Automatic Abduction of Qualitative Models

Ina C. Kraan
Dept. of Artificial Intelligence
University of Edinburgh
inak@aipna.ed.ac.uk

Bradley L. Richards
Dept. of Computer Sciences
University of Texas, Austin
bradley@cs.utexas.edu

Benjamin J. Kuipers[1]
Dept. of Computer Sciences
University of Texas, Austin
kuipers@cs.utexas.edu

Abstract. We describe a system, MISQ, which automatically abduces qualitative models from descriptions of behaviors. MISQ generates models in the form of qualitative differential equations suitable for qualitative simulation by QSIM from either quantitative or qualitative data. Constraints are generated and filtered both by comparison with the input behaviors and by dimensional analysis. If the user provides complete information on the input behaviors and complete information on the dimensions of the input variables, the resulting model is unique, maximally constrained, and guaranteed to reproduce the input behaviors. If the user provides incomplete information, MISQ may not derive a unique model, but will produce a set of models, all of which reproduce the input behaviors.

I. Introduction

Qualitative simulation of physical systems provides researchers with insights by giving an overview of system behaviors without the deluge of detail inherent in quantitative simulation. Perhaps even more important, it may be possible to develop a qualitative simulation where developing a quantitative one would be impossible due to inexact knowledge of the system's internal workings. But even with the power of qualitative simulation systems like QSIM ([Kuipers, 1986, 1989]), developing qualitative models remains something of an art. For this reason, many researchers are investigating automatic model building. The most common approach is to construct models from given model fragments ([Forbus, 1984], [Forbus, 1986], and [deKleer and Brown, 1984]).

In this paper, we present a system, MISQ, which takes a bottom-up approach to model building. Given some or all of the behaviors[2] exhibited by a particular system, we abduce a model which reproduces those behaviors. A central result of this paper is that, if the user provides sufficient information on the input behaviors, the resulting model is ***unique***, ***maximal***, and ***correct*** in the sense that it reproduces the input behaviors. The models MISQ builds are qualitative differential equations (QDEs) suitable for use by QSIM.

The remainder of this paper is organized as follows: Section II summarizes related work in abduction, machine learning, and model building. Section III provides an overview of the design and operation of MISQ. Section IV proves the theorem central to the correctness of MISQ. Section V presents several examples of models which MISQ automatically constructed. Finally, Section VI gives our conclusions and suggests directions for further research.

II. Related Work

Abduction. Abduction is the process of determining what assumptions are necessary to allow a proof of a desired conclusion. Abduction is used extensively in domains which can be readily represented in logic (e.g., [Ng and Mooney, 1990] and [Ourston and Mooney, 1990]). The same abductive framework, however, can be applied to the constraint satisfaction domain of qualitative reasoning. MISQ does exactly this: given a set of desired behaviors and a theory of qualitative simulation, MISQ abduces the constraints necessary to produce the behaviors.

[1]Benjamin Kuipers' research is supported in part by NSF grants IRI-8905494 and IRI-8904454, by NASA grant NAG 2-507, and by the Texas Advanced Research Program under grant no. 003658-175.

[2] A ***behavior*** is a continuous time-ordered sequence of values for all system variables.

Machine learning. From a machine-learning point of view, MISQ is similar to the generalizing half of the Version Space algorithm described in [Mitchell, 1982]. Mitchell presents a method of deriving logical descriptions by looking at a series of examples. Given a set of positive instances[3], Version Space constructs the most specific conjunctive expression which includes those instances. MISQ constructs the most constrained model (essentially a conjunction of constraints) which reproduces all the input behaviors.

Model building. A number of researchers have worked in the area of automatic model building. However, none have directly addressed the problem of deriving a qualitative model from known behaviors. [DeCoste, 1990] presents a system for maintaining a qualitative understanding of a dynamic system from continuous quantitative inputs. Since DeCoste begins with a qualitative model, he is able to handle missing variables and other data deficiencies more easily than MISQ can. [Hellerstein, 1990] discusses the process of obtaining quantitative predictions of system performance in the absence of exact knowledge of the target system. And [Forbus and Falkenhainer, 1990] combines quantitative and qualitative models to produce "self-explanatory simulations," which produce quantitative predictions along with qualitative explanations of overall system behavior. But, again, Forbus and Falkenhainer require system models as input and exploit the relationship between the quantitative and qualitative models, rather than deriving the qualitative model from the input data.

