

Sonic Zoom: A zoomable mapping of a musical parameter space using Hilbert curves

Released with the kind permission of Computer Music Journal. This preprint, unedited version was accepted for publication as part of the special issue on “Advances in the design of mapping for computer music”, Computer Music Journal Volume 38, Number 3 (Fall 2014).

Abstract

This article presents an interface for navigating a musical parameter space. The entire combinatorial space of a ten parameter synthesiser is laid out as a two-dimensional surface on a multi-touch screen. The surface can be scrolled and zoomed using touch screen swipe and pinch gestures, reminiscent of a maps application. The user can place markers on the surface to flag favourites, and explore different sized regions around these points. The mapping from the two dimensional surface to the high dimensional parameter space uses a space-filling curve. Hilbert curves constructed from Gray codes with long bit runs can be used to preserve locality as much as is possible, whilst still maintaining access to all possibilities. A user study was performed to compare with a more traditional one slider per parameter interface. Questionnaire responses indicate that different mapping strategies suit different stages of the creative process. The combination of the two interfaces was deemed more useful than either individually, reinforcing the notion that a combination of divergent and convergent processes is important for creative tasks.

«Start article»

Creativity is often characterised as a combination of idea creation and idea selection.

Whilst this is a very simplistic model, these two processes usually form the core of more sophisticated multi-stage process models (Sawyer 2012, p. 88 and 132). We shall refer to these contrasting processes as “divergent” and “convergent”, after Guilford (1967).

To express this idea in terms of search strategies in parameter space: idea creation would be a scattered exploration of a large neighbourhood creating as many candidate points as possible. Idea selection would be evaluating and choosing the best of those candidates. This strategy resembles optimisation meta-heuristics that are used to search large, complex solution spaces, such as Monte Carlo methods, particle swarms and genetic algorithms. Musicians, particularly those working in the electronic domain, say that a large proportion of the raw material they use comes from unplanned, emergent phenomena. Many claim that in the process of making music, accidents will occur and sounds will be discovered that were never intended, but prove invaluable. This would imply that some of the divergent process has been outsourced to the computer. Use of computers to induce “happy accidents” via stochastic processes has a long history (Xenakis (1971); Jones (1981)). One of the earliest exhibitions of computer art and music was entitled “Cybernetic Serendipity” (Reichardt 1968). Generating large amounts of widely scattered data quickly is certainly not difficult, but computers are rather bad at artistically evaluating those data: listening to candidate sounds becomes the bottleneck.

[Figure 1 and 2 about here side by side]

Figure 1 shows one view of the creative process. The artist does the creative thinking, and arrives at the technology with a clear idea in mind. They then proceed to realise this idea within the computer via the interface. This conforms to a traditional view of the recording studio as a tool for recording pre-composed, pre-rehearsed material. The

interface is therefore judged by how faithfully and painlessly the artist can realise their vision. Increasingly though, the computer studio is used to experiment and generate ideas (Duignan et al. 2010). Figure 2 illustrates an alternative view of the creative process, where the computer can be used to produce many divergent ideas, and indeed divergences may occur at any point within the interaction loop of brain, interface and software. In this picture the technology is judged not on ease of brain-to-computer input, but on how many good ideas emerge from the hybrid human-computer system.

Our main hypothesis is that standard computer music interfaces do not support multi-stage creativity as well as they could. The majority of interactions consist of adjusting individual parameters, leading to navigation “Etch A Sketch” style: one dimension at a time. This means that to move to a new point always involves travelling at least the Manhattan distance, and requires discrete analytical decisions to switch parameters. This method of navigation would seem to be best suited to the convergent stage of the design process. Divergent ideation, however, would seem to require the ability to travel in any direction, fast, with varying levels of predictability.

Prior Work

Much research has been carried out into reducing the difficulty of navigating timbre parameter space. Indeed, almost all music technology must address this problem in one way or another, as it is impossibly laborious to specify by hand an entire musical signal.

The simplest way to make a space quickly navigable is simply to save the coordinates of preferred points, these are referred to here as “presets”. Once a set of presets has been created, a low dimensional subspace can be created from them, the simplest being a line that interpolates between two preset points. The presets can then be “morphed” by navigating the subspace using a gestural controller. Given a D dimensional controller, and a P dimensional parameter space, $D + 1$ presets can be used to form a D

dimensional subspace within \mathbb{R}^P . In-depth treatments of the geometry of these mappings can be found in Goudeseune (2002) and Van Nort et al. (2004). Applications that have implemented these ideas include SYTER (Allouis and Bernier 1982), Bencina's metasurface (Bencina 2005), and the "nodes" object in Max/MSP.

Researchers have tried to create dimensions that correspond to high-level perceptual descriptors of the character of the sound, using techniques such as multidimensional scaling to create a "timbre space" (Grey 1977; Wessel 1979; Arfib et al. 2002). This approach certainly fits many users expectations, but the nature of timbre is extremely hard to quantify (Aucouturier and Pachet 2004). Useful dimensions may vary widely between musical styles and different users. Other examples of zoomable interfaces make use of this timbre space approach: SoundExplorer by Yee-King (2011) uses an MFCC based timbre similarity metric and multidimensional scaling to create a 2-D zoomable timbre map, and the ISEE (Intuitive Sound Editing Environment) by Vertegaal and Bonis (1994), where zooming in to a region of timbre space would take the user further down a hierarchy of instrument categories.

Most techniques throw away a large proportion of the space. Even with the best subspace finding algorithm, many interesting settings will become inaccessible.

