Volume Rendering Strange Attractors

Dietmar Saupe and Wayne Tvedt

Abstract

We consider approximation and rendering techniques for strange attractors that arise in the study of chaotic dynamical systems. We propose that the ideal represention of a strange attractor is a volume rendering of its invariant probability measure, and provide efficient data structures and convergence criteria for the task.

Keywords: scientific visualization, volume rendering, strange attractors, chaos

Introduction

The primary task of complex visualization in mathematics is creating meaningful images of objects which defy intuition. It is more than an issue of approximation, because one has not only to define what quantities can be visualized, but to be conscious of their topological properties.

One of the most confounding topological structures to come into currency is that of the strange attractor. Guided by mathematical development, physicists and mathematicians were led to believe that the long term behavior of dynamical systems would always run into simple patterns of motion, such as a rest point or a limit cycle. The discovery of strange attractors by the meteorologist Edward N. Lorenz in 1962 disproved this belief. Strange attractors are those patterns which characterize the final state of dynamical systems that are highly complex and show all the signs of chaos. They are indeed strange, and yet they are now proven to be all around us. Moreover, strange attractors are the point where chaos and fractals meet in an unavoidable and most natural fashion: as geometrical patterns strange attractors are fractals; as dynamical objects strange attractors are chaotic. Researchers in the natural sciences became aware of the subject and concentrated on the irregular patterns of processes which they had previously dismissed as misfits. There is now a whole new experimental and theoretical industry dealing with strange attractors and their reconstruction from experimental data. Scientists hope to be able to crack the mysteries of our planet's climate as well as the secrets of turbulence or human brain activity through the metaphor of strange attractors.†

† For an introduction to the topic of chaos and strange attractors for the nonspecialist, see [Peit92].

In this paper we discuss methods for rendering strange attractors of continuous systems in three-dimensional phase space. The methods can also be adapted to other systems, which may be discrete or may live in a higher-dimensional space.

For the purposes of this paper, we present an intuitive and working definition of a strange attractor for continuous or discrete dynamical systems in Euclidean spaces (given by differential or difference equations of motion). The final mathematical definition which provides the 'correct' way to deal with attractors is still outstanding.† A set, A, is considered a strange attractor if the following four conditions are satisfied:

there is a neighborhood R of A such that R is a trapping region, i.e., each trajectory started in R remains in R for all time; moreover, the orbit becomes close to A and stays as close to it as we desire. Thus, A is an *attractor*;

orbits started in R exhibit sensitive dependence on initial conditions; this makes A a *chaotic attractor*;

the attractor has a fractal structure and is thus called a *strange attractor*;

A cannot be split into two different attractors; there is an initial point in R such that the corresponding trajectory gets arbitrarily close to any point of the attractor A.

In the usual way to draw a strange attractor we begin by computing a trajectory that starts in the trapping region, discarding the transient phase needed to get sufficiently close to the attractor. It then lets that orbit run along the attractor until we have a satisfactory image. Though perhaps good suggestions as to the general shape of the attractor, such pictures are misleading, for the best you can come up with is a spotty, skeleton-like cover without any depth cues. They are not truthful pictures, by the obvious fact that you get a different skeleton when a different initial point is picked. And, letting the trajectory run for a longer time does not improve things: it starts to fill up the silhouette as a solid and we lose all internal detail. For example, the continuation of the trajectory from the Rössler attractor shown in Figure 1 would just be seen from above as a solid disk with a hole in the center.

The trajectory technique also falls short of actually showing the real structure of the attractor, with all its bends and folds, the bandedness (or lack thereof) of the distribution of trajectories on the attractor and the spatial texture, smooth in some directions but fractally distributed in others. For instance, we found that the attractors of Lorenz and Rössler, which seem to have the same texture with the trajectory method, are actually quite different in texture.

In short, two ideals of adaptive data rendering are missed:

the final image should be some kind of precise, meaningful statement about the object — a quantity to look at;

even if we can never get a 'true' image and must settle for some series of approximations, we should be able to buy more quality of image (that is,

---

†See, for example, the discussion in [Guck83], pp. 255–259.

