Hybrid phase-portrait analysis in automated system identification

Matthew Easley and Elizabeth Bradley*
University of Colorado at Boulder
Department of Computer Science
Boulder, Colorado 80309–0430
{easley,lizb}@cs.colorado.edu

Abstract

We present a new representation for hybrid phase-portrait analysis, called the *qualitative state/ parameter space*, wherein a physical system's dynamics are classified into discrete regions of qualitatively identical behavior. This classification is performed using automated phase-portrait analysis techniques and an abstraction scheme from the dynamical systems literature called *cell dynamics*. This hybrid representation is useful for input-output modeling of dynamical systems; among other things, it is a very natural way to reason about multiple sets of observations over a given system. Issues about the transitions between these discrete regions are analogous to many of the issues found in the hybrid systems literature. Our representation is an essential element of the input-output modeling component of the PRET automated system identification program.

Hybrid Qualitative Phase-Portrait Knowledge

One of the goals of the qualitative reasoning (QR) community(Forbus 1997) is to abstract specific instances of behavior into more-general descriptions of a system. An 80kg adult bouncing on the end of a bungee cord, for instance, will produce a different time series from a 50kg child, but both produce similar damped oscillatory responses. Reasoning about these two behaviors in their time series form can be difficult, as it requires detailed examination of the amplitude decay rate of and the phase shift between two decaying sinusoids. The phase-space representation, which suppresses the time variable and plots position versus velocity, brings out the similarity between these two behaviors in a very clear way. Both bungee jumps, for example, manifest on a phase-space plot as similar decaying spirals. Automated phase-portrait analysis techniques(Bradley 1995; Yip 1991; Zhao 1993), which combine ideas from computer-vision, dynamical systems, discrete mathematics, and artificial intelligence, generate qualitative descriptions that effectively capture this kind of information.

A discretized version of the phase-space representation can abstract away many low-level details about the dynamics of a system while preserving its important qualitative properties. The cell-to-cell-mapping formalism(Hsu 1987), for instance, discretizes a set of n-dimensional state vectors onto an n-dimensional mesh of uniform boxes or *cells*. The circular phase-space trajectory in Fig. 1(a), for example, — a sequence of two-vectors of floating-point numbers — can be represented as the following *cell sequence*

$$[\ldots(0,0)(1,0)(2,0)(3,0)(4,0)(4,1)(4,2)(4,3)\ldots]$$

Because multiple trajectory points are mapped into each cell, this discretized representation of the dynamics is significantly more compact than the original series of floating-point numbers and therefore much easier to work with. Using this representation, the dynamics of a trajectory can be quickly and qualitatively classified using simple geometric heuristics — in this case as a *limit cycle*. Part (b) of the figure shows a different trajectory with identical topology; this, too, would be classified as a limit cycle by the cell dynamics algorithm. A key concept here is that a set of geometrically different and yet qualitatively similar trajectories — an "equivalence class" with respect to some important dynamical property — is classified as a single coherent group of phase-space portraits.

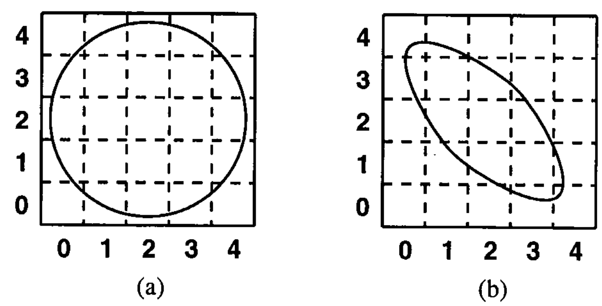

Figure 1: Identifying a *limit cycle* with phase-portrait analysis.

*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.

Dynamical systems can be extremely complicated; attempting to understand one by analyzing a single behavior instance is generally inadequate. Rather, one must vary a system's inputs and control parameters and study the change in the response. Even in one-parameter systems, however, this procedure can be difficult(Abelson 1990); as the parameter is varied, the behavior may vary smoothly in some ranges and then change abruptly ("bifurcate") at critical parameter values. A thorough representation of this behavior, then, requires a "stack" of phase portraits: at least one for each interesting and distinct range of parameter values.

