Tools for the Writing of *Tamonontamo* (2012): A new way to relate concatenative synthesis and spatialization

Maurilio Cacciatore
Ircam, Paris, France
maurilio.cacciatore [at] ircam.fr
http://www.mauriliocacciatore.com

Diemo Schwarz
Real-Time Music Interaction Team (IMTR),
IRCAM–CNRS–UPMC, Paris, France
diemo.schwarz [at] ircam.fr

Tamonontamo (2012) is a piece for amplified vocal quartet, choir of 24 singers and live electronics. The focus of this article is the work made in collaboration with the Real-Time Music Interaction Team of Ircam. Augustin Muller, computer music programmer working with the first author at Ircam for this project, upgraded the CataRT software adding a Spat~ module to relate the corpus-based synthesis with the spatialization. In most works, the spatialization of sounds depends on the aesthetic choice of the composer. The Spat~ software allows the drawing of linear movements of a source among a pre-selected number of loudspeakers with a user-friendly graphical interface. In this patch, we used Spat to spatialize the sounds in a non-linear way. The logic of spatialization depends on the pair of audio descriptors chosen in the set-up of the CataRT graphical display. In this sense, the aesthetic ideas of the composer refer not locally to each sound but on the choosing of the audio descriptors used for the x, y axis. A further Spat-related display has been implemented to split the space used for analysis/playing from the space used for diffusion of sounds. The implementation of the *Unispring* algorithm in the *mnm.distribute* external Max/MSP object allows the distribution of grains, rescaling their position inside a pre-drawn sub-space. The interpolation between different shapes or the change of audio descriptors on the x, y axis for both displays can be programmed and made in real time. The storage of the synthesis as database (in a text file), allows the analysis to be recalled, recovering the shapes drawn previously. Hiding and permanent deletion of some selected corpora in the database is also possible. The use of colour to work on the z axis is possible and is among the next steps of this work. The upgrade of the use of Spat~ to an ambisonics system for diffusion of sounds represents the possibility to make—through the use of concatenative synthesis—a genuine work of 3D-sound-sculpting.

The project “Identité et environnement sonore”

Tamonontamo[1] (2012, Suvini Zerboni Edition, Milan, Italy) is a piece for amplified vocal quartet, choir of 24 singers (6S-6A-6T-6B) and live electronics. It has been commissioned by the Choir *Les Cris de Paris*, conducted by Geoffroy Jourdain and it was created on the 16th June at the festival *ManiFeste* in Paris.

The electronics have been realized at the Ircam studios during a four-month residency in collaboration with the Real-Time Music Interaction Team[2]. The Max patch and the work on the live-electronics setup has been realized by the Computer Music programmer Augustin Muller.

The work is the result of a long-term project called “Identité et environnement sonore” in collaboration with the ARTE Radio of Paris. A selection of students aged between 6 and 15 recorded with the help of some advisors a series of sounds representing the daily life from several ambiances: home, subway, streets, churches, etc. These sounds have been used in some parts of the piece as basic material to be treated, as it will be described in the next paragraphs.

This article will focus on the strategies for the concatenative synthesis used in this work. A special way to relate this synthesis and the spatialization will be explained through its mathematical aspects and its application in Max/MSP.

Corpus-based concatenative synthesis

The recent technique of corpus-based concatenative sound synthesis (Schwarz 2007) builds up a database of pre-recorded or live-recorded sound by segmenting it into units, usually of the size of a note, grain, phoneme, or beat, and analysing them for a number of sound descriptors, which describe their sonic characteristics. These descriptors are typically pitch, loudness, brilliance, noisiness, roughness, spectral shape, etc., or meta-data, like instrument class, phoneme label, etc., that are attributed to the units. These sound units are then stored in a database (the corpus). For synthesis, units are selected from the database that are closest to given target values for some of the descriptors, usually in the sense of minimizing a weighted Euclidean distance. The selected units are then concatenated and played, possibly after some transformations have been applied.

Corpus-based concatenative synthesis and related approaches are summarized in a survey (Schwarz 2006) that is constantly kept up-to-date on-line[3].