In the mapping literature there tends to be a focus on the problem of controlling synthesis algorithms via physical controllers. But for this study, rather than facilitating expressive musical performances, we apply mapping concepts to a different stage of the creative process: namely the initial idea generation phase. Here predictability is deemed less important, and access to the full space of sonic possibilities more so. We are attempting to investigate, and speed up, the transitions between the exploration, evaluation and refinement stages of creativity.

Desirable Properties for Mappings

The following properties are desirable for controller mappings:

Low dimensionality

Control devices often have fewer parameters than synthesis engines. Given the brain's limited conscious multi-tasking abilities and working memory capacity, simple controllers are preferable.

Locality, or distance preservation

Having travelled a certain distance in control space, we want that to be reflected in the distance travelled in parameter space, and ideally perceptual distance too.

Revisitability

If we return to the same point, we wish it to sound the same. The location of preset points should be stable.

Continuity

If a point is adjacent to another point on the low-dimensional surface, they should be adjacent in the high-dimensional space.

Smoothness

Continuous higher derivatives are desirable to eliminate sudden changes in direction, this has relevance to the predictability of a control.

Linearity

When a gesture, such as a scroll, occurs it will have a certain effect on that sound, more extreme versions of this gesture should produce more of the same effect. This property is hard to achieve with any dimensionality reduction method, however smoothness

implies some linearity in the immediate neighbourhood.

It is difficult to perfectly satisfy all the above requirements with any mapping method. We can nonetheless experiment with mappings with different points of failure and ask if they can assist the creative process in different ways. Since little work has been done using mappings that can access the entire combinatorial space, we shall preserve this at the expense of linearity and smoothness and see how users fare.

The Hilbert Curve

A space-filling curve is a continuous parameterised function that maps a line segment to a continuous path in a unit N-cube. The curve can approach any point in the space arbitrarily closely as the iteration parameter is increased. The curve is constructed recursively, and is self-similar. These mappings have proved to be useful in all kinds of applications such as clustering, data indexing, parallel computing and even a computationally cheap solution to the travelling salesman problem (Bartholdi and Platzman 1982), due to their locality preservation properties. A 3-D colour space can be distributed onto a 2-D chart (Jaffer 2005), a good example of dimension reduction increasing usability. Hilbert curves in particular have good locality (Gotsman and Lindenbaum 1996). This property relates very well to the zooming idea: the further the user zooms in, the smaller the accessible sonic neighbourhood will become.

[Figure 3 here]

Figure 3 shows four iterations of a Hilbert curve in \mathbb{R}^2 . In its first iteration the curve simply visits each corner of the unit square. For the next iteration, sub-squares are formed at these 4 corners, and the corners of each of these are visited in a similar manner, but the bottom left and right sub-squares are rotated to ensure the continuity of the line.

The process is iterated until the plane is filled to some desired resolution. There are always upper bounds for distance in \mathbb{R}^N given a certain distance in \mathbb{R}^1 . Unfortunately distance preservation in the opposite direction can be worse, as can be seen if the x direction is traversed at the bottom of figure 3 (d). In the application this corresponds to moving a slider a small amount, but yet jumping to a distant point on the zoom surface.

For a higher dimensional curve, the basic unit is an N-dimensional hypercube. Each hyper-cube must have all of its corners visited exactly once, and the path must only run along the edges of the cube. Such paths are called “Hamiltonian paths”, and if the corners of the cube are labelled with binary coordinates the sequence forms a “Gray code” (Figure 4). Gray codes are binary numeral systems where only one digit changes at a time (Gray 1953). Figure 5 (a) shows the binary numerals for five digits, and Figure 5 (b) shows the reflected Gray code.

For the purposes of parameter mapping, there is still a problem with this Gray code: the rightmost bit flips sixteen times whereas the leftmost only flips twice. This would be rather frustrating for the user as one parameter will flip very fast and another will hardly ever change. We wish to achieve some kind of smoothness by distributing these transitions more evenly.

[Figure 4 about here]

[Figure 5 about here]

Gray codes with two particular properties can mitigate this issue. The first is a “balanced” Gray code (Bhat et al. 1996). Here the transitions are spread as evenly as possible between dimensions (Figure 5 (c)). The second property is “minimum run length” or MRL (Goddyn et al. 2003). If the MRL is maximised then the bit swaps never

occur within a certain distance of one another (Figure 5 (d)). This is also desirable: when one parameter goes high we would not want it to immediately flip low again.

Beyond three dimensions there are rapidly increasing numbers of alternative Gray codes and Hilbert curves. It is beyond the scope of this paper to investigate these in detail, as their construction seems to follow no simple method. For practical purposes we assume that the long MRL code shown in figure 5 (d) will be indistinguishable from better balanced codes.

The Algorithm

The 2-D surface is displayed as a grid. Each dimension uses a separate 5-D Hilbert curve, so that moving in the x direction will change the first five sliders, and moving in the y direction will change the other five. Each axis is split into 32 subdivisions representing the corners of a 5-cube. Crossing a grid line at the largest scale will flip a single slider up or down by half their range, the sequence of the flips being determined by the Gray code. These are further split into 32 subdivisions that flip parameters by a quarter of their range and so on, down to the smallest subdivisions, which change parameters by a single unit.

