The LoM Mapping Toolbox for Max/MSP/Jitter

Doug Van Nort , Marcelo M. Wanderley
Sound Processing and Control Laboratory, Input Devices and Music Interaction Laboratory
Music Technology Area - McGill University
doug@music.mcgill.ca, mwanderley@music.mcgill.ca

Abstract

This paper presents the Library of Maps[1] toolbox to aid in the mapping of control parameters to sound synthesis parameters via strategies that result from a geometric representation of control. A set of objects have been created for Max/MSP and Jitter that allow the user to map arbitrary high-dimensional data from control to sound parameter space, and to visualize this through the use of Jitter and OpenGL. The mapping implementations are discussed and related to existing work.

1 Introduction

In the context of live performance of computer-based music, the coupling between the energy that a performer puts into an instrument by way of musical gestures is divorced from the eventual sonic energy that the instrument produces. This de-coupling is both a blessing and a curse, as it allows one to re-think the notion of "instrument" while at the same time it provides a challenge in regaining the gesture-sound causality and the level of expressiveness that we've always enjoyed with acoustic instruments. The strategy that one employs to assign the various control signals that a performer produces to proper sound processing parameters is generally referred to as *mapping*. This issue has garnered increased attention in the past several years (Wanderley 2002) as an important aspect of the design of digital musical instruments in determining the overall *feel* of such instruments.

The notion of mapping in a given design context is tied to one's approach to the performance interface as a whole. If a systemic viewpoint is taken in which a performer is guiding high-level musical processes, then mapping relates to the composition of the music as much as to the composition of the performance system. If one's goal is to construct an interface that is closer to the notion of instrument in a traditional sense - wherein a performer's input energy is transduced into acoustic energy at the output - then mapping is a strong determinant of one's ability to articulate musical passages and maintain a high level of expressiveness. Quite naturally the design of an interface for musical performance is quite fluid and can fall between these two extremes.

As has been discussed elsewhere (Hunt and Wanderley 2002), mapping can be subdivided into several categories. In regards to the choice of input/output parameters to associate, a mapping can be constructed so that it is *one-to-one*,*one-to-many* or *many-to-one*, with a combination of these different approaches resulting in a complex *many-to-many* mapping. Several results have shown a many-to-many mapping to be more intuitive in a real, immersive musical situation, most notably in (Hunt and Kirk 2000). Further, a mapping can be *explicit* or *implicit*. The former refers to a situation in which the mapping is known and can be expressed analytically, while the latter is based on internal adaptation of a system, and can be seen as a "black box" model. This latter approach is promising in that it can allow one to adapt a performance system to their unique gestures, with examples given in (Lee and Wessel 1992) and (Fels and Hinton 1995). The explicit approach is beneficial in that having knowledge about the way that the mapping occurs allows one to tune, alter, and expand it over time and for different musical contexts. Several such strategies have been implemented for computer-based musical control, and yet there are not many readily accessible tools for constructing mappings between large parameter sets. We will discuss our contribution towards this end in the form of the LoM toolbox.

2 Related Work on Explicit Mappings

As soon as we have an expression that explicitly associates control and sound parameters, we can consider it from a functional point of view. This is a useful approach as we can have a basis of comparison between potential mapping strategies, allowing us to consider their essential qualities, see how they are inherently similar or different and to help us choose one that fits a given musical context. This sort of

[1] This title is inspired by the series of texts of the same name by Moira Roth.

exposition was given in (Van Nort, Wanderley, and Depalle 2004), in which several mapping strategies that can be represented geometrically were compared. This approach is useful as it allows one to think of the set of all possible control and sound parameters as a collection of Euclidean spaces, which further allows one to think spatially about how one organizes musical materials, to visualize this in two and three dimensions and to explore the structure - perceptual and physical - of various sonic spaces (Wessel 1979).

In practice, the techniques involved in constructing such associations between spaces are a combination of interpolation, extrapolation and regression methods. There are several works that deal with interpolation between and extrapolation of control/sound presets including piecewise linear techniques relative to a triangularization of parameter space (Goudeseune 2002) or a lattice constructed in this space (Bowler, Purvis, Bailey, and Manning 1990) (Choi, Bargar, and Goudeseune 1995), a multilinear interpolation between points spaced in a grid (Wanderley, Schnell, and Rovan 1998) and a regularized spline-based technique that generates variable smoothing between points (Van Nort, Wanderley, and Depalle 2004). Each of these techniques have been incorporated in the LoM toolbox and are discussed in more detail in the next section.

