In G. Biswas and S. McIlraith (Eds.),
Hybrid Systems and AI: Modeling Analysis and Control of Discrete+Continuous Systems.
AAAI Spring Symposium Series, Stanford, March 1999.

Monitoring Piecewise Continuous Behaviors by Refining Trackers and their Models

Bernhard Rinner
Institute for Technical Informatics
Technical University Graz
A-8010 Graz, AUSTRIA
b.rinner@computer.org

Benjamin Kuipers
Department of Computer Sciences
University of Texas at Austin
Austin, TX 78712, U.S.A.
kuipers@cs.utexas.edu

Abstract

We present a model-based monitoring method for dynamic systems that exhibit both discrete and continuous behaviors. MIMIC (Dvorak & Kuipers 1991) uses qualitative and semi-quantitative models to monitor dynamic systems even with incomplete knowledge. Recent advances have improved the quality of semi-quantitative behavior predictions, used observations to refine static envelopes around monotonic functions, and provided a semi-quantitative system identification method. Using these, we reformulate and extend MIMIC to handle discontinuous changes between models. Each hypothesis being monitored is embodied as a tracker, which uses the observation stream to refine its behavioral predictions, its underlying model, and the time uncertainty of any discontinuous transitions.

Introduction

Physical systems are by nature continuous. However, it is natural to simplify models by abstracting isolated regions of rapid change to instantaneous discontinuities separating regions of continuous behavior (Iwasaki *et al.* 1995; Nishida & Doshita 1987). Systems which exhibit both continuous and discrete behaviors are called *hybrid systems*, where a continuous segment of the system's behavior is called a *mode of operation* and a discontinuous change is called a *transition* between modes.

Model-based monitoring relies on a comparison between the predicted behavior of a model and the observed behavior of a physical system. Traditional monitoring approaches typically use a single precise model of the physical system. However, even if the system is behaving properly, precise parameter values and functional relationships are often not known. More importantly, monitoring systems are designed to detect unexpected events or faults, after which knowledge of the system is by definition incomplete. A reliance on precise models leads to overly-specific predictions, sacrificing accuracy and coverage exactly when it is most important for the monitoring system to consider all possible scenarios.

The MIMIC framework (Dvorak & Kuipers 1991) addresses this need, first by using qualitative and semi-quantitative (SQ) models in the QSIM representation (Kuipers 1994) to express incomplete knowledge with a guarantee that all possible real-valued behaviors are covered; and second, by tracking multiple qualitatively-distinct hypotheses in parallel. SQSIM (Kay 1998) extends the semi-quantitative inference power of QSIM by deriving and reasoning with dynamic envelopes guaranteed to bound the real behaviors consistent with an SQ model. SQUID (Kay 1996; 1997) is a semi-quantitative system identification method based on SQSIM that assimilates a set of observations to an SQ model over a single continuous mode.

Time uncertainty at a mode transition has a particularly explosive effect on the uncertainty of predictions from the SQ model after the transition. Therefore, we focus our attention first on getting the most out of SQUID-based tracking of a continuous mode hypothesis, and second, on detecting the mode transition and refining its time uncertainty. In our approach, the monitoring system starts with a coarse description of the physical system and uses the observation stream to refine the behavior prediction, its underlying model, and the time uncertainty of any discontinuous transition. After presenting the details of our extension and reformulation of MIMIC, we present a non-trivial example and discuss related work.

Tracking Piecewise Continuous Behaviors

A *tracker* embodies a continuous mode hypothesis and confirms or refutes the hypothesis by unifying its predictions with the observed behavior. When the observations provide sufficient new information, the tracker may be able to refine the uncertainty in the underlying model, and thus make more precise predictions in the future.

SQ System Identification

A tracker is based on SQUID, which refines an imprecise model (SQDE) by a process called *trend matching*. Uncertainty in the SQDE is represented by numerical intervals bounding possible values of unknown parameters, and by static envelopes – functions bounding the possible graphs of unknown monotonic functions.

