrams, and sample-based playback in one interface. Over the course of the project, 136 B♭ clarinet and 130 bass clarinet multiphonics were systematically recorded, categorized, and incorporated into the user interface, allowing composers to search and select multiphonics via tags. ChordsNest offers immediate audio feedback for each multiphonic and automatically displays the corresponding fingering diagram on the score, removing the manual workaround traditionally needed. By leveraging MaxScore's message-driven architecture and its integration in the Max environment, the system seamlessly unites notation, visualization, and playback. This significantly enhances the workflow for composers working in contemporary and electroacoustic idioms. The paper outlines the development methodology, implementation details, and discusses future extensions of the ChordsNest environment.

1. INTRODUCTION

Multiphonics are extended instrumental techniques in which a performer produces more than one pitch simultaneously on a typically monophonic instrument, such as the clarinet [1]. These complex, often fragile sounds have become increasingly common in contemporary music, offering rich sonic textures and expressive potential. However, working with multiphonics presents challenges in both notation and performance.

Traditionally, composers refer to catalogues such as Bartolozzi's *New Sounds for Woodwind* [2] or Richards' *The Clarinet of the Twenty-First Century* [3] to identify usable fingerings. The actual sonic result, however, is often unpredictable, and performers may not reliably reproduce the intended sound. Notation typically involves representing the resulting chord and manually inserting fingering graphics – often via external graphic tools or custom scripting solutions. These workflows are cumbersome and fragmented, and the actual sonic result is usually only heard at rehearsal.

ChordsNest was developed to address these issues by providing an integrated solution that unifies chordal notation, fingering diagrams, and immediate playback within the MaxScore environment. It is designed to support composers and performers alike, reducing the technical overhead of working with multiphonics and streamlining their use in both composition and performance.

1.1. Existing Notation Tools and Multiphonic Notation

Multiphonics are increasingly common in new music: i.e., Fallowfield [4] analyzes complex multiphonic fingerings for cello, and specialized clarinet catalogs by Roche [5][6] document dozens of multiphonic fingerings (with ease-of-production ratings and audio examples). In traditional engraving practice, however, there is no single standardized method for notating multiphonics across mainstream notation software. Composers and performers often resort to ad-hoc solutions – writing multiple simultaneous notes in separate voices, drawing custom fingerings by hand, or attaching graphics or text glyphs – leading to extra work and potential ambiguity in the score.

Most notation applications support multiple simultaneous voices, but none include a ready-made multiphonic fingering system. For instance, Sibelius [7] explicitly instructs users to write multiphonics by using a second voice on the same staff and selecting alternate notehead shapes for those notes. MuseScore [8] likewise allows up to four voices per staff, and users often treat multiphonics as chords or secondary voices with modified noteheads; however, any specialized fingering diagram must be created as an image or composed from basic fonts. LilyPond [9] can express simultaneous notes via multiple Voice contexts and supports engraved woodwind fingering diagrams (introduced in v2.14). However, each multiphonic must still be constructed manually, as there is no predefined library of multiphonic charts in the source distribution.

Dorico is designed for advanced notation, yet as of its latest release it offers no dedicated multiphonic tool. Its official documentation explains that users import custom fingering graphics or design bespoke symbols via the "Creating custom playing techniques" interface [10], and forum discussions confirm that multiphonic graphics are typically defined as custom playing techniques in personal libraries [11].

In the Max environment, notation has begun to see more real-time and flexible solutions. MaxScore allows scores to be generated and updated on-the-fly [12]. Agostini and Ghisi's bach library introduced a suite of objects for creating and manipulating notation within Max, but even here there is no specialized multiphonic support – users assemble noteheads or attach static images manually [13]. ChordsNest fills this niche: it provides MaxScore with a dedicated palette of fingering charts and symbols for woodwind[1] multiphonics. By selecting from this palette, a user can spawn correctly formatted multiphonic notation objects directly in the score, without leaving the Max/MaxScore environment. This integration is novel because it combines real-time score editing with a high-level, instrument-specific notation interface.

2. METHODS AND MATERIALS

Given the absence of existing tools that seamlessly combine multiphonic notation, categorization, and playback, this project was motivated by the need to create a resource grounded in real-world instrumental practice and informed by performer expertise. From the outset, we sought to develop ChordsNest not only as a technical solution, but as a collaborative endeavor bridging compositional needs with realities of performers.