The CataRT system

The CataRT software system[4] (Schwarz / Beller / Verbrugghe / Britton 2006) is a collection of patches for Max/MSP that realises corpus-based concatenative synthesis in real-time. It is a modular system based on the freely available FTM&Co extensions (Schnell / Borghesi / Schwarz / Bevilacqua / Müller 2005), providing optimized data structures and operators in a real-time object system, including the Gabor library (Schnell / Schwarz 2005) for arbitrary rate signal processing, and the MnM library for matrix processing and statistics (Bevilacqua / Muller / Schnell 2005)[5].

CataRT was created in 2005 and is used in various contexts of sound design, music composition and performance, and installations (Schwarz / Britton / Cahen / Goepfer 2007; Schwarz / Cahen / Britton 2008). The most straightforward interaction modality is based on a simple interface consisting of a 2D projection of the descriptor space, and navigation with the mouse, where the grains are selected and played depending on geometric proximity. When the navigation is controlled by gestural input devices, CataRT can become a genuine musical instrument, as examined in (Schwarz 2012).

The CataRT software allows to store, recall and merge the corpus-based concatenative synthesis as text-files. In this sense, the choice and storage of parameters as a database of numerical information is independent from the storage of audio file. The merge of several analyses does not ask for bouncing the treated audio files together.

Many creative, industrial and pedagogic applications rely on the interactive use of sound databases. Data as diverse as full-length recordings, samples, sound grains, used for instance by corpus-based concatenative synthesis methods (Schwarz 2007) or even non-audio elements such as synthesizer presets, are used to achieve the specific form of expressivity required by the application. A sound representation is needed to allow the user to interact with the database elements: in the past years, feature-based representations of sound have attracted much attention. Closely related to the field of content-based music information retrieval, they allow greater insight to the data than the usual classification by keywords. Such representations can efficiently be used for interacting with sound and relating the representation of the synthesis with the representation of the space of the performance (Lallemand / Schwarz 2011).

CataRT has been integrated in Max/MSP and it can be used by sending and receiving signals and data from other signal processing inside a patch. Even though it allows its use as standalone application, it can be integrated as a module in a more complex Max/MSP patch.

Figure 1 shows the main workspace of the software. In both the standalone and the modular version, this window appears in this way; each parameter is customizable and can be stored within a text file. The green circle indicates the grains played when the DSP is on. Several kinds of reading (distribution, connection with the mouse movements, speed, etc.) of the activated corpora are possible.

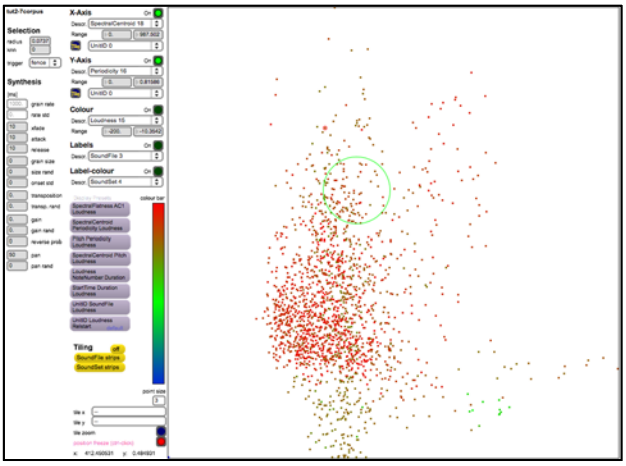

Figure 1. CataRT main workspace. The corpus has been distributed with the XY pair of audio descriptors SpectralCentroid/Periodicity. The colours correspond to the loudness of each grain.

Audio descriptor analysis

Extraction of descriptors from audio signals is an essential technique in many fields of computer music research and production, and music information retrieval. It is becoming increasingly important for the next generation of real-time processing systems, such as content-based processing (Amatriain 2003), corpus-based concatenative synthesis, and multi-modal applications including gesture analysis.

An audio descriptor is a value that describes a certain quality of a sound, which can evolve over time or be constant. There are a multitude of useful audio descriptors, and each can be calculated in a number of ways and variants, see e.g. Peeters (2004) for an overview.

The descriptors calculated in CataRT are the fundamental frequency, aperiodicity, loudness, and a number of spectral descriptors: spectral centroid, sharpness, spectral flatness, high and mid frequency energy, high frequency content, and the first order autocorrelation coefficient (which expresses spectral tilt).

