From: AAAI-87 Proceedings. Copyright ©1987, AAAI (www.aaai.org). All rights reserved.

Extracting Qualitative Dynamics from Numerical Experiments

Kenneth Man-kam Yip
MIT Artificial Intelligence Laboratory
NE 43 - 438
545 Technology Square,
Cambridge, MA 02139.

Abstract

The Phase Space is a powerful tool for representing and reasoning about the qualitative behavior of nonlinear dynamical systems. Significant physical phenomena of the dynamical system – periodicity, recurrence, stability and the like – are reflected by outstanding geometric features of the trajectories in the phase space. Successful use of numerical computations to completely explore the dynamics of the phase space depends on the ability to (1) interpret the numerical results, and (2) control the numerical experiments. This paper presents an approach for the automatic reconstruction of the full dynamical behavior from the numerical results. The approach exploits knowledge of Dynamical Systems Theory which, for certain classes of dynamical systems, gives a complete classification of all the possible types of trajectories, and a list of bifurcation rules which govern the way trajectories can fit together in the phase space. These bifurcation rules are analogous to Waltz's consistency rules used in labeling of line drawings. The approach is applied to an important class of dynamical system: the area-preserving maps, which often arise from the study of Hamiltonian systems. Finally, the paper describes an implemented program which solves the interpretation problem by using techniques from computational geometry and computer vision.

I. Introduction

The theory of any functions begins naturally with its qualitative aspect, and thus the problem which first presents itself is the following: Construct the curves defined by differential equations.
- Henri Poincare

Qualitative Physics is a young field. Progress is made when researchers formalize and implement their understanding of how certain qualitative reasoning tasks, such as prediction of future behavior, and explanation of how the behavior comes about, are being performed in particular problem domains. Two domains, among others, have received much attention: circuit analysis and design in the engineering domain, and simple boilers and fluid flow in commonsense physics. Early works in Qualitative Physics primarily dealt with incremental deviation from equilibrium states where time evolution is not explicitly considered [De Kleer, 1979]. More recent works attempt to extend DeKleer's qualitative algebra and incremental analysis to handle time-varying behavior [Forbus, 1984, Williams, 1984, Williams, 1986, Kuipers, 1984].

The machineries developed for qualitative reasoning – qualitative state vector, quantity space, and limit analysis – are largely applicable to systems which are piecewise well-approximated by low-order linear systems or by first order nonlinear differential equations. The behavior of linear systems is particularly simple: the complete input-output behavior can be summarized in a single system transfer function. Consequently, if the response to one type of input is known, no more information is needed to determine responses for other input signals.

The situation in a nonlinear system is completely different: essential changes in the qualitative behavior of the system may occur as the amplitude of the input signal changes, or as the starting conditions are varied. More importantly, nonlinear systems have a far richer spectrum of dynamical behavior. Simple equilibrium points, periodic and quasiperiodic motion, limit cycles, chaotic motion as unpredictable as a sequence of coin tosses – these are some of the behavior found in a typical nonlinear system.

Unfortunately, these nonlinear characteristics do not show up in first order nonlinear differential equations. This is because the continuity and (local) uniqueness of flow severely constrain the kind of behavior possible on the real line: the flow either tends towards an equilibrium, or goes off to infinity.

In this research, I therefore propose to look at dynamical systems – those typically encountered in Physics – to provide a new source of examples for investigation into the fundamental issues of descriptive language, style of reasoning, and representation techniques in qualitative reasoning about nonlinear dynamical systems. Specifically, I will consider two-dimensional discrete dynamical systems defined by area-preserving maps containing a single control parameter. The study of area-preserving maps – transformations of the plane which preserves area – began with the venerable problem of the stability of the solar system. I choose to investigate this simplest nontrivial type of conservative system because many important problems in physics – the restricted 3-body problem, orbits of particles in accelerators, and two coupled nonlinear oscillators, just to mention a few – can be reduced to the study of area-preserving maps.