Central to this process was our collaboration with clarinetist Heather Roche, whose expertise in multiphonics was essential to the project. Through regular online meetings, we discussed technical possibilities, developed the categorization system, and recorded a comprehensive library of B♭ clarinet and bass clarinet multiphonics. This close cooperation ensured that ChordsNest reflects both practical playability and artistic potential.

In designing ChordsNest, we envisioned a tool that would offer composers, performers, and educators flexible entry points tailored to their specific needs. Composers often require a way to explore sounds intuitively, such as searching by approximate pitch content, dynamic range, or timbral characteristics to find multiphonics that fit a musical idea. Performers and educators, on the other hand, may wish to look up specific fingerings or assess technical feasibility based on difficulty levels. We aim to support these varied workflows in our designs. This approach was intended to make the exploration of multiphonics both practical and creatively engaging, accommodating a range of artistic and pedagogical contexts.

2.1. Recording and Categorization

The dataset includes 266 recorded multiphonics: 130 for bass clarinet and 136 for B♭ clarinet. Each sample was systematically categorized by its fundamental and harmonic content to document pitch, by dynamic level ranging from *ppp* to *fff*, by a difficulty rating from 1 (easy) to 3 (challenging), and by characteristic tags such as "fragile," "buzz," "spectral," and "glissando" to support detailed classification.

2.2. Interface Design and Implementation

The user interface was designed to prioritize intuitiveness and efficiency. It enables users to select the instrument (B♭ clarinet or bass clarinet), browse available multiphonics through either an interactive staff diagram or a categorized tag-based search, and hear immediate audio playback of any selection.

The Max patch integrates several core components: a *jweb* object that loads a static HTML page containing an SVG *viewbox* for interface display and interactions, a *js* object that manages queries to an SQLite database, a *js* object that handles interactions with the MaxScore instance, multiple *dict* objects that store intermediary data such as search results and active filter selections, and a *playlist~* object that handles preview playback. Together, these elements provide a responsive interface that connects the multiphonic database, visual rendering, and playback engine in real time.

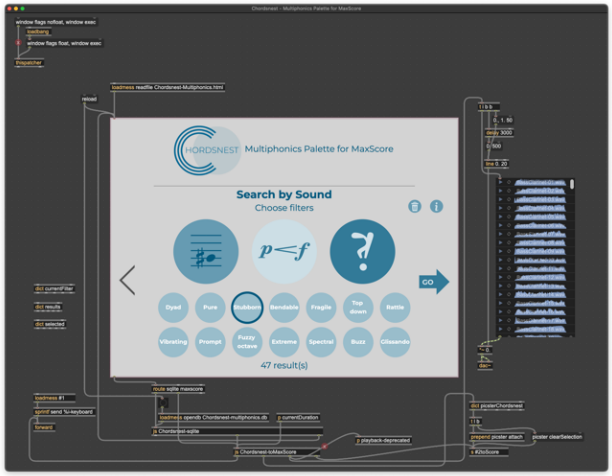

Figure 1. The ChordsNest Max patch

2.3. Database

The library of available multiphonics is provided by our collaborating clarinetist, Heather Roche. Several solutions, including JSON, were considered for storing the information, but ultimately SQLite was chosen because of its simplicity in handling search queries. This was further supported by the availability of an *SQLite* object in the default Max JavaScript implementation. The information for individual multiphonic samples is organized in an SQLite

[1] As of now, the implementation includes two instruments: the clarinet and the bass clarinet.

database stored in a file called “Chordsnest-multiphonics.db”. The columns include:

Field	Type / Range	Description
index	Integer (1–266)	Unique identifier for each multiphonic sample
instrument	String (“clar-bb”, “clar-bass”)	Indicates B♭ clarinet or bass clarinet
dynamicsMin	String (e.g. “pp”)	Lower end of plausible dynamic range
dynamicsMax	String (e.g. “ff”)	Upper end of plausible dynamic range
dynamicsCategory	Array of integers (1–3)	Derived category from **dynamicsMin** and **dynamicsMax**, used to facilitate interface searches
difficulty	Integer (1–3)	Indicates the relative difficulty of producing the multiphonic
tags	Array of strings	Characteristic tags (e.g. “fragile,” “buzz,” “spectral”). See **Table 3**
fingering	String	Fingering used to produce the multiphonic (e.g. “r023456f¡€”), each character corresponding to a unique key being pressed
pitchWritten	Array of strings	Perceived pitches as identified by H. Roche
pitchMidicent	Array of integers	Perceived pitches converted to midicents, derived from **pitchWritten**
pitchClassApprox	Array of integers (0–11)	Approximation of pitches to nearest MIDI key class (mod 12) for interface search, derived from **pitchMidicent**

