SKETCHINGSONICTRAJECTORIES
A IANNIX TOOL FOR COMPOSING THE ELECTROACOUSTIC SPACE

Julian SCORDATO (`julian.scordato@conservatoriopollini.it`)[1], **Nicola PRIVATO** (`nicola.privato@gmail.com`)[2], and **Nicola RACCANELLI** (`raccanelli.nicola@gmail.com`)[2]

[1]*SaMPL - Sound and Music Processing Lab / Department of New Technologies and Musical Languages*, **Conservatory of Padua**, Italy
[2]*Department of New Technologies and Musical Languages*, **Conservatory of Padua**, Italy

ABSTRACT

This paper provides a report on the development of SketchingSonicTrajectories (SST), a software tool for sound spatialization that simplifies the user interaction in a specific usage of IanniX graphic sequencer: SST integrates multi-track audio playback, Ambisonics spatial audio processing, and IanniX score and performance management.

In addition to a project description and details on the user interface design, several strategies for the composition of the electroacoustic space with SST will be proposed, also with reference to the score examples included in the software package. The suggested exemplification is intended to illustrate the current software features and guide the user in the design and articulation of advanced movements of sound sources to achieve a multi-channel sound projection without getting entangled in software technicalities.

1. INTRODUCTION

Over the years, several panning techniques such as WFS [1], Ambisonics [2], and VBAP [3] have been developed with the aim of simulating the position and movement of sound sources in space accurately, and different approaches on their use in design and compositional practices have been adopted [4]. From the user side, strategies for sound spatialization commonly foresee automations on multitrack sequencers including software plugins, algorithms developed on programming environments, and performative practices involving a mixing console or a controller. In addition to standalone applications, certain implementations of spatialization algorithms can be integrated in software environments for real-time sound synthesis and processing (e.g., Max/MSP, Pure Data, or SuperCollider) and digital audio workstations. Among the notable tools available for free are Ircam Spat [5], ICST Ambisonics [6], HOA Library [7], Zirkonium [8], and SpatGRIS [9]; these offer a high degree of customization. However, it is worth mentioning that Zirkonium and SpatGRIS are currently incompatible with Windows, while spatialization libraries such as Spat, ICST and HOA depend on programming environments for their implementation, thus requiring specific skills from the user. Software plugins that rely on multitrack sequencers have the advantage of an easier approach, but limited control over the design of spatial movements; their representation is often trivial consisting in the variation of single parameters versus time. In different ways, these tools face the difficulty of interfacing the user with the notation and the representation of space as a compositional parameter.

In the ecosystem of sound spatialization tools, SST arises from specific questions: how to find a handy way to formalize and reproduce spatial parameters from the micro to the macro-form, in relation to the audio content; how to facilitate the design of sound trajectories for multi-channel electroacoustic works and sound design projects; how to offer flexibility to different live / studio contexts and broad compatibility with software and operating systems. The goal was to give access to a potentially wider user community by developing a tool capable of offering multiple approaches to notation and performance, and to bring attention to design and artistic issues instead of retaining the user on technical and programming aspects.

We chose to consider the Ambisonics equivalent panner included in the ICST Tools[1] for its flexibility in terms of parameter customization, suitability to a wide range of contexts, and licensing under the Revised BSD License. In particular, the uncoupling of the spatialization project from the electroacoustic configuration makes the system adaptable to different speaker setups, with the only requirement of having the same kind of speakers preferably arranged in a regular angular way. In real-life scenarios, loudspeaker amount and placement are usually different in relation to the host organization's choices and other factors.