As part of the model building process, MISQ performs dimensional analysis on the input variables. This analysis is similar in concept to [Bhaskar and Nigam, 1990], which uses dimensional analysis to derive qualitative functions. However, [Bhaskar and Nigam, 1990] requires that user dimensions be stated in terms of predefined fundamental types, whereas MISQ allows dimensions to be user-defined or even to remain unspecified.

III. System Description

Overview. MISQ's model generation is broken into three major phases. In the first phase, MISQ converts quantitative data into qualitative behaviors. It is also possible to input qualitative behaviors directly. In the second phase, MISQ generates and tests individual constraints, creating a set of all constraints which are individually consistent with the input behaviors. In the third phase, MISQ constructs models, in the form of QDEs, from the set of constraints generated in the second phase.

Conversion of quantitative data. MISQ can execute on two forms of quantitative input: high-resolution sensor data or hand-generated quantitative behaviors. If the input is high-resolution sensor data, MISQ converts the data to the required numeric precision and aligns events which occur in different variables at insignificantly different times. MISQ then discards all but the "interesting" points in the data; i.e., points where some variable reaches a maximum, a minimum, or zero. Hand-generated quantitative behaviors are the analog of processed sensor data: quantitative behaviors which include only interesting time points.

There are two requirements which high-resolution sensor data must meet in order to guarantee that MISQ can convert it to meaningful qualitative behaviors[4]. First, numeric precision must be sufficient to guarantee that consecutive values of a variable only have the same value if the variable is steady or has reversed its direction of change. Second, the quantitative data must include values for all times at which any variable reaches a steady point or is equal to zero.

MISQ converts quantitative behaviors into qualitative behaviors. For each variable at each time point, the quantitative value is turned into a qualitative value consisting of a qualitative magnitude and a direction of change. The qualitative magnitude is constructed by generating a landmark value and, if it is a new landmark, inserting it

---

[3]A *positive instance* is an example of the concept of interest. For example, if Version Space is to derive a logical description of birds, positive instances might include descriptions of robins and cardinals. In the domain of qualitative simulation, a positive instance is a behavior exhibited by the system we wish to derive a model for.

[4] These restrictions reflect the fact that our processing of high-resolution sensor data is simplistic. For example, we do not deal with the very real problems of limited precision or noise. The emphasis of our research is qualitative model building, and in a realistic application we would expect sensor data to be preprocessed by a system designed to deal with these problems.

into the qspace constructed so far[5]. The direction of change is determined by comparing the numeric value of the variable at the current time point with those of the preceding and subsequent time points. Furthermore, MISQ will add qualitative states to behaviors as needed. If, for example, a variable is at a minimum at one time point and at a maximum at the next, the qualitative state for the interval during which the variable is increasing is added to the behavior.

Examples of models generated from sensor data and hand-generated quantitative input are given in Section V.

Constraint generation. The input to the second phase of MISQ is a set of consistent qualitative behaviors, the landmark values (qspaces) of the initial state, and dimensional information. In the first step of this phase, MISQ selects an arbitrary behavior and generates all constraints satisfied by any combination of variables. This is done by generating tuples of variables and testing their values against the satisfaction conditions for each constraint type.

MISQ currently covers the following constraint types[6]: arithmetic constraints (add, mult, and minus), differential constraints (d/dt), functional constraints (M^+ and M^- for strictly monotonically increasing and decreasing functions), and direction-of-change constraints (constant). These constraints are a subset of the constraints provided in QSIM. Nevertheless, they are expressive enough to build many types of interesting qualitative models.

The satisfaction conditions are similar to those in QSIM, though somewhat simpler. Since the input behaviors are assumed to be correct, we need not check the continuity criteria from a state to its successor. And, since the satisfaction criteria within a state are the same for time points and intervals, MISQ does not have to distinguish between time points and intervals. The constraint satisfaction criteria are based on the magnitudes, signs, directions of change, and corresponding values of the variables; these criteria are defined in detail in [Kuipers, 1986]. For example, the constraint (M^+ x y) is satisfied if the directions of change of x and y (expressed as increasing, decreasing, or steady) are always identical, and there are no corresponding values which are non-functionally related. If, for instance, there are corresponding values at (x_1 y_1) and (x_1 y_2), and y_1 and y_2 are known to be distinct values, the relationship between x and y cannot possibly represent a function. The constraint (d/dt x y) is satisfied if the direction of change of x is increasing, decreasing, or steady and the sign of y is +, -, or 0, respectively. Similarly, the conditions for qualitative addition and multiplication involve checking signs, directions of change, and corresponding values.