The algorithm *Unispring* and its integration in CataRT

Mathematical aspects

The *Unispring* algorithm (Lallemand / Schwarz 2011) has been developed by Ianis Lallemand during a Master's degree internship at Ircam[6] between March and June 2010. It allows the spreading the data points within a user-defined region. As it will be shown below, this region corresponds to the space of sound diffusion.

The physical algorithm part of *Unispring* adapts the physical model used in *Distmesh*[7], a MatLab Toolbox (Persson

/ Strang 2004) to relocate previously existing points (our initial scatterplot), whereas *Distmesh* is aimed at generating a mesh over a blank region.

The user can specify the final data points density by providing a desired length function $h(x,y)$. If (x,y) are the coordinates of the middle of two points, $h(x,y)$ gives a target distance between these points that should be reached after applying the algorithm. The resulting distances are in fact proportional to those specified by $h(x,y)$, which actually gives the relative distribution over the region. Making the density uniform can thus be obtained by providing any uniform length function.

The algorithm works through these steps. In a list of points with a 2D position $i_{(x,y)}$ (Figure 2):

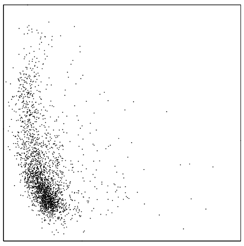

Figure 2. Starting distribution of points

1. sort coordinate value list $(x_i)_{1\le i\le N}$ (resp. $(y_i)_{1\le i\le N}$). For each position i, it gets the positions $n(i)$ in the sorted list;

1b. fill output coordinate values list $(x'_i)_{1\le i\le N}$ such as $x'_i = n(i)\ \left(\text{resp.}\ (y'_i)_{1\le i\le N}\right)$;

2. normalize the resulting coordinate values so that all data points are inside the user-specified region (Figure 3);

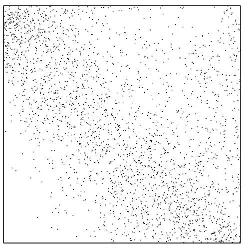

Figure 3. Distribution of points after normalization of the resulting coordinates

3. perform a Delaunay triangulation of the data points. The triangulation of data points appears only if the mesh points have moved more than the maximum allowed distance during the previous iteration. The triangulation defines a truss structure where edges of the triangles (the connections between pairs of points) correspond to bars, and points correspond to joints of the truss. Each bar has a force-displacement relationship $f(l,l_0)$ depending on its current length l and its unextended length l_0 (which is computed using the desired length function). To help points spread out across the whole used-defined region, only repulsive forces are allowed:

$$\begin{cases} f(l,l_0) = k(l_0 - l) & \text{if } l < l_0 \\ f(l,l_0) = 0 & \text{if } l > l_0 \end{cases}$$

4. update data points positions according to a physical algorithm;
5. points that go outside the region during the step 4 are moved back to the closest boundary point. This corresponds to the physical image of external reaction forces on the truss structure, that act normal to the boundary. Hence points can move along the boundary, but not go outside;
6. if all interiors points move less than a significant distance, the algorithm exits the loop, otherwise it goes back to step 3 (Figures 4 and 5).

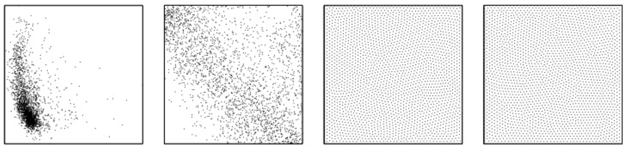

Figure 4. Distribution of an example corpus: from left to right, *original distribution, uniform, spring, unispring.*

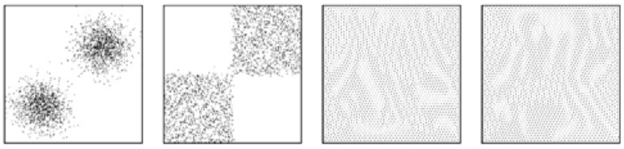

Figure 5. Gaussian distribution: from left to right, *original distribution, uniform, spring, unispring.*

In the above figures, *uniform* refers to the pre-uniformisation steps of *unispring*, and *spring* to the physical algorithm without pre-uniformisation.