As for the graphic design features, we decided to use IanniX[2] and discard the objects included in the ICST package. IanniX proposes a poly-temporal and multi-formal open-source sequencer also useful for the creation and the performance of message-emitting control scores for sound spatialization [10]. Through the basic functions of the IanniX GUI, it is possible to notate and store various data such as the 3D position of sound sources and speakers, to design custom spatial movements and patterns with

[1] https://www.zhdk.ch/forschung/icst/software-downloads-5379

[2] https://www.iannix.org/en/whatisiannix/

precise event timing, and to visualize a clear representation of global behavior in time and space. In addition to the GUI, the OSC and JavaScript approaches to score can be combined together to define the path of various types of curves available in IanniX (freehand, linear, Bézier, parametric and circular curves) [10]; this can significantly facilitate the spatialization design process according to the different abilities and needs of users. Most of the tools available for spatialization design provide one-way and single-strategy approaches, flat and non-reactive representations of the current position and level of sound sources, and simple and stereotyped motions such as rotation, translation, and random walk. In some cases, trajectory descriptions must be provided in a breakpoint format (time, coordinates) that forces the user to a clunky approach, especially when designing a macro-form.

In order to interface the IanniX environment with the spatialization algorithm, we have developed a macOS and Windows application that also includes a multitrack reproduction device. The SST GUI was intended to facilitate the interaction with IanniX, guiding the user in a step-by-step project definition, proposing advanced customizable models of sound trajectories, allowing reactive and bidirectional management of the spatialization project, and avoiding dependence on DAWs. However, the user is given the option to choose the preferred audio input from external hardware or third-party software (via virtual audio drivers), which makes SST suitable for both studio and live projects.

After its first iteration [11], SST is currently under development at the Department of New Technologies and Musical Languages of the Conservatory of Padua. The software package is freely available for download[3]. An updated project overview accompanied by a description of the system components is presented in the Section "Project description". The user interface has been adapted for improved usability, adding IanniX score controls, transport synchronization, event handling and visualization. Through this, we aim to further facilitate the production and reproduction of a graphic score for sound spatialization (see "User interface design"). Several score examples are proposed and discussed in relation to their functionality, in order to demonstrate different procedures related to the use of SST and the included trajectory library (see "Compositional and performance strategies"). Additionally, the use cases within our Living Lab Music showcase have been taken into account (see "Use cases"). Finally, we conclude with some considerations and perspectives for the future development of the project (see "Conclusions").

2. PROJECT DESCRIPTION

The proposed system for the creation and reproduction of graphic scores for sound spatialization involves three main logical sections – SST, IanniX, and Mira – as described below. Communication is handled via OSC (to and from IanniX) and Bonjour protocol (between SST and Mira).

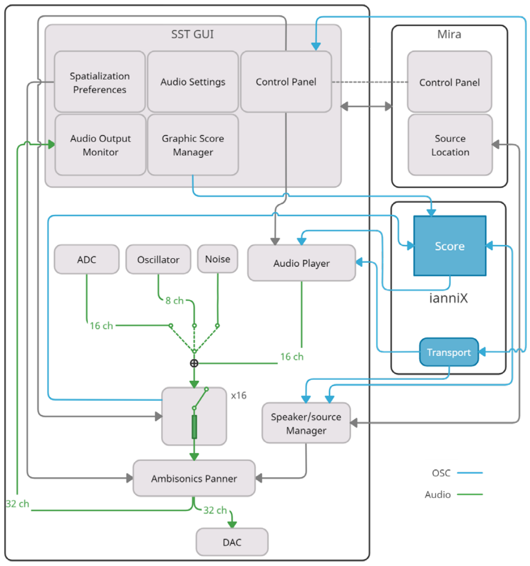

Figure 1. Block diagram.

2.1 SST

SST is an application developed in Max/MSP; it includes a Graphical User Interface (GUI) for managing the score and related audio content, a series of devices for interfacing with the *IanniX score* and *Transport*, and a signal processing stage for audio routing, playback and spatialization (see Fig. 1). The SST GUI provides access to several components: *Graphic score manager*, *Control panel*, *Audio settings*, *Spatialization preferences*, and *Audio output monitor*.

