NAVIGATING VARIATION: COMPOSING FOR AUDIO MOSAICING

Diemo Schwarz
Ircam-CNRS-STMS
Centre Pompidou, Paris
diemo.schwarz@ircam.fr

Benjamin Hackbarth
CRCA
University of California, San Diego
hackbarth@ucsd.edu

ABSTRACT

We present a method, applicable to corpus-based concatenative synthesis and specifically to audio mosaicing, that assists the composer in exploring the relationship between the parameterization of a concatenative algorithm and the resulting similarity between the output sound and the original target soundfile. Rather than focus solely on straightforward imitation, our work is predicated upon the notion that similarity can be manifest in a variety of perceptually meaningful ways and that both semblance and dissemblance have compositional utility. Our method consists of visualizing a collection of concatenated outputs, each of which is a unique solution to the problem of matching the same target soundfile with the same sound database but using a different combination of descriptor weights. We create a solution space where the location of each output is modeled by its similarity to the target as well as its similarity to each other solution. Visualization and navigation of this space is made possible through a multi-dimensional scaling algorithm, permitting 2D browsing, aural feedback, and the composition of paths through the solution space. This meta-control framework helps to give the composer a more comprehensive understanding of concatenative potential. By arranging concatenated outputs into different regions of similarity and dissimilarity, the solution space provides a rich and expansive terrain for compositional exploration and discovery.

1. INTRODUCTION

The heightened ability to design and manipulate sonic morphology is an alluring aspect of electronic music composition. Much music in the fixed media tradition relies on ordering sonic chunks intuitively, by hand, in order to create temporal structures which evoke perceptually singular morphologies. Recent advances in corpus-based concatenative synthesis (CBCS) have created alternate approaches which permit the temporality and spectral profile of morphological continuities to be generated with other types of control data. This scenario permits the composer's attention to shift away from piecemeal sound selection to higher-level control structures proffered by concatenative algorithms.

Corpus-based concatenative synthesis in general matches segments of sounds in a database of soundfiles, which are analyzed for audio descriptors (also called *features*), to a sequence of target audio descriptors [5]. We call a segment of sound with its descriptors a *unit*, and the database of units the *corpus*.

Audio mosaicing, a special case of CBCS, is premised on the idea of using a soundfile as the target, who's amplitude profile and analyzed descriptor values drive the selection of sound segments from the corpus. The use of a target sound creates a layer of abstraction which assists the composer in managing the selection and assemblage of sound segments. While this enables the composer to more readily prescribe overarching sonic characteristics, it also creates the need for tools to help explore and elucidate the relationship between the parameterization of a sound selection algorithm and the sonic outcome.

While the normative goal of audio mosaicing is to reconstitute the target soundfile such that the resulting concatenation is as similar to the target as possible, the very notion of similarity, especially from a creative standpoint, is multivalent by nature. We hold that similarity is not *a thing*, but rather a degree of likeness framed by one of many possible experiential perspectives. From this point of view, a target sound and database may be used to create a nearly infinite variety of concatenated outcomes which, taken as a whole, form numerous categories of similarity with respect to the target. Rather than assume that a computationally closest match is the user's goal, our work seeks to provide a more expansive understanding of the range of concatenative possibilities originating from a single target and single database.

In the following sections, we present a method, applicable to audio mosaicing and other CBCS systems, and actually any audio resynthesis method, that gives the user feedback on the affect of sound search criteria over the resulting deviation from the target soundfile. Thus, rather than thinking about unit selection as being predicated upon imitation, the target is better thought of as an abstract gesture-template consisting of feature contours and their time-dependent correlations, somewhat akin to the research and compositional work by Bob Sturm [8, 9].

Our method is based on AUDIOGUIDE[1], a program for differed-time concatenative synthesis written in Python. AUDIOGUIDE provides a customizable framework for experimenting with concatenative algorithms, discussed in detail in a journal article [2].