The algorithm to converting an x or y coordinate to P N-bit parameters, giving $2P$ parameters for each point on the surface, is as follows:

Each coordinate is first expressed as a base- 2^P N-tuple

$$a = (a_{N-1}, \dots, a_1 \dots a_0), \quad (1)$$

where the individual base 2^P digits are calculated like so

$$a_n = \lfloor \frac{x}{2^{Pn}} \rfloor \pmod{2^P}. \quad (2)$$

Each of the a_n are then converted to P-digit binary numbers $(b_{n,P-1}, \dots, b_{n,1}, b_{n,0})$ using the Gray code $G()$, via a look-up table such as the code in figure 5 (d).

$$b_{n,p} = G(a_n) \quad (3)$$

Then the parameter control values c_p can be then built up by treating the N different scales as standard binary digits.

$$c_p = \sum_{n=0}^{N-1} b_{n,p} 2^n \quad (4)$$

The number of points along one coordinate necessary for full resolution is $(2^P)^N$. In our implementation we require ten 7-bit MIDI control parameters, so $P = 5$ and $0 \leq x < 32^7$.

So far we have yet to ensure correct rotation of the sub-cubes such that a true Hilbert curve can be constructed from a long bit run Gray code, as the standard method assumes certain symmetry properties of the original reflected Gray code (Hamilton and Rau-Chaplin 2008). Therefore the jump at transitions between sub-cubes is sometimes audible, which will be addressed in future work.

Implementation

Whilst we envisaged this mapping being used for timbre exploration, it was initially felt that a single tone would become fatiguing for users. So five of the parameters were used for a simple melodic pattern generator. This creates a 16-step sequence based on five sine waves of integer frequencies; this ‘‘Frequencer’’ is detailed in Tubb (2012). The other five

sliders control a simple FM/subtractive synth. Figure 6 shows the interface.

For the “standard” interface, ten sliders (or faders) were used. These appeared as an overlay on the left of the screen. When both the sliders and the zoomer are on screen together, movements in one space are immediately reflected in the other. The “listen point” location is represented as a cross-hair in the middle of the screen. The absolute locations of touch points have no bearing on the sound. A single finger drags the surface: changing the coordinates of the listen point, and hence the positions of the sliders. A two finger pinch-out gesture will zoom into an area around the listen point, whilst keeping the listen point stationary. As the user zooms, smaller sub-divisions of the grid become visible. The sub-divisions are coloured according to their Hilbert curve iteration level.

The “lock sequencer” and “lock synth” buttons constrain the surface to move in only the x or y directions. In this case only the five sliders exclusively relating to the sequencer or synth will change. Once a preset is saved, it appears as a coloured dot on the surface. If the listen point moves near to a preset, it will snap to the preset coordinates. Without snapping it is impossible to line the preset up precisely.

[Figure 6 here]

Interface Evaluation Experiment

The experiment consisted of three five minute intervals, where the sliders, the Zoomer, and a combination of both were presented. The order was randomised for each session. All user interactions were recorded and sent over a WIFI internet connection to a logging server. As many concurrent users were anticipated for the public release, the event data were thinned to 15 Hz before being sent, but care was taken to preserve the start and end points of each scroll, zoom or slider change. Whilst assigning specific

search tasks may make for a more controlled experiment, this behaviour is only relevant for “convergent” interaction modes, therefore users were instructed to explore the space however they wished, but to intentionally search for sounds they liked and save them as presets. At the end of the session, users were presented with a short questionnaire.

The Lickert-style questions on the questionnaire fell into two categories. Statements that participants rated on a five point agreement scale are shown in Table 1. Questions regarding which interface was best for a certain task, with a five point scale for interface preference, are shown in Table 2.

For the pilot study, twenty PhD students participated. Most were familiar with computer music, but unaware of the focus of the experiment.

Results

[Figure 7 about here]

In figure 7 the strongest agreement was for statements AD 4 and AD 7. This indicates that spatial visualisation of preset points on the surface is a good feature, and that people definitely felt that the combination of the two navigation methods was more powerful than either individually. The improvement over a randomiser (AD 6) was also a good result for the Zoomer. Most people got an impression of the distance preservation property (AD 3), though this was not conclusive, and two users disagreed strongly.

SZ 10 and SZ 11 were questions specifically relating to convergent processes.

Unsurprisingly people felt much more in control using the sliders, and would prefer them to fine tune a sound. The statements about “finding interesting sounds quickly” (question SZ 9 and SZ 13) and “generating new ideas” relate to divergent thinking, were

AD 1	I am familiar with music software and sound synthesis.
AD 2	The correspondence between the sliders and the grid was understandable.
AD 3	Scrolling greater distances on the grid made larger differences in the sound.
AD 4	The ability to see other presets laid out on the grid was useful.
AD 5	The sounds were too limited/poor quality to be able to judge the interface.
AD 6	The Zoomer was an improvement on just using a randomiser.
AD 7	The combination of Zoomer and Sliders was better than either individually.

Table 1. Questions requiring a 5 point agree/disagree answer.

SZ 8	The best interface to get a feel for the possibilities of the synth was...
SZ 9	The best interface for discovering interesting sounds quickly was...
SZ 10	The best interface for fine tuning a sound was...
SZ 11	The interface that I felt more in control using was...
SZ 12	The interface that felt more creative was...
SZ 13	The interface better for generating new ideas was...
SZ 14	The interface better for performing live would be...
SZ 15	Overall, the interface I preferred using was...

Table 2. Questions requiring a 5 point sliders vs. Zoomer answer.

slightly in favour of the Zoomer. The question about which felt more creative (SZ 12) narrowly came down in favour of the Zoomer, but with very large variance. Questions SZ8 and SZ15 generated no clear results.

Surprising neutral results include the fact that performing live (question SZ 14) came out as evenly balanced between the two interfaces. One would expect the more predictable sliders to be preferred. The understandability of the mapping (AD 2) was neutral, but expected to be strongly disagreed with, due to the complex nature of the Hilbert curve.