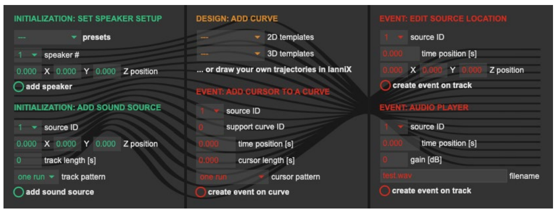

Figure 2. *Graphic score manager*.

The *Graphic score manager* defines the possible strategies for the creation and facilitated handling of objects within a IanniX score, aiming at the design of a spatialization project (see Fig. 2). It allows the user to set the speakers' number and position and to initialize the virtual displacement of audio signals and files into the ianniX score by organizing them into *sound sources* (see §3.2). Also, it suggests a series of 2D and 3D templates for the insertion of trajectory paths, and facilitates the interaction with the inserted objects by configuring their attributes and behavior as well as the visualization in IanniX of relevant information such as labels and audio level meters.

[3] https://www.julianscordato.com/projects.html#sst

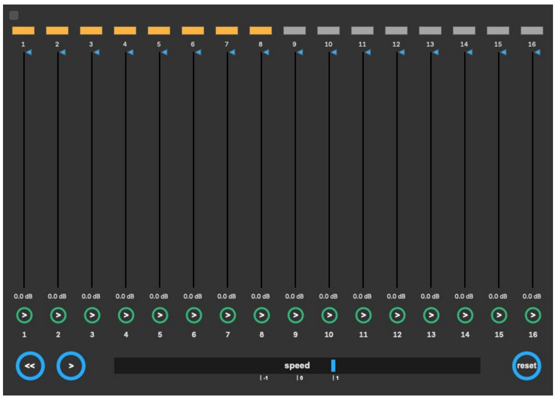

Figure 3. *Control panel.*

The *Control panel* can be accessed from both the SST application and the Mira app (see §2.3). It contains 16 potentiometers and playback buttons, one for each available sound source (see Fig. 3). This panel is synchronized with the *Transport* information in IanniX, giving the ability to combine score reproduction with real-time playback controls. Gain and mute controls for each audio channel can intervene at the input stage before the audio signal is routed to the *Ambisonics panner*.

The *Audio settings* module includes driver selection and audio I/O mappings. A further window is dedicated to the *Spatialization preferences* for adjusting the Ambisonics directivity and the amplitude attenuation pattern in the *Ambisonics panner*, which is based on the ambipanning~ object for Max/MSP [6]. The *Ambisonics panner* output is sent to the DAC as well as to the *Audio output monitor* in order to provide the user with visual feedback of the channel levels during score playback.

SST application offers the possibility to choose different types of audio inputs: *Audio player*, *ADC*, *Oscillator,* and *Noise*. The *Audio player* is responsible for the reproduction of the audio files assigned to each of the 16 available *sound sources* through the *Graphic score manager.* Activation of audio file playback with a desired gain level can be set in the score at any time position along a track. It is also possible to use the dedicated playback controls located in the *Control panel* (see Fig. 3). Except for the *Audio player*, the other inputs are mutually exclusive and can be switched via a selector; it is therefore possible, for example, to couple an external analog input with the playback of audio files using the same channel. The ADC can provide 16 channels from audio hardware as well as from virtual audio devices, in order to interface third-party audio applications with SST in real-time. To assist the user in testing the spatialization system, two ready-to-use audio inputs are selectable: a white noise generator and a polyphonic sawtooth oscillator. Available on channel 1, the noise generator can be conventionally used for a speaker test. The polyphonic oscillator, on the other hand, can be a useful tool in the design of spatialization models that involve multiple sources: it matches a single tone of the major scale to each of the sound sources, allowing the user to postpone the assignment of audio content.

Formatting of IanniX speaker and source position messages for the *Ambisonics panner* is done by the *Speaker/Source manager.* This device also turns on the audio inputs when the *Transport* starts and turns them off when score playback is stopped. In order to automatically update the configuration with any intervention on the score between two consecutive reproductions, the *Speaker/Source manager* forces IanniX to output the current status of all the elements present in the score following the *Transport* activation.