Similar kinds of problems arise in the hybrid systems literature. Hybrid modeling techniques describe continuous nonlinear behavior using an ontology of piecewise-continuous regimes and discrete inter-regime transitions(Mosterman, Zhao, & Biswas 1998). In this representation, if a control parameter is changed or a state variable moves into a prescribed phase-space region, a *transition function* moves or "jumps" the hybrid model into that new operating regime and simultaneously invokes the appropriate governing equations. If one attempts to use this representation to capture the type of complex behavior described in the previous paragraph, however, the requirement that different operating regimes occupy physically distinct phase-space regions poses some serious problems. The same phase-space region may exhibit radically different behaviors for different control parameter values, and the simple hybrid system representation cannot handle this.

An example from (Branicky, Borkar, & Mitter 1994) makes this clearer. A manual transmission can be modeled by:

$$\frac{d}{dt}x_1(t) = \dot{x_1} = x_2 \quad ; \quad \dot{x_2} = \frac{[-f(x_2) + throttle]}{1 + gear}$$

where x_1 is the velocity, x_2 is the engine RPM, f is some function of x_2, *throttle* position varies between 0 and 1, and $gear \in \{1, 2, 3, 4\}$. It is difficult to build a model like this from observed behavior — say, velocity x_1 — because it is impossible, without knowing the engine RPM, to distinguish whether the car is in first or second gear. At 15 MPH the engine could be at 3500 RPM in first gear or 2000 RPM in second gear, and an external observer would not be able to distinguish between these two states. Even if the engine RPM could be measured, model building might still be impossible, as the *throttle* position may depend upon loading factors (e.g., pulling a heavy load or going down a hill). The problem is that there is a *family* of phase-space portraits for each gear, parameterized by velocity, RPM, and throttle position. If only a subset of these parameters are observable — say, velocity and RPM — then the families of phase portraits for each gear can overlap, which makes identification of the actual system very difficult.

In this paper, we propose an approach that solves some of these problems. In particular, we use a combined *state/parameter space* and decompose into discrete regions, each associated with an equivalence class of dynamical behaviors, derived qualitatively using Hsu's cell dynamics formalism. This collection of discrete regions describes the behavior of the system in a uniquely powerful way. Because each trajectory is effectively equivalent, in a well-known sense, to all the other trajectories in the same region, one can describe the behavior in that region in a significantly simpler way, which results in ease of analysis — and great computational savings.

Consider, for example, the driven pendulum system described by the ODE model

$$\ddot{\theta}(t) + \frac{\beta}{m}\dot{\theta}(t) + \frac{g}{l}\sin\theta(t) = \frac{\gamma}{ml}\sin\alpha t$$

with mass (m), arm length (l), gravity constant (g), damping factor (β), drive amplitude (γ) and drive frequency (α). m, l, g and β are constants; the state variables of this system are θ and $\omega = \dot{\theta}$. In many experimental setups, the drive amplitude and/or frequency are controllable: these are the "control parameters" of the system. The behavior of this apparently simple device is really quite complicated and interesting. For low drive frequencies, it has a single stable fixed point; as the drive frequency is raised, the attractor undergoes a series of bifurcations between chaotic and periodic behavior. These bifurcations do not, however, necessarily cause the attractor to *move*. That is, the qualitative behavior of the system changes and the operating regime — in the standard hybrid systems view — does not. Traditional analysis of this system would involve constructing phase portraits of the system, like the one shown in Fig. 1, at closely spaced control parameter values across some interesting range. The standard hybrid