[1] http://crca.ucsd.edu/~ben/audioGuide

2. MODELING THE VARIATION SPACE

Unit selection in AUDIOGUIDE is made according to Euclidean distance calculations using continuously valued amplitude measurements, spectral features, and monophonic pitch estimates. Evaluating time varying features permits the preservation of the morphological profile of target and corpus units. Corpus units which overlap in time are selected according to an algorithm which approximates the composite of descriptors, thus enabling the selection of polyphonic mixtures which match the target's morphological contour.

When parameterizing a concatenation, the user may prescribe any number of features with different weights in order to influence each feature's saliency during unit selection. In addition to varying the weights of features, the user may specify different normalization and transformation strategies for each feature, permitting more expressive control over the sonic results.

Despite decreasing the fidelity of imitative similarity, these normalization and data transformation strategies are significant in that they permit the user to shape and sculpt the target's representation in order to alter the concatenated output. Thus, rather than considering the target soundfile a fixed object for imitation, these normalization and transformative tools allow the user to deploy the gestural profile of the target with an enhanced degree of creative freedom. This encourages the composer to treat the target soundfile as a correlated set of feature contours (a gesture-template) which can be mapped on to the database with differing degrees of likeness and semblance. As such, a large array of concatenated variations can be obtained with a single target and a single database.

Consequently, the resulting *parameter space* which affects a resynthesis has a rather high number of dimensions; we will nevertheless try to make it amenable for efficient musical exploitation. AUDIOGUIDE provides a choice among 32 continuously valued sound descriptors. For the purposes of this article, we selected 6 independently weighted features to parameterize the selection algorithm, namely *zero crossing rate*, *spectral flatness*, *spectral centroid*, *spectral kurtosis*, *mel-scale band amplitudes*, and *MFCC Envelope*. Together, these form a 6-element parameter vector p.

The *solution space* of the AUDIOGUIDE algorithm can be represented as a (not necessarily Euclidean) space populated by the set of possible solutions of a resynthesis of one target sound. In order to organise that space, we define two abstract distances d and v:

The *target distance* d_i expresses the dissimilarity of each solution i to the target, and the *inter-solution distance* or *variation* v_{ij} corresponds to the dissimilarity between two solutions i and j,

Together, these two distances will allow us to express the relative likenesses of all solutions to the target (d), and thus the compositional dimension of semblance and dissemblance. The degree of difference between solutions is captured by v and is significant here since even if two solutions have the same computational distance d to the target, they may be perceptually divergent in different ways.

The above two abstract distances can be realised in different ways, depending on the details of the resynthesis algorithm. In our case, we propose two approaches, contingent on the intermediate calculations of the AUDIOGUIDE algorithm. In a first attempt, we took as inter-solution similarity distance simply the Euclidean distance in the parameter space of the algorithm, i.e.

$$v_{ij} = ||p_i - p_j|| \tag{1}$$

The closeness to the target is given by the final sum of feature (mel-band amplitude) differences between the target spectrum t and the database unit i's spectrum u_i, which are calculated by AUDIOGUIDE while performing unit selection:

$$d_i = ||t - u_i|| \tag{2}$$

This approach to derive the distance is computationally cheap, since it completely relies on values calculated anyway by the selection algorithm. However, the perceptual validity of the parameter-based inter-solution distance v in equation (1) is questionable, since it compares the feature weights used during the selection of one solution, and not the timbral characteristics of the solutions themselves. Thus, this distance is situated on a conceptual level, possibly remote from the perceptual implications of the parameterisation of the selection.

Therefore, a second attempt is using the timbral differences between solutions, as expressed by the mel-spectra amplitude differences, which have to be calculated in an extra step of the algorithm (based on the already calculated spectra themselves):

$$v_{ij} = ||u_i - u_j|| \tag{3}$$

In our implementation, these differences are calculated, once all solutions have been generated, from the stored spectra of them, and are saved to a distance matrix V. The target distance d is the same timbral difference from equation (2) as used before.

3. NAVIGATING THE VARIATION SPACE

Now that we have defined the solution space, we populate it for a specific target and corpus, by running the AUDIOGUIDE algorithm on a large number of combinations of parameters. In our tests, we sample each of the 6 parameters in 3 steps, leaving us with $3^6 = 729$ resyntheses. Figures 1 and 2 show the distribution of the target distances d and variation distances v, respectively.