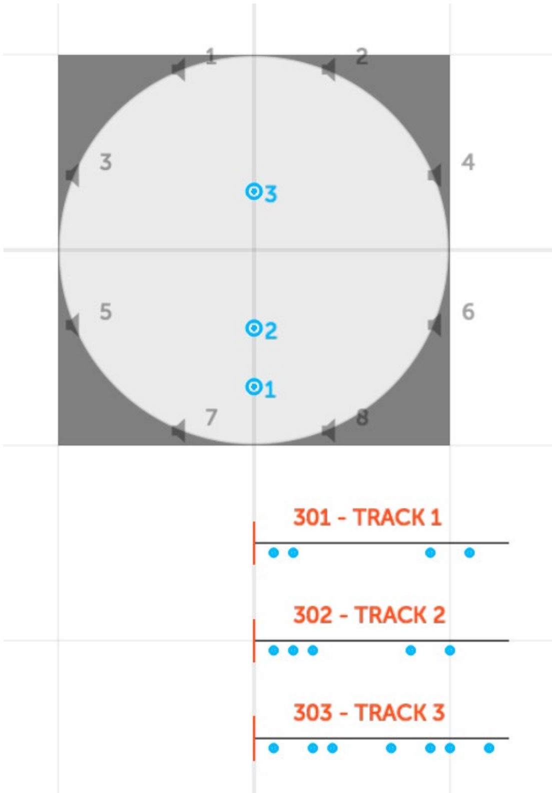

Figure 4. Example of static source positioning in a score included in SST software package (*sequence_1.iannix*).

2.2 IanniX

We have designated IanniX as the core sequencing engine and graphics platform for SST, due to its inherent poly-temporal and multi-formal sequencing capabilities and the flexibility it offers in terms of graphic notation [10]. For the description, two blocks are distinguished: the *Transport* and the *IanniX score* (see Fig. 1).

The *Transport* sends the score playback status to various devices: *Speaker/source manager*, *Audio player*, and *Control panel*. Available directly from the IanniX GUI as well as from the *Control panel*, the *Transport* represents the main control for the global reproduction of the graphic score and for the synchronization of routines such as input/output activation and source position refresh (see §3.3). Transport information is also sent to the *Audio player* in order to enable the reproduction of the audio files added to the project via the *Graphic score manager*.

The *IanniX score* stores the data and graphically represents the objects entered by the user through the *Graphic score manager*. The multi-formal representation space of the score includes a coincident reproduction of the Ambisonics 3D coordinate system [6] – which in turn can be considered as a convenient approximation of the electroacoustic space (see Fig. 6) – and up to 16 source-related automation tracks (see Fig. 4). The *IanniX score* acts as an interface that allows the user to intervene on the virtual position of speakers and sound sources, to manage the automation tracks and to set advanced attributes of the IanniX objects (see §4).

2.3 Mira

New to this SST iteration, this latter section consists of a touch-control interface for real-time source positioning which also provides global transport and audio playback controls. Developed in Max/MSP, this interface makes use of the Mira app for gestural input and visualization on a compatible mobile device [12].

Figure 5. *Source location* tab in Mira.

The proposed interface allows the user to combine the SST functionalities for score designing with the possibility of real-time intervention into the score during its performance; it contains two main tabs, both accessible via the Mira app: *Control panel* and *Source location*. Optionally accessible also from the SST GUI, the former provides audio controls for each channel (see §2.1). The latter allows real-time spatial manipulation of up to 4 selectable sound sources (see Fig. 5). Since the source position and playback controls can be set gesturally from a compatible mobile device, the user is free to move to the "sweet spot" while maintaining a certain degree of control over the score.