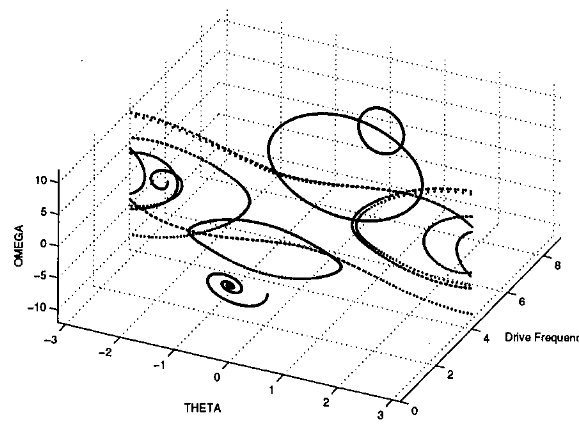

Figure 2: The state/parameter (S/P) space portrait of the driven pendulum: a parameterized collection of phase portraits of the device at various drive frequencies. Each (θ, ω) slice of this *S/P-space portrait* is a standard phase portrait at one parameter value.

representation does not handle this smoothly, as the operating regimes involved are not distinct. If, however, one adds an axis to the space, most of these problems vanish. Fig. 2 describes the behavior of the driven pendulum in this new *state/parameter space* (S/P space).

Each θ, ω slice of this plot is a phase portrait; the control parameter varies along the *Drive_Frequency* axis.

Combining this state/parameter space idea with the qualitative abstraction of Hsu's cell dynamics, yields the new *qualitative state/phase space* (QS/P space) representation that is one of the topics of this paper. A QS/P-space portrait of the driven pendulum is shown in Fig. 3. This representation is similar to the state/parameter space portrait shown in the previous figure, but it groups qualitatively similar behaviors into equivalence classes, and then uses those groupings to define the boundaries of qualitatively distinct regions of state/parameter space.

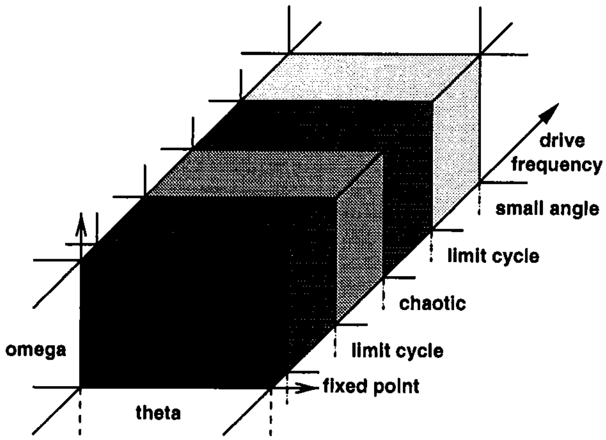

Figure 3: The *qualitative state/phase-space* (QS/P-space) portrait of the driven pendulum. This is an abstraction of the state/parameter space, wherein qualitatively similar behaviors are grouped into equivalence classes and those groupings are used to define the boundaries of qualitatively distinct regions of state/parameter space.

This qualitative state/ parameter-space representation is an extremely powerful modeling tool. One can use it in the traditional hybrid-systems approach, identifying each operating regime, creating a separate model in each, and then using a finite-state machine to model transitions between them. More importantly, however, the QS/P-space representation lets the model builder leverage the knowledge that its regions — e.g., the five slabs in Fig. 3 — all describe the behavior of the *same system*, at different parameter values. This is exactly the type of knowledge that one needs for *input/output modeling*, in which one attempts to learn more about a system by changing its inputs and observing the results. The remainder of this paper expands upon this, describing how the QS/P-space representation can aid input/output modeling of dynamical systems.