The integration of *Unispring* in CataRT

The *Unispring* algorithm has been implemented in the external Max/MSP object *mnm.distribute*.

Combined with some controls in the patch, it provides the ability to organize all points within some pre-formatted shapes (homogeneous distribution on the whole space, circle at the centre of the space, triangles with the base on one side) or within a shape drawn manually.

The logic of distribution inside the drawn sub-space keeps the logic of distribution made by the audio descriptors because, as mentioned before, the algorithm works through a process of rescaling points.

Two “empty” audio descriptors have been added on the list (we called them $DistX, DistY$). When exporting the database as text file, the rescaling of points within the subspace is stored, so it may be recalled automatically in a further moment.

If some sound units are not suitable, they can be deactivated or deleted from the text file that stores all data.

The interpolation between two shapes can be programmed with a customizable time.

The IRCAM Spat~

The Spatialisateur project started in 1991 as collaboration between Espaces Nouveaux and Ircam. Its goal is to propose a virtual acoustics processor, which allows composers, performers or sound engineers to control the diffusion of sounds in a real or virtual space. This project stems from research carried out within the Ircam room acoustics laboratory on the objective and perceptive characterization of room acoustic quality. It also incorporates research done at Télécom Paris on digital signal processing algorithms for the spatialization and artificial reverberation of sounds. Spat~ is an effort to organize and optimize the experimental patches developed in the Spatialisateur project, in order to make them accessible to musicians and researchers who work with Max/MSP. The current release allows reproduction on multi-channel loudspeaker systems in studios or concert halls. It also integrates 3D stereo reproduction modes for headphones (binaural) or 2 loudspeakers (transaural), as well as Vector Based Amplitude Panning and Ambisonics[8] (Figure 6).

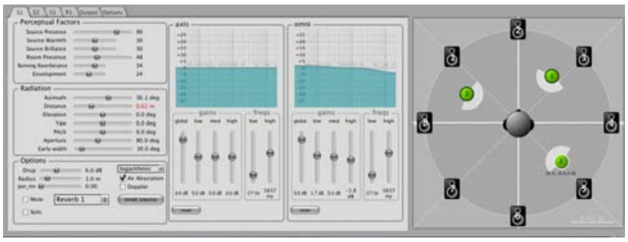

Figure 6. Spat~ graphical display, source parameters window with interface of control of the source movements. It opens through the Max object *spat.oper*.

The Spat~ software is a collection of external objects in Max/MSP. All parameters above and others in the hidden windows can be controlled through some messages in patch (Figure 7); the output values can be in degrees or in metres. The software allows a high-level control of the diffusion of sounds based on perceptual factors.

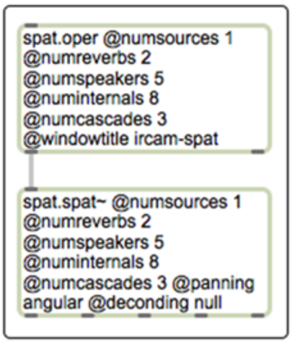

Figure 7. Example of setup of a *spat.oper* Max object connected with a *spat.spat~* module. The opening and closing of the graphical display is activated through the message objects *open* and *wclose*.

A number of reverbs can be added on each module. Figure 8 shows the graphical display of one reverb. Both the position of the loudspeakers and of the listener are customizable.

Figure 8. Spat graphical display, reverberation window. Each reverb set up in the *Spat.oper~* module has an independent window.

All parameters can be programmed by patching and can be controlled through human interfaces. Both direct and reverberated sound can work in 2D or in 3D: the parameter "elevation" works on the z axis. In *Tamonontamo*, Spat~ has been used in 2D, according with the actual functionality of CataRT.

The integration of the Spat~ in CataRT

The compositional need I had was to make the space of CataRT correspond to an exhibition space, such as the concert hall where the piece has been performed.