Table 1. SQLite database structure and description

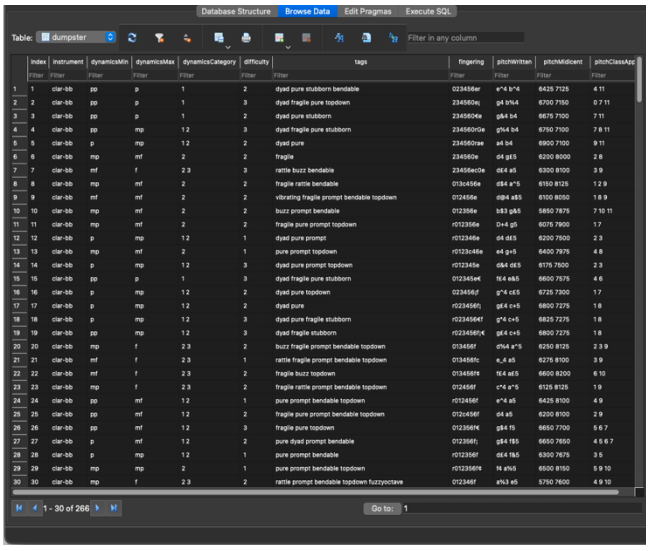

Figure 2. The SQLite database when opened with DB Browser

2.4. Search Methods

After selecting the instrument, the user is given two primary search options:

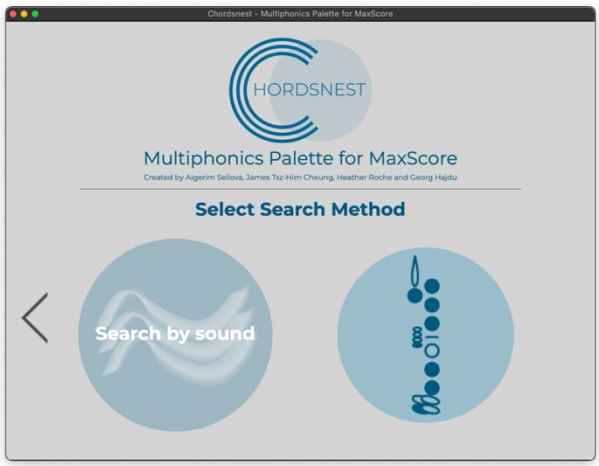

Figure 3. Interface view showing the two primary search modes: (left) search by sound, and (right) search by fingering

2.4.1.Search by Sound

This path allows filtering by the following parameters:

Parameter	Description
Pitch	Users select a pitch class (e.g., "D"). The system returns all multiphonics containing an approximate match within an octave-transposed range around that pitch class (e.g., D-half-sharp to D-half-flat).
Dynamics	Three dynamic categories are provided: soft (*pp*), middle (*mf*), and loud (*ff*). Selecting one includes all multiphonics playable within the chosen dynamic level.
Difficulty	Ranges from 1 (easy) to 3 (difficult), allowing users to filter based on playability.
Characteristic Tags	Tags help describe the sonic behavior and playing characteristics of each multiphonic, provided by Roche. These are user-selectable filters and defined in **Table 3** below.

Table 2. SQLite database structure and description

These filters can be stacked, and the number of available results with the current selected filters is shown in the bottom of the page.