Hybrid Phase-Portrait Analysis in Model Building

System identification (SID), the process of inferring an internal ordinary differential equation (ODE) model from external observations of a system, is an ideal test case for hybrid phase-portrait analysis using the QS/P space. The computer program PRET(Bradley & Stolle 1996), a QR modeling tool that automates the SID process by building an AI layer around a set of traditional system identification techniques, constructs ODE models of lumped-parameter continuous-time nonlinear dynamic systems. It first uses domain knowledge to combine model fragments into ODEs, then observes the target system using sensors, and finally tests those ODEs against the sensor data using a body of mathematical knowledge encoded in first-order logic(Stolle & Bradley 1998). In order to interact with the target system, PRET makes use of sensors and actuators, as shown in Fig. 4. Distilling available sensor information into qualitative form is reasonably straightforward(Bradley & Easley 1998), but reasoning about the information so derived is subtle and difficult. If the target system has 34 state variables, for example, and one can only measure one of those 34 signals, it would appear that the conclusions that one can draw from the sensor data are fundamentally limited. This is control theory's *observer problem*: the task of inferring the internal state of a system solely from observations of its outputs. A general solution to this is an open problem; PRET uses *delay-coordinate embedding* to effect a partial solution. Reasoning about actuators is even harder because of the nonlinear control theory that is involved. Determining what experiments one can perform from the system's present state involves complicated reasoning about *reachability*. That is, given a black-box system, a partial measurement of its current state, some knowledge about the available actuators, and some preliminary ideas about a candidate model, PRET must reason effectively about what experiments are (1) possible and (2) useful. This is a difficult, open problem for nonlinear systems, and the hybrid view of qualitative phase-portrait analysis described in the previous section plays a critical role in the knowledge representation and reasoning involved in PRET's solutions to this input/output modeling problem.

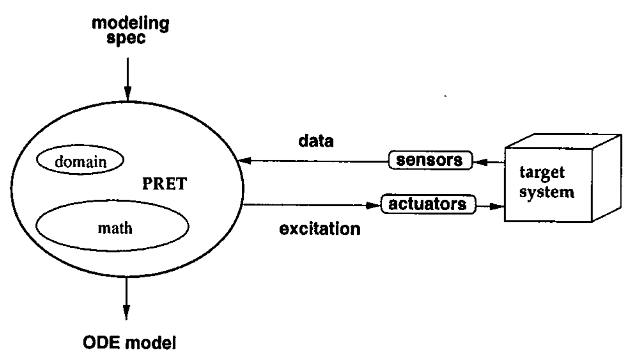

Figure 4: PRET uses sensors and actuators to interact with target systems in an *input-output* approach to dynamical system modeling.

Input-Output Modeling in PRET

The goal of input-output modeling is to apply a test input to a system, analyze the results, and learn something useful from the cause/effect pair. In this section, we show how PRET reasons about this process.

PRET's knowledge representation and reasoning framework is designed specifically for model building. PRET follows a generate-and-test paradigm, using domain knowledge (e.g., force balances) to combine user-specified model fragments ("hypotheses") into ODE models, and then testing those models against the known behavior of the system ("observations") using ODE theory rules like "the divergence of a dissipative system is negative." I/O modeling via the QS/P representation contributes to this process in a variety of ways. Firstly, it allows PRET to reason effectively about test inputs; a good test input excites the behavior in a useful but not overwhelming way, and choosing such an input is nontrivial. The I/O modeling techniques described in the previous section also allow PRET to reason about sensible hypothesis combinations — a process without which the generate phase would be reduced to blind enumeration of an exponential number of candidate models. Finally, qualitative I/O modeling techniques help PRET reason about state variables and observations — information whose sole source would otherwise be the user.

The "input" part of PRET's input-output reasoning takes place in the *intelligent sensor data analyzer*(Bradley & Easley 1998). This module first reconstructs the hidden dynamics from the sensor and then analyzes the results using geometric reasoning. The first of these two steps is necessary because fully *observable* systems, in which all of a system's state variables can be measured, are rare in normal engineering practice. Often, some of the state variables are either physically inaccessible or cannot be measured with available sensors. Delay-coordinate embedding(Abarbanel 1995), PRET's solution to this problem, creates an m-dimensional *reconstruction-space* vector from m time-delayed samples of data from a single sensor. The central idea is that the reconstruction-space dynamics and the true (unobserved) state-space dynamics are topologically identical. This provides a partial solution to the observer problem, as a phase portrait reconstructed from a single sensor is qualitatively identical to the true multidimensional dynamics of the system[1]. Given a reconstructed phase portrait of the system's dynamics, the intelligent sensor data analyzer's second phase distills out its qualitative properties using the cell dynamics paradigm discussed in conjunction with Fig. 1. The results of reconstructing and analyzing the sensor data are a set of qualitative observations similar to those a human engineer would make about the system.