Navigational Data Analysis

Data analysis is in an initial phase, the release to the public will enable a much larger dataset to be analysed. However some trends have already emerged. Figure 8 shows a plot of the total dwell time at various zoom scales for all users. The scale unit corresponds to the 7 iteration scales of the Hilbert curve. There is an obvious peak at the

Interface	Sliders	Both	Zoomer	Stage 1	Stage 2	Stage 3
Total saves	64	61	47	66	56	50

Table 3. Total number of presets saved for the three interface views, and by timed stage.

larger scales: the smaller the scale the less time people spent exploring it. Levels 1, 2 and 3 are largely avoided: presumably because the sound does not change rapidly enough.

[Figure 8 about here]

An interesting question is whether the clear peak at large scales corresponds to a real optimum scale for divergent searches. If this is so, the results seem surprising in that the preferred scale is smaller (about level 4) for the combination of the two interfaces than for the Zoomer only (level 6).

[Figure 9 about here]

Figure 9 shows the distribution of distances travelled between listen points. Any point that was hovered over for more than 0.2 seconds is assumed to have been assessed, at least for timbre, and is categorised as a “evaluation point”. A histogram was constructed for the distances between evaluation points. As you would expect, this distance is related to the scale factor dwell time, and the two curves show a more or less similar distribution. Again, the sharp drop off below a distance of 32^4 implies that fine resolutions can be discarded. However this would abandon the one-to-one mapping, and probably render presets obtained with the sliders inaccessible to the Zoomer.

The other quantity of interest was the rate of preset discovery. The save rate could be taken as a measure of how effective an interface is for discovery. The total number of presets saved in each different interface mode are shown in Table 3.

The zoomer seems to fare worst by this measure. Despite the positive results from the questionnaire, the actual sound discovery rate seems highest with the sliders. It could be that people were too busy exploring the interface itself to fully concentrate on sound design: the Zoomer is far less familiar than the sliders. However the rate of preset saving actually went down slightly over the duration of the experiment, suggesting that unfamiliarity does not have a negative effect on the save rate. The other possibility is that given more different options, people's expectations rise and they become more selective about what they save.

An average of 15 seconds elapsed between users switching interfaces when both were on screen. Whilst not a rapid process, it indicates that both navigational methods were alternated many times before new sounds were located.

User Comments and Suggestions

Some people felt that it was very hard to find the ideal scale at which to scroll around. To feel one is "making progress" a large scale is needed, but the sounds change too suddenly and unpredictably. On the other hand at smaller scales you can be lost in a sea of minor variations. This issue can be expected to get considerably worse as more parameters are added. People also felt it too long to zoom between the highest and lowest scales.

Many users suggested smoother transitions, both in a general sense, and between specific presets. Future implementations will make it possible to interpolate between various complexity levels: detail will only reveal itself as one zooms in.

There were frequent requests for an easier way to return to where you had visited before, but had not saved. It is too easy to accidentally overshoot something that caught

your ear. Two users came up with the intriguing suggestion of a visual, scrub-able undo trail for these situations. Session histories can promote new ways of thinking by providing an overview of one's own creative process (Shneiderman 2000).

A featureless grid was not enough to make use of visual spatial memory. Colour coded lines were not sufficient for people to instantly grasp what scale they were at, though this can be learnt with use. One user suggested using audio feature extraction techniques to create a texture that would convey the nature of the sounds underneath. Alternatively, patterns or shapes could be associated with presets and then morphed, as in Van Wijk and van Overveld (2004). Real geographical data could be displayed from a database such as Openstreetmap (Haklay and Weber 2008). Most users expressed some confusion over the sequencer parameters, and another suggested that a set list of MIDI sequences of various genres should be provided, and the surface left just for timbre exploration. However one user was most interested in the sequencer, and spent some time constructing a selection of points that enabled sequences to be navigated to generate a chord sequence.

Conclusion

This study indicates that different ways of navigating parameter space are suited to different stages of the creative process. For divergent exploration the Zoomer was preferred, for convergent fine tuning the sliders were preferred. Users responded very positively to the assertion that the combination of the the two interfaces was better than either individually. This could mean that being able to alternate navigation styles is valuable, however preset save rates do not support this assertion.

Whilst no single mapping can be perfect, the Hilbert curve is far from an ideal instrumental mapping due to its lack of smoothness and linearity. It can however be

very useful in cases where a representation of the complete parameter space is desired, and was shown to be preferred to a randomiser.

The fact that people tended to use very high zoom levels indicates that an overview of the entire space is useful feature, but access to every level of detail is not. Given that users prefer to fine-tune with the sliders, we can eliminate some of parameter space with no noticeable ill effect. Further testing should be done to establish what users feel is the ideal scale, but certainly the lowest two levels seem disposable: leading to at least a millionfold reduction in the area of the surface.

Evaluating creativity is a hard task, and this study has not addressed many issues, for instance the quality of the saved points. Further work could utilise a “consensual assessment technique” (Amabile 1996), by enabling other user’s presets to appear on the grid and be rated. Once the application is made available to the general public, analysis may reveal clearer statistical relationships: such as how experts perform compared to non-experts, and if distinctive behaviour occurs immediately prior to saving a preset.

Many electronic musicians rely heavily on technology as an extension of their artistic thought process. If creativity does indeed involve rapid alternation between idea creation and idea selection, systems should be designed with this fact in mind. Divergent or convergent features on their own may be less effective than a well integrated combination of the two.

