Qualitative Reasoning for Automated Exploration for Chaos

Toyoaki Nishida
Graduate School of Information Science
Nara Institute of Science and Technology
8916-5 Takayama, Ikoma, Nara 630-01, Japan
nishida@is.aist-nara.ac.jp

Abstract

Chaos is ubiquitous in our everyday life and even a simple system may manifest chaotic behaviors. Chaos has been a challenge to the methodology of qualitative reasoning as well as classic science and engineering, due to unpredictability and complexity of behavior.

In this paper, I claim that associating continuous domain with symbolic representation, a basic principle of qualitative reasoning, is vital for automating analysis of chaos, as long as it is properly formalized. As an empirical support to this claim, I present a computer program called PSX3 that can semi-automatically explore for chaotic behavior of a given system of piecewise linear ordinary differential equations with three unknown functions. The power of PSX3 originates from an ability of reasoning about smooth surfaces that implicitly exist in the phase space. PSX3 is implemented using Common Lisp and Mathematica^{TM}.

Introduction

Chaos is ubiquitous in everyday life. Theoretically, it is known that even a simple system may exhibit chaos. Chaos has been a challenge to classic science. The source of difficulty is twofold: unpredictability due to sensitive dependence on initial conditions and complexity of geometry resulting from fractal structure.

Huberman and Struss (Huberman and Struss, 1989) have taken chaos as a serious challenge to the whole methodology of qualitative reasoning, for the existence of chaos severely limits applicability of various filtering techniques and effect of landmark-based representation, that have been popularly used in qualitative reasoning systems. They propose to regard chaos rather a peculiar phenomenon and carefully separate them from commonsense and qualitative reasoning.

In this paper, I show that associating continuous domain with symbolic domain, a basic principle of qualitative reasoning, is vital for analysis of chaos, as long as it is properly formalized. I claim that qualitative reasoning techniques, with adequate generalization, provide a powerful means for automating analysis of chaos.

As an empirical support to this claim, I present a computer program called PSX3 that can semi-automatically explore for chaotic behavior of a certain class of ordinary differential equations (ODEs) and generate structural description of the behavior. The power of PSX3 originates from an ability of reasoning about smooth surfaces that implicitly exist in the phase space. PSX3 is implemented using Common Lisp and Mathematica^{TM}.

In what follows, I take a system of piecewise linear ODEs that exhibits chaotic behavior and discuss issues related to analysis of chaos. Secondly, I describe how PSX3 analyzes chaotic behaviors. Thirdly, I characterize reasoning about smooth surfaces as a generalization of conventional qualitative reasoning techniques. Finally, I discuss the strength and limitation of the current technique and suggest future direction.

A Glimpse of Chaos

In dynamical systems theory (Guckenheimer and Holmes, 1983), it is known that even a simple dynamical system manifests chaotic behavior. For example, Matsumoto and Chua (Matsumoto *et al.*, 1985) have shown that a simple continuous dynamical system consisting of three subsystems of linear ODEs with three unknown functions:

$$\begin{cases} \frac{dx}{dt} = \begin{cases} -1.8x + 6.3y - 2.7 & (x < -1) \\ 0.9x + 6.3y & (-1 \le x \le 1) \\ -1.8x + 6.3y - 2.7 & (1 < x) \end{cases} \\ \frac{dy}{dt} = 0.7x - 0.7y + z \\ \frac{dz}{dt} = -7y \end{cases} \quad (1)$$

exhibits chaotic behavior.

Applied mathematicians study chaotic behavior, by investigating geometric and topological features of trajectories in the phase space spanned by a given set of unknown functions (or *state variables*). In the case of (1), Matsumoto and Chua have found that trajectories (or *orbits*) tend towards a chaotic attractor with a "double scroll" structure, two sheet-like thin rings curled up together into spiral forms.[1] Orbits approach

[1] Roughly, an attractor is a dense collection of orbits that nearby orbits approach as $t \to \infty$. The reader is referred to (Guckenheimer and Holmes, 1983) for complete definition and detailed discussion.