Finally, for each derivative and arithmetic constraint generated, MISQ checks whether the dimensions of the variables in each individual constraint are reasonable. If, for example, the constraint (d/dt x y) has been generated, MISQ will test whether the dimensions of x can be the dimensions of y divided by time. This ensures that derivative and arithmetic constraints are abstractions of equations potentially representing real physical systems. A functional constraint, of course, imposes no a priori restrictions on the dimensions of its arguments. Since we are working with qualitative data, dimensions are not stated in traditional units of measurement such as meters or grams, but rather in terms of fundamental types like mass, time, or length.

Users are free to define their own fundamental types, except for ***time***; MISQ is only interested in the ***relationship*** between the dimensions of different variables. Ideally, the dimensions of every variable should be specified in terms of fundamental dimensions that allow MISQ to relate it appropriately to other variables in the behaviors. However, if the user does not know the relationship between the dimensions of some variables, they may remain unspecified.

In some cases, the user may be able to reduce the number of constraints in the final QDE by making the fundamental dimensions even more specific. For example, if a system has amounts of both oxygen and water, and the user knows that it makes no sense to combine amounts of oxygen and water in this simulation, the user can use different fundamental dimensions, such as amount_of_oxygen and amount_of_water.

After generating all possible candidate constraints, MISQ tests all behaviors against these constraints, eliminating any constraint that violates any satisfaction condition.

---

[5] A *qspace* is a totally ordered set of landmark values. ***Landmarks*** are values which break the domain of a variable into qualitatively distinct intervals. For example, the qspace of the temperature of a pot of water might be {*absolute-zero, freezing, boiling, infinity*}.

[6] In our notation, the addition constraint (add x y z) means $x + y = z$, (d/dt x y) means $dx/dt = y$, (M^+ x y) means that a strictly increasing monotonic function holds between x and y, and so forth.

Model generation. If the user provided complete information on the input behaviors, MISQ will produce a single model which consists of the set of all constraints output from the second phase. If, on the other hand, the user leaves directions of change or dimensions unspecified, the set of constraints generated in the second phase may be inconsistent as a model. In this case, MISQ constructs QDEs which are the maximal consistent subsets of the constraints[7].

As an example of model generation with complete information, consider an empty bathtub with a finite capacity, a constant inflow, and a constant drain opening. This simple bathtub model exhibits three possible behaviors: reaching equilibrium at a level below the top of the tub, reaching equilibrium exactly at the top, and overflowing. We simulated the simple bathtub using QSIM, and presented a complete qualitative description of one behavior to MISQ (equilibrium point less than full). This behavior is shown in Figure 1. MISQ abduced the exact QDE used to produce the behavior, with the addition of two redundant constraints (see Figure 2).

Variable	Initial Qspace	Units
Inflow	0, in1, ∞	mass/time
Outflow	0, ∞	mass/time
Netflow	-∞, 0, net1, ∞	mass/time
Amount	0, ∞	mass

State 0 (initial state) — successors: state 1

Variable	Magnitude	Direction of Change
Inflow	in1	steady
Outflow	0	increasing
Netflow	net1	decreasing
Amount	0	increasing

State 1 — successors: state 2

Variable	Magnitude	Direction of Change
Inflow	in1	steady
Outflow	(0, ∞)	increasing
Netflow	(0, net1)	decreasing
Amount	(0, ∞)	increasing

State 2 — successors: none

Variable	Magnitude	Direction of Change
Inflow	in1	steady
Outflow	out1	steady
Netflow	0	steady
Amount	amount1	steady

Figure 1. Single qualitative behavior of the simple bathtub. This information can be automatically generated from high-resolution sensor data.

If the user does not provide complete information on the input behaviors or dimensions, the second phase of MISQ may generate and retain too many constraints; hence the output QDE from phase two will contain conflicting constraints.

In this case, the third phase will generate and test subsets of the constraints, subject to a few heuristic criteria which ensure that a subset has the potential to form a valid model. These criteria also serve to limit the exponential explosion inherent in subset generation. Specifically, the subset must contain at least one derivative constraint, form a connected graph, and constrain all input variables.

```
(constant inflow)
(add outflow netflow inflow)
(M- outflow netflow)
(M+ outflow amount)
(M- netflow amount)
(d/dt amount netflow)
```

Figure 2. Output model for simple bathtub.

