Using Chaos to Generate Choreographic Variations

Elizabeth Bradley and Joshua Stuart[1]

Department of Computer Science
University of Colorado
Boulder, Colorado, USA 80309-0430
[lizb,stuartj]@cs.colorado.edu *Fourth Experimental Chaos Conference*, Boca Raton FL; 1997

Abstract

We describe a method for introducing variations into predefined dance sequences using a chaotic mapping. A symbol sequence representing a dance piece is mapped onto a chaotic trajectory, establishing a symbolic dynamics that links the dance sequence and the attractor structure. A choreographic variation on the original piece is created by generating a trajectory with slightly different initial conditions, inverting the mapping, and using special corpus-based interpolation schemes to smooth any abrupt transitions. Sensitive dependence guarantees that the variation is different from the original; the attractor structure and the symbolic dynamics guarantee that the two resemble one another in both esthetic and mathematical senses.

1 Introduction

This paper describes a chaotic mapping technique that creates choreographic variations. A sequence of special symbols representing the body positions in a dance piece is mapped onto a chaotic attractor, establishing a symbolic dynamics that links the dance progression and the attractor geometry. To create a variation, one generates a new trajectory and then inverts the mapping. Sensitive dependence guarantees that the variation is different from the original; the attractor structure and the symbolic dynamics guarantee that the two resemble one another in both the esthetic and the mathematical senses.

To establish a mapping between an N-move dance sequence and a chaotic attractor, we first integrate a chaotic ODE system numerically from some initial condition x_0. We then use a Voronoi diagram to partition the state-space region occupied by the ω-limit set $\phi_\omega(x_0)$ of this trajectory into N cells. Finally, we label the *itinerary* of cells traced out by $\phi_\omega(x_0)$ with special symbols that represent the body positions in the dance sequence. To create a variation, we then generate a new trajectory $\phi(x')$ from an initial condition x' near the attractor and invert the mapping: each time $\phi(x')$ enters a new cell, the corresponding dance move is sent to an animation tool.

[1] Supported by NSF NYI #CCR-9357740, ONR #N00014-96-1-0720, and a Packard Fellowship in Science and Engineering from the David and Lucile Packard Foundation.

This work was catalyzed by a paper by Diana Dabby[1] that proposes a similar mapping that creates *musical* variations. The core idea here is the same, but many of the issues and tactics — and much of the mathematics — are very different. The symbol set is one obvious distinction. There is a simple, well-established notational scheme for music, but body positions are much harder to represent. We use techniques from rigid-body mechanics to solve this problem. The mathematics of the mapping is also very different; Dabby uses a simple metric on a one-dimensional trajectory projection to define cells, whereas we work with a full, formal symbolic dynamics derived using computational geometry techniques. Finally, while musical instruments can play arbitrary pitch sequences, kinesiology and dance style impose a variety of constraints on consecutive body postures. To smooth any abrupt transitions introduced by the chaotic mapping, we have developed a class of corpus-based interpolation schemes that capture and enforce the dynamics of a particular dance genre.

The results produced by these mapping and interpolation algorithms have intrigued both dynamicists and dancers. The chaotic variations bear an obvious resemblance to the originals[2], and yet they are also clearly different; broadly speaking, the variations resemble the originals with some shuffling of coherent subsequences. When contrasted with random shuffles of the same sequences, the properties of this scheme become even more apparent: the temporal structure of randomized "variations" is completely unlike that of the originals. It is impossible to appreciate these results from a textual description; please see the animations at `http://www.cs.colorado.edu/~lizb/chaotic-dance.html`.

2 Linking Attractor Geometry and Dance Structure

A point in a state-space trajectory of a dynamic system can be described at different precisions, ranging from a tuple of real numbers to a symbol that identifies a large state-space region. Though a symbolic representation abstracts away much detailed information about the dynamics, it preserves many of its invariant properties; see, e.g., Hao[5] for details. Establishing a symbolic dynamics presents two problems: the partition and the ordering. This paper offers novel and unusual solutions to both: we use computational geometry techniques on points of a trajectory to obtain a good partition, and we use the natural progression of body positions in a dance sequence to induce the symbol order.