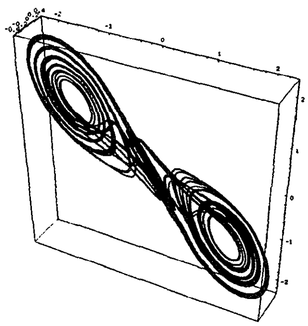


Figure 1: A trajectory of Matsumoto-Chua's equations (1) near the double scroll attractor reported in (Matsumoto *et al.*, 1985)

the attractor as time goes and manifest chaotic behavior as they irregularly transit between the two rings, as illustrated by a sample trajectory shown in Figure 1.

Investigation of chaos can be roughly divided into three stages. The first stage is search for a condition under which a given dynamical system exhibits an interesting behavior. The second stage is qualitative analysis whose purpose is to identify the structure of the behavior in focus. The third stage is quantitative analysis, by applying known measurements such as Lyapunov exponents or fractal dimension (Moon, 1987) to obtain quantitative support of findings.

Generally, as one proceeds to the later stages, the more well-studied, sophisticated mathematical techniques are available, some of which have been automated. In contrast, the earlier stages depend on more general capability of humans, such as visual perception and spatial reasoning, as pointed out in (Yip, 1991b). Even mathematicians have to go through trial and errors in early stages of analysis. It is worth developing a computational model of earlier stages of dynamical systems analysis, for (a) automated search does help both applied mathematicians in search for interesting phenomena and engineers who do not have ample knowledge about chaos, and (b) analysis and modeling of experts' intellectual behavior as an integration of various cognitive processes are an important subject of AI research.

Critical issues here are, (a) high-level representation of information such as topology and geometry of orbits, and (b) its application to intelligently controlling numerical analysis. In (Nishida, 1993), I proposed flow mappings as a solution to the first issue. In this papers, I show that qualitative analysis of chaos can in fact be automated, using flow mappings as central representation. I have implemented my theory as a program called PSX3. In the next section, I describe how PSX3 works. Then, I characterize my theory as an advanced formalization of a qualitative reasoning principle.

Exploration for Chaos by PSX3

PSX3 takes a specification of system of piecewise linear ODEs with three unknown functions and a region of analysis, and produces a qualitative and quantitative description of the behavior ranging from the regular to the chaotic, which can be used as a prescription for quantitative measurements.

The procedure incorporated into PSX3 is roughly divided into local analysis, global analysis, and prescription generation for detailed quantitative measurements.

Local Analysis

In local analysis, PSX3 classifies trajectories in each intersection (*cell*) of region of analysis and a linear region, into coherent bundles of orbit intervals. Roughly, a coherent bundle of orbit intervals is a collection of orbit intervals from/to a singly connected region of a cell surface or the same fixed point (Nishida, 1993). Figure 2a shows[2] how PSX3 partitions the collection of orbit intervals contained in a cell `cell-1`: $(-3 \leq x \leq -1, -2 \leq y \leq 2, -3 \leq z \leq 3)$ made by intersecting a linear region $(x \leq -1)$ and a given region of analysis $(-3 \leq x \leq 3, -2 \leq y \leq 2, -3 \leq z \leq 3)$. Figure 2b shows subsidiary partitioning on the surface of the cell, where each region of the cell surface is identified by the attached number. Figure 2c shows several bundles of orbits that PSX3 has identified in `cell-1`. Orbits running through regions #83, #95, and #87 on the right side plane $(x = -1)$ of `cell-1` are qualitatively different with respect to this cell in the sense that they are running out of the cell from different sides of the cell (#77 on top plane $(z = 3)$, #66 on the rear plane $(x = -3)$, and #94 on the right side $(x = -1)$, respectively).

PSX3 represents the result of partitioning as

- $\{\phi_i\}$, where ϕ_i stands for either a fixed point or a two-dimensional region of the cell surface; and
- a set of *flow mappings* $\{\phi_i \rightarrow \phi_j\}$ which represents the structure of *flow* (the collection of orbits), where a flow mapping $\phi_i \rightarrow \phi_j$ means that ϕ_i (a two-dimensional region on the cell surface or a fixed point) is mapped to ϕ_j (another two-dimensional region or a fixed point) by the flow underlying the cell.