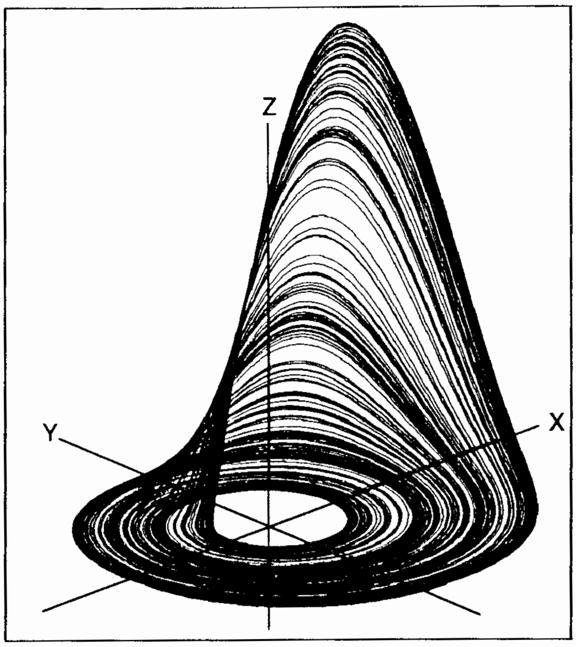

Figure 1. A trajectory of the Rössler system plotted in three-dimensional phase space as a first picture of the Rössler attractor.

convergence of the image with the 'true' object) with time, and we should have some idea as to how fast the quality improves with time.

We seek the same order of confidence in rendering strange attractors. The most natural quantity to look at is the attractor's ***natural measure***. Roughly, this measure of a region represents the portion of time that a trajectory, moving chaotically over the strange attractor, spends in that region. For example, let us assume that a given trajectory $X(t), t \geq 0$ generates the attractor. Then the measure μ of a cube C in phase space is the fraction of time the solution $X(t)$ passes through the cube

$$\mu(C) = \lim_{T \to \infty} \frac{1}{T} \int_0^T \mathbf{1}_C(X(t))dt \qquad (1)$$

where $\mathbf{1}_C$ is the indicator function having value 1 in C, and 0 otherwise.

There is an analogy here to physics. One way to think about an electron of, for example, a hydrogen atom is in terms of a particle having a certain location and impulse at each point in time. In other words, we can imagine an electron as a particle rapidly spinning around on a shell about the nucleus of the atom. However, a view which is more appropriate in many respects is that of the electron as a charge distribution on the entire shell. The older, mechanical type of interpretation corresponds to that of representing strange attractors as portions of short or moderately long trajectories, while the other has similarities to the natural measure on strange attractors.†

Actually, the first oscilloscope traces of strange attractors, looked at by the first explorers in the field, were decent approximations of their natural measures. The output of one or more variables of an electronic circuit representing the equations of motion is plugged into a standard laboratory oscilloscope. The 'deposited' trajectories are always fading away, but slowly. If the sweep/fadeout rate is tuned right, one can capture the high-density regions and show the smooth connecting 'tissue' in between. However, these images are not quantitatively correct — the trajectories are weighted according to how recently they were deposited, with a bright spot at the leading point — and are always shimmering and hard to photograph.

By recording a histogram of trajectories numerically, we hope to achieve the same smoothness, but with measurable precision and manipulation capability.

Computing Strange Attractors

We consider a chaotic flow moving (depositing its measure) around a regular Cartesian lattice of mesh size ε and aim at representing the attractor as the set of cubes (voxels) having nonempty intersection with the set. By a 'chaotic flow' we mean a system of differential equations in $\mathbb{R}^3$ with a strange attractor. For example, the Rössler system is given by the equations

$$\dot{x} = -(y+z) \quad \dot{y} = x + \frac{y}{5} \quad \dot{z} = \frac{1}{5} + (x-c)z \tag{2}$$

which yields the chaotic Rössler attractor for the parameter $c = 5.7$.

We consider a trajectory started on or very close to the attractor and determine all cubes C that the trajectory passes through at some time. Ideally we would want to accumulate the time spent by the trajectory in each cube, but a numerical 'trajectory' is actually a sequence of points $X_0, X_1, X_2, \ldots$, tolerably close to the mathematical trajectory, computed by some integration scheme. Rather than interpolating between points and measuring voxel intersections, we increment a counter at each voxel that is hit by the discrete orbit, which statistically should yield the same result. The integral in Eq. (1) can be approximated in this case by the sum

$$\mu(C) \approx \frac{1}{n} \sum_{0}^{n-1} 1_C(X_i) \tag{3}$$

If we let $n \to \infty$, then equality holds.

A note should be made about the choice of step length h in the integration scheme. Our approach here is to sample the solution such that at all times

† However, the natural measure of a strange attractor is different in the sense that its support is a fractal and it does not allow a density function; it is a singular measure.

the distance of two subsequent points is on the order of half the mesh size ε or less. In practice, though, maintaining the same step size h for the entire process is impractical, because the speed of the trajectory can vary greatly over the attractor. It is more economical to reparameterize the differential equation using arc length. Then we can set $h = \varepsilon/2$, and the trajectory will be sampled at the constant rate of twice the mesh size. The samples must be appropriately weighted according to speed corresponding to the original equation.