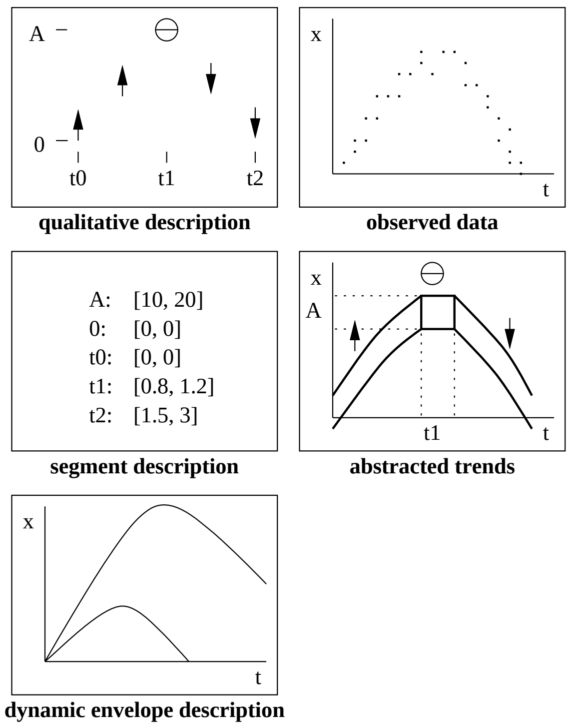


Figure 1: Abstracted description used by trend matching. SQSIM derives a behavior description at the qualitative, segment and dynamic envelope level. Trends describe the observed data at the same levels.

Trend matching compares semi-quantitative trajectory descriptions derived by SQSIM (the *SQ prediction*) and the corresponding properties of the observations (the *SQ trend*). To refine the underlying model, portions of the model space which cannot plausibly generate the observations are excluded.

There are three levels of abstracted properties of the trajectories, corresponding to the level of detail derived by the components of SQSIM: *qualitative* (QSIM), *segment* (Q2), and *dynamic envelope* (NSIM) descriptions (Figure 1). The qualitative description is defined by a sequence of symbols ($\downarrow$, $\ominus$ and $\uparrow$) representing the derivative's sign (qdir) of the trajectory at time points and intervals between time points. The segment description specifies intervals bounding the trajectory at particular time points, i.e., magnitude and time ranges. The dynamic envelope description bounds the trajectory by a lower and an upper envelope. A trend represents the abstracted properties of the observed data (Figure 1), i.e., symbols representing the qdir, bounding intervals on extrema and bounding envelopes for monotonic segments.

Tracker Architecture

Figure 2 presents the architecture of a tracker. The tracker is initialized with the SQ prediction and the underlying SQDE of the current mode. Information

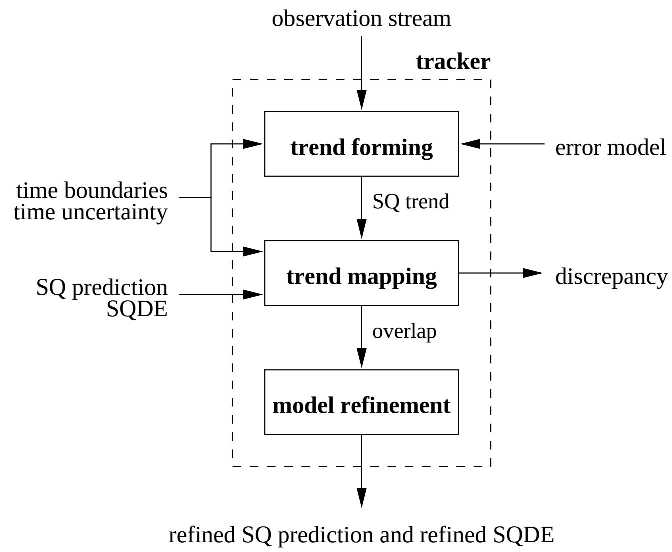


Figure 2: Tracker architecture.

about the time boundary and the time uncertainty of the current mode may be given. The tracker consumes an observation stream and it either produces a refined SQ prediction and SQDE, or detects a discrepancy between the observation and the prediction. The observation stream is a sequence of samples: numeric values for variables at specified times derived by possibly noisy sensors. Samples do not necessarily appear at a constant rate, and need not be synchronized across variables.

Trend forming generates an SQ trend describing each variable in the observation stream by breaking the samples into monotonic segments (Kay 1996). The segments are determined by computing the slope of a linear least-squares fit to the data within a sliding window over the samples. Dynamic envelope descriptions for the $\uparrow$ and $\downarrow$ segments are generated by MSQUID, a neural network-based estimator for monotonic functions (Kay & Ungar 1993), out to any given confidence bound. Each $\ominus$ segment is described by time and magnitude ranges.