Creating a partition for the purposes of re-mapping a dance sequence requires tiling the state-space region occupied by the attractor with N nonoverlapping cells, where N is the number of moves in the sequence. Voronoi diagrams[6] — the intersection of the perpendicular bisectors of adjacent pairs of points — are an effective solution to this problem; see figure 1 for an example. The actual implementation uses K-D trees[3] to reduce the computational complexity of a later nearest-neighbor step in the algorithm

[2] A well-known dynamicist opined "It looks like Al Gore doing the macarena."

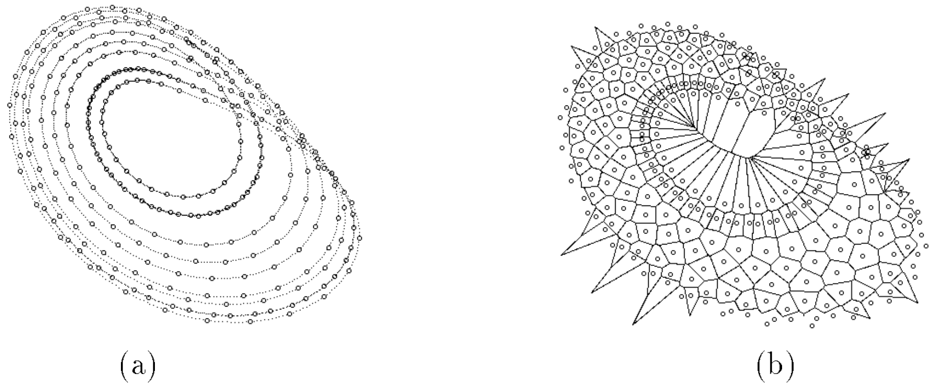

Figure 1: A Rössler trajectory for a 300-move dance sequence and the associated Voronoi-diagram partition of the attractor. The perpendicular bisectors of the circled trajectory points in (a) yield the Voronoi diagram in (b). Line segments that extend beyond the bounding box of part (b) have been omitted from this plot.

from $O(N^2)$ to $O(N \log N)$. Note that a tiling derived from a short trajectory may not cover the attractor, and numerical dynamics effects preclude solving this problem by simply increasing the timestep. We address this by fixing the timestep and trajectory length[3] and using a "skip" parameter to control the spacing of the trajectory points used to construct the Voronoi cells. In figure 1, for instance, every tenth trajectory point (shown circled) generates a cell.

The symbol set used here represents the position of each of the 44 primary joints in the human body with a *quaternion* — a standard representation in rigid-body dynamics, dating back to Hamilton[4]. A quaternion consists of a vector $\vec{u} \in R^3$ and a scalar r that specifies the angle of rotation around that vector. Thus, a body position symbol is quite complicated: 44 descriptors (`pelvis, right-wrist`, etc.), 176 floating-point numbers (four for each joint), and a variety of information about the position and orientation of the center of mass.

Given an N-move dance sequence, expressed in terms of the quaternion-based symbol set described above, and an N-cell Voronoi tiling of a chaotic attractor — like the one shown in figure 1 — establishing a mapping that links the attractor geometry and the dance structure simply amounts to equating indices: the first entry in the itinerary of cells traversed by $\phi_\omega(x_0)$ is labeled with the symbol that describes the first position in the dance sequence, and so on[4]. Using this mapping to create a variation is equally simple, but slightly more computationally expensive: we generate another trajectory $\phi(x')$ from an initial condition x' near the attractor, use the K-D tree to determine the cell in which its first point falls, output the associated body-position symbol to an animation tool, and repeat to the desired variation length.

[3]We choose values for these parameters that assure that the transient has died out, the attractor is covered, and the dynamics include no spurious numerical effects.

[4]This representation — a symbolic dynamics induced by that dance sequence — is conjugate to the R^n dynamics of $\phi_\omega(x_0)$, which is interesting in that it implies that the animations are formally equivalent, in a precise mathematical sense, to the "real" dynamics[5]

3 Interpolation

The re-mapping procedure described in the previous section introduces abrupt transitions in the chaotic variation — places where consecutive body positions would require physically impossible or stylistically illegal moves. The interpolation scheme described in this section inserts new body postures into these gaps in order to smooth the progression. Note that one cannot simply use splines or some other purely mathematical technique on the quaternion data, as such tactics would not obey stylistic or kinesiological requirements. Instead, we use a corpus of human movement to select a posture sequence that would naturally bridge the gap between the two positions in question.