Voxel techniques are always brutal memory consumers; but fortunately we can take advantage of the fractal distribution of the actual data set in a very concrete way. If the object is said to have dimension near 2.1, then we expect the number of cubes needed to span the object to scale as $\varepsilon^{-2.1}$ rather than ε^{-3}, which means that at high resolutions that data is closer to planar size (N^2) than spatial (N^3). Actual data structures always have pointer overhead, and in the case of octrees, the fact that you need to cover the lower leaves (at small mesh sizes where you reap the advantages of having a low fractal dimension) with parent nodes (at scales where you do not) means your savings are not quite so astronomical. But it is still necessary to keep the data structure manageable in a workstation environment. Running on an SGI Indigo with 16 Mb and all the window system overhead in the background, we were able do our stats on a 512 × 512 × 512 octree. Pointer overhead counted for about a third against the actual number of leaves storing histogram values.

One needs to store the histogram not only to capture the measure but also to know when to stop the computation. A standard procedure which is often used to compare images is derived from root-mean-square (rms) differences. It is straightforward to adapt the method to quantify differences between approximations of measures of the attractor. Not knowing the true measure of the attractor, we generate two measures simultaneously in the same octree, or rather two overlapping ones, and take the rms difference. This (almost) doubles the memory needed — 'almost', because although leaves are duplicated most of the parent nodes are shared.

Our empirical tests for the Rössler attractor show that the relative rms difference decreases by a factor of about 0.7 when the number of sample points is doubled. The computation can be stopped when this rms difference drops below a specified threshold, or when the allotted computing time is used up. At a final stage the rms difference yields an indication of the quality achieved by the approximation. We then average the two measures to get a slightly better fit, and project the result onto the viewing plane.

Rendering the Natural Measure

In principle we can proceed using standard methods of volume rendering. Here we only point out a few but important differences. It is uncommon in volume rendering to deal with high resolution but sparse data sets. In other words, we need to consider volumes of, say, 512^3 voxels, most of which carry a zero measure.

There are several possible ways to project the data. For us it was sufficient to project each nonzero voxel as a light source with intensity corresponding to the measure it carries, applying common histogram equalization and gamma correction to improve contrast.

Alternatively, we can interpret a data point as representing a partially opaque object in space, which reflects light from one or more external light sources. For the purpose of computing the reflected light, normal vectors need to be supplied. In volume rendering, normal vectors are usually obtained from gradients computed by central difference schemes. In the case of strange attractors, however, there is a more appropriate way to arrive at normal vectors. At every point of a trajectory there is also a maximal spreading direction, which indicates in which direction nearby trajectories of the attractor are most strongly repelled from the given one. In other words, the velocity vector at a point, together with the vector for the maximal spreading direction, gives us a tangential plane, from which we readily obtain a normal vector. These maximal spreading directions are related to the so-called Ljapunov exponents and can be conveniently computed along with the trajectory of the original system (see, e.g., [Peit92]).

As a first result of this research, we present two images for the invariant measure of the Rössler attractor (see Figure 2 and Plate 18). They are based on approximately four million points.

Open Problems and Extensions

The rms test is an experimental and intuitive one, and in some sense an arbitrary choice. Are there more suitable tests? Does the addition of noise in the maximal spreading direction perpendicular to the trajectory help to achieve a faster convergence toward a 'fuzzy' rendition by sacrificing a little detail in resolution? We can even go one step further and replace the trajectory points that enter the data structure by small line segments centered at the points and oriented in the maximal spreading direction. This amounts to drawing the attractor using a calligraphic pen with its fat edge along the spreading direction.

An alternative approach to the direct computation of the measure in state space is to consider only the measure in a transverse two-dimensional cross-section of the attractor, called a Poincaré section. Its computation is more accessible in two regards. We save in storage by using a planar lattice rather than a spatial one. Moreover, the integration of the differential equation can make full use of adaptive step-size methods. Thus, one can expect that convergence of the approximations of the natural measure associated with the Poincaré map can be achieved more rapidly. Additionally, methods can be devised to extend this measure to the full three-dimensional state space.

The final observation to be made is that, as with computing the digits of π to arbitrary precision, there is an inherent amount of work to be done in order to describe a complex object to the human senses. The order of 'work' needed — convergence time, memory requirements, program complexity — is an essential

Figure 2. The invariant measure of the Rössler attractor; darker parts indicate regions of high measure. The picture is based on 4 million points with a normed rms quality factor 0.04.

property of that object. In the case of strange attractors, perhaps the order of work can be tied to the objects' information dimension, or used to define a taxonomy of attractors according to their complexity.