The goal is to detect the qualitative dynamics of the underlying signal in the noisy observation. In the current implementation it is assumed that Gaussian noise of fixed mean and variance is superimposed on the "pure" signal. Each observed variable has an error model that specifies bounds on mean and variance for noise.

Trend mapping compares the SQ trend derived from the observations with the SQ predictions by stepping through both sequences. If an inconsistency is detected between the trend and the prediction, the current hypothesis is refuted, so the mapping process and the current tracker are aborted.

Qualitative mapping generates a correspondence between the qdirs in the SQ prediction and the SQ trend. A successful correspondence may fail to be one-to-one because (a) the samples in the observation stream may end before some of the qualitative changes in the SQ prediction take place; (b) the SQ prediction terminates with a mode change before the end of the current SQ trend, leaving data to correspond to the next mode; or (c) the SQ prediction may include small qdir changes which are not detectable in a noisy observation stream.

Segment mapping ensures consistency of corresponding behavior segments in the SQ trend and the SQ prediction, in the sense that their time and magnitude bounds overlap. Consistency of segments is checked by asserting the segment boundaries of the SQ trend to the corresponding segments of the SQ prediction and propagating these boundaries to the other variables in the SQDE using Q2's interval propagation.

Dynamic envelope mapping ensures consistency by intersecting the dynamic envelopes for corresponding monotonic segments of the trend and the prediction, and propagating using Q2.

Model refinement takes place when trend mapping decreases the bounds on some variables in the SQDE. Parameter uncertainty is refined by using Q2 to derive bounds on independent variables from dependent ones. Functional uncertainty is refined by excluding portions from the static envelopes that are inconsistent with the (refined) variable bounds. The trend mapping techniques guarantee that portions of the model space are ruled out only when they are inconsistent with the observations (Kay 1996).

If a mode change is manifested by a discontinuous change of an observed variable or a sudden sign change of its slope, the change becomes explicit in the purely qualitative trend, and is easy to detect. Otherwise, segment and dynamic envelope trend mapping should eventually refute the current model, but this depends on the amount of uncertainty in both data and model.

Once a mode change has been detected, segment and dynamic envelope mapping and model refinement are limited to the (qualitatively) mapped segments. The tracker for the next mode is initialized with the unmapped segments. The time estimate on the mode change specifies the set of trend segments that correspond to each mode. Time uncertainty in the mode change affects the current tracker and more dramatically the following tracker, in an important way.

Refining the Time Uncertainty of Discontinuous Changes

We assume that there are three possible causes for discontinuous changes in the model of a complex system: (i) the autonomous operation of the plant moves from one operating mode to another; (ii) the plant operator takes a known action; and (iii) an unexpected and externally caused event such as a failure takes place. In the first two cases, the current and following modes are known. In the third case, we assume that a separate diagnosis engine proposes a set of fault hypotheses, which MIMIC tracks in parallel.

A discontinuous change happens in an instant. Unfortunately, with imprecise models and noisy and finitely sampled observations, we may never be able to determine the precise instant when the change takes place. The best we can do is determine time bounds on the instant when the change occurred.

For matching a piecewise continuous model to a stream of observations, it is particularly important to make the time bounds on discontinuous changes as precise as possible. Time uncertainty on a mode change affects the entire correspondence between prediction and observation in the following mode, resulting in propagating uncertainty. Figure 5(c) shows how weak time bounds on a discontinuous change can result in extremely weak bounds on the prediction of the following mode. Therefore, we focus on improving these time bounds.

Intersecting Trend and Prediction

We focus here on refining time uncertainty of a mode change based on the intersection of SQ trend and SQ prediction. After semi-quantitative reasoning has provided bounds on the transition time, advanced filtering techniques based on statistical or digital signal processing may be applicable.

When there is time uncertainty, the mapping between the SQ trend and the SQ prediction is not fixed. The SQ prediction can be shifted relative to the SQ trend by any offset within the range of the time uncertainty.

However, a mapping is only valid if the SQ trend segment and the SQ prediction segment have a non-empty intersection for every time-point t in the SQ trend. This is exploited to derive refined bounds on the time uncertainty of discontinuous changes (Figure 3).