To illustrate the vocabularies that a physicist uses to describe the qualitative behavior of nonlinear systems, we can examine the following description of one of the phase portraits obtained from the results of numerical experiments with the *quadratic map* (see next section):

> *Figure 1a represents a number of trajectories (sequences of points) for* $\cos\alpha = 0.4 \ldots$ *a regular structure in a neighborhood of the elliptic fixed point at the origin, and farther away a chaotic zone. In many places the plotted points are so dense that they give the illusion of a continuous curve. Near the origin, the "curves" are almost circular. As we move outward, the curves become distorted. Just inside the outermost regular curve lies a chain of six closed curves, or "chain of islands". Successive points of a trajectory jump from one island to another by application of the mapping. Finally, as the curves become more distorted, there is a sudden break-up and the set of points no longer lies on a curve, but seems rather to fill a two-dimensional region.*

The description is an edited form of a passage from [Henon, 1981, pages 99–100]. What is striking in this description is that the language is entirely geometric: it contains terms like *fixed point, continuous curve, chain of islands, two-dimensional region* etc. The phase portrait is completely characterized by the spatial relationships among these geometric objects.

A goal of this research is to develop a clear understanding of the geometric language displayed in the above description. The product of the research will be a prototype program that uses numerical simulation to translate state equations into phase space pictures, and exploits knowledge of Dynamical Systems Theory to reconstruct a complete geometric description of the dynamical behavior from the numerical results. The significance of the research is two-fold: (1) to develop powerful tools that allow scientists to rapidly analyze dynamical models, and (2) to develop recognition and reasoning techniques based on the idea of representing qualitative behavior in a geometric form.

The paper is organized as follows. Section two defines the task. Section three reviews the knowledge – qualitative dynamics and bifurcations – required for the task. Section four describes my approach to the experiment control problem. This part has not been implemented. Section five, the main result of this paper, shows the major pieces of an implemented program which solves the interpretation problem.

II. The Task

Given an one-parameter area-preserving map defining a discrete dynamical system, I am interested in describing the qualitative long-term, or asymptotic, behavior of the system over its entire operating range. Specifically, if U is an open subset of the phase space $X \subseteq \Re \times \Re$, and $J \subset \Re$ an interval over which the control parameter varies, I want my program to automatically generate a family of phase portraits describing the main dynamical properties of the map for all initial conditions in U and parameter values in J.

To explore the complete dynamics of a nonlinear system over a large region of the phase space and parameter space is a fairly typical problem in the physics literature. A good illustration of this task is provided by Henon's well-known paper, "Numerical Study of Quadratic Area-Preserving Mappings" [Henon, 1969]. The goal of Henon's paper is to provide a description of the main properties of the *quadratic map*:

$$\begin{aligned} x_{n+1} &= x_n \cos\alpha - (y_n - x_n^2)\sin\alpha \\ y_{n+1} &= x_n \sin\alpha + (y_n - x_n^2)\cos\alpha \end{aligned} \quad (1)$$

where x and y are the state variables, and α is the control parameter. The main results of Henon's paper are shown in Figures 1(a)-(f), which display the output of many numerical simulations.

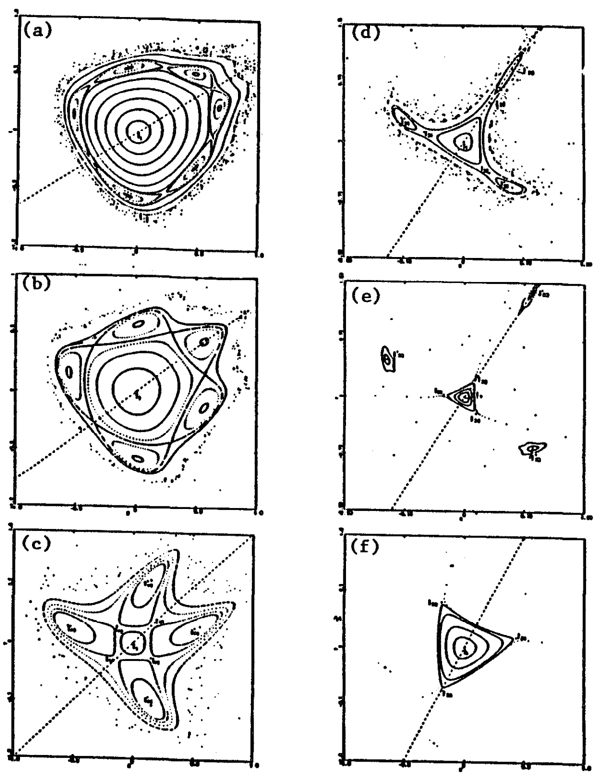

Figure 1: *A partial list of phase portraits from numerical experiments. The figures are generated by plotting several hundred of successive values of* (x_n, y_n). *(a)* $\alpha = 1.16$ *(b)* $\alpha = 1.33$ *(c)* $\alpha = 1.58$ *(d)* $\alpha = 2.0$ *(e)* $\alpha = 2.04$ *(f)* $\alpha = 2.21$. *Dashed line: axis of symmetry.*