To accomplish this, we construct a directed graph with one vertex corresponding to each body position in the corpus. Transitions between positions are represented by edges between the corresponding vertices; if no single edge connects vertices b_i and b_j, then the transition $p(t) = (b_i, b_j)$ did not appear in the corpus and is therefore deemed illegal. When such a transition is encountered in the chaotic variation, we use a forward-backward modified Dijkstra's algorithm[2] to find the shortest path between the respective vertices of the two body states and then insert the corresponding move sequence into the gap.

This scheme is quite coarse-grained; to better model and enforce the nuances of a dance genre, we need to improve it. Firstly, the atomic representational unit is a full body position; the next version will perform *joint-wise* interpolation instead. Unfortunately, the graph will then contain $O(10^{88})$ vertices in the worst case. To work around this, we are working on a *hierarchical* graph structure that exploits knowledge of how the body is connected. Secondly, we are adding weights to the graph edges that represent the probability of each inter-move transition. The techniques in this paragraph were inspired by solutions to similar problems that arise in molecular biology (DNA sequencing) and computational linguistics (learning a grammar from a corpus and then using it to construct sentences).

4 Results

Figure 2 shows a simple ballet jump sequence, a chaotic variation of that jump generated with the Lorenz system, and a smoothed version of that variation. The sequence shown in the middle row of figure 2 was derived from the original using the re-mapping scheme described in section 2. An abrupt transition is visible between the third and fourth moves of this variation; the interpolation scheme described in section 3 inserted two new moves (arrows) to produce the smoothed sequence shown at the bottom of the figure. Note that the inserted moves define a very natural way to move between the two body positions that frame the abrupt transition.

While it is clear from the figure that the jump positions are indeed shuffled and that the interpolated version is indeed smoother, it is impossible to fully appreciate these results from a static portrayal of such a short sequence; please see the web site listed at