For $\uparrow$ segments, the mapping is valid as long as (i) the upper envelope of the trend is above the lower envelope of the prediction and (ii) the lower envelope of the trend is below the upper envelope of the prediction. More formally, we can determine the lower bound t_{cmin} and the upper bound t_{cmax}, respectively, as follows:

$$\begin{aligned} t_{cmin} &= min\{t_s : \forall t, \overline{x_{tr}}(t) \geq \underline{x_{pr}}(t-t_s)\} \\ t_{cmax} &= max\{t_s : \forall t, \underline{x_{tr}}(t) \leq \overline{x_{pr}}(t-t_s)\} \end{aligned}$$

where $x_{tr}(t)$ is the observed trend for $x(t)$, $x_{pr}(t-t_s)$ is the prediction, shifted by t_s. Over- and under-bars represent the upper and lower dynamic envelopes, respectively. Similar conditions hold for $\downarrow$ and $\ominus$ segments.

This intersection process is applied to all segments of the mode. Improvements in time uncertainty propagate from segment to segment by interval arithmetic and intersection of bounding intervals.

Incremental Refinement

When there is a great deal of time uncertainty about the transition from one mode to another, many samples in

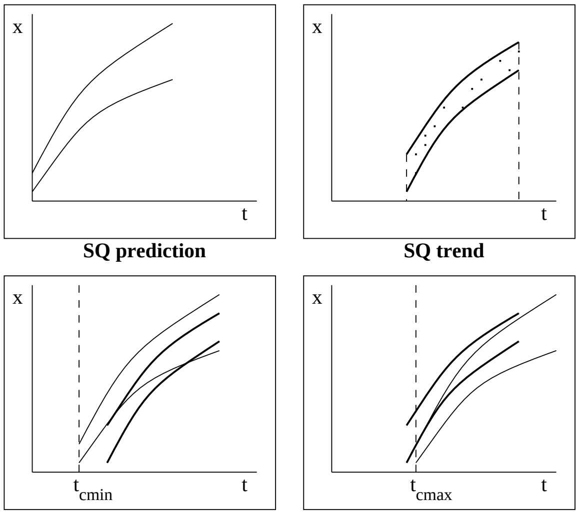


Figure 3: Deriving bounds on the time uncertainty by intersecting the SQ prediction and the SQ trend.

1 *Behavior prediction for mode$_{i+1}$*
Inherit the model refinements (variable bounds and static envelopes) from mode$_i$ to mode$_{i+1}$.
Determine the SQ prediction for mode$_{i+1}$ by SQSIM.

2 *Time uncertainty refinement of mode$_{i+1}$*
Perform a qualitative mapping of mode$_{i+1}$ to ensure consistency between the SQ trend and the SQ prediction.
Apply trend/prediction intersection to refine the time uncertainty t_c.

3 If the time uncertainty t_c has been decreased
Goto 4
otherwise stop and return t_c

4 *Re-tracking of mode$_i$*
Re-map the samples outside the uncertainty interval to mode$_i$ and mode$_{i+1}$.
Re-track mode$_i$ and refine its model.
Goto 1

Figure 4: Incremental refinement of the time uncertainty t_c of the transition between mode$_i$ and mode$_{i+1}$.

the observation stream fall within the uncertainty interval, and thus cannot be unambiguously assigned to one of the adjacent segments of monotonic change. After time uncertainty is decreased, some samples can now be assigned to a definite adjacent segment. The additional information helps refine the mode the segment belongs to, and its underlying model. Improvements to the adjacent modes can, in turn, lead to further decreases in time uncertainty of the transition. And so on until no further improvement results (Figure 4).

Overall Monitoring System

The overall monitoring system tracks multiple hypotheses in parallel. The hypotheses may represent different nominal or fault models of the plant, or they may represent different qualitative behaviors predicted from semi-quantitative simulation of a particular model.

A particular hypothesis is a sequence of modes $[H_1(t_0, t_1); H_2(t_1, t_2); \cdots; H_n(t_{n-1}, t_n)]$. The monitoring system alternates between tracking a particular mode $H_i(t_{i-1}, t_i)$ and refining the time uncertainty of the mode transition at t_i. Model refinements such as improved variable bounds or static envelopes may be inherited from one model H_i to the next H_{i+1} across the mode transition. In the current implementation, the user specifies the variables and functional relations whose refinements can be inherited.