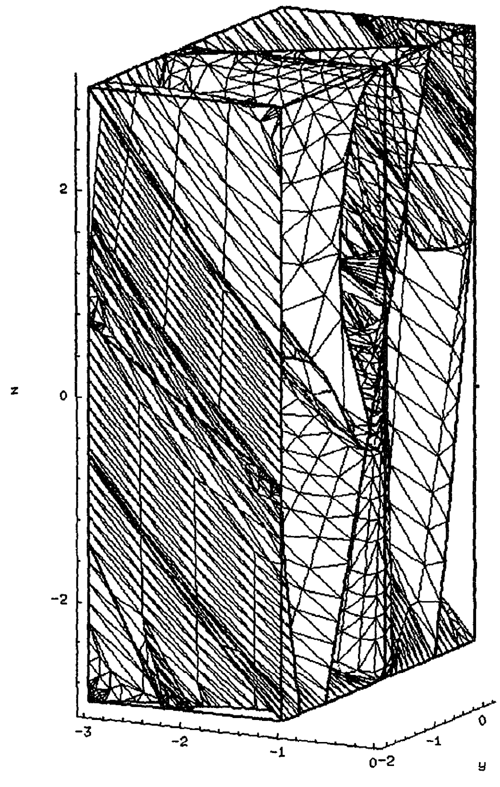
Figure 2d shows a set of flow mappings for `cell-1`. Figure 3 shows results of local analysis for the remaining two cells in the given region of analysis. In order to obtain the partitioning, nontrivial amount of qualitative and quantitative analysis is needed as described in (Nishida, 1993).

Global Analysis

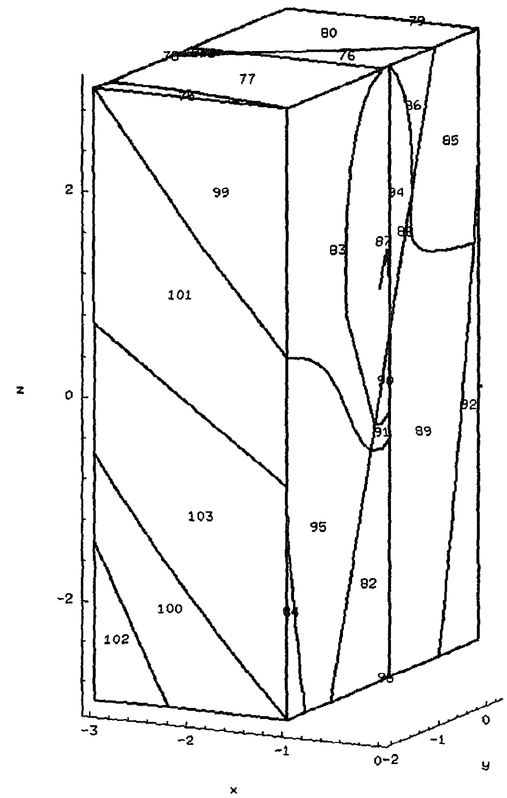
After it has generated a set of flow mappings for each cell, PSX3 proceeds to global analysis and takes a col-

[2] Although each surface partitioning bundles of orbit intervals is approximated by triangulation, they are only for demonstration purpose and not used for reasoning. In order to reason about the flow, PSX only refers to regions on the cell surface such as region #95 in Figure 2b.

(a) partitioning of the flow in `cell-1`:
$(-3 \leq x \leq -1, -2 \leq y \leq 2, -3 \leq z \leq 3)$
by a collection of smooth surfaces