Reasoning about actuators is much more difficult, so the development of PRET's *intelligent actuator controller* has been slow. The problem lies in the inherent difference between passive and active modeling. It is easy to recognize damped oscillations in sensor data without knowing anything about the system or the sensor, but using an actuator requires a lot of knowledge about both. Different actuators can have different properties (range, resolution, response time, etc.); consider the difference between the time constants involved in turning off a burner on a gas or an electric stove. Identical actuators can affect systems in radically different ways. Step, impulse, and random signals are common choices for test inputs in linear system analysis, but they elicit tremendously complicated responses from nonlinear systems, making output analysis very difficult. In nonlinear systems analysis, one typically applies *constant* inputs and ignores any transients. Reasoning about these issues involves a multitude of questions; e.g., Should the drive be a high frequency sine wave or a low one? A current or voltage source? These are the kinds of questions that we hope to use the qualitative state/phase space representation to answer.

Deciding how to use an actuator is only the first part of the problem. Experimenters must also consider the set of possible states — those that are reachable from the existing state with the available control input. The system state that one wishes to explore simply may not be *reachable* from the existing state with the available actuators. Finally, effective input-output modeling requires reasoning about *useful* experiments: those that increase one's knowledge about the target system in a productive way. The ultimate goal of PRET's intelligent sensor/actuator control module is to find and exploit the overlap between these sets of *useful* and *possible* experiments.

To perform these tasks, PRET uses hybrid system techniques to reason about multiple sets of observations about a system. Using the QS/P paradigm developed earlier, we treat changes of the control parameter — those that take the system into a new regime — as discrete transitions between operating regimes. This process, known in the dynamical systems literature as *bifurcation analysis*, involves varying a parameter to drive the system into different regimes and then analyzing these behaviors. To use this information to fit a model to a system, one simply matches up the boundaries between regimes[2]. The following example demonstrates how these ideas help PRET manage its sensors and actuators.

Table 1 displays the ODEs that describe the behavior of the driven pendulum in each of the five qualitative state/parameter-space regions shown in Fig. 3. Although four of these five ODEs are different, all five are,

[1] It also allows PRET to estimate the number of state variables in a system.

[2] This is similar to hybrid systems analysis procedures that reason about *controlled switching/jumping* or externally induced discontinuities(Branicky, Borkar, & Mitter 1994). PRET makes no use of *autonomous switching/jumping* (internally induced) discontinuities because the ODE theorems that are the source of its power and generality constrain it to continuously differentiable input signals.

in reality, instances of a single ODE that accounts for the physical behavior across the whole parameter range. PRET's goal is to identify these distinct QS/P-space regions, build a model for each regime, attempt to reconcile models across all the regimes, and finally unify this collection of ODEs into a single, globally valid model. This process continues though all regimes in the region of interest. In the driven pendulum, PRET analyzes the system in the small-angle regime[3] and discovers the model $\ddot{\theta}(t) = -\frac{g}{l}\theta(t)$. If it then analyzes the system in a limit cycle regime with a larger angle, the small angle solution will no longer hold, forcing a new model search, which would yield the model $\ddot{\theta}(t) = -\frac{g}{l}\sin\theta(t)$. PRET would then try to reconcile the two models, applying both of them in both regimes. Since $\ddot{\theta}(t) = -\frac{g}{l}\theta(t)$ is a special case of $\ddot{\theta}(t) = -\frac{g}{l}\sin\theta(t)$, the former will hold in only one of the two, whereas the latter will hold in both, so PRET discards the $\ddot{\theta}(t) = -\frac{g}{l}\theta(t)$ model and goes on to the next regime, repeating the model building/unification process. Once PRET finds a single model that accounts for all observed behavior in all regimes across the range of interest, its task is complete. Such a model may not, of course, exist; a system may be governed by completely different physics in different regimes, and no single ODE may be able to account for this kind of behavior. In this case, the models in the different regimes would be mutually exclusive, and PRET would be unable to unify them into a single ODE, and so it would simply return the list of regimes, models, and transitions — which is exactly the standard "hybrid model" of the system.