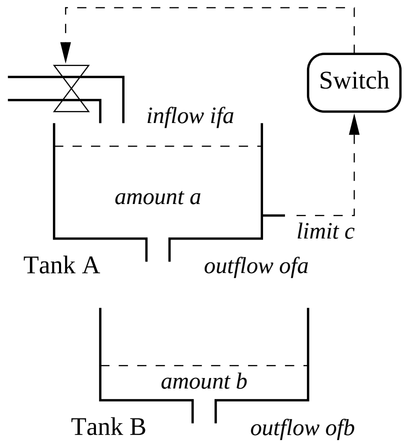
Experimental Results

We demonstrate the refinement capabilities of our monitoring system using a two tank system. In this example (Figure 5(a)), we start with a filled upper tank and an empty lower tank; the drains of both tanks are open and the upper tank is filled at a constant inflow rate. When the amount in the upper tank drops below a limit the inflow rate is increased. This scenario is modeled as a transition between two operating modes (Figure 5(b)). Only imprecise information is known about this scenario, i.e., intervals for variables and bounding envelopes for functional relations. Since both tanks remain unchanged, the refinements of the variables a and b as well as of the functional relations f and g are inherited from the first to the second mode.

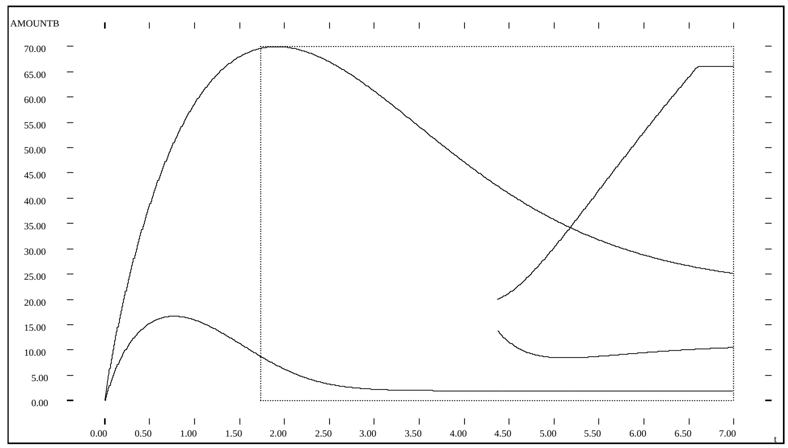
SQSIM predicts 9 different behaviors for the two tank scenario; 3 of them include the region transition. Only one is consistent with the SQ trend; the other trackers are refuted. Figure 5(c) shows the predicted dynamic envelopes for the amount in the lower tank for the surviving prediction. SQSIM predicts the time of the region transition as $[1.68, inf]$.

The observations are generated by numeric simulation of an ODE, adding Gaussian noise with fixed mean and variance to the samples. The exact model for deriving the samples is given as $a' = ifa - 9\sqrt{a}$, $b' = 9\sqrt{a} - 8\sqrt{b}$ with $a(t_0) = 95$, $b(t_0) = 0$, $c = 9$ and an inflow rate $ifa = 25$ before and $ifa = 60$ after the transition. The samples are derived at a rate of 20 Hz.

Figure 5(d) shows the samples and bounding envelopes derived by trend forming for the variable b. MSQUID constructs the bounding envelopes around the observations to achieve a certainty of 95%. This figure also presents the final refinement achieved by the monitoring system. The dynamic envelopes for b are refined to 18% of their initial area and the time uncertainty is refined to $[3.78, 4.14]$ after two iterations of the refinement algorithm. Observations for a, b, ofa and ofb, with noise $\mu = 0$ and $var = 1$, are used in this case. Note the propagation of refinements through the SQDE, i.e., the dynamic envelopes in the first segment are narrower than the bounding envelopes of the