Subsets which meet these initial criteria must be checked to ensure that assumptions for missing direction-of-change information are consistent among all constraints in a subset. Additionally, subsets are checked for global dimensional consistency to see that they make sense as a model of a physical system. For example, even without any dimensional information from the user, we know that the following two constraints are inconsistent:

(d/dt *a* *b*)
(add *a* *b* *c*)

These constraints are inconsistent because variables being added must have the same dimensions (whatever they may be), but the dimensions of a variable and its derivative differ by a factor of 1/time.

Finally, since we are interested in maximally constrained models, we discard any models which are subsets of other valid models. The final models output from the third phase carry the correctness guarantee discussed above: each model reproduces all input behaviors.

Presenting the simple bathtub to MISQ with no dimensional information results in six models. One of these is the desired model shown in Figure 2. The others reflect the fact that, without dimensional information, MISQ is unable to discard obviously impossible constraints. For example, MISQ is no longer able to distinguish between outflow

[7] MISQ generates maximal QDEs, which means that QDEs may contain redundant constraints. For example, since the M^+ constraint is transitive, if M^+ constraints hold between variables *a* and *b* and between variables *b* and *c*, then MISQ would also include a redundant M^+ constraint between variables *a* and *c*.

and amount, as they are qualitatively indistinguishable in the specified behavior. As a result, it produces models such as the one shown in Figure 3.

```
(constant inflow)
(add amount netflow inflow)
(M- outflow netflow)
(M+ outflow amount)
(M- netflow amount)
(d/dt outflow netflow)
```

Figure 3. Incorrect model for the bathtub, resulting from lack of units information.

Providing Incomplete Information. It may not always be possible to provide a complete description of the behaviors. There are three main areas where a description might be incomplete.

Dimensional information: The user may not be able to determine dimensional information for all variables, but can only identify which variables have the *same* dimensions, whatever those dimensions may be. For example, the user might identify all variables which are flows and assign them the same (unspecified) dimensions. With this partial dimensional information, in our test cases, MISQ still generated unique models.

Directions of change: In some cases, direction-of-change information may be unavailable for some variables ([Kuipers and Chiu, 1987]). MISQ can still generate constraints for these variables by using the limited direction-of-change information that can be derived from the qualitative magnitudes of the variables and by examining corresponding values.

Variables and values: There may be cases where the user does not know what set of variables are important to a system. Alternatively, the user may know of a variable but be unable to measure its value. Abducing QDEs when variables or values are omitted becomes difficult, and is the subject of ongoing research.

IV. Central Theorem

A central feature of MISQ is that, given sufficient information on the input behaviors, it will generate a unique maximal QDE which is guaranteed to reproduce the input behaviors. This further implies that, if the user presents all system behaviors as input, MISQ will produce a correct system model. This section presents the essential definitions and proves this central feature of MISQ.

Consistent set of behaviors. A set of behaviors is consistent if it represents (or potentially represents) a real physical system. This can be summarized by two specific criteria: First, relationships among variables must be qualitatively consistent among behaviors. For example, if two variables are (in reality) related by some constraint, then this constraint must be the same in all behaviors (e.g., not M^+ in one behavior and M^- in another). Second, dimensions must make sense (as they must in a real system). For example, we might imagine a system in which a variable and its derivative have the same dimensions and create behaviors for the system, but the system could never actually exist.

Complete description. A description of a behavior is complete if the following three criteria are met. First, all variables in the system are identified. Second, all variables have values at all given time points and intervals. If MISQ is working from high-resolution data, this means that continuous data is given for all variables. Finally, fundamental dimensions are given for all variables. Note that MISQ does not require that all behaviors of a system be given. However, specifying too few behaviors may result in a model which is too constrained to produce the behaviors of the system which were not given as input. The more behaviors that are given, the more constraints may be eliminated, thus making it less likely that the resulting model will be overconstrained.

Theorem. Given a complete description of a consistent set of behaviors, MISQ will produce the most constrained QDE which reproduces those behaviors. Furthermore, this QDE is unique.

Proof of Theorem. Given a fixed set of variables, two sets of constraints on these variables C_1 and C_2, and the behaviors consistent with these constraints $Beh(C_1)$ and $Beh(C_2)$, the set of behaviors consistent with both sets of constraints is given by the relation

$$Beh(C_1 \cup C_2) = Beh(C_1) \cap Beh(C_2) \tag{1}$$

Given a complete and consistent set of input behaviors, MISQ exhaustively generates all possible constraints which are individually consistent with the behaviors. By (1), these constraints can be combined, and their behavior set will include all input behaviors; this guarantees that a correct model in fact exists. Since MISQ generates and retains all consistent constraints, the resulting QDE is maximally constrained and unique.