An existing toolbox (MnM) for mapping within Max/MSP was presented in (Bevilaqua, Muller, and Schnell 2005). It is based on multiple linear regression techniques: given a set of control/sound parameter presets, the "surface" which represents all traversable regions of parameter space is a hyperplane that is situated near the preset points relative to some best-fit criteria. Whereas the aforementioned techniques are made up of one or many surfaces that pass through or very near each preset, this regression approach creates a single linear control/sound surface that may not pass through any preset value. This drawback is traded off with the ability to draw on vast resources from matrix algebra and linear systems theory, and to deeply utilize the matrix processing available in packages such as Jitter and FTM (Schnell, Borghesi, Schwarz, Bevilacqua, and Muller 2005). Therefore, rather than recreate any of the work put into the MnM toolbox, this current library of mapping strategies seeks to add to the available options by providing linear, piecewise linear, multilinear (hyperbolic) and spline-based strategies for interpolation and extrapolation.

3 LoM Toolbox

The purpose of the LoM toolbox is to allow artists and researchers access to tools for experimenting with different complex mappings that would be difficult to build from scratch (or from within Max/MSP) and which can be combined to create many different control possibilities. This includes rapid experimentation of mapping in the dual sense of choosing what parameters to associate between control and sound space as well as the mapping of entire regions of these spaces through interpolation. The former definition of mapping (as a pointwise association) clearly affects the design of a musical instrument, and the latter - which defines the sort of musical trajectories that are possible in sound space - is also an important determinant of the overall feel and expressiveness of an instrument (Van Nort and Wanderley 2006). The toolbox is currently based around three central interpolation strategies, and includes several externals and abstractions that provide variations, combinations and visualizations of these.

3.1 lom.si

A technique for mapping between an N-dimensional control space and M-dimensional sound parameter space (with N $\leq$ M) is presented in (Goudeseune 2002). The mapping begins with a pointwise association between points in control and sound spaces, followed by a partitioning of the sound space by the creation of a simplicial complex[2] that induces a similar complex in the lower-dimensional control space. At any given moment the state of the overall control system lies within one simplex within this partition, and this position can be defined relative to the N+1 nodes of the simplex that contains it. Specifically, given simplex nodes $(x_0, ..., x_N)$ the distance from input vector x to each face of the simplex determines the barycentric coordinates $(\lambda_0, ..., \lambda_N)$ so that

$$x = \sum_{j=0}^{N} \lambda_j x_j.$$

These scalar values $\{\lambda_j\}$ are then used to weight a linear combination of vectors in sound parameter space that correspond to the nodes of the simplex in this higher-dimensional space. The geometric surface produced in sound space is a collection of connected, continuous and piecewise-linear simplices that have non-differentiable edges, which makes it suitable for certain musical contexts (Van Nort, Wanderley, and Depalle 2004). This technique was previously available as a C++ library, but has now been ported to Max/MSP/Jitter and the LoM toolbox under the LGPL, allowing for rapid prototyping within the Max environment. Three related externals are included in the LoM package:

•lom.si: stores input and output data points and computes simplicial interpolation based on an input list of control parameters, outputting a list of sound parameters.

•lom.siw: same as above but ouputs list of barycentric coordinates rather than sound parameters.

[2] The reader is directed to http://www.music.mcgill.ca/musictech/spcl/mapping for technical definitions and details not covered in this paper.

•lom.jit.si: a Jitter-based external that uses OpenGL to provide a two dimensional display of triangles in control space, to allow visualization and interactive control.

Figure 1 illustrates an example of using both lom.si and lom.jit.si. First, lom.si is intialized by providing arguments for input and ouput dimensions - in this case a two dimensional control space is mapped to a nine dimensional space of granular synthesis parameters. The control data is sent to lom.si, which interpolates this input vector and outputs a list of sound parameters. The control space is visualized using lom.jit.si, which renders the triangles creates by lom.si. Each node of the triangular complex refers to one stored parameter set, and the point lets the user know where they are in parameter space.

3.2 lom.multi

The second set of objects are based on a multilinear interpolation of data points, wherein the mapping is defined on a given set of control and sound parameter input/output pairs, with the control points spaced in a grid. This strategy maps an N-dimensional input point to an M-dimensional list of sound features by finding which section of the grid that the point lies in, and then computing a weighted sum of the 2^N nodes of the enclosing hypercube. Regarding its properties, this technique is differentiable across different cells of the grid, and the space is hyperbolic rather than piecewise linear as with lom.si (i.e. curved rather than flat). Again, this is important to consider as the geometry of the mapping strategy can strongly affect the response of the instrument, and the extent to which this is true in a given context can be easily tested with the use of the LoM toolbox. There are three different objects included that are related to multilinear interpolation:

•lom.multi: stores input and output data points and computes multilinear interpolation based on an input list of control parameters, outputting a list of sound parameters.

