An intermediate mapping layer for interactive sequencing

Mathias Kirkegaard, Mathias Bredholt, and Marcelo M. Wanderley

Input Devices and Musical Interaction Laboratory - IDMIL
Centre for Interdisciplinary Research in Music Media and Technology - CIRMMT
McGill University, Canada
{mathias.kirkegaard, mathias.bredholt}@mail.mcgill.ca
marcelo.wanderley@mcgill.ca

Abstract. The definition of mapping strategies between input devices and sound generating algorithms is an essential process in computer music. Traditionally, mapping has been considered in the context of digital musical instruments where the control of low level sound features (note articulation, timbre) is the goal. This paper describes the motivation, implementation and example application of an intermediate mapping layer for interactive sequencing. Building upon the mapping literature, we expand it by exploring the control of predefined musical sequences, focusing on the ability to make spontaneous musical decisions, creative exploration by browsing, and easy mapping between devices. It involves a parameterization of rhythm, melody, and harmony along with collaborative mappings in both libmapper and Ableton Link.

Keywords: mapping · sequencing · collaborative performance

1 Introduction

The importance of mapping within design of DMI's has been elaborately discussed in [6] and [3], and it has been used in several contexts, synthesis engines [5], physical models [15], audio effects [14], and envelopes [7]. In these contexts, mappings are usually focused on the direct control of timbral parameters, to define the feel and responsiveness of a musical instrument. The work described in this paper focuses on expanding the concept of mapping into sequencing. The *sequencer*, a recording and playback system, which can send and receive the control and performance data needed to regenerate a series of musical events [11], lets music makers organize and control more layers and dimensions of musical structure than possible with traditional acoustic instruments, giving access to extended timescales of rhythm, polyrhythmic grids or extreme uptempo performance [11]. The concept of sequencing can be seen as a mapping problem by considering a sequencer as an intermediate mapping layer that translates a control input to multiple musical events. This is the idea behind the *algorithmic sequencing layer* (ASL), an intermediate mapping layer that maps 18 control parameters to real-time generated MIDI events, by means of a parameterization of rhythm, melody and harmony. The parameters can be mapped to a gestural controller with complex mappings such as one-to-many or many-to-one. In a performance situation, the generated MIDI events would trigger a synthesis engine, which generates the audio.

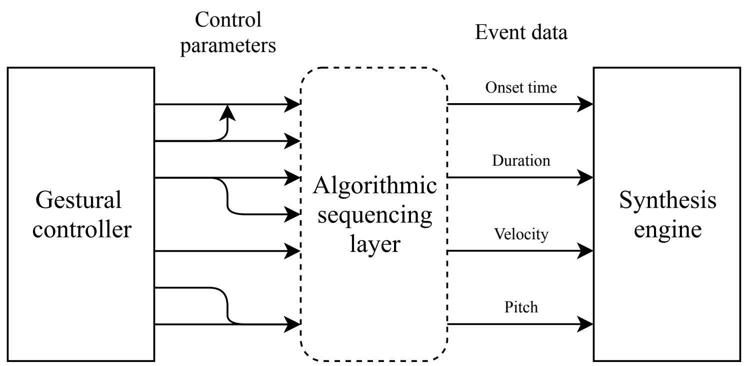

Fig. 1. An example of a complete sequencer application

The following keywords are used as guidelines for the mapping strategy

- **Spontaneity** The ability to make spontaneous musical decisions by controlling multiple musical events simultaneously.
- **Serendipity** Creative exploration by browsing.
- **Collaborative** Easy mapping between devices to increase musical coherence between multiple performers.

1.1 Spontaneity

In a live performance situation, the performer should be able to change the generated sequence spontaneously. It should be possible to quickly make drastic changes to tempo, event onset density, and melodic/harmonic content.

1.2 Serendipity

In [2] S. Fels proposes the use of metaphors to increase the transparency of a mapping. Fels argues that the perceived expressivity of an instrument depends on the transparency of the mapping, as the communication between the player and the audience involves an understanding of the link between the player's actions and the sounds produced [2]. The metaphor of browsing is used in the design of the ASL, in order to allow for "scrolling" through different rhythms and chords. This is related to the concept of serendipity, as the user can discover new musical ideas without having imagined them beforehand.

1.3 Collaboration

When gestures are mapped to timbral parameters as in [3,5,6,7,14,15], the auditory feedback is often instantaneous. On the contrary, when changing parameters for a musical sequence, the auditory feedback is distributed over time. This can result in lack of coherence between performers in a collaborative improvisational musical setting. The proposed strategy enables a mapping between multiple sequencers and gestural controllers, in order to accommodate this issue.