V. Examples

During development, we ran MISQ on a variety of common models, including the cascaded tanks example shown below. Once the system was complete, we ran the system on a more difficult problem: One of us was trying to develop a model by hand to simulate the opening and closing of stomata (pores) in plants. This model is also briefly discussed below.

Cascaded tanks. Cascading two tanks so that the drain from one provides the inflow to the next provides a more complex system than the simple bathtub. We ran MISQ on various types of input, i.e., with no, partial, and complete dimensional information for qualitative behaviors; with complete dimensional information for hand-generated quantitative information; and with complete dimensional information for high-resolution data. Graphs of the high-resolution data for four of the variables are shown in Figures 4 and 5. We did not provide direction-of-change information for the flow between the two tanks in any of the qualitative behaviors.

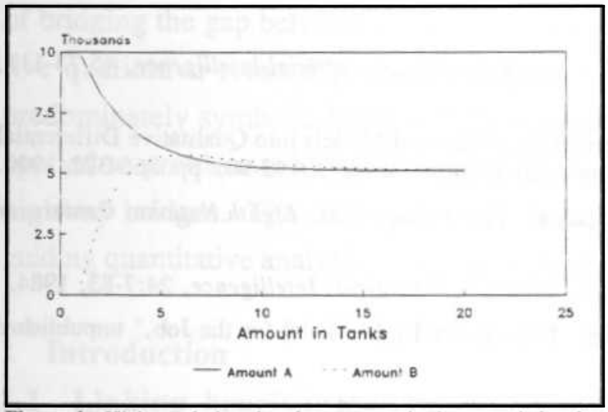

Figure 4. High-resolution data for amounts in the cascaded tanks.

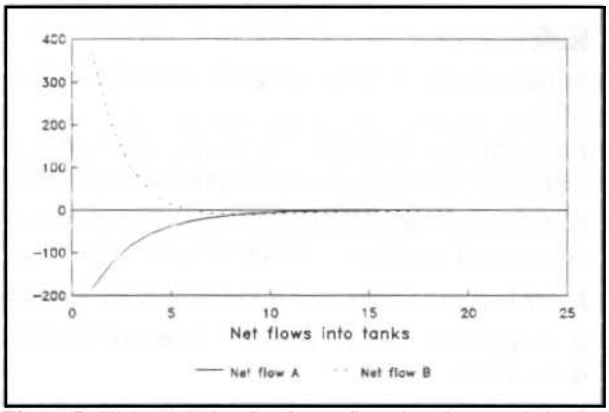

Figure 5. High-resolution data for net flows into the cascaded tanks.

The model produced for complete dimensional information and qualitative input, shown in Figure 6, is exactly the one we would expect.

When we provided no dimensional information at all, there were so many potential relationships among variables that MISQ produced over 300 QDEs. However, when we provided partial dimensional information, i.e., we specified each variable to be either a flow or an amount, but did not provide the fundamental dimensions for either, we obtained the same results as in Figure 6. When we provided high-resolution data, and when we provided hand-generated quantitative behaviors, MISQ again uniquely generated the model shown in Figure 6.

(constant inflow_a)
(add outflow_a netflow_a inflow_a)
(add outflow_b netflow_b outflow_a)
(M+ amount_a outflow_a)
(M- amount_a netflow_a)
(M+ amount_b outflow_b)
(M- outflow_a netflow_a)
(d/dt amount_b netflow_b)
(d/dt amount_a netflow_a)

Figure 6. Output for cascaded tanks, with full unit information.

Stomatal opening and closing. One of us was trying to develop a model of stomata (pores) in plants to simulate the influence of the levels of water and carbon-dioxide on stomatal opening and closing. This is a complex biological process with conflicting feedback loops. Given a qualitative description of the desired behaviors and full dimensional information, MISQ produced a unique, correct QDE with 22 constraints over 14 variables.

VI. Conclusions

Model building can be a difficult and time-consuming task. It can be simplified by automating some steps of the process. In this paper, we presented a system, MISQ, which automatically produces models from known behaviors. This approach is useful both in design and diagnosis.