The simplest approach to this problem is the *brute* *force* method: it divides the phase space and parameter space into small grids and tries every possible combinations of initial conditions and parameter values. A simple calculation will show that this method involves an enormous amount of computation. For instance, if we choose a uniform grid size of 0.01, we have to compute approximately $300 \times 300 \times 600 = 54$ million orbits. Assuming, on the average, 0.02 second is needed to compute a trajectory of 500 points, it will take over 300 hours of computation time to compute all the trajectories.

The brute force method suffers from two serious problems. First, it is grossly inefficient because most of the phase portraits computed will be qualitatively the same. Second, it is not reliable because there is always the danger of missing some important qualitative features when the change occurs at a resolution finer than the grid size.

A physicist often does much better than this. Figure 2 represents a flow-chart of what a professional physicist does during the numerical experiment. The flow-chart has two nested loops. The outer loop involves deciding when to stop the experiment; the inner loop, when to move on to next parameter value. Controlling what experiment to do next, and interpreting the results of the simulation – these are the two most important decisions the experimenter has to make.

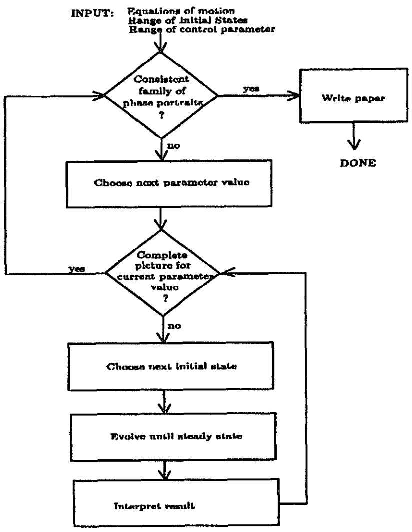

Figure 2: *Flow-chart which describes the process of experimenting with a dynamical system.*

The task of behavior prediction can now be summarized as this: to develop a picture of all possible solutions to the dynamical system from a limited amount of numerical experiments at a limited number of initial conditions and parameter values. The key observation is that knowledge of qualitative dynamics and their geometric manifestations in the phase space provides a strong constraint on the type of behavior possible. As we will see in the next section, this constraint translates into a dramatic reduction of the amount of search required to find those combinations of initial states and parameter values that lead to "interesting" phase portraits.

III. The Knowledge

A. Terminology

The purpose of this section is to introduce some concepts and definitions from Dynamical Systems Theory [Hirsch and Smale, 1974]. A dynamical system consists of two parts: (1) the system state, and (2) the evolution law. The **system state** at any time t_0 is a minimum set of values of variables $\{x_1, \ldots, x_n\}$ which, along with the input to the system for $t \geq t_0$, is sufficient to determine the behavior of the system for all time $t \geq t_0$. The variables which define the system state are called **state variables**. The conceptual n-dimensional space with the n state variables as basis vectors is called the **phase space.** A **state vector** is a set of state variables considered as a vector in the phase space. As the system evolves with time, the state vector traces out a path in the phase space; the path is called an **orbit** or a **trajectory**. Finally, a **phase portrait** is a partition of the phase space into orbits.

The **evolution law** determines how the state vector evolves with time. In a finite dimensional discrete time system, the evolution law is given by difference equations. The difference equation is specified by a function $f : X \rightarrow X$ where X is the phase space of the discrete system. The function f which defines a discrete dynamical system is called a **mapping,** or a **map**, for short. The **multipliers** of the map f are the eigenvalues of the Jacobian of f. An **area-preserving map** is a map whose Jacobian has a unit determinant.

The set of iterates of f – $x, f(x), f^2(x), f^3(x), ..., f^n(x)$ – as n becomes large is called the **orbit of x relative to f**; it captures the **history** of x as f is iterated.

Two types of point have the simplest histories – fixed point, and periodic point. The point x is a **fixed point** of f if f(x) = x. A fixed point x is called **stable**, or **elliptic**, if all the multipliers of f at x lie on the unit circle; it is called **unstable**, or **hyperbolic**, otherwise. The point x is a **periodic point** of period n if $f^n(x) = x$. The least positive n for which $f^n(x) = x$ is called the period of x. The set of all iterates of a periodic point forms a **periodic orbit**.

