

Towards Interactive Authoring Tools for Composing Spatialization

J r mie Garcia* Jean Bresson† Thibaut Carpentier‡
UMR 9912 STMS IRCAM-CNRS-UPMC

ABSTRACT

We present interactive tools designed to help music composers controlling sound spatialization processes in a computer-aided composition environment. We conducted interviews with composers to understand their needs and inform the design of new compositional interfaces. These interfaces support quick input, visualization and edition of three-dimensional trajectories, as well as the control of the temporal dimension in spatial scene descriptions.

Index Terms: H.5.5 [Information Systems]: Information Interf. and Presentation—Sound and Music Computing;

1 INTRODUCTION

Contemporary music composers extensively use sound spatialization techniques as a means to increase the formal or acoustic dimensionality of music [2]. These techniques create the illusion that sound sources come from various directions in space and organize them in three-dimensional scenes.

Despite the development of advanced spatial audio renderers [5], the control of sound spatialization remains a challenging task for composers [6]. Spatialization processes require users to define the location of sound sources, as well as a great number of other spatial features such as their orientation, the amount of doppler effect or the reverberating characteristics of the virtual space. As these parameters often vary to produce sound motions or other dynamic effects, composers also need to specify their temporal evolutions. In addition, the compositional focus implies this time determination to be carefully related with the score parts.

Many composers use visual programming environments such as OpenMusic [1] or Max to create and run programs that combine algorithmic processing with musical material or notation. Previous works explored tools and formats to generate, represent and manipulate spatial parameters within this kind of environment [3]. However they mostly focused on the integration of complex multidimensional data in compositional processes rather than providing interactive tools to support end user interactions.

Spatialization control requires interactions in a three-dimensional environment that represents the acoustic world. It presents interesting challenges for 3D user interfaces because it also involves a temporal dimension, sound perception factors, and the need to provide live aural feedback [7].

Several researchers already explored interaction techniques to select, move and interact with objects in space [4]. We focus on a different task: the musical composition of *temporal spatial scenes*. Our goal is to design interactive systems to support composers' work with spatialization. We aim at providing tools for defining sound sources trajectories in potentially complex scenes that can be connected to temporal structures and to other aspects of compositional processes.

2 OBSERVATIONS

We conducted interviews with four professional composers experienced in sound spatialization in order to better understand their needs and reflect on new interactive tools to facilitate the monitoring and writing of spatialization processes. We studied the representations and the software they use during their composition process and we focused on the problems they encounter.

We observed that composers primarily work with visual representations of the sound space, for sketches as well as for the final scores. Each composer follows a unique process, combining personal graphical representations with various musical concepts. They explained that they need such representations to explore ideas and articulate them with other musical elements usually described in the score.

All four composers mentioned the temporal specification of spatial processes as a major problem. They also emphasized their need for a global vision of the piece and a more detailed one, generally multidimensional (many sources and many parameters), in a unified context. They were all interested in potential interactive ways to capture, visualize and render trajectories as well as other spatial and temporal parameters. Two composers discussed the need to quickly input and hear trajectories to explore their ideas before developing more complex processes.

3 PROTOTYPES

Based on these observations, we decided to improve the tools available in the OpenMusic computer-aided composition environment for specifying 3D-trajectories [3] and describing temporal spatial scenes with multiple sound sources.

Figure 1 shows an OpenMusic visual program containing our prototype objects¹. OpenMusic "objects" are data structures built and used in visual programs. Composers can create them programmatically from lists of parameters or using graphical editors. Objects and functions connected together in visual programs constitute a data-flow graph corresponding to a compositional process.

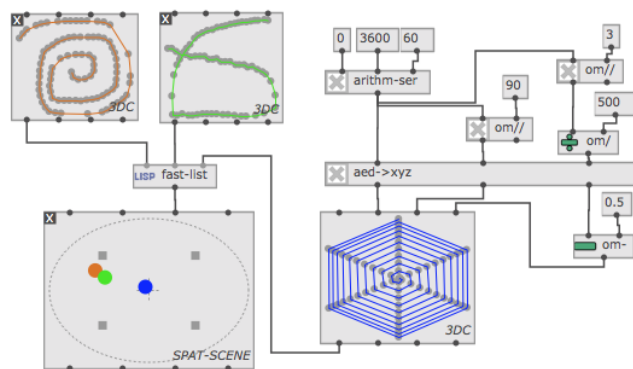


Figure 1: Construction of a spatial scene object in an OpenMusic visual program. Each 3DC object contains a trajectory. The SPAT-SCENE displays the initial positions of the three trajectories.

*e-mail: jgarcia@ircam.fr

†e-mail: bresson@ircam.fr

‡e-mail: thibaut.carpentier@ircam.fr

¹The OpenMusic objects are implemented in Common Lisp. We used OpenGL for the 3D rendering.

3.1 Editing / Visualizing Trajectories

The 3DC object allows to view a 3D curve and edit it with two synchronized 2D editors (top/front view). Figure 2 shows the graphical user interface of a 3DC object with the 3D and 2D views, as well as the parameters panel providing a number of visualization options. For instance, the user can map a color scale to the length or speed of the trajectory. These features are meant to help composers to easily understand the motion of a trajectory and to quickly recognize fast trajectories from slow ones.

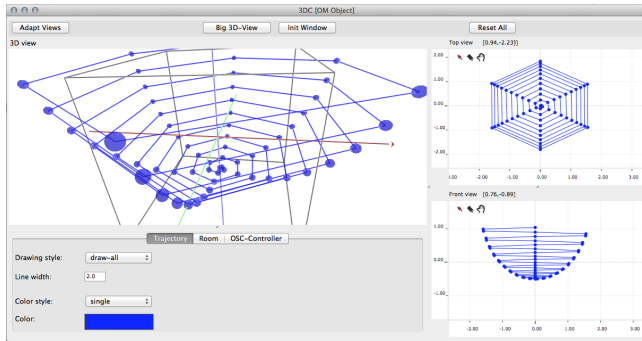


Figure 2: Graphical editor for a 3D trajectory object (3DC).