A video of Sonic Zoom in use can be found at <http://youtu.be/l1AAinjrTbI>

References

- Allouis, J., and J. Bernier. 1982. "The SYTER project: Sound processor design and software overview." In *Proceedings of the 1982 International Computer Music Conference*. pp. 232–240.
- Amabile, T. M. 1996. *Creativity and innovation in organizations*. Harvard Business School.
- Arfib, D., J. M. Couturier, L. Kessous, and V. Verfaillie. 2002. "Strategies of mapping between gesture data and synthesis model parameters using perceptual spaces." *Organised Sound* 71(2):127–144.
- Aucouturier, J.-J., and F. Pachet. 2004. "Improving timbre similarity: How high is the sky?" *Journal of negative results in speech and audio sciences* 1(1):1–13.
- Bartholdi, J., and L. Platzman. 1982. "An $O(N \log N)$ planar travelling salesman heuristic based on spacefilling curves." *Operations Research Letters* 1(4):121 – 125.
- Bencina, R. 2005. "The Metasurface - Applying Natural Neighbour Interpolation to Two-to-Many Mapping." In *Proceedings of the 2005 International Conference on New Interfaces for Musical Expression (NIME05), Vancouver, BC, Canada*. pp. 101–104.
- Bhat, G. S., C. D. Savage, et al. 1996. "Balanced Gray codes." *The Electronic Journal of Combinatorics* 3(1):25.
- Duignan, M., J. Noble, and R. Biddle. 2010. "Abstraction and activity in computer-mediated music production." *Computer Music Journal* 34(4):22–33.
- Goddyn, L., P. Gvozdjak, et al. 2003. "Binary Gray codes with long bit runs." *The Electronic Journal of Combinatorics* 10(27):1.
- Gotsman, C., and M. Lindenbaum. 1996. "On the metric properties of discrete space-filling curves." *IEEE Transactions on Image Processing* 5(5):794–797.

- Goudeseune, C. 2002. "Interpolated mappings for musical instruments." *Organised Sound* 7(2):85–96.
- Gray, F. 1953. "Pulse code communication." US Patent 2,632,058.
- Grey, J. M. 1977. "Multidimensional perceptual scaling of musical timbres." *J. Acoust. Soc. Am* 61(5):1270–1277.
- Guilford, J. P. 1967. *The nature of human intelligence*. McGraw-Hill (New York).
- Haklay, M., and P. Weber. 2008. "Openstreetmap: User-generated street maps." *Pervasive Computing, IEEE* 7(4):12–18.
- Hamilton, C. H., and A. Rau-Chaplin. 2008. "Compact Hilbert indices: Space-filling curves for domains with unequal side lengths." *Information Processing Letters* 105(5):155–163.
- Jaffer, A. 2005. "Color-Space Dimension Reduction." URL <http://people.csail.mit.edu/jaffer/Color/CSDR>.
- Jones, K. 1981. "Compositional applications of stochastic processes." *Computer Music Journal* 5(2):45–61.
- Reichardt, J. 1968. *Cybernetic serendipity: the computer and the arts: a Studio International special issue*. Studio international.
- Sawyer, R. R. 2012. *Explaining creativity: The science of human innovation*. New York: Oxford University Press.
- Shneiderman, B. 2000. "Creating creativity: user interfaces for supporting innovation." *ACM Transactions on Computer-Human Interaction (TOCHI)* 7(1):114–138.

Tubb, R. 2012. "Frequency domain patterns for live improvisatory performance." In *Proceedings of the International Conference on Live Interfaces (ICLI)*. URL <http://lipam.lurk.org>.

Van Nort, D., M. M. Wanderley, and P. Depalle. 2004. "On the choice of mappings based on geometric properties." In *Proceedings of the 2004 conference on New Interfaces for Musical Expression (NIME)*. Shizuoka University of Art and Culture, pp. 87–91.

Van Wijk, J. J., and C. van Overveld. 2004. "Preset based interaction with high dimensional parameter spaces." In G.-P. B. Frits H. Post, Gregory Nielson, (editor) *Data Visualisation: The State of the Art*. Kluwer Academic Publishers, pp. 391–406.

Vertegaal, R., and E. Bonis. 1994. "ISEE: An Intuitive Sound Editing Environment." *Computer Music Journal* 18(2):21–29.

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

Xenakis, I. 1971. *Formalized Music: thought and mathematics in composition*. Indiana University Press.

Yee-King, M. J. 2011. "Automatic sound synthesizer programming: techniques and applications." Ph.D. thesis, University of Sussex.

List of Figures

1	"Studio" view of the creative process.	21
2	Creative process involving feedback loop with technology.	21

3	Four iterations of Hilbert curve construction in two dimensions. Arrow <i>a</i> demonstrates a locality violation: a small High-D movement results in a large Low-D distance.	22
4	The Hamiltonian path visiting the 8 corners of a cube. The sequence of coordinates is a Gray code (right).	23
5	5 digit binary codes. (a): Standard binary numbers. (b): Gray code, a single bit flips from row to row. (c): “Balanced” Gray code: all the digits transition either 6 or 8 times. (d): “High MRL” Gray code: The minimum distance between flips is 4 steps.	23
6	Sonic Zoom app screenshot showing both interfaces. A preset is being hovered over in the centre of the screen. For the timed session, the sliders or the zoomer are hidden.	24
7	Questionnaire responses to Lickert items (left) and interface preferences (right). The median is marked by a dotted circle, thick bars indicate the first quartile, thin bars the third quartile.	25
8	Plot of total time dwelt on different zoom scales.	26
9	Frequency of journeys between evaluation points by 2-D distance.	27

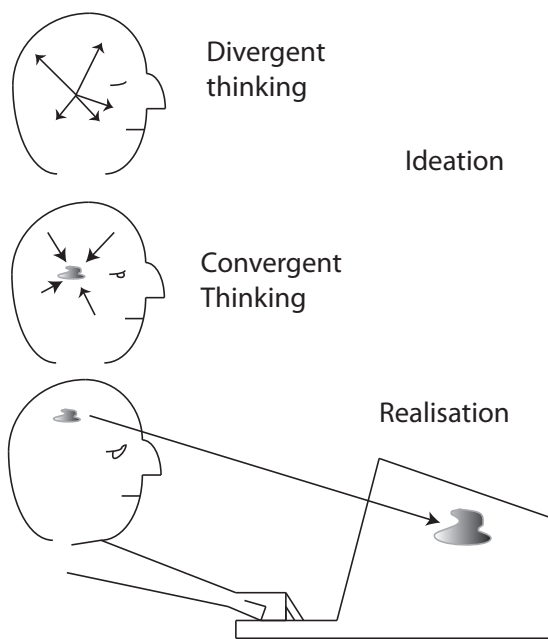


Figure 1. "Studio" view of the creative process.