3. USER INTERFACE DESIGN

This section addresses the design of fundamental aspects of the SST interface for the purposes we have set ourselves (see §1). In the first subsection we will present issues related to the visualization of the elements in the score, from the macro-form to the micro-form, with a particular focus on the latter since in our approach to spatialization design we consider the macro-form as a result of the articulation of individual events that unfold over time. The second subsection will cover notational aspects, in particular how the score composition process was designed and how the IanniX objects were used for this purpose. Finally, we will address the aspects related to score reproduction and the possible performative approaches.

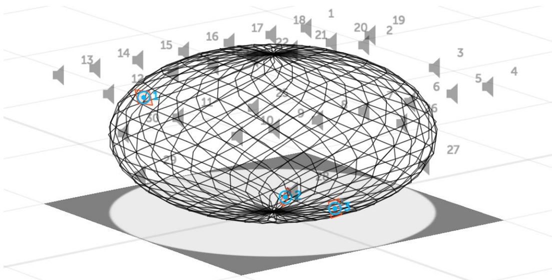

Figure 6. Example of a three-dimensional IanniX score.

3.1 Score representation

The three-dimensional Cartesian representation of the IanniX score in our specific application tends to be an approximation of a spatial configuration including both the actual speaker setup and the virtual positioning of the sound sources (see Fig. 6). The electroacoustic space is mapped according to the Ambisonics space definition [5], whose dimensions coincide with the absolute Cartesian coordinates of the rendering area of the IanniX score. A graphic texture has been placed to delimit conventional 2D boundaries. A similar two-dimensional representation is provided in the Mira interface for simultaneous control of up to 4 sound sources in live interventions (see Fig. 5).

Additionally, up to 16 source-related tracks are displayed in the lower area of the IanniX score (see Fig. 4). As in common sequencers, tracks provide additional timelines that normally serve the purpose of viewing and editing specific automations. In our application they constitute the support for the formal articulation of several types of events (see §3.2).

For the purpose of spatialization design, the content of a score includes a user-defined number of speakers, sources, and events notated as IanniX triggers; spatial trajectories are notated as IanniX curves; trajectory reading heads are notated as IanniX cursors with the function of driving the triggers associated with the same source.

During playback, the visualization in the main score area is synchronized with the events set along the tracks. Furthermore, to relate the audio content to the current position, a localized level meter is proposed for each active source by means of a HSV color variation in the related IanniX trigger.

3.2 Notation

In the design of the *Graphic score manager*, we distinguished three consequential phases in the user approach to creating a score: from the initialization of the electroacoustic setup (in terms of available sound sources and speakers), through the sketch of the sound trajectories, up to the addition of control events and audio content (see

Fig. 2). For the purpose of writing a spatialization score in SST, the basic IanniX objects (i.e. triggers, curves and cursors) take on certain functions.

In IanniX, a trigger is an "*object with the ability to send individual output messages*" [10]. Within the SST interface it can assume various connotations:

- *Speaker*, i.e. a graphical representation of the loudspeaker position with a custom texture (see Fig. 6); up to 32 speakers can be entered and displaced in a score;
- *Sound source*, namely an Ambisonics input (see Fig. 1) virtually positioned in the electroacoustic space; up to 16 sources can be displaced in the score; it is also possible to assign a single input to multiple sources through the I/O mappings included in the *Audio settings* window;
- *Event*: *Audio player* (see Fig. 2) creates a marker placed on a source-related track to start playing an audio file and set its volume level;
- *Event: Source location* creates a marker placed on a source-related track to set the three-dimensional displacement of a sound source in the score (see Fig. 4);
- any other trigger that causes an event in the score using a IanniX loopback message formatted as a command [13].

A IanniX curve is a "*graphical representation of a function or a vector-based path within the score*" [10]. Curves are used in the following cases:

- *trajectories*, i.e. the three-dimensional paths involving one or more sound sources located in the main score area (see Fig. 6);
- *tracks*, consisting of source-related timelines coupled with cursors aimed at activating events (see Fig. 4); in the SST GUI, the user can instantiate a track while initializing the related sound source (see Fig. 2);
- track *automations* that define the desired variation of up to two parameters as a function of time, considering the Y and Z axes (see Fig. 7).