How can we make this meta-corpus accessible for composition? Our approach is to use interactive dimensionality reduction algorithms to embed the high-dimensional solution space into a 2D navigation space. We use here the multi-dimensional scaling (MDS) algorithm [7], based on a mass–spring–damper physical model.

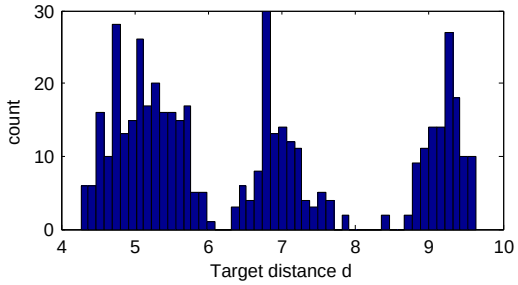

Figure 1. Histogram of target distances d.

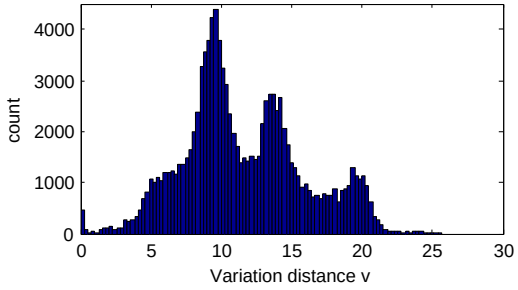

Figure 2. Histogram of variation distances v.

In this approach, we map the high-dimensional dissimilarities between two elements to the nominal lengths of simulated springs linking two masses. The model is then run iteratively in 2D, updating force, speed, friction, viscosity at each step, until it converges (or until the user stops the iterations).

Here, the target is mapped to a fixed mass, and the target distances d are mapped to a link to each of the n solutions. For initialisation, we place the solutions as movable masses in circles around the target, with a radius corresponding to the target distance and random angle. See figure 3 for an example of this placement. Then, we create links between each solution and all others (729^2 in our example), with lengths given by the inter-solution variation distance matrix v.

Running the mass–spring–damper model, the similar-sounding solutions will try to get close to each other, and push back dissimilar solutions, thus laying out the solution space. See figure 4 for a layout that shows the 4 clusters of variation also visible in figure 2. Figure 5 shows a surprisingly beautiful layout on a different target and corpus.

During the layout process, the user has one parameter to control the relative weight of the target distances versus the variation distances, that influences the stiffness of the inter-solution links. This way, he or she can favour if the precise target (dis)similarity should always be respected, or if the solutions could express their similarity more freely.

4. IMPLEMENTATION

The *variation explorer* within which the navigation of the variation space takes place is implemented in MAX/MSP based on CATART [6][2] and the FTM [3], MNM [1], and GABOR [4] extension libraries,[3] taking advantage of FTM&CO's advanced data structures such as matrices and dictionaries, and the real-time optimised operators that work on these.

Figure 3. Initial placement of the solutions around the target, with only the target links drawn.

Figure 4. Organisation of the solution space in 4 clusters.

It allows to load the result data and distance matrices from a run of AUDIOGUIDE, to interactively control the layout process by manipulating the parameters of the mass–spring–damper model while it is running, and to browse the corpus of solutions by moving over the points with the mouse, using CATART's *fence* or *click* mode.

These browsing movements can be recorded and the AUDIOGUIDE parameters corresponding to each closest solution written to a multi-break-point-function file. For smoother transitions, an interpolation between the param-

[2] http://imtr.ircam.fr
[3] http://ftm.ircam.fr

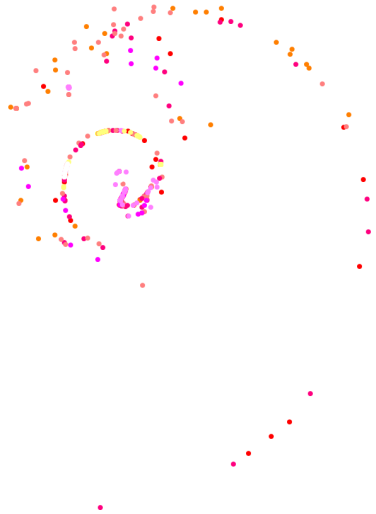

Figure 5. Organisation of the solution space in a spiral.

eters of the three nearest neigbor points could also be output.