2 Algorithmic Sequencing Layer (ASL)

The ASL was developed as part of the hardware sequencer *T-1*, which features 18 control parameters for algorithmic sequencing. This section describes the implementation of the most important parameters of the ASL, specifically the parameterizations of rhythm, melody, and harmony.

2.1 Rhythm

In this work the concept of musical rhythm is considered as a sequence of events, defined by their amplitude, duration, and onset time. A parameterization of rhythms can be regarded as a divergent mapping to multiple rhythmic events, that results in a sequence with musically relevant properties. This allows for an interaction, which uses the metaphor of browsing. By changing the parameters of the rhythm, a user can browse through different musically relevant rhythms. In [13] G. Toussaint studies a large range of traditional rhythms, and especially those of Sub-Saharan African music. They share an important property, that their onset patterns are distributed as evenly as possible. Toussaint also demonstrates, how the *euclidean algorithm* can be used to generate rhythmic sequences that share this property. This is the basis for the parameterization of onset time in the ASL.

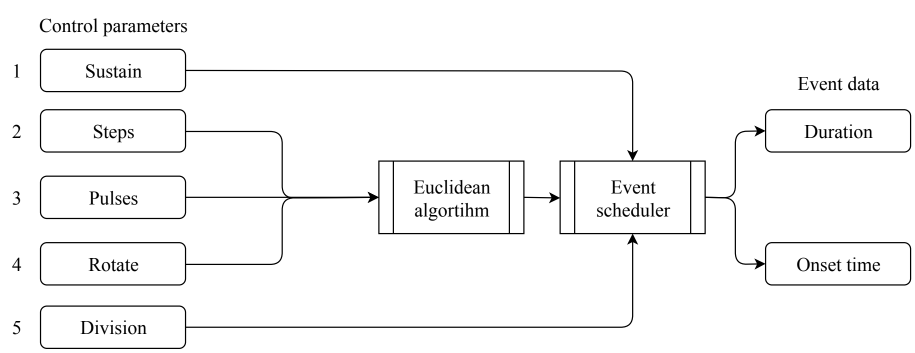

Fig. 2. Internal mappings to onset time and duration of musical events

Euclidean rhythms are represented by 2 parameters, *steps* and *pulses*. For a euclidean rhythm $E(n, k)$, there is n steps, $k < n$ pulses, and $k - n$ rests. The euclidean algorithm, a procedure that computes the greatest common divisor, is used to place the rests among the pulses as evenly as possible. The position of a pulse determines the onset time of a rhythmic event, and the duration of each event corresponds to the number of rests succeeding it.

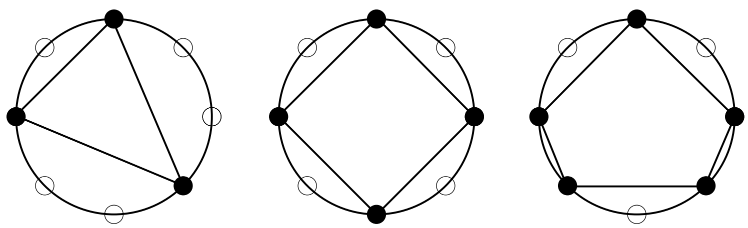

Fig. 3. Illustration of 3 euclidean rhythms with different number of pulses. From left to right: $E(8, 3), E(8, 4), E(8, 5)$

The ASL control inputs that maps to duration and onset time include:

1 ***Sustain*** Percentage of the full duration of a rhythmic event

2 ***Steps*** Length of the euclidean rhythm

3 ***Pulses*** Number of onsets in the euclidean rhythm

4 ***Rotate*** Starting position of the euclidean rhythm, which corresponds to a rotation of the graphical representation in fig. 3

5 ***Division*** Temporal resolution of the euclidean rhythm

Accentuation An important feature of rhythmic patterns is accentuation, i.e. amplitude modulation over time, as it allows for emphasizing certain events by means of a louder sound. P. Pfordresher concludes that accents function as temporal landmarks that listeners can use when tracking the time structure of musical patterns [10]. In the ASL, a parametrization of accentuation is obtained by considering the amplitude envelope of a rhythmic phrase as discrete steps in an accent phrase. An overview of the accentuation mapping is seen in fig. 4. A collection of perceptually relevant accent phrases can be used as input to the ASL.