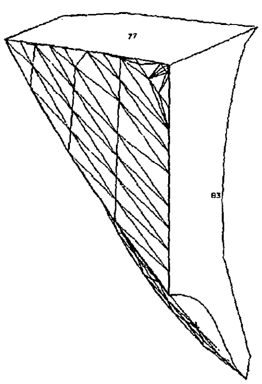


(b) resulting partitioning on the surface of `cell-1`

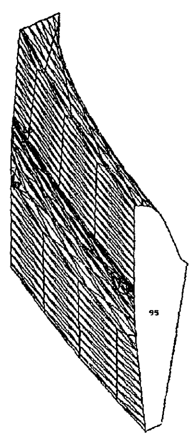


(c) bundles of orbit intervals

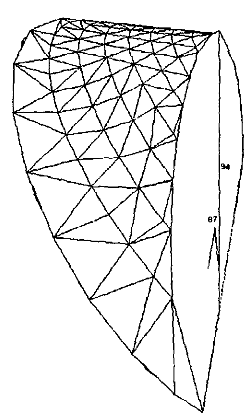
(1) #83



(2) #95



(3) #87

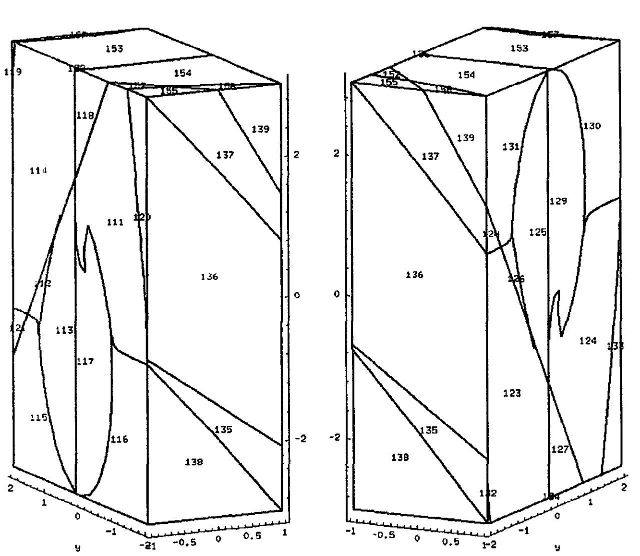


(d) representation of the flow by flow mappings

61 $\to$ 70
71 $\to$ 73
65 $\to$ 78
63 $\to$ 81
82 $\to$ 72
83 $\to$ 77
80 $\to$ 85
76 $\to$ 86
62 $\to$ 89
90 $\to$ 88
91 $\to$ 74
87 $\to$ 94
95 $\to$ 66
97 $\to$ 92
79 $\to$ 98
99 $\to$ 75
101 $\to$ 68
84 $\to$ 103
104 $\to$ 100
105 $\to$ 102
96 $\to$ 106
107 $\to$ 67
69 $\to$ 108
109 $\to$ 64
93 $\to$ 110

Figure 2: Local analysis PSX has produced for `cell-1`

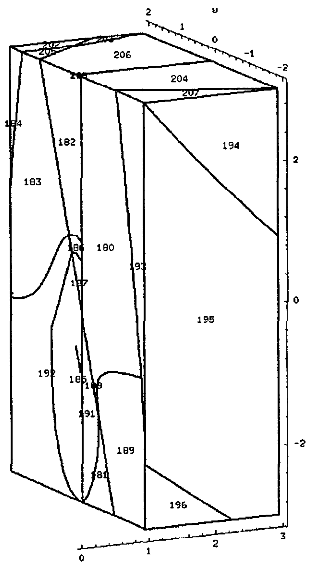
(a) partitioning the surface of cell-2: $(-1 \leq x \leq 1, -2 \leq y \leq 2, -3 \leq z \leq 3)$



(b) flow mappings for cell-2

113 → 117
123 → 111
114 → 124
126 → 118
112 → 127
125 → 129
132 → 135
136 → 120
140 → 116
121 → 141
115 → 142
144 → 134
145 → 138
119 → 147
148 → 133
149 → 143
151 → 146
128 → 152
153 → 130
131 → 154
137 → 155
156 → 122
157 → 150
139 → 158

(c) partitioning of the surface of cell-3: $(1 \leq x \leq 3, -2 \leq y \leq 2, -3 \leq z \leq 3)$



(d) flow mappings for cell-3:

168 → 167
174 → 159
169 → 177
179 → 163
170 → 180
161 → 181
182 → 178
183 → 173
186 → 162
187 → 188
166 → 189
185 → 191
192 → 160
195 → 193
165 → 196
197 → 175
200 → 164
184 → 201
202 → 198
203 → 199
176 → 204
205 → 172
206 → 171
194 → 207
190 → 208

Figure 3: Result of local analysis for `cell-2` and `cell-3`

lection of flow mappings for each cell and looks for minimal sets of mutually transitioning flow mappings defined below.