The first step of this upgrade has been to relate the Spat~ window with the CataRT graphical display (Figure 9):

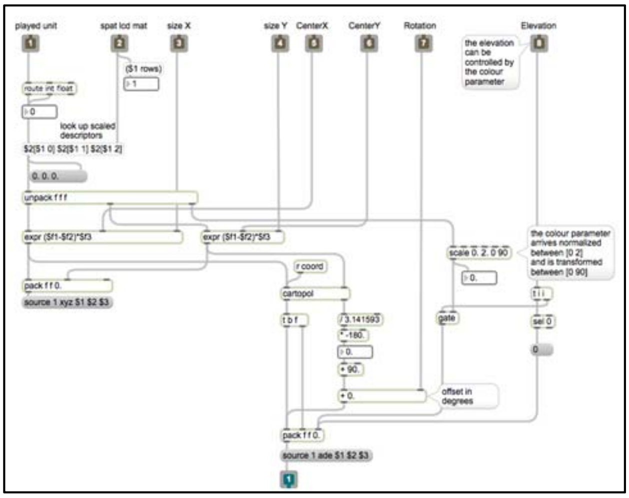

Figure 9. Sub-patch with the connections between CataRT and Spat~

A second graphical display has been implemented in the next step of work on the Max/MSP patch. The strict correspondence between the space of synthesis and the space of exhibition gave while reading the corpus inside CataRT a "linear" result, because of the exact correspondence of the grains activated through the moving of the cursor on screen and their position on the lcd corresponding with the audio descriptors used for the x, y.axis.

Splitting the actions on two independent spaces, we decided to connect the external object *mnm.distribute*[9], which contains practically the Unispring algorithm, to the Spat-related graphical display. Through this solution, the main space of CataRT works as the space of analysis and reading of grains whose position depends according with the couple of audio descriptors selected for the axis x, y; the Spat-related graphical display works as space of physical-distribution of the grains in the exhibition space. The diffusion of the sounds depends from the position of sub-spaces drawn on this display and the factors programmed in the Spat~ (distance of loudspeakers, reverb, filters, etc.).

As shown in the patch above, the Spat~ used for this sound treatment works with one single source; it corresponds with the grain selected each moment by the cursor on the CataRT graphical display.

The exploration with the cursor on the first graphical display allows in this way a non-linear reading of the space of exhibition, if on the Spat-related display two different audio descriptors have been selected. The energy of this spatialization by points keeps a strict logic inside itself because the spatial distribution of the grains respects the distribution made by the audio descriptors.

The merge of different shapes (Figure 10) is possible and gives the opportunity to mix corpora coming from different audio files in the same space of playing. Each list of sounds has an independent shape that makes the final result more complex.

The images of the Spat-related display at the end of the article show the steps to arrive to merge different shapes within the same space of diffusion.

Strategy of use

In *Tamonontamo* we used this technique of concatenative synthesis for the treatment of the sounds collected by the project with ARTE Radio to make a list of multichannel sound files to diffuse during the piece.

The strategy of use can be summarized in these steps:

1. Choose the audio descriptors on the main CataRT graphical display to organize the corpora and set up the parameters for the granulation;
2. Draw a shape that corresponds to a subspace of diffusion in the Spat-related graphical display and rescale the points. When doing this, the choice of audio descriptors in the CataRT main display is not only technical but also aesthetic, because the organization of points within the shape depends on the distribution on the main window;
3. Choose one feature of exploration of the CataRT main display (size of the activation area, activation of corpora when mousing[10] or automatic, speed of reading the points within the activation area if it does not move, etc.) and start to record;
4. To change the internal organization of distribution of grains keeping the shape, change the audio descriptors after specifying a time of interpolation; to change the space of diffusion, make an interpolation between two shapes (or from a shape, i.e. with the descriptors $DistX, DistY$, to some other audio descriptors) after specifying a time of interpolation;
5. To fill the space with more corpora, repeat the steps 1–3 importing other audio files or recalling some previous databases of sounds, then record a multichannel sound file.

Next steps

In the future we hope to be able to use this technique of spatialization in real time and passing from 2D to 3D.

The patch is almost ready to use the colour to manage the elevation of sounds but some tests about filtering of sounds and reverberation are necessary.

An improvement of the reverberation technique is necessary for the spatial impression of sounds when diffused. The passage to the Ambisonics system of spatialization could be useful to make the perception of sub-spaces of diffusion more natural.

The use of two or more CataRT objects would give the possibility to adopt different grain parameters to the sub-spaces, associating the n subspaces to n sources of the Spat~.