B. Qualitative Dynamics and their Geometry

1. Types of Orbit

In this section, I give a brief outline of the most important types of orbits in area-preserving maps.

There are four ways in which an orbit generated by infinitely many iterations of the map can be explored in the phase space:

1. A finite number of N points are encountered repeatedly, corresponding to a periodic orbit of period N.
2. The iterates fill a smooth curve, which is a topological circle, in the phase space. This curve is called an **invariant curve** because the whole curve maps onto itself under the action of the map.
3. The iterates can form a random splatter of points that fills up some area of the phase space. This happens when the orbit evolves in a *chaotic* manner whose detail depends sensitively on the initial conditions.
4. The iterates leave the phase space after a finite number of iterations and escape to infinity in the end. These points are called **escape points**.

Since the dynamics of area-preserving maps and Hamiltonian systems have a lot in common, these four types of geometric orbit have important physical interpretations. Due to space limitation, I just list the interpretations below. The explanation of these interpretations can be found in [Yip, 1987].

periodic points	$\Longleftrightarrow$	periodic motion
invariant curve	$\Longleftrightarrow$	quasiperiodic motion
chaotic region	$\Longleftrightarrow$	chaotic motion
escape points	$\Longleftrightarrow$	unbounded motion

2. Bifurcations: Qualitative changes in the phase portrait

Two phase portraits are **qualitatively equivalent** if there exists a homeomorphism between them which preserves fixed points, periodic points, invariant curves, and their stability. **Bifurcation** is said to occur when the dynamical system goes through a qualitative change in its phase portrait as the control parameter is varied. I will focus on one important type of bifurcation: appearance and disappearance of periodic orbits.

Meyer [Meyer, 1970] gives a complete classification of the generic bifurcations of periodic points for one-parameter area-preserving maps. Meyer has shown that generic bifurcations occur when the multiplier $\lambda = n^{th}$ root of unity where n = 1, 2, 3, 4, and ≥ 5. The five types of generic bifurcation are: (1) extremal, (2) transitional, (3) phantom 3-kiss, (4) phantom 4-kiss, and (5) emission. Because of space limitation, I only discuss the case of *phantom 3-kiss* as an illustration of what the bifurcation geometry is. Again, detail of this can be found in [Yip, 1987].

Phantom 3-kiss occurs when the multiplier λ of the map is a cube root of unity. The region of stability of the elliptic fixed point shrinks to zero as the hyperbolic points of an unstable period-3 cycle "kiss" at the origin. After the "kiss", the fixed point turns elliptic again, and a new unstable period-3 cycle is emitted. Note the change in orientation of the *triangular* region around the elliptic point. The phantom 3-kiss is often preceded by extremal bifurcations in a region a bit further away from the original elliptic fixed point, resulting in the formation of a pair of elliptic and hyperbolic period-3 points.

The dynamics of the area-preserving maps severely constrain the way orbits can fit together in the phase portraits. These constraints, which are encoded in the bifurcation patterns, are thus analogous to the consistency rules for line labeling in Waltz's thesis [Waltz, 1975]. As we shall see later, the geometry of bifurcation allows us to decide whether a collection of phase portraits is consistent, and gives us clue to the types and locations of orbits that the program should be looking for.

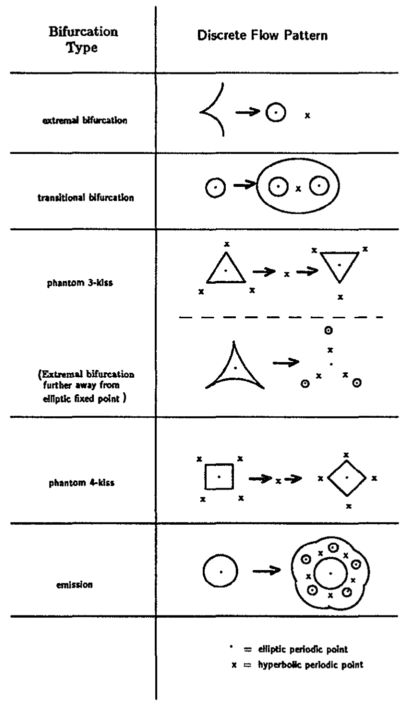

Figure 3: *Five Generic types of Bifurcation Geometry. A periodic point bifurcates whenever its multipliers pass through an n-th root of unity.*

IV. The Control Problem

A. How to start the numerical experiment?

Elliptic fixed points are good places to start. We expect that the orbits near an elliptic fixed point, where the linear terms of the map dominate, will be mostly invariant curves. We then search radially outward until we encounter island chains, and eventually chaotic regions.