In design, researchers often want models to produce specified quantitative or qualitative behaviors; MISQ can eliminate the need to handcraft these models. In diagnosis, MISQ can derive a model which reproduces a faulty behavior. Comparing the model of the faulty behavior with the correct model may show where the system fault lies. The fact that MISQ can work directly with the available quantitative information is particularly helpful in this context.

There are several promising directions for further research. First, MISQ can easily be extended to include other types of constraints like the QSIM *S* and *U* constraints. Second, MISQ could be enhanced to allow the user to leave variables or values unspecified; this would be particularly helpful when dealing with real physical systems where it may not be possible to obtain measurements of all essential variables. Third, when MISQ is given incomplete information and generates many potential models, additional filters could eliminate many of the proposed models (for example, by eliminating isomorphic variants); specifying behaviors which should not be produced by the model could also help to eliminate incorrect models. Fourth, inconsistent input behaviors may represent a system which is crossing a transition. Modelling such a system would require constructing multiple models connected by well-defined transitions. Lastly, MISQ could be more closely tied to qualitative systems which work with partial quantitative information; rather than converting quantitative inputs to a purely qualitative model, MISQ could retain the quantitative information and pass it, along with the model, to a system like Q2 ([Kuipers and Berleant, 1988]).

References

R. Bhaskar and A. Nigam, "Qualitative Physics Using Dimensional Analysis," *Artificial Intelligence*, 45:73-111, 1990.

J. Crawford, A. Farquhar, and B. Kuipers, "QPC: A Compiler from Physical Models into Qualitative Differential Equations," *Proceedings of the Eighth National Conference on Artificial Intelligence* (AAAI-90), pp. 365-372, 1990.

D. DeCoste, "Dynamic Across-Time Measurement Interpretation," *Proceedings of the Eighth National Conference on Artificial Intelligence* (AAAI-90), pp. 373-379, 1990.

J. deKleer and J. Brown, "A Qualitative Physics Based on Confluences," *Artificial Intelligence*, 24:7-83, 1984.

B. Falkenhainer and K. Forbus, "Compositional Modeling: Finding the Right Model for the Job," unpublished draft, 1990.

K. Forbus, "Qualitative Process Theory," *Artificial Intelligence*, 24:85-168, 1984.

K. Forbus, "The Qualitative Process Engine," Technical Report, Department of Computer Science, University of Illinois, 1986.

K. Forbus and B. Falkenhainer, "Setting Up Large-Scale Qualitative Models," *Proceedings of the Seventh National Conference on Artificial Intelligence* (AAAI-88), pp. 301-306, 1988.

K. Forbus and B. Falkenhainer, "Self-Explanatory Simulations: An integration of qualitative and quantitative knowledge," *Proceedings of the Eighth National Conference on Artificial Intelligence* (AAAI-90), pp. 380-387, 1990.

J. Hellerstein, "Obtaining Quantitative Predictions from Monotone Relationships," *Proceedings of the Eighth National Conference on Artificial Intelligence* (AAAI-90), pp. 388-394, 1990.

B. Kuipers, "Qualitative Simulation," *Artificial Intelligence*, 29:289-338, 1986.

B. Kuipers, "Qualitative Reasoning: Modelling and Simulation with Incomplete Knowledge," *Automatica*, 25:571-585, 1989.

B. Kuipers, *Qualitative Reasoning: Modeling and Simulation with Incomplete Knowledge*, unpublished draft, 1990.

B. Kuipers and D. Berleant, "Using Incomplete Quantitative Knowledge in Qualitative Reasoning," *Proceeding of the Seventh National Conference on Artificial Intelligence* (AAAI-88), pp. 324-329, 1988.

B. Kuipers, and C. Chiu, "Taming Intractible Branching in Qualitative Simulation," *Proceedings of the 10th International Joint Conference on Artificial Intelligence* (IJCAI-87), pp. 1079-1085, 1987.

T. M. Mitchell, "Generalization as Search," *Artificial Intelligence*, 18:203-226, 1982.

H. T. Ng and R. J. Mooney, "On the Role of Coherence in Abductive Explanation," *Proceedings of the Eighth National Conference on Artificial Intelligence* (AAAI-90), pp. 337-342, 1990.

D. Ourston and R. J. Mooney, "Changing the Rules: A Comprehensive Approach to Theory Refinement," *Proceedings of the Eighth National Conference on Artificial Intelligence* (AAAI-90), pp. 815-820, 1990.