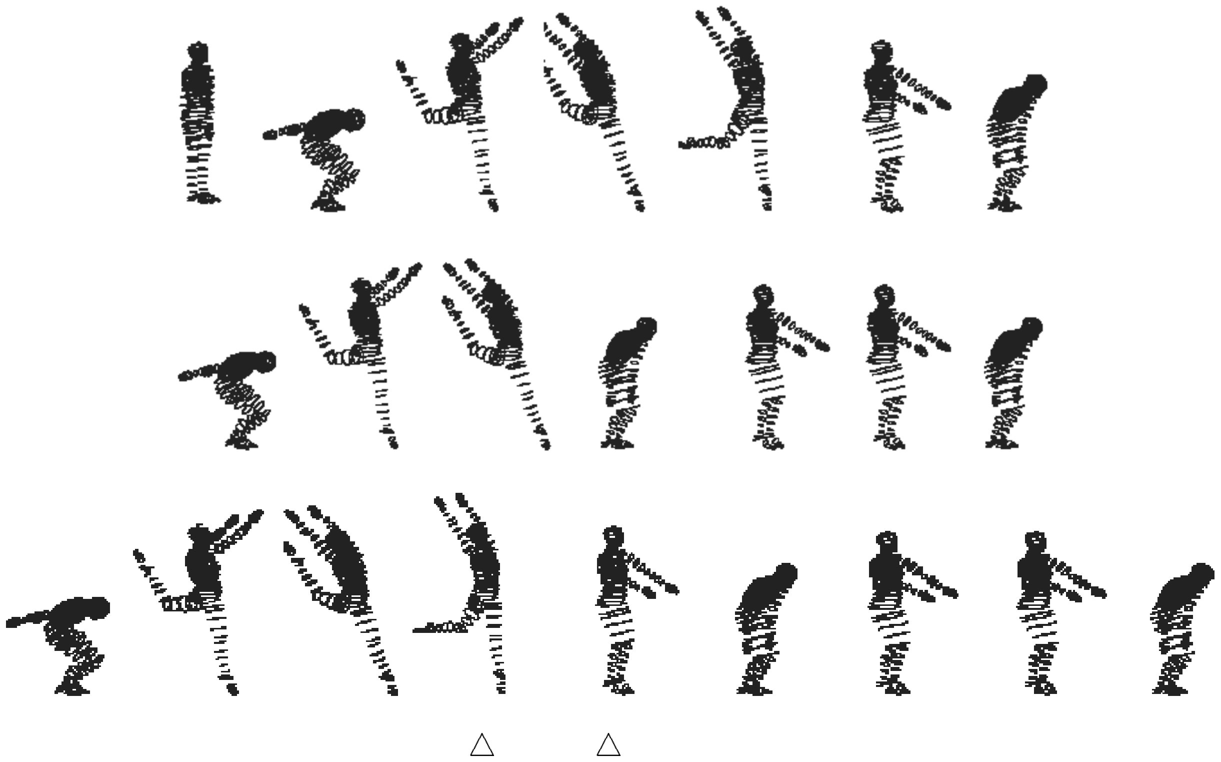

Figure 2: A ballet jump: the original sequence (above), a chaotic variation (middle), and an interpolated version of that variation (below). The moves identified by arrows in the lower sequence were inserted by the interpolation scheme to smooth an abrupt transition between the third and fourth moves in the variation above it.

the end of the introduction for animations of a variety of different movement sequences and chaotic variations[5]. Loosely speaking, the latter resemble the former with some shuffling of coherent subsequences. Randomly shuffled versions (also available on the website) clearly demonstrate, by contrast, how much structure is retained by the chaotic variation scheme.

These results set off a host of interesting questions. For instance, a shorter dance sequence implies larger cells and hence a "coarser" symbolic dynamics; this has interesting effects on how smoothly the cell itinerary of $\phi(x')$ moves and shifts along the original attractor, with corresponding implications for the animation and its resemblance to the original piece. The attractor geometry plays an obvious, visible, and interesting role in the character of the variation. Space limitations preclude more discussion here; please see the website for a longer preprint, along with a simple animation package and the re-mapping code itself. We encourage the readers (and their students) to create new animations and/or try different ODE systems, initial conditions, and so on.

[5]Variations were constructed using both Lorenz and Rössler equations in order to show how the attractor geometry affects the variation.

5 Conclusion

Evaluation of these results is necessarily subjective. We have shown these animations to hundreds of people, including dozens of dancers. The consensus is that the variations not only resemble the original pieces, but also are in some sense pleasing to the eye. They are both different from the originals and faithful to the dynamics of the dance genre: there are no jarring transitions or out-of-character moves. This is a non-trivial accomplishment. A previous attempt to use mathematics to generate choreographic variations — a subsequence randomization scheme used by the well-known choreographer Merce Cunningham — met with an extremely negative reception in the dance world.

From a scientific viewpoint, this scheme is interesting for several reasons. It involves a formal (albeit unusual) application of symbolic dynamics, the properties of chaotic attractors, and rigid-body mechanics: the partition for the symbolic dynamics is generated automatically using computational geometry techniques and the natural order of the dance sequence, and the symbol set relies on a representational device invented by Hamilton himself. By applying methods from graph theory, statistics, and computational linguistics to a corpus of dances from a particular genre, the interpolation scheme proposed here smooths awkward transitions in a physically *and stylistically* coherent fashion. Last, but certainly not least, showing these results in a classroom is an enormously effective way to motivate students to learn the mathematics of rigid-body dynamics and chaos.

Acknowledgements: Apollo Hogan contributed the K-D tree code and a variety of good ideas to this project and Meenakshy Chakravorty helped set up the website and write the associated animation code. The authors would like to thank David Capps of the University of Colorado Department of Theater and Dance for helpful conversations and Margo Seltzer of Harvard for suggesting — and teaching us — the karate sequences.

References

[1] D. S. Dabby. Musical variations from a chaotic mapping. *Chaos*, 6:95–107, 1996.

[2] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959.

[3] J. H. Friedman, J. L. Bentley, and R. A. Finkel. An algorithm for finding best matches in logarithmic expected time. *ACM Transactions on Mathematical Software*, 3:209–226, 1977.

[4] H. Goldstein. *Classical Mechanics*. Addison Wesley, Reading MA, 1980.

[5] B.-L. Hao. Symbolic dynamics and characterization of complexity. *Physica D*, 51:161–176, 1991.

[6] F. P. Preparata and M. I. Shanos. *Computational Geometry: An Introduction*. Springer-Verlag, New York, 1985.