This way, a path through the 6-dimensional parameter space of AUDIOGUIDE can be composed based on audition of example solutions, and navigating in terms of semblance to the example target. The accompanying webpage[4] gives an example of browsing through the solution space along a path from one cluster of very dissimilar solutions, passing by the closest solutions, and reparting into a very different realm, also shown in figure 6. In an actual resulting composition, the target would of course vary over time, and the path might be stretched out over the duration of a part of the piece.

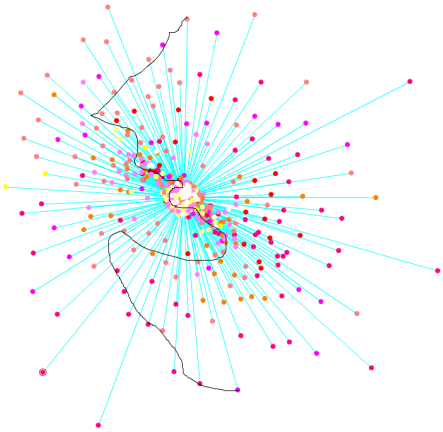

Figure 6. A composed path through the solution space.

[4] http://imtr.ircam.fr/imtr/Variation_Explorer

5. CONCLUSION

The work presented here is an encouraging first realization of a meta-control framework for better integrating audio mosaicing into a compositional workflow. By organizing a collection of concatenated outputs according to their sonic character as well as their relation to the target soundfile, composers may explore different categories of similarity as well as the relationship between search criteria and the degree of likeness to the target. While current research in corpus-based concatenative synthesis routinely emphasizes the importance of optimizing the closest match, our work seeks to help the composer explore the concept of similarity more deliberately and with a heightened degree of nuance.

6. REFERENCES

[1] F. Bevilacqua, R. Muller, and N. Schnell, "MnM: a Max/MSP mapping toolbox," in *New Interfaces for Musical Expression*, Vancouver, 2005, pp. 85–88.

[2] B. Hackbarth, N. Schnell, P. Esling, and D. Schwarz, "Composing Morphology: Concatenative Synthesis as an Intuitive Medium for Prescribing Sound in Time," *to appear in Contemporary Music Review*, 2012.

[3] N. Schnell, R. Borghesi, D. Schwarz, F. Bevilacqua, and R. Müller, "FTM—Complex Data Structures for Max," in *Proc. ICMC*, Barcelona, 2005.

[4] N. Schnell and D. Schwarz, "Gabor, Multi-Representation Real-Time Analysis/Synthesis," in *Digital Audio Effects (DAFx)*, Madrid, Spain, 2005.

[5] D. Schwarz, "Corpus-based concatenative synthesis," *IEEE Signal Processing Magazine*, vol. 24, no. 2, pp. 92–104, Mar. 2007, special Section: Signal Processing for Sound Synthesis.

[6] D. Schwarz, R. Cahen, and S. Britton, "Principles and applications of interactive corpus-based concatenative synthesis," in *Journées d'Informatique Musicale (JIM)*, GMEA, Albi, France, Mar. 2008.

[7] D. Schwarz and N. Schnell, "Sound search by content-based navigation in large databases," in *Sound and Music Computing (SMC)*, Porto, 2009.

[8] B. L. Sturm, "MATConcat: An Application for Exploring Concatenative Sound Synthesis Using MATLAB," in *Digital Audio Effects (DAFx)*, Naples, Italy, Oct. 2004.

[9] ——, "Adaptive concatenative sound synthesis and its application to micromontage composition," *Computer Music Journal*, vol. 30, no. 4, pp. 46–66, 2006.