The goal is to make a genuine 3D-sound-sculpting possible: an improvement of the human–computer interface is already in progress by the second author and the Real-Time Music Interaction Team of Ircam; it is necessary to make the performances with CataRT more user-friendly, improving the idea of CataRT as musical instrument that looks at the performer as composer and vice versa.

Conclusion

CataRT is a music software for concatenative synthesis by databases of sounds. We showed how the integration of the algorithm Unispring gives the possibility of rescaling the distribution of points in the space or in some manually chosen sub-spaces.

The addition of a second graphical display splits the space of analysis from the space of diffusion and both are customizable and controlled in the same Max/MSP patch.

This technique of spatialization does not refer to an instinctive sound spacing just supported by the will of the composer but keeps the logic of the audio descriptors that rule the distribution. The will and the aesthetic of the composer does not work directly on the distribution of points in the space but on the choice of the audio descriptors; then, the complexity of the space depends on the merging of several sub-spaces where the points of some databases of sounds have been rescaled.

The integration with Spat~ offers the possibility to relate the diffusion of grains to the acoustical parameters of this software.

Both in studio and in live performing, CataRT goes toward the role of musical digital instrument where sound sculpting and spatialization work together.

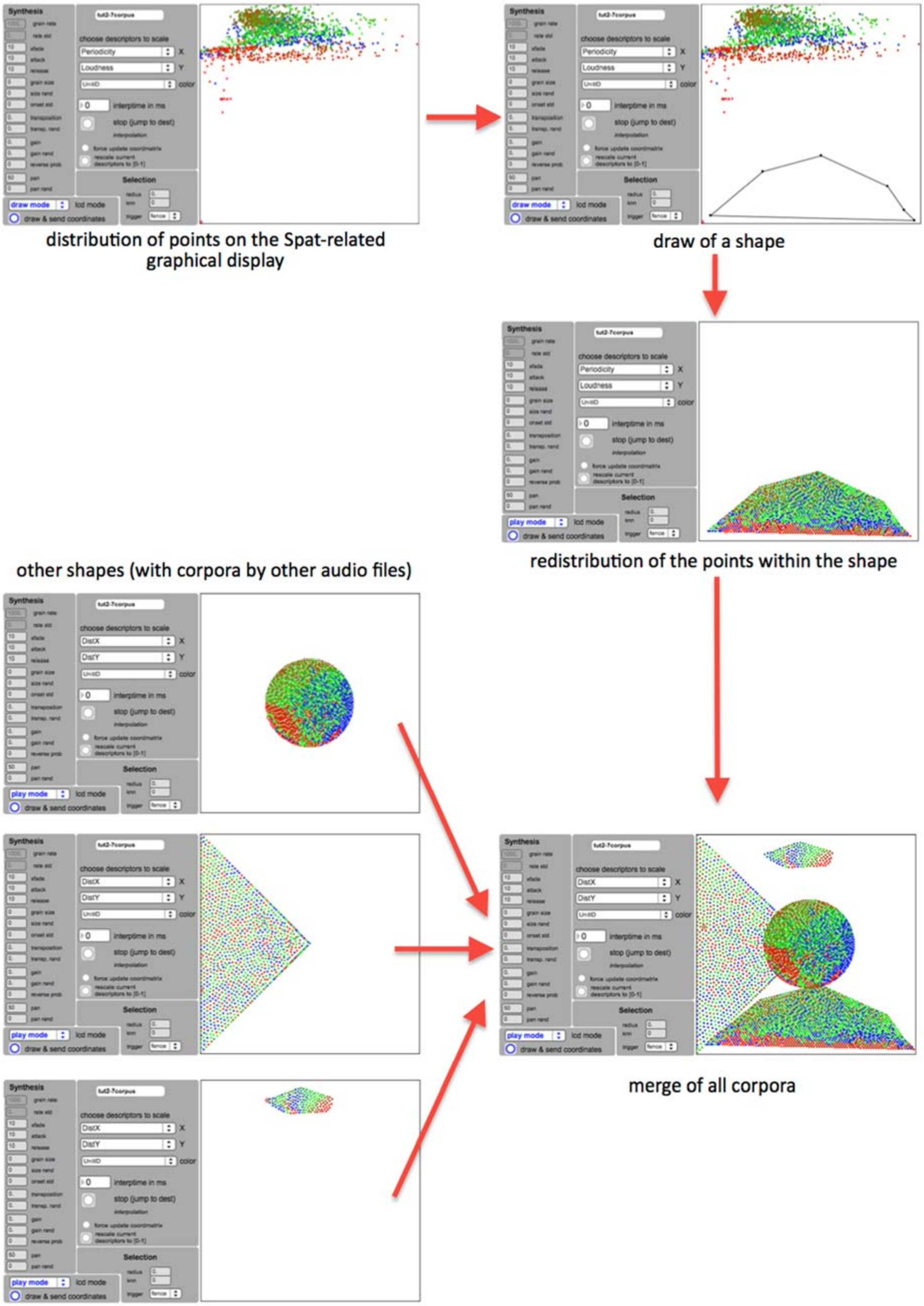

Figure 10. Process of rescaling points within a shape and merge of four corpora

References

Amatriain, X. / Bonada, J. / Loscos, A. / Arcos, J. / Verfaille, V. (2003). Content-based transformations. *Journal of New Music Research* 32/1: 95–114.

Bevilacqua, F. / Muller, R. / Schnell, N. (2005). MnM: a Max/MSP mapping toolbox. In: *Proceedings of the 2005 International Conference on New Interfaces for Musical Expression* pp. 85–88. Vancouver, Canada.

Lallemand, I. / Schwarz D. (2011). Interaction-optimized sound database representation. In: *Proceedings of the 14th Conference of Digital Audio Effects (DAFx-11)* DAFx-292–DAFx-299. Paris, France.