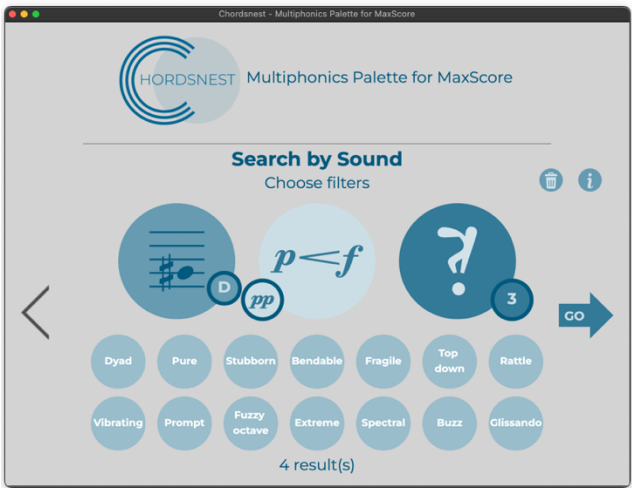

Figure 4. Search by Sound options

Tag	Description
dyad	A soft close dyad multiphonic with two prominent pitches, no more than an octave apart.
pure	A pure, clean sound.
stubborn	Requires strong diaphragm and embouchure support to sound.
bendable	Top pitch can be bent downwards.
fragile	Pitches fluctuate or don't remain stable.
topdown	Can be started on the top pitch, unlike typical bottom-up multiphonics.
rattle	Contains a rattling, buzzing texture.
vibrating	Similar to a rattle but more pulsed.
prompt	Speaks quickly with both pitches immediately audible.
fuzzyoctave	Nearly octave multiphonics (rare for clarinet).
extreme	Among the most difficult to produce reliably.
spectral	Overblown, spectral-sounding with a stable fundamental.
buzz	Very distorted and buzzy.
glissando	Allows upward pitch sliding.

Table 3. Characteristic Tags implemented in ChordsNest

2.4.2.Search by Fingering

For users already familiar with a specific multiphonic fingering, the "Search by Fingering" method enables direct access. Users interactively toggle keys to match a fingering chart; the tool then displays the corresponding multiphonic. Only one result is returned for each unique fingering.

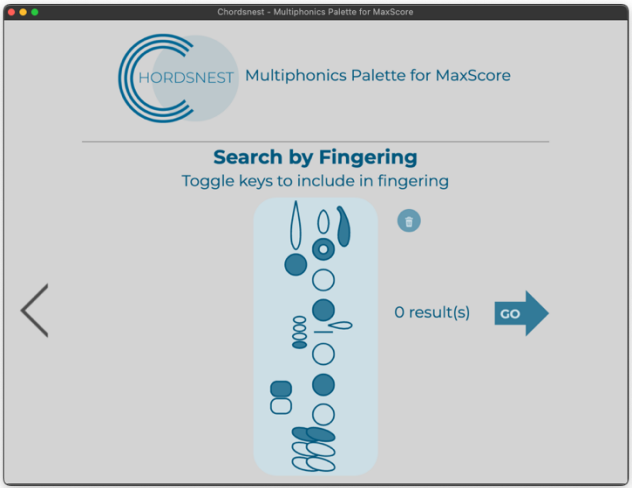

Figure 5. The Search by Fingering page

2.4.3.Result cards

Regardless of search method, the results will be shown after clicking on the “Go” button, as rectangular result cards containing all information of the multiphonic samples, containing the index, fingering, pitches (transposed), plausible dynamic range, characteristic tags and difficulty.

The fingering diagrams are generated using SVG graphics, with path data adapted from Bret Pimentel’s online fingering diagram builder [14]. CSS variables are used to control the visibility and fill state of individual keys or key groups.

The notation considers the transposition of the instrument (not concert pitch).

Figure 6. Search results page displaying multiphonic result cards, each containing fingering diagram, tags, and playback link

2.4.4.Preview playback

When a result card is clicked on, the multiphonic sample will be played back through Max audio as a preview. This is achieved internally in the *ChordsNest-multiphonics* patch using the Max *playlist~* object.

2.5. Integration into MaxScore Editor

ChordsNest is now included in the installation of MaxScore, accessible via the MaxScore Editor as an Entry Tool [15] within the Pitch Utility tab. Entry Tools are interfaces that provide interactive pitch input, primarily designed for microtonal composition. For example, the Scala Browser allows users to browse microtonal scales available in the Scala database. ChordsNest leverages the flexibility of custom Entry Tools, enabling the system to add notes and intervals, with attached fingering charts as SVG graphics containing playback instructions, by sending messages to the MaxScore Editor instance.

In the ChordsNest interface, after selecting and previewing a result card, clicking on “Add” inserts the selected multiphonic into the chosen measure, using the duration set previously in MaxScore. A note and its interval are generated according to the corresponding pitchMidicent values from the SQLite database.

Fingering diagrams are inserted in MaxScore as Picster elements, which are custom graphical components created using an interactive Max patch, with SVG or text-based shapes stored as JSON and linked to playback expressions. [16]. These include JSON representations of SVG graphics and Expressions for playback control. The SVG paths are based on diagrams created with Bret Pimentel’s online fingering diagram builder [14].

