Aaron Krajeski

MUMT 620

Summary of Week 8 Readings

**Hunt and Kirk (2000) Mapping Strategies for Musical Performance**

**Overview:** Examines mapping strategies for live performance, experiment, certain forms of “continuous multiparametric mapping” are best.

**Introduction:** Two modes of though *analytical and holistic*. *Performance mode* more suitable to real time control.

1. **Modes of thought for Musical Performance** – Choice based, computer is in charge of the dialog.
   1. Cognitive modes: analytical holistic. Analytical breaks down and decodes information, goal oriented. Holistic is more perceptible, intuitive, harder to define.
   2. Older instruments are holistic, computer interfaces are analytical. There should be no fixed ordering, human has control, instant response.
   3. Users “play around with” and discover, are not taught. Advanced users (of traditional instruments) do not rely on visual feedback.
2. **The Design of Computer Instruments –** “Instruments” are non real time editing machines.
3. **Multiparametric Interfaces and mapping strategies** – In order to control in real time, human must have access to all parameters. No set of options, instead continuous control. Should utilize operator’s energy.
   1. One to one vs one to many vs many to one mappings explained.
   2. Energy required in acoustic instruments, why not digital?
   3. Weight mappings, create a highly non-linear device.
   4. DMIs should have: continuous control of many parameters in real time, more than one conscious body control, parameters are couples together, and the user’s energy is used.
4. **Comparison of User Interfaces for Real-time Control** – Data from the university of york on the effectiveness of different interfaces.
   1. Choice – sliders controlled by mouse, physical sliders, multiparametric interface controlling: pitch, volume, timbre and panning.
   2. Mouse drags sliders, time multiplexed. One to many mapping.
   3. Physical sliders - Hand moves sliders, one to one. Space multiplexed.
   4. Multiparametric interface – no visual cues, sound only made when mouse is moved. Complex mappings.
5. **Implementation of the MIDAS system**  - Used to create audio algorithms. Each setting above is described. Matched to an ideal sound made by the computer (is an instrument really just about matching sounds?).
6. **Cross-sectional user interfaces** – Each user spends 15 minutes with each interface. Reproduce 24 sound examples. Stepwise, trajectories, complex changes (Group A, B, C)). Stored and graded for timing, parameter and trajectory accuracy. Human marking for 3456 tests (needs to get a nightlife?). Subjects interviewed and taped.
7. **Results of the Cross-sectional tests** – Ugly graphs. Score highest for mouse (all tests). Mouse wins groups A and B. Multi wins group C.
8. **Analysis of taped interviews** – Mouse interface is easy to start with, multiparametric volume is difficult. Multiparametric is subconscious, fun, more long term potential.
9. **Longitudinal User-Interface Test** – Longer period of time, subjects could train themselves in the Muli mode. 3 subjects over 10 sessions. Oh boy, 3d plots! Scores get better for more complex test and are eventually much higher than other two interfaces.
10. **Major Conclusions** – Real time control is enhanced by multiparametric control, mouse is better for little training and simple tasks. Multi is more engaging. SOME people prefer to think in separate parameters.
11. **Further applications** – Might be a good way to improve mixing and synthesis.

**Hunt – Towards a Model for Instrumental Mapping in Expert Musical Interaction**

1. **Introduction** – Devices are now separated, must be coupled with mapping.
   1. Mapping “refers to the liaison or correspondence between *control parameters* (derived from *performer actions*) and *sound synthesis parameters*.
   2. Specific feature of a composition, integral part of an instrument
   3. Generative mapping (Neural networks) Explicit mapping through design.
2. **Explicit mapping strategies** – One to one, one to many, etc. Euclidean analogies (point to a curve equals one to many).
3. **Review of previous works** –
   1. Favilla - Nonlinear for pitch
   2. Roan – Work in timber space? Beginner simple, skilled complex
   3. Mulder – Geometrical shapes reduce cognitive load?
   4. Description of study above (this is why we read in temporal order!)
4. **Towards a general model of mapping** – Based on vowel sounds?
   1. Wanderly – ESCHER mapping layer into *two independent* layers.
   2. Garnett – automatically generate perceptual parameters with timbre rover
   3. Multi levels may allow users and composers to fit their ideas better
5. **Conclusions** – Reviewed literature, topic deserves more study in its own right. Recommend complex mappings.

**Goudeseune (2002) – Interpolated mappings for musical instruments**

Compound mapping continuous function from Rd to Re for arbitrary integers *d* and *e* where 1<=d<e. *d* control values and *e* parameter values (sounds). “This ‘pointwise’ mapping can then be extended through a geometric technique called simplicial interpolation to produce a continuous mapping, which can be adjusted and refind by simply moving or adding new pairs of ‘points.’” Open source C++ is available.

1. Introduction – Static over dynamic, simple over complex. Must be smooth with several degrees of freedom (mappings). Pointwise map: “when a performer does *this*, the instrument should sound like *this*. *High dimensional interpolation* (HDI) – performer controls a large number of parameters with a smaller number.
2. Controls and Driving graphs – Control’s value is its instantaneous state (scalar, switch). Control drives dimension. Gain is the strength of coupling.
   1. Order measures directness: order of zero is a direct mapping, one is control’s value relates to rate of change of dimensions value. -1 is vice versa. (This seems incredibly arbitrary).
   2. Need not be linear, but needs to be based on a continuum
   3. Orders greater than 1 (or less than -1) are less important
   4. Derivative controls are bad at value, good at motion, integral the opposite
3. Inputs to a mapping – Primary controls, secondary controls (mutes, organ drawbars). Sounds like modifiers to me.
   1. Sliders and multisliders – multisliders are higher dimensional (pads? Joysticks). Diagonal lines are hard to draw on an Etch-a-Sketch. Physical limitations are hard barriers, attentional limitations may be overcome with practice.
   2. Multisliders and cross-coupling – only successful if the task is integral. *Cross coupled* parameters cannot be separated. (Violin elevation and pitch) “Simpler, les cross-coupled, mappings can help novices learn to play; technically, cross-coupled conrols are harder to learn because part-task training on individual controls transfers poorly to the whole task…cross-coupling can produce a better controller, once learned.
   3. Input devices – *pressure sensors, ribbon controllers, light pens* etc. Builds a space based on selection overhead and property sensed asa dimensions.
4. Interpolation
   1. Classical – “interpolation is ‘the performance of a numerical procedure that generates and estimate of functional dependence at a particular location, based upon knowledge of the functional dependence at some surrounding locations.’” Give it a goal, it finds intermediate solutions. (*f* has nice properties). Lots of great multi-dimensional calculus that I tried so hard to forget from my undergrad.
   2. High-dimensional interpolators – “’How can I control *e* parameters with only *d* scalar controls (*d<e*)?’ We can restate this question as ‘how can I make a *desirable* collection of gestures in **R**e using only *d* degrees of freedom?’ **R**e are sounds, d are gestures. Must specify desirable points, reduce size of space or else things get overwhelming and incomprehensible.
   3. Automatic generation of pre-image points – different methods for finding optimal methods are described, evaluated for features (avoiding local minima) and run times.
   4. Simplicial interpolation – All depends on “triangles” in **R**e. Builds multidimensional pyramids in mapping space, as opposed to simple one-to-one directional arrows.
   5. Other variations on interpolators are discussed in multi-dimensional space.
5. Conclusion – Reducing dimensions of control makes instruments friendlier. Avoids redundancy inherent in the exponential growth of increasing dimensionality! How well does a dimension-reducing controller lose information?

**This is so damn cool! Are there examples?**