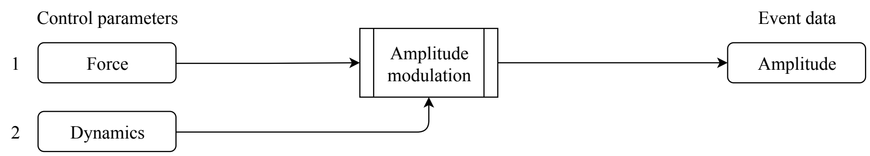

Fig. 4. Internal mapping to amplitude of rhythmic events

The ASL control inputs that maps to amplitude include:

1 ***Force*** Addition and subtraction to the accent phrase, effectively controlling the average amplitude of a rhythmic sequence.

2 ***Dynamics*** Multiplication of the accent phrase, effectively controlling the amount of accentuation.

2.2 Melody

By considering a musical phrase of a melody as a discrete sequence of degrees in a musical scale, the first phrase of the melody *Brother John* can be represented as

$$[0, 1, 2, 0, 0, 1, 2, 0]$$

which in the C-major scale translates into the sequence of pitches

$$[\mathrm{C}, \mathrm{D}, \mathrm{E}, \mathrm{C}, \mathrm{C}, \mathrm{D}, \mathrm{E}, \mathrm{C}]$$

In this paper, this is defined as the *scale degree representation* of a musical phrase, and it allows for doing arithmetic operations on melodies. Addition and subtraction corresponds to the concept of diatonic transposition i.e. transposing a melody sequence within the given musical scale. Multiplication and division corresponds to changing the melodic range of the melody i.e. the distance between the lowest and highest note. As the scale degree representation is independent of the chosen musical scale, a phrase can easily be translated between major and minor. As with accent phrases, a collection of melody phrases can be used as input to the ASL. The control inputs that map to pitch include:

1 ***Scale*** Musical scale (major, minor, pentatonic etc.)

2 ***Ambitus*** Multiplication of the melody phrase, effectively controlling the melodic range of the phrase

3 ***Degree*** Addition and subtraction to the melody phrase, effectively doing diatonic transposition

2.3 Harmony

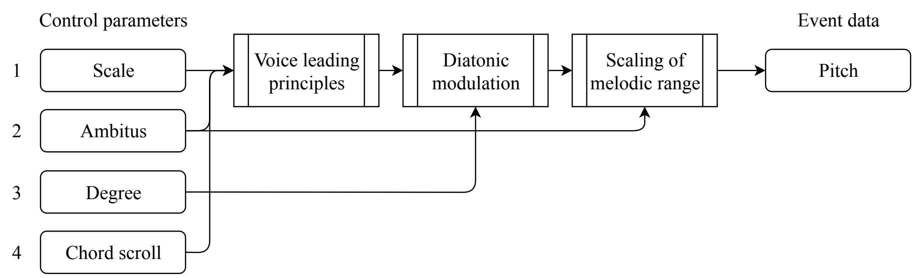

Fig. 5. Internal mapping to pitch height

Harmony can be parameterized using voice leading principles i.e. the theory of progressions of individual melody lines and how they interact with each other to create harmony. An important principle within voice leading is that a good chord progression has common tones between each successive chord. In scale degree representation this means that each successive chord should have at least 1 integer in common. This chord sequence follows that principle

$$\begin{bmatrix}0\\2\\4\end{bmatrix} \rightarrow \begin{bmatrix}0\\2\\5\end{bmatrix} \rightarrow \begin{bmatrix}0\\3\\5\end{bmatrix}$$

In C-major the chords above represent the progression

$$\mathrm{C} \rightarrow \mathrm{Am} \rightarrow \mathrm{F}$$

The ASL includes a simple algorithm that adds or subtracts to a single tone, T, in the chord at a time. After each addition/subtraction, T points to the next tone in the chord, such that a new chord is created at each step. The new chord always has 2 common tones with the previous chord. As with the euclidean algorithm, this algorithm allows an interaction which uses browsing as metaphor. By stepping through the algorithm a user can scroll through musically relevant harmonic progressions. This is implemented with the control input *Chord scroll*:

4 *Chord scroll* A single step of the voice leading algorithm

3 Hardware platform

T-1 is a hardware platform, developed by the two first authors, that utilizes the ASL to implement a MIDI sequencer with 16 polyphonic tracks [12]. Parameterizations of rhythm, melody, and harmony, enables the design of a sequencer interface where all control inputs are space multiplexed, and thus immediately accessible. This limits the need for a display, and instead animations of multi-colored LED's are used as visual feedback.