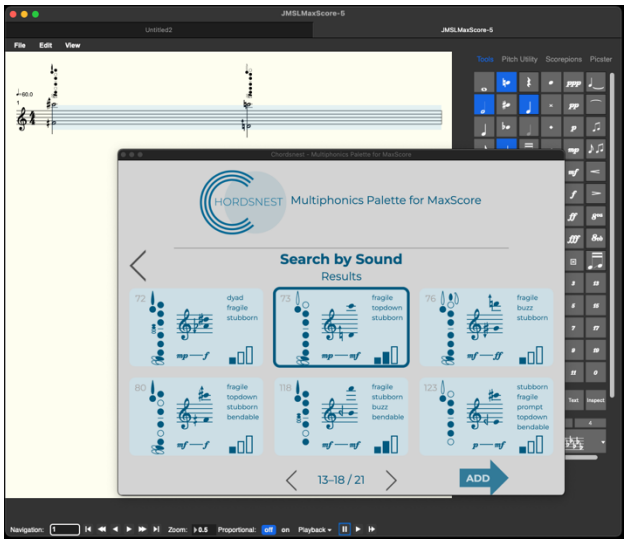

Figure 7. ChordsNest resides in MaxScore as a pitch entry tool

2.6. Playback using MaxScore Sampler

MaxScore Editor works with MaxScore Sampler via the *maxscore.makenote* object, which manages incoming MIDI messages and instrument changes. ChordsNest adopts this paradigm by utilizing a dedicated bank file containing two instruments: *BassClarinet-Multiphonics* and *Clarinet-Multiphonics*. This single bank file defines each instrument as a collection of .wav audio samples, with accompanying metadata such as root key, key and velocity zones, envelopes, and playback parameters. The samples within these banks are ordered according to their indices from the SQLite database. The MIDI pitch of each sample corresponds directly to its sample index, taking advantage of the MaxScore Sampler's ability to handle MIDI pitches above 127 (the standard limit).

When generating a multiphonic dyad, the bottom note is muted (velocity set to 0), while the top note employs the MaxScore note attribute "originalPitch" to define a MIDI playback pitch (corresponding to the sample index) that differs from the notated pitch on the score. The attached Picster Element, which displays the fingering diagram as SVG graphics, also contains an expression to facilitate a temporary instrument change in the Sampler.

This mechanism is analogous to how a pair of Picster Elements labeled "pizz" and "arco" can induce instrument changes on a string staff. A particular challenge arises because, after playing back a multiphonic sample, there is no subsequent Picster Element to instruct the Sampler to revert to the previous instrument, for example an ordinario clarinet sampler instrument. To address this, a new message "multiphonic [instrument]" has been introduced for the *maxscore.makenote* object, enabling one-time, temporary instrument changes.

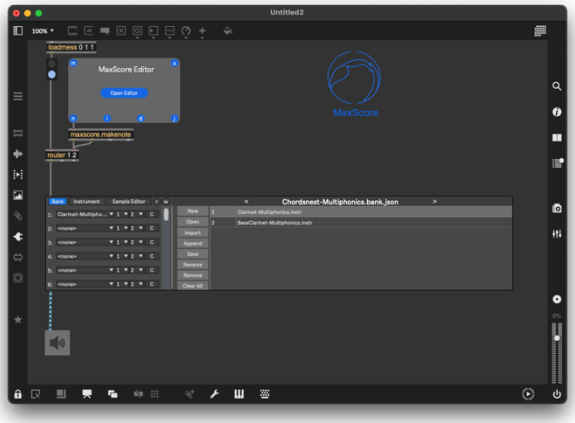

Figure 8. MaxScore combo template, with the ChordsNest clarinet multiphonics sound bank loaded

3. RESULTS

ChordsNest integrates into MaxScore and provides a comprehensive library of multiphonics with high-quality audio samples. It includes an interface for notation and playback, along with a tagging system that facilitates navigation. Playback is managed through the MaxScore Sampler, using a sound bank containing all samples recorded by Heather Roche.

3.1. Preliminary Evaluation

To assess workflow improvements, two composers tested ChordsNest while preparing short clarinet studies. They reported a significant reduction in time spent preparing multiphonic notation compared to manual entry and other software solutions. The interface was described as "intuitive" and "well integrated." Future work will include a broader usability study to quantitatively measure these improvements.