Definition 1 *Given a set of flow mappings* $\{\phi_i \rightarrow \phi_j\}$ *and intersection relations* $\{\langle\phi_m, \phi_n\rangle \mid \phi_m \cap \phi_n \neq \text{empty}\}$, *an extended set of flow mappings* $\{\phi_p \rightarrow^* \phi_q\}$ *is defined as follows:*

(1) $\phi_i \rightarrow \phi_j \Rightarrow \phi_i \rightarrow^* \phi_j$
(2) $\phi_i \rightarrow \phi_j \wedge \phi_k \rightarrow \phi_m \wedge \phi_j \cap \phi_k \neq \text{empty} \Rightarrow \phi_i \rightarrow^* \phi_m$.

Definition 2 *Given a set of flow mappings* $\{\phi_i \rightarrow \phi_j\}$ *and intersection relations* $\{\langle\phi_m, \phi_n\rangle \mid \phi_m \cap \phi_n \neq \text{empty}\}$, $\Phi = \{\phi_i \rightarrow \phi_j\}$ *is a minimal set of mutually transitioning flow mappings iff*

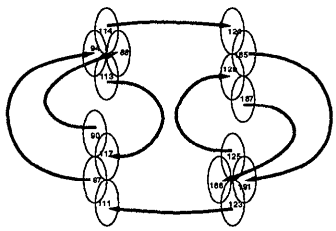
$$\forall(\phi_i \rightarrow \phi_j) \in \Phi\ [\phi_j \rightarrow^* \phi_i].$$

An algorithm for searching for a minimal set of mutually transitioning flow mappings is implemented using a simple graph search algorithm.

In the case of Matsumoto-Chua equations, PSX3 has found a minimal set of mutually transitioning flow mappings, as shown in Figure 4a. The predicted set of mutually transitioning flow mappings suggests that it is likely that the attractor is a composition of several recurrnt cycles, rather than a single cycle.

As long as the phase space is partitioned into cells in such a way that each non-point attractor intersects more than one cell, mutually transitioning flow mappings tell the approximate location and structure of the attractor. In other words, the surface of the union of bundles of orbit intervals corresponding to a minimal set of mutually transitioning flow mappings serves an envelope wrapping a candidate of a non-point attractor, as shown in Figure 4b.

(a) structural analysis



(Transition 90 → 88 and 187 → 188 will eventually be removed at the prescription generation stage)

(b) envelope of the candidate of an attractor

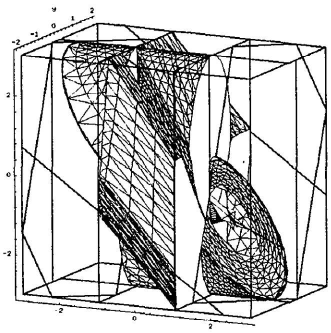


Figure 4: Result of global analysis produced by PSX3

Prescription Generation for Quantitative Measurement

Finally, PSX3 assembles a structured report on its findings as a prescription for further quantitative analysis.