3.1 Interface

The interface consists of 18 rotary encoders and 23 rubber keypad buttons with LED's. A drawing of the interface is seen in fig. 6. The 18 encoders are mapped in a one-to-one manner to the ASL, effectively controlling the rhythm, melody and harmony generation of a selected track. The 16 buttons to the bottom left represent the 16 tracks, which can be edited by pressing the corresponding button. The 7 buttons to the bottom right represent the *performance tools*, which activate a one-to-many mapping between the interface and the ASL.

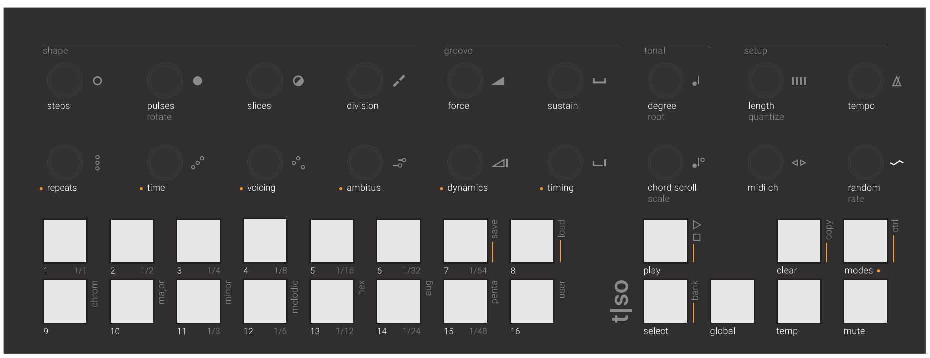

Fig. 6. Front panel of the T-1

3.2 Performance tools

Controlling 16 tracks simultaneously can be both time consuming and confusing in a musical performance. In order to allow for spontaneous musical decisions, a set of tools for multi-track editing are developed with the aim of minimizing *speed of performance* for tasks that are deemed typical in a musical performance.

Global Apply relative changes to all enabled tracks simultaneously i.e. adds or subtracts the same value to all parameters

Temp Apply temporary changes to different tracks one at time, and revert all changes simultaneously

Mute Mute/unmute multiple tracks simultaneously

4 Collaborative sequencing

The parametrization implemented in the ASL, enables the possibility of a shared reference of musical parameters. Each parameter can be represented as a signal, which can be shared with peers in an collaborative musical setting. This allows for the creation of one-to-one and many-to-many mappings between multiple instances of the ASL. A one-to-one mapping is relevant for parameters such as tempo, musical scale/key and meter, where uniformity is desired. A many-to-many mapping is relevant for rhythmic and melodic parameters where coherence but not necessarily uniformity is desired. Some examples include:

Constant note density A mapping of *pulses* across performers. When a performer increases their *pulses* parameter, the number of pulses decrease on the their peers thereby keeping the combined note density constant.

Pizzicato A mapping between the sustain of one performer to velocity of all performers, where the sustain is inversely proportional to the velocity, effectively implementing a pizzicato mode when sustain is low.

Shared global mode A mapping between all the parameters of multiple ASL's, effectively implementing the global mode explained in section 3.2 across multiple performers.

4.1 Technology

Collaborative mappings are obtained by utilizing *libmapper* [8] to connect multiple sequencers. libmapper is a library for making connections between data signals on a shared network. For sharing tempo and meter, *Ableton Link* [4] is used. Ableton Link is a technology that synchronizes tempo and beat across applications running on multiple devices connected on a shared network. The 2nd author has ported the Ableton Link and libmapper technologies to the *ESP32* microcontroller platform to allow for synchronization and mapping between multiple instances of the T-1 and other music applications [1].

Collaborative performance with 2 T-Sticks During the porting of libmapper a demonstration of mappings on a shared network in a collaborative performance and was created. The demonstration used two T-Sticks [9], a gestural controller made from a pipe equipped with a range of sensors. Two performers had one T-Stick each and controlled different aspects of a single sequence of sounds. By using libmapper, it was possible to experiment with and dynamically change mappings between the gestures of multiple performers and a single stream of musical events.