•lom.multiw: same as above but ouputs a list of 2^N weights rather than sound parameters.

•lom.jit.multi: Rather than accepting lists of data parameters, this object accepts grid points in the form of matrices.

3.3 lom.rst

The final set of objects calculate a surface based on the regularized spline with tension (rst) technique as described in (Mitasova and Mitas 1993). This is an approximation strategy that differs from standard spline-based techniques in that it possesses smoothing and tension parameters that allow one to "tune" the surface and avoid large overshoots of data points. These parameters can be changed in real time to alter the character of the mapping surface. Unlike the previous set

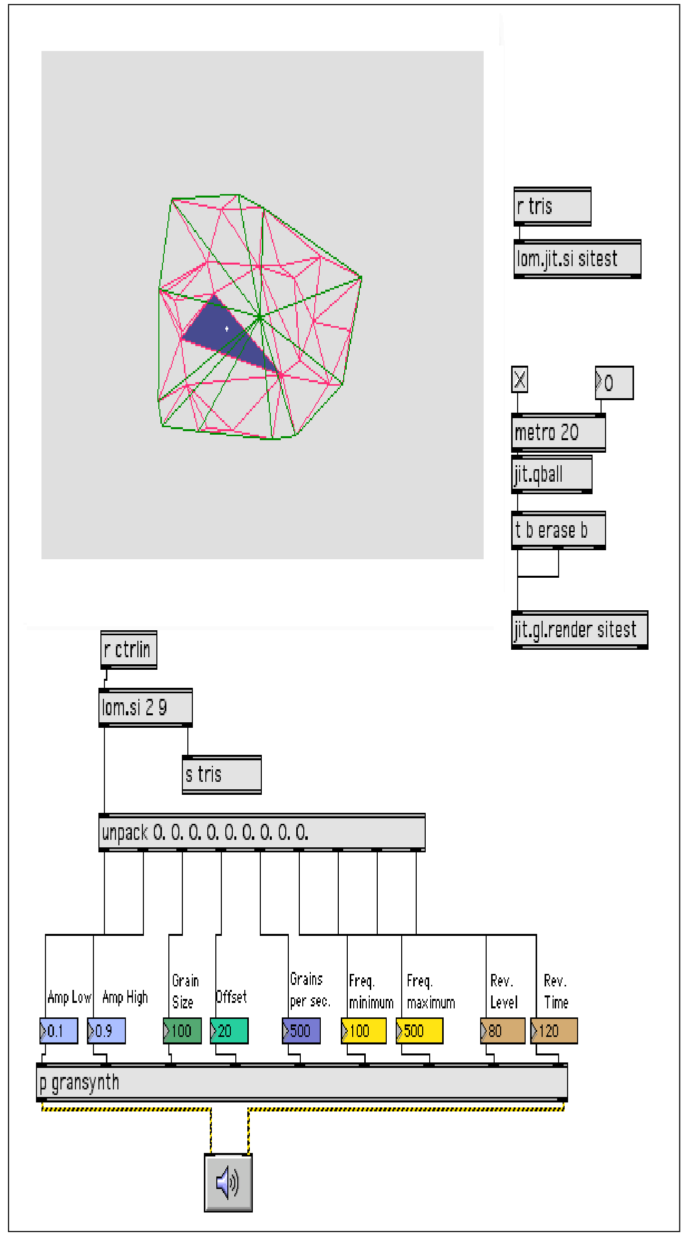

Figure 1: Example: control input from ctrlin is interpolated by lom.si and list of sound parameters is output - in this case to control granular synthesis parameters. Control space is rendered in OpenGL using lom.jit.si.

of objects, this is a global interpolation strategy in which all stored parameter points contribute to a single output point. Further, the output surface is constrained to be of size N+1 for a control space of size N. However, this mapping can be combined with the other objects or with the linear transformations of the aforementioned MnM toolbox in order to map to higher dimensional sound parameter spaces. Therefore, it is suggested that this set of objects be used to map to an intermediate space of *meta-parameters* that control higher-level attributes of sound processing. The two objects associated with the rst technique are:

•lom.rst: stores input and output data points and computes rst-based interpolation as a function of an input list of control parameters, outputting a list of sound parameters.