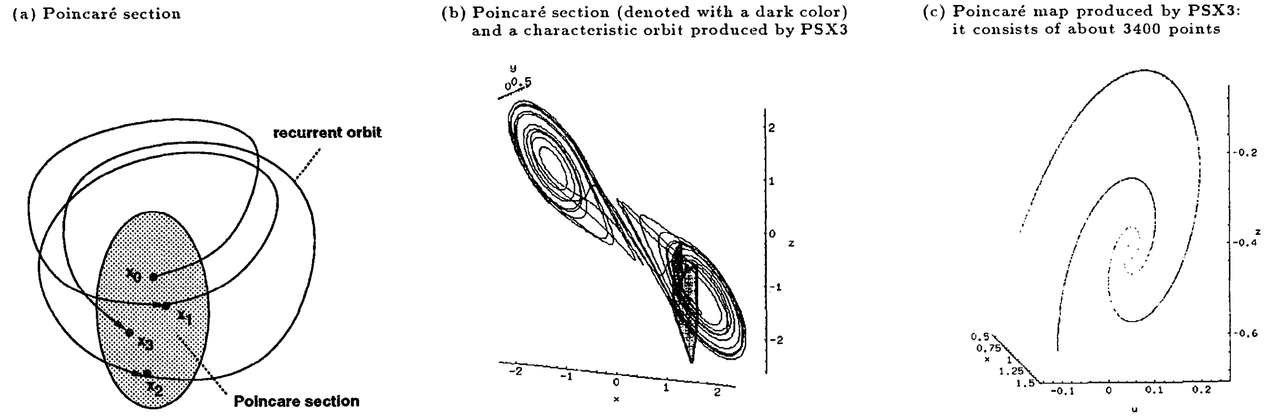


Figure 5: Prescription generation

A critical information added at this stage is a Poincaré section and mappings, which are popularly used by applied mathematicians as a basis of quantitative analysis. Poincaré section is a plane that cuts across a recurrent orbit, as illustrated in Figure 5a. A Poincaré map is represented as $x(t) = x_t \mapsto x(t+1) = x_{t+1}$, where x_i is the i-th point at which the recurrent orbit penetrates the Poincaré section. It is possible to make preliminary diagnosis of a given recurrent orbit by examining the shape of a Poincaré map as a collection of points. The more diverged and fuzzier the Poincaré map becomes, the more likely the orbit in question is chaotic.[3] PSX3 automatically generates a Poincaré section by analyzing critical section of a minimal set of mutually transitioning flow mappings. PSX3 determines where to start tracking an orbit, by intersecting the proposed Poincaré section and the internal region delimited by an envelope wrapping a candidate of a non-point attractor. Figure 5b shows a Poincaré section and an orbit that PSX3 has actually proposed and computed. Figure 5c shows a resulting Poincaré map, which conforms to the one reported in (Matsumoto *et al.*, 1985).

Reasoning about Smooth Surfaces

One of the fundamental principles of qualitative reasoning is associating continuous domain with symbolic representation by aggregating coherent objects. The techniques embodied by PSX3 fit this schema. PSX3 aggregates orbit intervals (continuous geometric objects) into bundles based on coherency relations defined with respect to a cell, associates each bundle with a flow mapping (symbolic representation), and uses the resulting set of flow mappings to set up a plan for more detailed quantitative measurements. PSX3 integrates qualitative and quantitative analysis for doing these. In particular, the success of PSX3 can be attributed to its ability of constructing representation of smooth surfaces implicitly existing in the phase space, by intelligently controlling numeric and symbolic computation. PSX3 combines numerical analysis and reasoning about smooth surfaces to establish delimiting surfaces in a cell. For example, when encountered with a set of fragmentary observations, PSX3 will consult a library of smooth surface interaction patterns (Nishida, 1993), set up a hypothesis about the underlying geometric structure, and try to verify it in subsequent analysis.

Techniques incorporated into PSX3 can be regarded as an extension of conventional qualitative reasoning techniques in which real numbers are associated with a set of symbols, such as $\{-, 0, +\}$, by a set of landmarks. Although it has made various AI techniques such as GDE (de Kleer and Williams, 1987) applicable to continuous domains, the approach has flawed in several ways, especially in the case of reasoning about chaos.

In contrast, my formalization is based on partitioning the continuous domain by smooth surfaces. Reasoning about smooth surfaces is more sophisticated and powerful, while more computationally expensive.[4]

Related Work and Discussion

This work is considered to be an effort of developing a computational model of dynamical systems theory (Guckenheimer and Holmes, 1983). Previous work in this direction involves: POINCARE (Sacks, 1991), PSX2NL (Nishida and Doshita, 1991; Nishida *et al.*, 1991), Kalagnanam's system (Kalagnanam, 1991), MAPS (Zhao, 1993), and Perfect Moment(Bradley, 1992). KAM (Yip, 1991b) is one of the frontier work, though it is for discrete systems (difference equations), as opposed to continuous systems addressed in this paper. Unfortunately, the techniques used in these systems except MAPS (Zhao, 1993) and Perfect Moment

[3] I have not yet automated the evaluation of Poincaré map. Automation could be possible using techniques pioneered by Yip (Yip, 1991a).

[4] However, it should be noted that the amount of redundant computation is significantly reduced for the sake of its expressive power.

(Bradley, 1992) are severely limited to two-dimensional flows whose geometry is significantly simpler than those in three dimensional phase spaces. Indeed, continuous flow in two-dimensional Euclidean phase space never exhibits chaos.