4. DISCUSSION

4.1. Implications for Composition and Performance

ChordsNest addresses a notable gap in contemporary music composition by providing a practical tool for integrating multiphonics into notation and playback workflows. Performers also benefit from access to systematically categorized and playable examples, supporting detailed preparation and interpretation. ChordsNest can also be useful for educators introducing different categories of multiphonics and their characteristics.

4.2. Potential for Expansion

The project's framework allows for future development to include additional instrumental families such as oboe, bassoon, saxophone etc. The tagging system and interface are designed to be scalable, making it possible to accommodate a broader range of multiphonic resources. While the

current implementation focuses on clarinet and bass clarinet, its applicability to other woodwind instruments remains untested and could be explored in future research, along with the integration of live-processing capabilities.

Further development could also extend MaxScore's existing message-based control, enabling programmatic generation of multiphonic notation and playback, and thus supporting algorithmic composition with extended techniques.

In addition, more features can be potentially developed, including the capability for users to save certain multiphonic samples in a "favourites" list, or for example, for users to define custom multiphonics not currently included in the fixed database.

5. CONCLUSION

The ChordsNest project represents a significant step forward in the integration of multiphonics into compositional practice. By combining a detailed library, intuitive UI, and seamless integration into MaxScore, the tool addresses key challenges in contemporary music composition. With the potential for further expansion, ChordsNest lays the groundwork for a new standard in multiphonic accessibility and usability.

Acknowledgment

We would like to acknowledge the generous support of the Gerhard Trede-Stiftung, which made this research possible.

6. REFERENCES

1. C. Beare, *Cultivating the Contemporary Clarinetist: Pedagogical Materials for Extended Clarinet Techniques*, Ph.D. dissertation, Univ. of Washington, Seattle, WA, 2021.
2. B. Bartolozzi, *New Sounds for Woodwind*, Oxford University Press, 1982.
3. E. Michael Richards, "The Clarinet of the Twenty-First Century: New Sonic Resources Based on Principles of Acoustics", E & K Publishers, 1992.
4. E. H. Fallowfield, "Cello Multiphonics: Technical and Musical Parameters," *Tempo*, vol. 74, no. 291, 2020, pp. 51–69. [Online]. Available: https://irf.fhnw.ch/. Accessed: Jul. 14, 2025.
5. H. Roche, "27 Easy Bb Clarinet Multiphonics," heatherroche.net, Sep. 13, 2018. [Online]. Available: https://heatherroche.net//. Accessed: Jun. 20, 2025
6. H. Roche, "Spectral Multiphonics (Bb and Bass Clarinet)," Multiphonic Monday, Sep. 26, 2016. [Online]. Available: https://heatherroche.net/
7. Avid Technology Inc., Sibelius User Guide, 2022. [Online]. Available: https://kb.avid.com Accessed: Jun. 15, 2025.
8. MuseScore, *MuseScore Handbook*, 2023. [Online]. Available: https://musescore.org. Accessed: Jul. 10, 2025.
9. LilyPond Development Team, "Woodwind fingering diagrams," *LilyPond Documentation*, v2.14, 2021. [Online]. Available: https://lilypond.org. Accessed: May 20, 2025.
10. Steinberg Media Technologies GmbH, "Creating custom playing techniques," Dorico Pro 5.0 Help, 2025. [Online]. Available: https://www.steinberg.help/. Accessed: Jul. 20, 2025.
11. D. C. Adams, "Multiphonics flute," *Steinberg Forums*, Apr. 2, 2021. [Online]. Available: https://forums.steinberg.net/. Accessed: Jul. 20, 2025.
12. G. Hajdu and R. Gottfried, "The Symphony in the St. Pauli Elbe Tunnel," *Contemporary Music Review*, Mar. 19, 2025, pp. 1–23.
13. A. Agostini and D. Ghisi, "A Max Library for Musical Notation and Computer-Aided Composition," *Computer Music Journal*, vol. 39, no. 2, June 2015, pp. 11-27.
14. B. Pimentel, "Woodwind Fingering Diagram Builder," [Online]. Available: https://bretpimentel.com/
15. G. Hajdu and N. Didkovsky, "MaxScore: Recent Developments," in *Proc. Int. Conf. Technologies for Music Notation and Representation (TENOR '18)*, Montreal, Canada, 2018, pp. 138–146.
16. G. Hajdu, and N. Didkovsky, "MaxScore – Current State of the Art," in *Proceedings of the International Computer Music Conference,* Ljubljana, Slovenia, 2012, pp. 156–162.