B. How to decide what experiment to try next?

Knowing the generic bifurcation patterns is valuable for controlling numerical experiments. To begin with, it is difficult to locate the value of the control parameter at which bifurcation occurs: it is of probability almost zero that a randomly chosen point in the (x, y, α) space will be the bifurcation point. But the pattern of flow near a periodic point just before and after the bifurcation occurs in a finite range of the control parameter; hence the pattern is easier to detect during the experiments.

Once a given flow pattern is found to match some parts in our library of bifurcation geometries, it will give us strong evidence that the corresponding bifurcation exists, and we should be able to locate the rest of the flow patterns as given by the generic bifurcation. The pre-stored knowledge about these bifurcations gives us the complete information about what geometric objects, and approximately where in the control parameter space to look for.

To take an example, consider the *phantom 3-kiss* seen in figure 1d. The local flow pattern around the fixed point matches that in figure 3. According to the bifurcation pattern, the regular region around the stable fixed point will shrink in size, becoming an unstable fixed point; eventually, a new stable fixed point is born. So, we should expect to see figure 1f at some α slightly greater than two.

C. How to decide when to terminate the experiments?

Besides imposing a strong constraint on what can be expected to happen in the phase portrait, the generic bifurcations also provide an answer to the problem of termination: *a simulation experiment is incomplete unless all the major qualitative features in the phase portrait can be explained by this finite list of local generic bifurcations.* An example is the change of stability of a fixed point. Suppose we have numerically located the 3-island chain and the center elliptic point at some $\alpha = \alpha_0$ as in figure 1d. We know that the family of phase portraits is yet incomplete because we expect a *phantom 3-kiss* bifurcation to occur. In particular, we need to try at least two more experiments to obtain two phase portraits: first, at $\alpha = \alpha_1$ when the triangular region is flipped, and second, at $\alpha_2 \in (\alpha_0, \alpha_1)$ when the region becomes vanishingly small, indicating instability of the fixed point.

V. Interpretation Problem

The **Interpretation Problem** consists of the following sub-problems:

1. *Orbit Type.* How can one recognize the orbit type – a 0-dimensional finite point set whose elements are encountered repeatedly, a 1-dimensional smooth curve, or a 2-dimensional region – of a set of iterates?
2. *Clustering.* How can one determine the number of islands in an island chain? This number gives the period of the enclosed periodic point.
3. *Area and Centroid.* How can one estimate the centroid and area enclosed by the curve? The centroid is a good approximation of the location of the enclosed periodic point. The area gives a measure of saliency of the island chain.
4. *Shape.* How can one recognize the shape of a curve? For example, is it a 3-sided figure resembling a triangle?

In the following, I will show how these four problems can be solved by applying techniques from computational geometry and computer vision. Euclidean minimal spanning tree (EMST) [Preparata and Shamos, 1985], and scale space image [Witkin, 1983] – these are the two important data structures used by the interpretation program. The main processing steps are as follows (see figure 4).

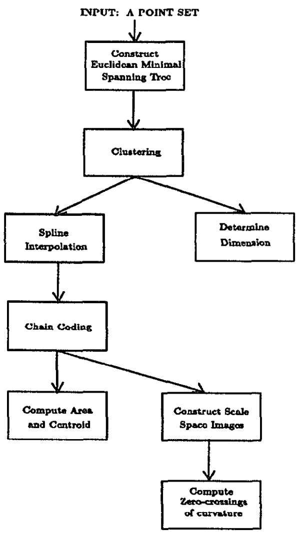

Figure 4: *Main processing steps for solving the interpretation problem*

- *Step 1.* The program computes a EMST from the input point set using the Prim-Dijkstra algorithm.
- *Step 2.* The program detects clusters in the EMST by looking for edges in the tree that are significantly longer than nearby edges. Such

edges are called *inconsistent* [Zahn, 1971]. The criterion of edge inconsistency suggested by Zahn is used to detect inconsistent edges. Inconsistent edges are then deleted, breaking up the EMST into connected sub-components. These sub-components are collected by a depth-first tree walk.