Drive	ODE
None	$\ddot{\theta}(t) = -\frac{\beta}{m}\dot{\theta}(t) - \frac{g}{l}\sin\theta(t)$
Low	$\ddot{\theta}(t) = -\frac{g}{l}\sin\theta(t)$
Med.	$\ddot{\theta}(t) + \frac{\beta}{m}\dot{\theta}(t) + \frac{g}{l}\sin\theta(t) = \frac{\gamma}{ml}\sin\alpha t$
High	$\ddot{\theta}(t) = -\frac{g}{l}\sin\theta(t)$
V. High	$\ddot{\theta}(t) = -\frac{g}{l}\theta(t)$

Table 1: Valid models of the driven pendulum in different behavioral regimes.

Relationship to Related Work

Most of the work in the AI/QR modeling community focuses on qualitative models by combining a set of descriptions of state into higher-level abstractions or *qualitative states*(de Kleer & Brown 1984; Forbus 1987). Many tools also reason about equations at varying levels of abstraction from *qualitative* differential equations (QDEs) in QSIM(Kuipers 1986) to standard ODEs in PRET. PRET's approach differs from many of these other tools in that it works with noisy, incomplete sensor data from real-world systems, and attempts not to "discover" the underlying physics, but rather to find the simplest ODE that can account for the given observation.

The QR research that is most closely related to PRET is (Capelo, Ironi, & Tentoni 1993), who build ODE models by evaluating time series using qualitative reasoning techniques and then use a parameter estimator to match the resulting model with a given observed system. Capelo *et al.*'s modeling tool selects models from a set of pre-enumerated solutions in a very specific domain (linear visco-elastic systems). PRET is much more general; it works on *all* linear *and nonlinear* lumped-parameter continuous-time ODEs and uses *dynamic* model generation to handle arbitrary devices and connection topologies.

The idea of building a more-accurate model in a piecemeal fashion is similar to work done in the hybrid systems community. The **dstool** simulation tool, for instance, takes a set of differential equations and their operating regimes and uses numerical integration techniques to display a phase portrait of that system(Back, Guckenheimer, & Myers 1992). **dstool** has been used to analyze a hopping robot with four behavior phases — flight, compression, thrust and decompression — where each phase is governed by a distinct ODE model. Like our driven pendulum, the hopping robot has a controllable parameter as well — the thrust force — which can switch the system from regime to regime and model a variety of behaviors, ranging from a simple periodic "hopping" response (a single limit cycle) to a "limping gait" response (a period doubling bifurcation of the single limit cycle). PRET could reproduce this work, finding ODEs for each separate operating regime and then using a finite state machine to govern transitions between models. However PRET's primary goal is to find a single model that unifies *all* of these behaviors — if such a model exists.

Future Work

PRET's sensor-related reasoning is essentially complete, but its reasoning about the relationship between models and excitation sources — as well as final design decisions about how to treat actuator knowledge in an explicit way — are still under development. PRET currently uses very little domain knowledge about its target systems; instead, it relies upon *general* mathematics and physics — principles that are broadly applicable and supported by a well-developed, highly formalized body of mathematical knowledge that applies in *any domain*. The point of this decision was to make PRET easily extensible to other domains; because of this choice, refitting PRET for some new domain is simply a matter of a few lines of Scheme code. However, as we extend PRET into more network-oriented domains, such as electrical circuits, we are discovering that effective use of domain theory may be critical to streamlining PRET's generate phase(Easley & Bradley 1999). This poses another difficult representation problem; to solve

[3] where $\sin\theta \approx \theta$ and the system acts like a simple harmonic oscillator

it, we are exploring technique that combine generalized components from bond graphs(Karnopp, Margolis, & Rosenberg 1990) and network theory(Bose & Stevens 1965). This is a reasonable compromise between requiring detailed, domain-specific knowledge and our current goal of keeping PRET clean, general, and thus broadly applicable. These ideas will not only help PRET generate better models, but also provide a framework for automated reasoning about the relationship between input excitations and output responses — a critical task in any active modeling process.