In MAPS, the flow pipe model is used. A flow pipe represents a homotopy equivalence class of orbits and hence is essentially equivalent to a bundle of orbit intervals,[5] despite some differences at the implementation level.[6] MAPS constructs the information structure in a bottom-up fashion. Unfortunately, this is not enough for exploring for chaos. A guideline is needed to plan numerical computation for constructing a critical flow pipe which may serve as an envelope of a chaotic attractor. The method incorporated into PSX3 is a kind of exhaustive search. PSX3 partitions a given region of analysis into a set of bundles of orbit intervals and searches for a candidate of an envelope of a chaotic attractor. Perfect Moment handles chaos with adaptive grids, which causes several problems due to discretization.

Current implementation of PSX3 is limited in a couple of ways. Firstly, PSX3 is specialized to flow in three-dimensional phase space. This is a rather serious limitation, for most applications of practical interest are defined in higher dimensional phase spaces. The limitation might be overridden either by developing a method of extracting a lower dimensional dynamics from a given system of ODEs, or by taking a direct step towards generalizing the current method. The latter direction requires substantial efforts on reducing computational cost.

Secondly, the subsystem for reasoning about smooth surfaces is "hard wired": there is no separation between a knowledge-base and a general reasoning engine. In developing PSX3, I have manually classified geometric interactions among smooth surfaces and hand-crafted a reasoning algorithm. However, the same approach is intractable for more general classes of problems, namely hyper-surfaces in n-dimensional space. An interesting open problem is to develop a meta system which can generate a computational theory of reasoning about smooth surfaces in general n-dimensional spaces.

Conclusion

In this paper, I have shown that automated analysis of chaos can in fact be possible, and I have described PSX3 as an empirical support of the claim. The power of PSX3 originates from an ability of reasoning about smooth surfaces that implicitly exist in the phase space.

[5] These two models were independently developed and widely published in the summer of 1991.

[6] The flow pipe model is implemented using (a) polyhedral approximation for representing the shape of flow pipes and (b) relational graph representation for representing the topology of the phase portrait.

Acknowledgments

I am grateful to Feng Zhao for useful comments.

References

Bradley, Elizabeth 1992. *Taming Chaotic Circuits.* Ph.D. Dissertation, M.I.T.

de Kleer, Johan and Williams, Brian C. 1987. Diagnosing multiple faults. *Artificial Intelligence* 32:97–130.

Guckenheimer, John and Holmes, Philip 1983. *Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields.* Springer-Verlag.

Huberman, Bernard A. and Struss, Peter 1989. Chaos, qualitative reasoning and the predictability problem. presented at 3rd International Workshop on Qualitative Physics, Stanford.

Kalagnanam, Jayant 1991. Integration of symbolic and numeric methods for qualitative reasoning. Technical Report CMU-EPP-1991-01-01, Engineering and Public Policy, CMU.

Matsumoto, Takashi; Chua, Leon O.; and Komuro, Motomasa 1985. The double scroll. *IEEE Transactions on Ciruits and Systems* CAS-32(8):798–818.

Moon, Francis C. 1987. *Chaotic Vibrations — An Introduction for Applied Scientists and Engineers.* John Wiley & Sons.

Nishida, Toyoaki and Doshita, Shuji 1991. A geometric approach to total envisioning. In *Proceedings IJCAI-91.* 1150–1155.

Nishida, Toyoaki; Mizutani, Kenji; Kubota, Atsushi; and Doshita, Shuji 1991. Automated phase portrait analysis by integrating qualitative and quantitative analysis. In *Proceedings AAAI-91.* 811–816.

Nishida, Toyoaki 1993. Generating quasi-symbolic representation of three-dimensional flow. In *Proceedings AAAI-93.* 554–559.

Sacks, Elisha P. 1991. Automatic analysis of one-parameter planar ordinary differential equations by intelligent numeric simulation. *Artificial Intelligence* 48:27–56.

Yip, Kenneth Man-kam 1991a. *KAM — A System for Intelligently Guiding Numerical Experimentation by Computer.* The MIT Press.

Yip, Kenneth Man-kam 1991b. Understanding complex dynamics by visual and symbolic reasoning. *Artificial Intelligence* 51(1-3):179–222.

Zhao, Feng 1993. Computational dynamics: Modeling and visualizing trajectory flows in phase space. *Annals of Mathematics and Artificial Intelligence* 8:285–300.