3.2 Gestural 3D Input

To support gestural input from external controllers, the 3DC interface embeds an OSC² server receiving position messages from various input devices. The protocol consists in two main messages: the *move* message (move a sound source in the 3D space) and the *draw* message (append points to the trajectory). A spatial rendering engine provides real-time audio feedback and the graphical user interface displays a green or a red sphere in the 3D view when the user respectively moves or “draws” in the trajectory (Figure 3.a).

We experimented this system with a *GameTrak* controller. This controller provides two 3D positions localized from the length and angle of two wires articulated on a base. The wires are tied to two fingers of one hand. Figures 3.b and 3.c present the two hand postures used to switch between *move* and *draw* modes. The posture detection is made using a distance threshold between the points. This simple interaction technique allows the user to explore sound source movements with audio-visual feedback and record the gesture by joining fingers and performing mid-air hand movements.

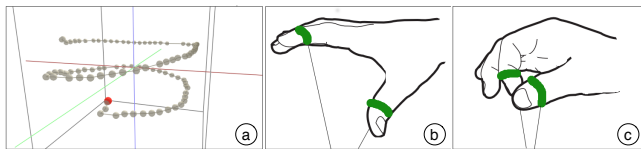


Figure 3: Gestural input of a new trajectory using the *GameTrak* controller: a) 3D view with visual feedback; b) free move hand posture; c) draw mode hand posture.

3.3 Temporal Spatial Scenes

SPAT-SCENE is an other OpenMusic object used to create, view and edit spatial sound scenes and their evolution in time. This object is built from a set of sound sources and 3DC objects. Figure 1 contains a SPAT-SCENE object at the bottom-left. Figure 4 shows its graphical editor.

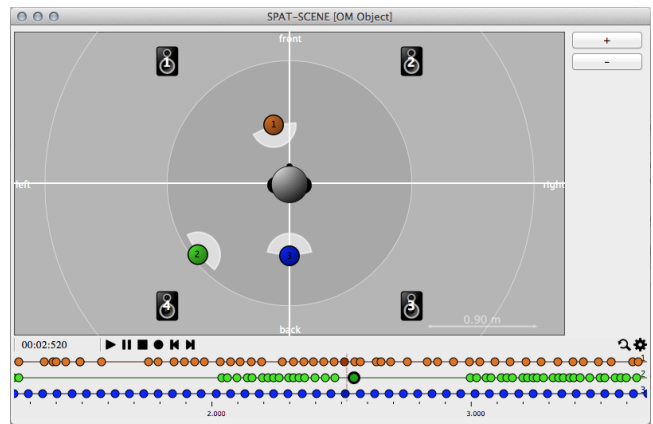


Figure 4: Graphical editor for the SPAT-SCENE object.

The SPAT-SCENE editor is divided in two main areas: the sound scene (top) and the *timeline* (bottom). The sound scene view controls the sound sources and the speakers positions at a given moment. The *timeline* contains, for each source, the times of all source position changes. These time positions are equivalent to *keyframes* in video editors. Users can determine the rendering interpolation rules for the source positions between successive keyframes.

4 CONCLUSION AND PERSPECTIVES

We presented interactive tools for authoring sound spatialization processes in the OpenMusic computer-aided composition environment. Interviews with four professional composers helped us understanding their needs for input and visualization of complex temporal scenes with different levels of details. Our prototypes address these needs for specific scenarios which can be generalized and adapted to other devices or environments.

The tools can be integrated into composers’ workflow to control the production of spatialized audio while exploring alternatives with real-time audio-visual feedback. We believe this approach will enhance composers’ creative processes by fostering the expression and evaluation of musical ideas. Our future plan is to collaborate with composers to refine and assess these tools in real music production contexts.

ACKNOWLEDGEMENTS

We thank the composers for their time and valuable insights. This work is part of the EFFICACE project (ANR-13-JS02-0004-01).

REFERENCES

- [1] G. Assayag, C. Rueda, M. Laurson, C. Agon, and O. Delerue. Computer Assisted Composition at IRCAM: From PatchWork to OpenMusic. *Computer Music Journal*, 23(3), 1999.
- [2] M. A. Baalman. Spatial composition techniques and sound spatialisation technologies. *Organised Sound*, 15(03):209–218, 2010.
- [3] J. Bresson. Spatial Structures Programming for Music. In *Spatial Computing Workshop (SCW'12)*, Valencia, Spain, 2012.
- [4] J. Jankowski and M. Hachet. A Survey of Interaction Techniques for Interactive 3D Environments. In *Eurographics*, Girona, Spain, 2013.
- [5] J.-M. Jot. Real-time Spatial Processing of Sounds for Music, Multimedia and Interactive Human-Computer Interfaces. *Multimedia Systems*, 7(1):55–69, 1999.
- [6] N. Peters, G. Marentakis, and S. McAdams. Current technologies and compositional practices for spatialization: A qualitative and quantitative analysis. *Computer Music Journal*, 35(1):10–27, 2011.
- [7] J. Thibaut, B. Gilles, L. Eric, F. Raphaël, and C. Etienne. A design space for three-dimensional curve edition. In *IHM'14, 26e conférence francophone sur l'Interaction Homme-Machine*, Lille, France, 2014.

²OpenSoundControl: <http://opensoundcontrol.org>