A cursor in IanniX is a "*time-based graphical object that moves along the path of a linked curve and performs local and autonomous sequencing functions*" [10]. In the context of the SST user interface, cursors are applied for dynamic positioning of sound sources: using IanniX loopback messages, a trigger can be forced to follow the current position of a cursor; through this, it is possible to define the temporal behavior of a sound source in the score and make the source move along the path of a three-dimensional curve (see Fig. 6). Taking into account a modular temporal approach, multiple cursors can be associated to the same sound source to articulate different spatial-temporal behaviors in the perspective of a macroform. Secondly, IanniX cursors are used as part of source-related timelines for the activation of any colliding triggers (see Fig. 4), as well as for reading the values of any colliding curves (see Fig. 7).

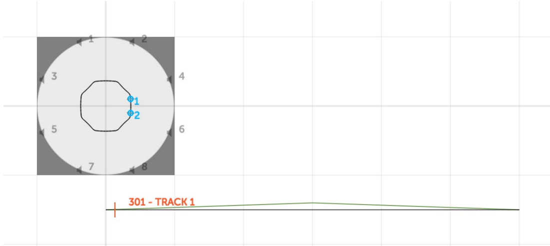

Figure 7. Example of track-level automation in a score included in SST software package (*supershape.iannix*).

3.3 Reproduction

As part of our goal of making SST suitable for both studio and live applications, we propose two main approaches related to the performance of a score. The former exploits the functions of the integrated *Audio player* (see §2.1), and is based on fixed audio content to be reproduced by means of events located on source-related tracks. The latter takes advantage of the ADC input for a more flexible configuration that allows the connection of external audio devices such as acoustic instruments, as well as other software and VSTs (using a virtual audio driver); this also led us to integrate the Mira interface into the system (see §2.3), in order to allow the user greater control over the audio content.

Once the score has been set properly, the audio inputs and outputs can be configured in the SST GUI from the *Audio settings* window. The audio input level can be checked on the meter incorporated in the *sound source* representation only after starting the score playback. The *Transport* (see §2.2) can be activated via the *play* button which is synchronized between IanniX and the SST GUI. The activation involves updating the status of the IanniX objects representing sources and loudspeakers, which are forced to send control messages to SST. Any active events or trajectory-related cursors will reposition the sound sources on the planned paths or position. When score playback is stopped, audio input routing is interrupted accordingly.

4. COMPOSITIONAL AND PERFORMANCE STRATEGIES

This section is focused on the score prototypes included in the software package with the aim of exemplifying some possible SST applications both on the basis of the *Graphic score manager* functionality and through a more advanced use of IanniX (e.g. including the loopback interface and JavaScript functions).

The trajectories library accessible in the *Graphic score manager* offers a potential starting point for a design process aimed at the dynamic spatialization of sound sources (see Fig. 2). Indeed, the proposed 2D and 3D curve templates rely on basic parametric equations that can easily transform and reshape the objects in IanniX and therefore adapt to the user's needs. The temporal behavior of a single trajectory, as well as the articulation of different movements in a macro-form is made possible by the definition of cursor's parameters: *time position* [s] and *cursor length* [s]. The former establishes the starting point in relation to global transport in IanniX; the latter defines the cursor's duration [s] along its support curve. A more advanced approach may also foresee the use of cursor acceleration patterns, recursion, or JavaScript functions, as described below. The positioning of a *sound source* can also be achieved by means of a single-step movement, for example in the event of a preset change or initial position setting. This can be practically implemented through track-level automations or direct displacement of the related trigger in the IanniX score (see Fig. 4) as well as through a gestural approach using the Mira app (see §2.3).