Nguyen / Burkardt, J. / Gunzburger, M. / Ju, L. / Saka, Y. (2009). Constrained CVT meshes and a comparison of triangular mes generators. *Computational Geometry* 42/1: 1–19.

Peeters, G. (2004). *A large set of audio features for sound description (similarity and classification) in the Cuidado project*. Ircam – Centre Pompidou, Tech. Rep. Retrieved from http://recherche.ircam.fr/anasyn/peeters/ARTICLES/Peeters_2003_cuidadoaudiofeatures.pdf on August 30, 2013.

Persson, P. / Strang, G. (2004). A Simple Mesh Generator in Matlab. *SIAM Review* 46/2: 329–345.

Schnell, N. / Borghesi, R. / Schwarz, D. / Bevilacqua, F. / Müller, R. (2005). FTM-Complex Data Structures for Max. In: *Proceedings of the International Computer Music Conference (ICMC)*. Retrieved from http://recherche.ircam.fr/equipes/temps-reel/articles/ftm.longer2005.pdf on August 30, 2013.

Schnell, N. / Schwarz, D. (2005). Gabor, Multi-Representation Real-Time Analysis/Synthesis. In: *Proceedings of the COST-G6 Conference on Digital Audio Effects (DAFx)* DAFX-1–DAFX-5. Madrid, Spain.

Schwarz, D. (2006). Concatenative sound synthesis: The early years. *Journal of New Music Research* 35/1: 3–22. (Special Issue on Audio Mosaicing)

_________. (2007). Corpus-based concatenative synthesis. *IEEE Signal Processing Magazine* 24/2: 92–104. (Special Section: Signal Processing for Sound Synthesis)

_________. (2012). The Sound Space as Musical Instrument: Playing Corpus-Based Concatenative Synthesis. In: *New Interfaces for Musical Expression (NIME)* pp. 250-253 Ann Arbour.

Schwarz, D. / Beller, G. / Verbrugghe, B. / Britton, S. (2006). Real-Time Corpus-Based Concatenative Synthesis with CataRT. In: *Proceedings of the COST-G6 Conference on Digital Audio Effects (DAFx)* DAFX-279–DAFX-282). Montreal, Canada.

Schwarz, D. / Britton, S. / Cahen, R. / Goepfer, T. (2007). Musical applications of real-time corpus-based concatenative synthesis. In: *Proceedings of the International Computer Music Conference (ICMC)*. Copenhagen, Denmark.

Schwarz, D. / Cahen, R. / Britton, S. (2008). Principles and applications of interactive corpus-based concatenative synthesis. In: *Journées d'Informatique Musicale (JIM)*. GMEA, Albi, France.