Conclusion

The goal of this project is to automate the type of input-output analysis that expert engineers apply to modeling problems, and to use that technology to improve the PRET modeling tool, which automatically constructs ODE models of nonlinear dynamical systems. The approach proposed in this paper solves some of the problems that arise in phase-portrait analysis of complex systems by combining a state/parameter space representation with the qualitative abstraction of cell dynamics to obtain a useful new representation for I/O modeling. This *qualitative state/parameter space* representation, which allows a system's dynamics to be classified into discrete regions of qualitatively identical behavior, provides a useful framework for reasoning about multiple sets of observations about a given system. The largest obstacle in the I/O modeling path is the set of control-theoretic problems involved in automating interactions with actuators and sensors. First, PRET must autonomously manipulate a control parameter in order to perform a bifurcation analysis and find the regime boundaries. Then, it must use knowledge about the behavior in those different regimes to reason about what experiments are useful and possible. Finally, PRET must use this information to perform the experiments and analyze the results.

Acknowledgments: Joe Iwanski, Reinhard Stolle, Apollo Hogan, and Brian LaMacchia also contributed code and/or ideas to this project, and the AAAI reviewers' comments helped focus the content of this paper.

References

Abarbanel, H. 1995. *Analysis of Observed Chaotic Data.* Springer.

Abelson, H. 1990. The Bifurcation Interpreter: A step towards the automatic analysis of dynamical systems. *Inter. J. of Computers & Math. with Appl.* 20:13.

Back, A.; Guckenheimer, J.; and Myers, M. 1992. A dynamical simulation facility for hybrid systems. In *Hybrid Systems*, 255–267. Berlin: Springer.

Bose, A., and Stevens, K. 1965. *Introductory Network Theory.* New York: Harper and Row.

Bradley, E., and Easley, M. 1998. Reasoning about sensor data for automated system identification. *Intelligent Data Analysis* 2(2).

Bradley, E., and Stolle, R. 1996. Automatic construction of accurate models of physical systems. *Annals of Math. & Artif. Intel.* 17:1–28.

Bradley, E. 1995. Autonomous exploration and control of chaotic systems. *Cybernetics & Systems* 26:299–319.

Branicky, M.; Borkar, V.; and Mitter, S. 1994. A unified framework for hybrid control. In *IEEE Conf. on Decision & Control*, 4228–4234.

Capelo, A.; Ironi, L.; and Tentoni, S. 1993. A model-based system for the classification and analysis of materials. *Int. Sys. Eng.* 2:145–58.

de Kleer, J., and Brown, J. 1984. A qualitative physics based on confluences. *Artif. Intel.* 24:7–83.

Easley, M., and Bradley, E. 1999. Generalized physical networks for automated model building. Technical Report CU-CS-878-99, Univ. of Colorado. In review for IJCAI-99.

Forbus, K. D. 1987. Interpreting observations of physical systems. *IEEE Trans. on Systems, Man, and Cybernetics* 17(3):350–359.

Forbus, K. D. 1997. Qualitative reasoning. In A. Tucker, J., ed., *Computer Science and Engineering Handbook.* Boca Raton, FL: CRC Press.

Hsu, C. S. 1987. *Cell-to-Cell Mapping.* New York: Springer.

Karnopp, D.; Margolis, D.; and Rosenberg, R. 1990. *System Dynamics: A Unified Approach.* New York: Wiley. 2nd edition.

Kuipers, B. 1986. Qualitative simulation. *Artif. Intel.* 29(3):289–338.

Mosterman, P.; Zhao, F.; and Biswas, G. 1998. An ontology for transitions in physical dynamic systems. In *AAAI-98*, 219–223.

Stolle, R., and Bradley, E. 1998. Multimodal reasoning for automatic model construction. In *AAAI-98*, 181–188. Madison, WI.

Yip, K. 1991. *KAM: A System for Intelligently Guiding Numerical Experimentation by Computer.* MIT Press.

Zhao, F. 1993. Computational dynamics: Modeling & visualizing trajectory flows in phase space. *Ann. of Math. & Artif. Intel.* 8:285–300.