5 Discussion and future work

In section 2, it is mentioned that the ASL uses accent and melody phrases as input for the parameterization of accentuation and melody. On the T-1, phrases are generated using a pseudo random algorithm. More work should be put into making the generation and collection of phrases motivated by music- and perceptual theory. Alternatively, it would be interesting to develop a system that allows for recording these phrases on a gestural controller, to add more expressive possibilities to the system.
The euclidean algorithm is used for parameterizing rhythms, and while it offers a large range of rhythmic patterns, it also simplifies the musical output to sequences with an even distribution of onsets. This limits the possibilities of making fills and uneven rhythms. Additional features, not described in this paper, were added to the T-1 [12] to accommodate this issue.
The work described in this paper simplifies musical sequences to discrete musical events. This simplification resulted from the development process of the T-1, which outputs MIDI messages, that are defined by only pitch, velocity, sustain, and onset time. Thus, perceptually important information regarding the temporal evolution of a musical event is disregarded. Future work could combine the findings in [7] on parameterizations of temporal envelopes, with the ASL, in order to increase the dimensionality of the sequenced musical events.
Another simplification is the assumption of independence between the dimensions of musical events. For instance, some acoustic instruments has inherent internal mappings between sustain and pitch, or amplitude and sustain, which are not accounted for here. Though, as the proposed mapping system enables internal mappings (through libmapper), an interesting next step would be to experiment with inter-dependency of these dimensions, by means of complex many-to-many mappings.

6 Conclusion

In this paper the motivation of an intermediate mapping layer for interactive sequencing is presented. The proposed mapping layer, the algorithmic sequencing layer (ASL), provides a mapping between linear control variables and discrete musical events. A detailed description of how the ASL parameterizes rhythm, melody, and harmony is provided, and the hardware sequencer T-1 is presented as an example application that utilizes the ASL. Additionally, a mapping strategy for collaborative performance with multiple sequencers is proposed. Finally, the simplifications of the suggested framework is discussed, emphasizing that the musical output of this work has a limited range.

7 Acknowledgements

The ideas presented in this paper are the result of the more than 3 years of development of the T-1 sequencer by Torso Electronics, a company founded by Mathias Bredholt, Mathias Kirkegaard, Lars Buchholtz, and Jonas Kenton. The first two authors send a big thanks to our colleagues, Jonas and Lars, for helping us with the development.

References

1. Bredholt, M.: libmapper for arduino, https://github.com/mathiasbredholt/libmapper_arduino, last accessed 24 Jan 2020
2. Fels, S., Gadd, A., Mulder, A.: Mapping transparency through metaphor: Towards more expressive musical instruments. Organised Sound **7**, 109–126 (09 2003). https://doi.org/10.1017/S1355771802002042
3. Garnett, G.E., Goudeseune, C.: Performance factors in control of high-dimensional space. In: International Computer Music Conference (ICMC) (1999)
4. Goltz, F.: Ableton link – a technology to synchronize music software. In: LAC. pp. 39–42 (2018), http://dx.doi.org/10.14279/depositonce-7046
5. Hunt, A., Wanderley, M.: Mapping performer parameters to synthesis engines. Organised Sound **7**, 97–108 (2002), https://doi.org/10.1017/S1355771802002030
6. Hunt, A., Wanderley, M.M., Paradis, M.: The importance of parameter mapping in electronic instrument design. In: New Interfaces For Musical Expression (NIME) (2002), https://doi.org/10.1076/jnmr.32.4.429.18853
7. Levitin, D.J., McAdams, S., Adams, R.L.: Control parameters for musical instruments: a foundation for new mappings of gesture to sound. Organised Sound **7**(2), 171–189 (2002), https://doi.org/10.1017/S135577180200208X
8. Malloch, J., Sinclair, S., Wanderley, M.: Libmapper (a library for connecting things). In: CHI (2013), https://doi.org/10.1145/2468356.2479617
9. Malloch, J., Wanderley, M.: The t-stick: From musical interface to musical instrument. In: New Interfaces for Musical Expression (NIME) (01 2007). https://doi.org/10.1145/1279740.1279751
10. Pfordresher, P.: The role of melodic and rhythmic accents in musical structure. Music Perception: An Interdisciplinary Journal **20**, 431–464 (06 2003). https://doi.org/10.1525/mp.2003.20.4.431
11. Roads, C.: Composing electronic music, a new aesthetic. Oxford University Press (2015)
12. Torso electronics homepage, https://www.torsoelectronics.com, last accessed 02 Jan 2020
13. Toussaint, G.: The euclidean algorithm generates traditional musical rhythms. In: BRIDGES: Mathematical Connections in Art, Music and Science. pp. 47–56 (2005)
14. Verfaille, V., Wanderley, M.M., Depalle, P.: Mapping strategies for gestural and adaptive control of digital audio effects. Journal of New Music Research **35**(1), 71–93 (2006), https://doi.org/10.1080/09298210600696881
15. Wang, S., Wanderley, M.M., Scavone, G.: The study of mapping strategies between the excitators of the single-reed woodwind and the bowed string. In: Conference on Sound and Music Technology (CSMT). pp. 107–120 (2019), https://doi.org/10.1007/978-981-15-2756-2_9