---

[1] This work has been partially funded by the Agessa (*http://www.agessa.org*). The participation at the KEAMS 2013 annual conference has been partially funded by the Conservatory "A. Steffani", Castelfranco Veneto, Italy. *http://www.conscfv.it*.

[2] *http://imtr.ircam.fr*

[3] *http://imtr.ircam.fr/imtr/Corpus-Based_Sound_Synthesis_ Survey*

[4] CataRT is released as free open source software at *http://imtr.ircam.fr*

[5] *http://ftm.ircam.fr*

[6] This work has been partially funded by the *Agence National de la Recherche* within the project *Topophonie*, ANR-09-COORD-022. *http://topophonie.fr*

[7] The *Distmesh* algorithm generates unstructured triangular meshes using a physical algorithm. It is based on a simple mechanical analogy between a triangular mesh and a 2D truss structure, or equivalently a structure of springs. It provides a mathematical framework that allows the user to specify the internal geometry of the mesh as well as the region over which it has to be generated.

[8] *http://redmine.spatdif.org/projects/spatdif/wiki/IRCAM_Spatialisateur*

[9] This object has been programmed by Ianis Lallemand in collaboration with the Real-Time Music Interaction Team of Ircam, 2011-2012.

[10] In the studio, the CataRT main graphical display was explored and payed with a joystick.

[Abstract in Korean | 국문 요약]

《타모논타모Tamonontamo》의 작곡 도구: 연쇄 합성과 공간화를 이어 주는 새로운 방법

마우릴리오 카차토레 / 디에모 슈바르츠

《타모논타모》는 증폭된 인성 4중주와 24인의 합창단, 라이브 일렉트로닉스를 위한 작품이다. 이 글의 초점은 이르캄IRCAM의 실시간 음악 상호작용 팀과의 합작으로 이루어진 작업에 있다. 본 연구 과제의 제1저자와 작업한 이르캄의 컴퓨터 음악 프로그래머 아우구스틴 뮐러Augustin Muller는 소리뭉치기반 합성corpus-based synthesis을 공간화와 연관시키기 위해 스팟~Spat~ 모듈을 덧붙여 카타알티CataRT 소프트웨어를 개량하였다. 대부분 작품에서 소리의 공간화는 작곡가의 미적 선택에 의존한다. 스팟~ 모듈은 사용자에게 친숙한 그래픽 인터페이스로 미리 선택된 수만큼의 스피커 사이에서 음원의 선형 이동을 그려내는 것을 가능하게 한다. 이 패치patch에서 스팟은 비선형적 방식으로 소리를 공간화하도록 사용되었다. 공간화 논리는 카타알티 그래픽 표시의 설정에서 선택된 오디오 기술어audio descriptor에 좌우된다. 이러한 점에서 작곡자의 미적 의도는 국부적인 개개의 소리가 아닌 x, y축에 사용되는 오디오 기술어를 선택하는데 적용된다. 그 이상의 스팟 관련 표시는 분석 및 연주에 사용되는 공간에 소리를 확산시키는 데 사용되는 공간으로부터 분리하는 것을 가능하게 한다. 맥스 외부 개체인 엠엔엠.디스트리뷰트mnm.distribute의 유니스프링 알고리즘Unispring algorithm 구현은, 미리 그려진 하위 공간 내에서 소리 낟알들grains의 위치가 비율적으로 조절되며 분포되는 것이 가능하도록 한다. 상이한 형태의 보간interpolation 혹은 x, y 축에서의 오디오 기술어의 변화 역시 프로그래밍되거나 실시간으로 제어될 수 있다. 합성 내용은 텍스트 파일 내에 데이터베이스로 저장되어 재분석되거나 이전 형태로 되돌릴 수 있다. 데이터베이스에서 선택된 소리뭉치 일부는 숨기거나 영구 삭제할 수 있다.

색상 제어를 위한 z축 사용도 가능하며 이는 이 작업의 다음 단계 중 하나이다. 소리의 확산을 위한 앰비소닉스 시스템ambisonics system으로의 스팟~ 모듈의 개량은 –연쇄 합성concatenative synthesis의 사용에 의해– 3차원 소리 조각의 본격적인 작업을 보여 준다.