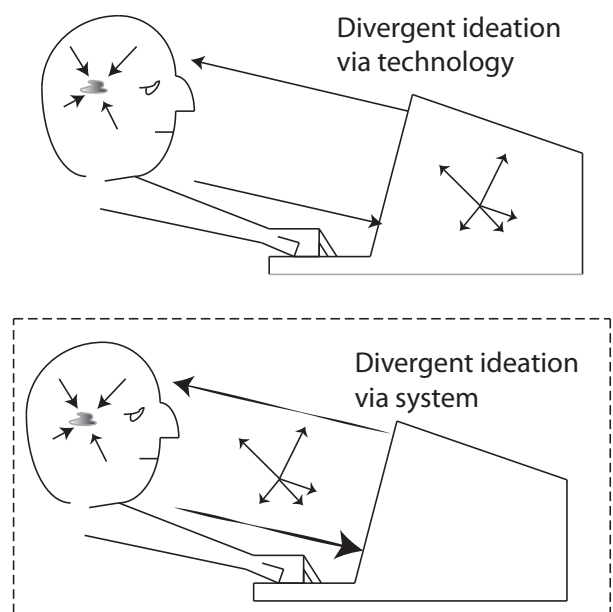


Figure 2. Creative process involving feedback loop with technology.

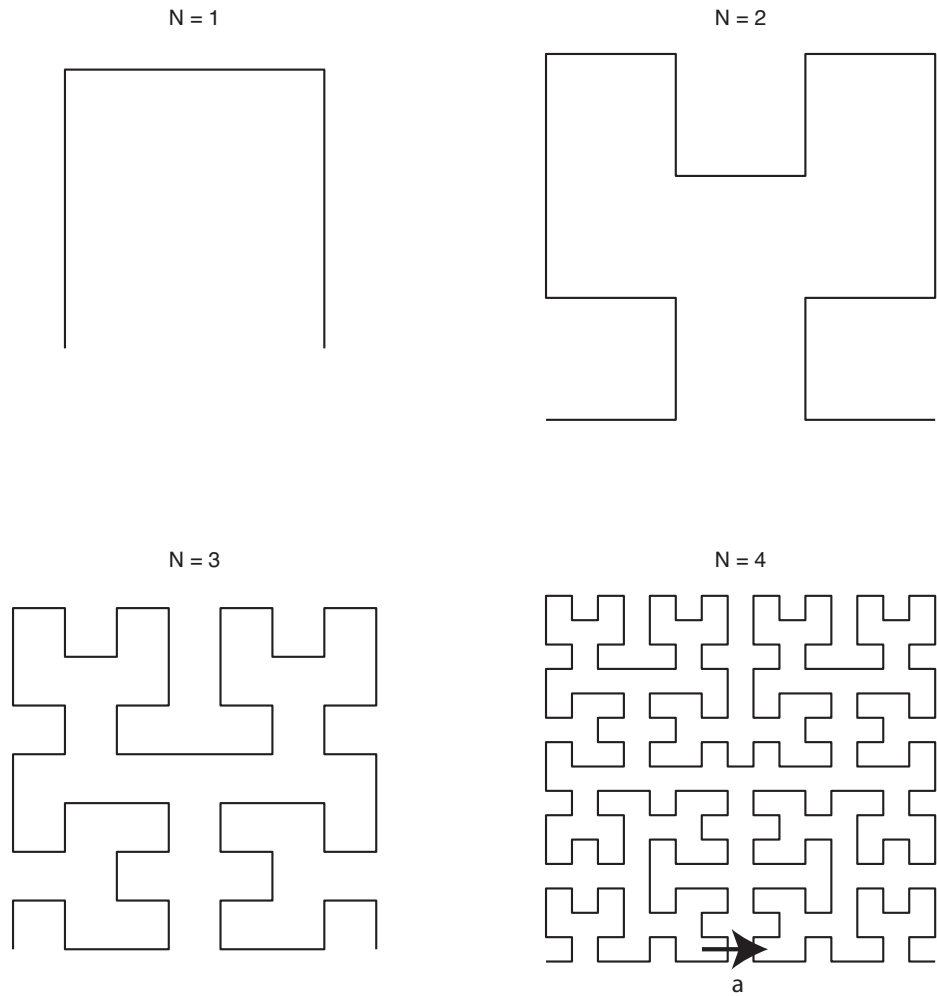


Figure 3. Four iterations of Hilbert curve construction in two dimensions. Arrow *a* demonstrates a locality violation: a small High-D movement results in a large Low-D distance.