Overall, we have identified some significant usage strategies, not with the intention of being exhaustive but rather to illustrate some modular paths that from the instance of individual objects proceed towards a potential complete project of sound spatialization. These are listed as follows:

- constant speed trajectory system with a predetermined and predictable space-time behavior resulting in a "control score" [10]; the cursors instantiated via SST can move either along a single curve or different curves intended as their spatial path; also, they can be temporarily synchronized or not, according to the cursor parameters set in the SST graphic score manager (see *solar_system.iannix* and *DNA.iannix*);
- trajectories with non-linear speed defined by cursor acceleration patterns; each cursor can behave in a specific and independent way within the score (see *circles.iannix*); advanced object attributes such as cursor acceleration can be set via the Inspector panel, which is part of the IanniX GUI;
- variable trajectories according to a parameter value defined by track-level automation (see *supershape.iannix*); as with traditional sequencers, the automations offer the possibility of defining the variation of mapped parameter values as a function of time (see Fig. 7);
- variable trajectories according to a colliding object that modifies the parametric equation defining the cursor path; automations and events can be set directly in the representation space of the score, thus exploiting the IanniX-specific three-dimensional and poly-temporal sequencing features (see *sphere.iannix*);
- variable trajectories according to themselves in a "recursive score" setting [10]; the IanniX loopback interface allows the user to route an output message to the IanniX input for a score command instance; a command can affect a single object as well as a group of objects, as in *spirals.iannix*;
- static source positioning by means of track-level automation (see Fig. 4); the source positioning event set via the *Graphic score manager* is designed in the score at the track level; this allows the user to have a view of the entire macro-form relating to single-step movements of the sound sources; the same approach applies to the triggering of audio files via the internal player (see *sequence_1.iannix*);
- static source positioning using random functions that produce a "generative score" [10]; track-level automations can be read in unconventional ways by going beyond the notion of linear timeline; different strategies are foreseen, such as the JavaScript implementation of a random function in a trigger-related output message that sets the object's position (see *random_points.iannix*) or the use of a time-related IanniX command [13] that affects the reading position on the timeline (see *sequence_2.iannix*);
- dynamic source positioning via gestural input using the Mira interface (see §2.3); this approach produces an "interactive score" [10] in which predetermined temporal behaviors reproduced in IanniX can be combined with live controls;
- creation of trajectories and performance control through other external inputs; specific OSC/MIDI input messages sent from third-party devices can be related to adding and controlling objects in a "reactive score" [10]; this requires a JavaScript approach to the *onIncomingMessage* method included in a IanniX score (see *touchosc.iannix*).

The operational proposals listed above can be combined functionally and conveniently in a score in order to produce more complex behaviors intended as articulations of simple events in the perspective of a macro-form (i.e. the spatialization project as a whole).

5. USE CASES

SST has been used extensively in an artistic and technological scenario of *Living Lab Music*[4], a SaMPL showcase that combines contributions from established artists and researchers with artistic products from the Department of New Technologies and Musical Languages of the Conservatory of Padua. Due to the restrictions of the COVID-19 pandemic, no audience could take part in the 2021 edition. However, binaural audio and video recordings were made with the aim of disseminating the content online, while still returning an acoustic image of the specific setup and venue. Living Lab Music 8 took place at the *Pase Platform*[5] in Venice, which provided a 30-channel speaker system whose near-hemispherical arrangement (see Fig. 6) was particularly suitable for advanced spatialization. We could observe that the ICST Ambisonics equivalent panning algorithm presented a satisfactory response after adjusting the parameters (Ambisonics order and level attenuation curves, in particular). A compromise was needed between the angular precision given by the Ambisonics order increase

[4] https://vimeo.com/sampllab/albums

[5] https://pase-platform.com

and the uniformity of the audio level in all points of space, which instead required a wider polar pattern.

Ranging from fixed-media works to multimedia performances, the applications of SST in this context covered both the approaches identified in §3.3 (sometimes used in combination) and all the strategies proposed in §4, allowing us to test the full operation of SST. Early stage electronic music students could easily implement SST in their stand-alone or interactive projects after a minimal training period, only occasionally encountering issues in the interaction with the IanniX score (see §6). However, thanks to IanniX's graphic sequencing capabilities combined with the ability to control the spatialization of sound sources in real-time via Mira or other interfaces, SST has proven flexible to the diverse performative and compositional needs of the students and artists involved.