•lom.jit.rst: Rather than accepting lists of parameters, this object accepts stored parameter points in the form of matrices.

3.4 Combinations

In addition to the set of externals discussed, the toolbox contains abstractions that combine these mappings in order to illustrate the benefit of a multi-layered approach. For example, one can map from control space into an intermediate parameter space using lom.rst, providing a smooth transition through this space, and this trajectory can then be mapped into a high-dimensional sound space via the lom.si object. The latter mapping defines the sub-region of sound parameter space that can be explored, and the former determines both the part of control space that can be accessed as well as the nature of the trajectory through this space. This allows one to construct mappings strategies that consider the perceptual structure of control and sound parameter space separately.

4 Discussion and Future Work

The toolbox has been developed in the context of our research into the perception of different control strategies and the effect of mapping on the ”feel” of an instrument. Current developments include the incorporation of other mapping strategies such as natural neighbor (Bencina 2005) and sparse grid interpolation as well as further two and three dimensional visualizations.

This work has been necessitated by the fact that there are currently a lack of tools available to modularly construct mapping strategies, particularly ones that map entire continuous regions of high-dimensional parameter spaces. Thus, this toolbox should help composers and instrument designers alike to more easily explore sound parameter spaces and construct complex strategies for controlling sound.

References

Bencina, R. (2005). The Metasurface: Applying Natural Neighbor Interpolation to Two-to-Many Mappings. In *Proc. of 2005 Conference on New Interfaces for Musical Expression (NIME)*, pp. 101–104.

Bevilaqua, F., R. Muller, and N. Schnell (2005). MnM: a Max/MSP Mapping Toolbox. In *Proc. of 2005 Conference on New Interfaces for Musical Expression (NIME)*, pp. 85–88.

Bowler, I., A. Purvis, N. Bailey, and P. Manning (1990). On Mapping N Articulation onto M Synthesiser-Control Parameters. In *Proc. of 1990 International Computer Music Conference*, pp. 181–184.

Choi, I., R. Bargar, and C. Goudeseune (1995). A Manifold Interface for a High Dimensional Control Space. In *Proc. of 1995 International Computer Music Conference*, pp. 181–184.

Fels, S. and G. Hinton (1995). Glove Talk II: An Adaptive Gesture-to-Formant Interface. In *Proc. of the Conference on Human Factors in Computing Systems (CHI'95)*, pp. 456–463.

Goudeseune, C. (2002). Interpolated Mappings for Musical Instruments. *Organised Sound 7*(2), 85–96.

Hunt, A. and R. Kirk (2000). Mapping Strategies for Musical Performance. In *Trends in Gestural Control of Music*, pp. 231–258. IRCAM – Centre Pompidou.

Hunt, A. and M. Wanderley (2002). Mapping Performance Parameters to Synthesis Engines. *Organised Sound 7*(2), 97–108.

Lee, M. and D. Wessel (1992). Connectionist Models for Real-Time Control of Synthesis and Compositional Algorithms. In *Proc. of 1992 International Computer Music Conference*, pp. 277–280.

Mitasova, H. and L. Mitas (1993). Interpolation by Regularized Spline with Tension: I. Theory and Implementation. *Mathematical Geology 25*(6), 641–655.

Schnell, N., R. Borghesi, D. Schwarz, F. Bevilacqua, and R. Muller (2005). FTM - Complex Data Structures for Max. In *Proc. of 2005 International Computer Music Conference*.

Van Nort, D. and M. Wanderley (2006). Exploring the Effect of Mapping Trajectories on Musical Performance. In *Proc. Sound and Music Computing Conference (SMC 06)*, pp. 19–24.

Van Nort, D., M. Wanderley, and P. Depalle (2004). On the Choice of Mappings Based On Geometric Properties. In *Proc. of 2004 Conference on New Interfaces for Musical Expression (NIME)* ., pp. 87–91.

Wanderley, M. (Ed.) (2002). *Mapping Strategies in Real-Time Computer Music*. Organised Sound 7(2).

Wanderley, M., N. Schnell, and J. Rovan (1998). Escher - Modeling and Performing Composed Instruments in Real Time. In *Proc. IEEE International Conference on Systems, Man and Cybernetics*, pp. 1040–1044.

Wessel, D. (1979). Timbre Space as a Musical Control Structure. *Computer Music Journal 3*(2), 45–52.