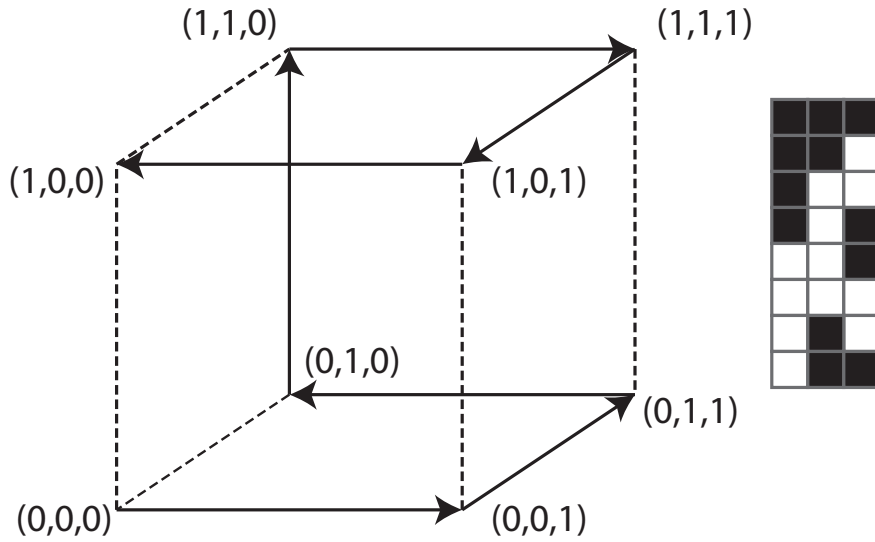


Figure 4. The Hamiltonian path visiting the 8 corners of a cube. The sequence of coordinates is a Gray code (right).

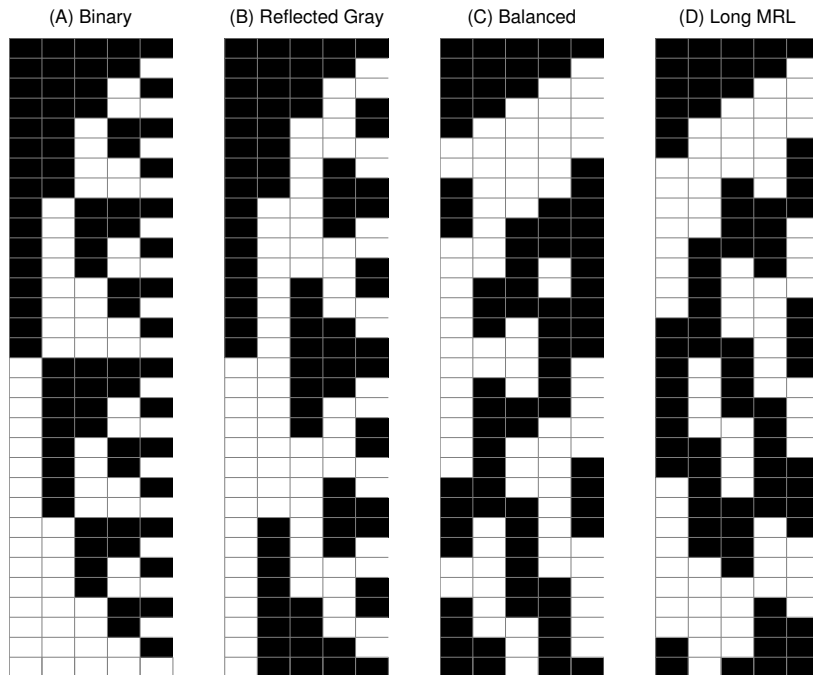


Figure 5. 5 digit binary codes. (a): Standard binary numbers. (b): Gray code, a single bit flips from row to row. (c): “Balanced” Gray code: all the digits transition either 6 or 8 times. (d): “High MRL” Gray code: The minimum distance between flips is 4 steps.



Figure 6. Sonic Zoom app screenshot showing both interfaces. A preset is being hovered over in the centre of the screen. For the timed session, the sliders or the zoomer are hidden.