- *Step 3.* For each sub-tree of the EMST, the program examines the degree of each of its nodes, where the degree of a node is the number of nodes connected to it in the sub-tree. For a smooth curve, the EMST consists of two terminal nodes of degree one; the rest, degree two. For a point set that fills an area, its corresponding EMST consists of many nodes having degree three or higher.
- *Step 4.* To compute the area and centroid of the region bounded by a curve, the program generates an ordered sequence of points from the EMST, and spline-interpolates the sequence to obtain a smooth curve. The smooth curve is encoded using chain coding [Freeman, 1961]. Straightforward algorithms are then applied to compute the area and centroid.
- *Step 5.* A curve is parameterized by $C(s) = (x(s), y(s))$ where s is the arc length along the curve. The two functions x(s) and y(s) are computed from the chain code representation. Then, x(s) and y(s) are smoothed by the Gaussian and its first two derivatives at multiple spatial scales. Finally, the zero-crossings of the curvature function $\kappa(s)$, and the signs of $\dot{\kappa}(s)$ are computed to determine the locations and type of the extrema.

Examples of orbit recognition can be found in [Yip, 1987].

VI. Summary

In this paper, I have studied the task of qualitative analysis of nonlinear area-preserving map by numerical experiments. I have also described how to approach the two major problems in automating the experimenting process: (1) experiment control, and (2) result interpretation. The basic idea is that knowledge of qualitative dynamics and bifurcations provides a strong constraint on the type of behavior possible. Finally, I have described a program which solves the interpretation problem by using techniques from computational geometry and computer vision.

ACKNOWLEDGMENT

The content of this paper benefited from many discussions with Gerald Sussman, Hal Abelson, and Jack Wisdom. Ken Forbus and Brian Williams taught me their views of Qualitative Physics. I also thank General Electric for their financial support during this research.

References

[De Kleer, 1979] Johan De Kleer. *Causal and teleological reasoning in circuit recognition.* AI-TR 529, Massachusetts Institute of Technology, Artificial Intelligence Laboratory, 1979.

[Forbus, 1984] Kenneth Dale Forbus. Qualitative process theory. *Artificial Intelligence*, 24:83–168, 1984.

[Freeman, 1961] H. Freeman. On the encoding of arbitrary geometric configurations. *IRE, Trans. Electron. Comput.*, EC-10, 1961.

[Henon, 1969] M. Henon. Numerical study of quadratic area-preserving mappings. *Quarterly of Applied Mathematics*, 27, 1969.

[Henon, 1981] M. Henon. Numerical exploration of hamiltonian systems. In *Chaotic behavior of Deterministic Systems*, North-Holland, 1981.

[Hirsch and Smale, 1974] M.W. Hirsch and S. Smale. *Differential Equations, Dynamical Systems, and Linear Algebra.* Academic Press, 1974.

[Kuipers, 1984] Benjamin Kuipers. Commonsense reasoning about causality: Deriving behavior from structure. *Artificial Intelligence*, 24:169–204, 1984.

[Meyer, 1970] K.R. Meyer. Generic bifurcations of periodic points. *Transactions of American Mathematical Society*, 149, 1970.

[Preparata and Shamos, 1985] Franco Preparata and Michael Shamos. *Computational Geometry.* Springer-Verlag, 1985.

[Waltz, 1975] David Waltz. Understanding line drawings of scenes with shadows. In *The Psychology of Computer Vision*, McGraw-Hill, 1975.

[Williams, 1984] Brian Williams. Qualitative analysis of mos circuits. *Artificial Intelligence*, 24:281–346, 1984.

[Williams, 1986] Brian Williams. Doing time: putting qualitative reasoning on firmer ground. In *Proceedings AAAI-86*, American Association for Artificial Intelligence, 1986.

[Witkin, 1983] Andrew Witkin. Scale-space filtering. In *Proceedings IJCAI-83*, International Joint Conference on Artificial Intelligence, 1983.

[Yip, 1987] Kenneth Yip. *Extracting Qualitative Dynamics from Numerical Experiments.* AIM 950, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 1987.

[Zahn, 1971] C.T. Zahn. Graph-theoretical methods for detecting and describing gestalt clusters. *IEEE Transactions on Computers*, 20, 1971.