Acknowledgments. This paper was inspired by some renderings of a natural measure as a (projected, two-dimensional) histogram done by Scott Hotton as part of a class project at the University of California, Santa Cruz, in the study of driven oscillators. We also thank Carl Evertsz for useful discussions on the convergence of fractal measures.

References

[Guck83]
Guckenheimer, J., and Holmes, P., *Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields,* New York: Springer-Verlag, 1983.

[Peit92]
Peitgen, H.-O., Jürgens, H., and Saupe, D., *Chaos and Fractals,* New York: Springer-Verlag, 1992.

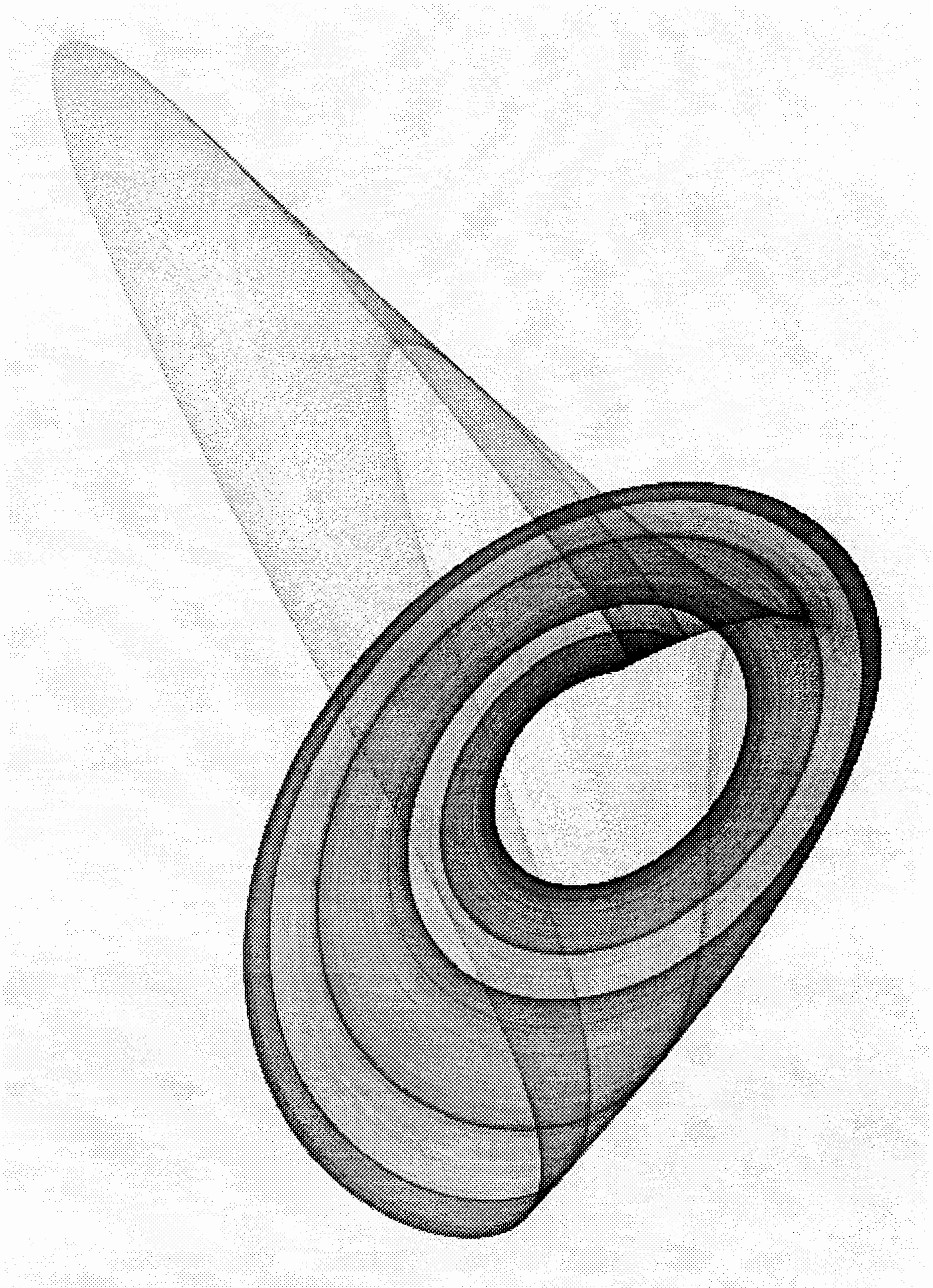

Figure 2