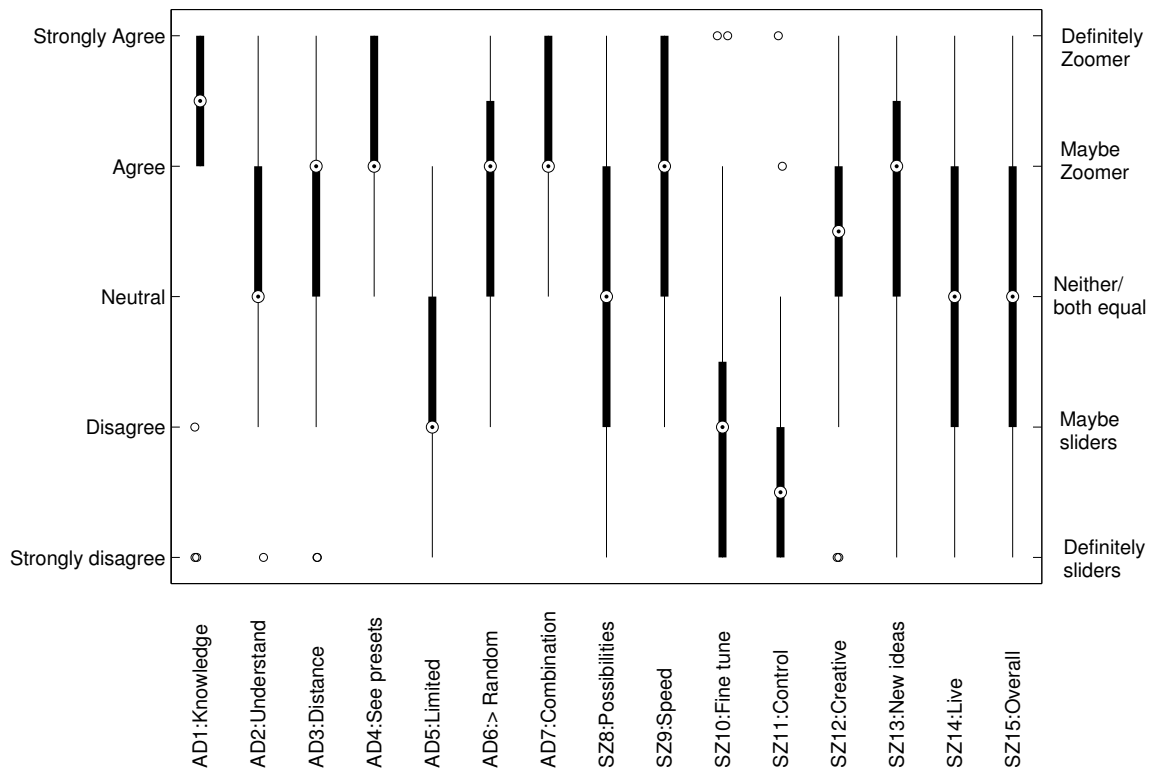


Figure 7. Questionnaire responses to Lickert items (left) and interface preferences (right). The median is marked by a dotted circle, thick bars indicate the first quartile, thin bars the third quartile.

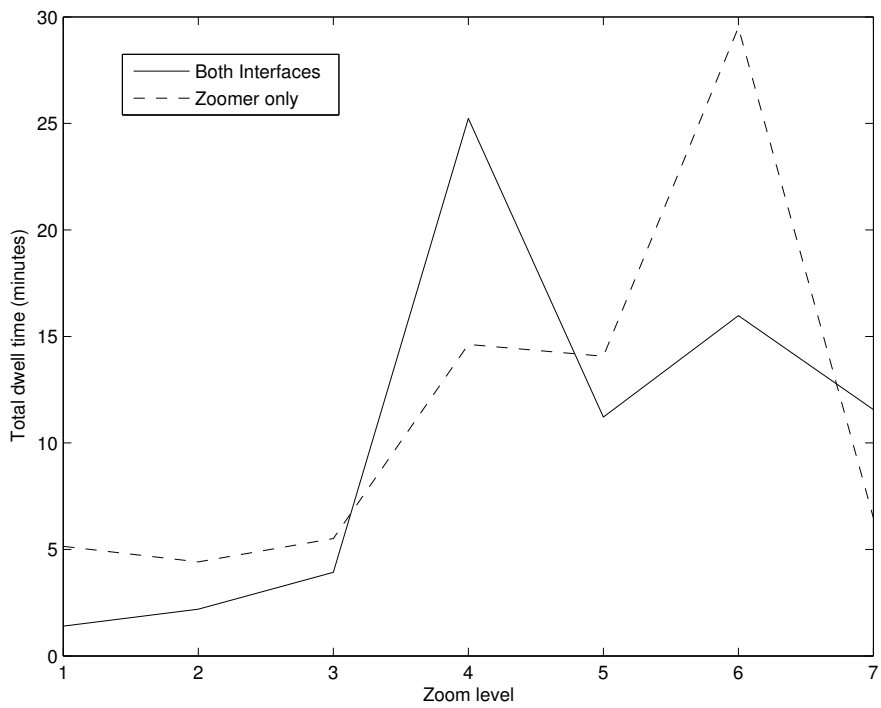


Figure 8. Plot of total time dwelt on different zoom scales.

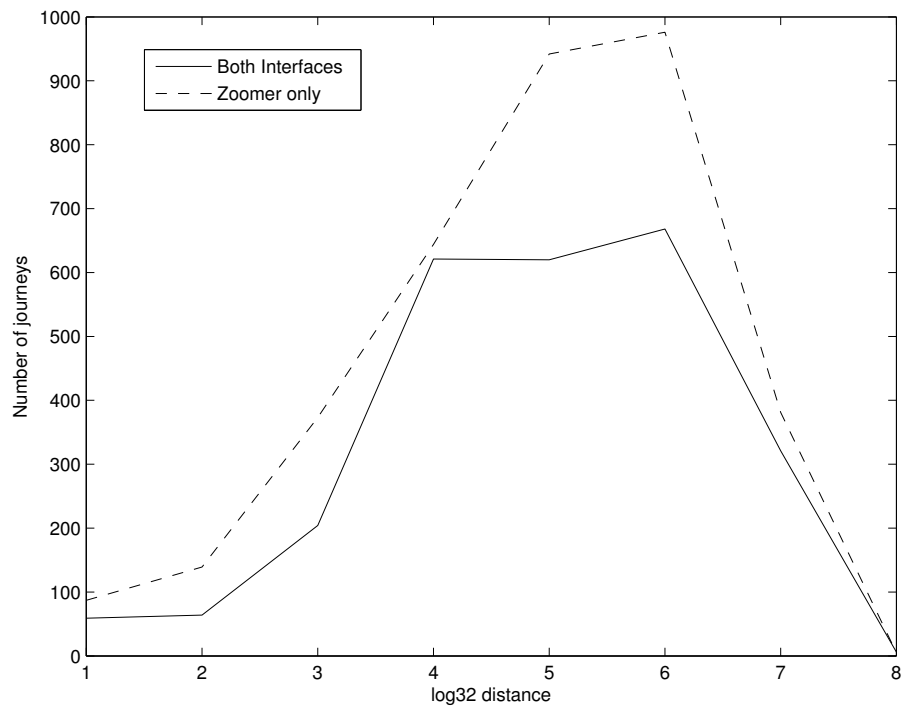


Figure 9. Frequency of journeys between evaluation points by 2-D distance.