6. CONCLUSIONS

Through SST we intend to propose an easy-to-use IanniX tool aimed at the design of multi-channel sound projections for musical works and sound design projects, but also to favor an approach to electroacoustic composition stimulated by the formalization of virtual sound trajectories organized in space and time. In a future version of the software we plan to focus on several additions and bug fixes, including:

- a reverb implementation for improved distance simulation and room treatment;
- advanced management of Ambisonics directivity using independent parameters for each sound source;
- better management in the instance of IanniX objects, which can currently generate bugs related to the programming logic of IanniX score files;
- an improved trajectory library and related score examples; we realize that we have not been able to consistently systematize a curve library, and have struggled to find satisfactory models in the literature to draw from;
- the possible inclusion in the score of a further representation of cursor-related events at the track level (see §3.2); in the articulation of different trajectories, it is currently not possible to have a view on the entire macro-form, as the cursors appear only during their movement.

7. REFERENCES

[1] A. J. Berkhout, D. de Vries, and P. Vogel, "Acoustic control by wave field synthesis," The Journal of the Acoustical Society of America, 93(5), 1993, pp. 2764-2778.

[2] D. G. Malham and A. Myatt, "3-D sound spatialization using ambisonic techniques," Computer Music Journal, 19(4), 1995, pp. 58-70.

[3] V. Pulkki, "Virtual sound source positioning using vector base amplitude panning," Journal of the Audio Engineering Society, 45(6), 1997, pp. 456-466.

[4] J. Garcia, T. Carpentier, J. Bresson, "Interactive-Compositional Authoring of Sound Spatialization," Journal of New Music Research, 46 (1), 2017.

[5] T. Carpentier, "A new implementation of Spat in Max," in Proceedings of the 15th Sound and Music Computing Conference (SMC), Limassol (Cyprus), 2018, pp. 184-191.

[6] J. C. Schacher, "Seven years of ICST Ambisonics Tools for MaxMSP - A brief report," in Proceedings of the 2nd International Symposium on Ambisonics and Spherical Acoustic, IRCAM, Paris, 2010.

[7] A. Sèdes, P. Guillot, and E. Paris, "The HOA library, review and prospects," in Proceedings of the 40th International Computer Music Conference and 11th Sound and Music Computing Conference, Athens (Greece), 2014, pp. 855-860.

[8] C. Miyama, G. Dipper, and L. Brümmer, "Zirkonium MKIII - A toolkit for spatial composition," Journal of the Japanese Society for Sonic Arts, 7(3), 2015, pp. 54–59.

[9] D. Ledoux et al., "SpatGRIS/ServerGRIS, Creative tools for 2D and 3D sound spatialization," in Proceedings of the 43rd International Computer Music Conference (ICMC), Daegu, 2018, pp. 291-297.

[10] J. Scordato, "Novel perspectives for graphic notation in IanniX," in L. Brümmer, S. Kanach, and P. Weibel (eds.), *From Xenakis's UPIC to Graphic Notation Today*. ZKM | Karlsruhe and Hatje Cantz, 2020, pp. 578-587.

[11] J. Scordato, M. Barberis, and N. Matteo, "Sketching Sonic Trajectories," in Proceedings of the Sound, Image and Interaction Design Symposium (SIIDS), Madeira, 2020.

[12] S. Tarakajian, D. Zicarelli, and J. Clayton. "Mira: Liveness in iPad Controllers for Max/MSP," in Proceedings of the International Conference on New Interfaces for Musical Expression (NIME), Daejeon, 2013, pp. 421-426.

[13] Messages and commands - IanniX Wiki - GitHub https://github.com/buzzinglight/IanniX/wiki/4.-Messages-and-commands#42-received-commands