(a) Two tank scenario



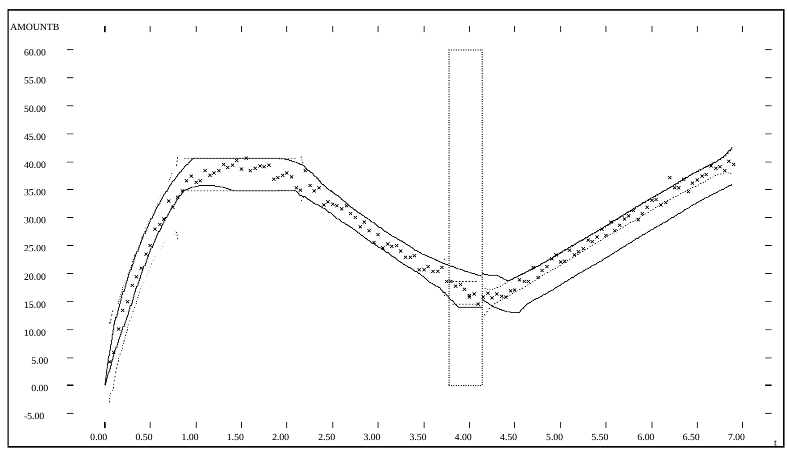
(c) Predicted behavior

a' = ifa - f(a)
b' = f(a) - g(b)

a(t0) = [90, 100] b(t0) = [0, 0] c = [8, 12]
*f(a) = [8*sqrt(a), 12*sqrt(a)] = ofa*
*g(b) = [8*sqrt(b), 12*sqrt(b)] = ofb*

$$ifa = \begin{cases} [24, 26] & \text{if } a<c \\ [60, 65] & \text{otherwise} \end{cases}$$

(b) Imprecise model with region transition



(d) Refined behavior and time uncertainty

Figure 5: A two tank scenario (a), modeled as a region transition (b), its prediction (c) of the upper and lower bounds (dynamic envelopes) for b, and the refined bounds and time uncertainty (d). In both graphs, the solid lines present the predicted or refined dynamic envelopes. The dotted box represents the time uncertainty of the mode change; the dashed lines represent the bounding envelopes of the observations (d). Due to the time uncertainty of the discontinuous change the prediction of the second mode can start at any time within the dashed box. For the sake of readability, they start in the middle (c) and at the right end (d) of the dashed box, respectively.

observation of b. Due to time uncertainty, the dynamic envelopes in the second mode are wider than the bounding envelopes of the observation.

The effect of observability is shown in Table 1. This table presents the achieved refinements dependent on the observed variables. The degree of refinement is defined by the ratio of the predicted and refined *areas* for variables and functional relations. For the variables a and b, the area is specified by the dynamic envelopes over the observation time. For the functional relations f and g, the area is specified by the bounding functions (static envelopes) over the range of f and g. The achieved refinement of the time uncertainty is represented by t_c. As the number of observed variables increases the refinement improves and extends to more variables and functional relations. Table 2 presents the reduced refinements caused by an increase of noise in the observation.

obs. vars.	$a(t)$	$b(t)$	f	g	t_c
none	1	1	1	1	[1.68, inf]
b	1	0.53	1	1	[2.70, 4.20]
a, b	0.29	0.22	0.68	1	[3.73, 4.14]
a, b, ofa, ofb	0.26	0.18	0.28	0.40	[3.78, 4.14]

Table 1: Achieved refinements dependent on the observed variables including noise with $\mu = 0$ and $var = 1$.

Conclusion

We have presented a method for monitoring dynamic systems that exhibit both discrete and continuous be-

obs. vars.	$a(t)$	$b(t)$	f	g	t_c
b	1	0.70	1	1	[2.50, 4.25]
a, b	0.64	0.56	0.96	1	[2.87, 4.25]
a, b, ofa, ofb	0.61	0.47	0.57	0.58	[2.98, 4.25]

Table 2: Achieved refinements dependent on the observed variables including noise with $\mu = 0$ and $var = 2$.

haviors. The monitoring system refines the behavior prediction, the underlying model and the time uncertainty of discontinuous changes. The hypothesis is refuted and pruned from the tracking set when refinement eliminates all possible values for any parameter.

Trend matching uses a statistical best fit to observed data, plus bounding envelopes out to any desired confidence bound. Portions of the model space are removed only when they are inconsistent with these bounds. This gives a good (and adjustable) compromise between aggressiveness and robustness in handling noise and uninformative data.

Related work has been done by (Mosterman *et al.* 1998). In their framework for model-based diagnosis the physical system is modeled by a temporal causal graph derived from a bond graph representation. Reasoning is only performed at the qualitative level and no model refinement is done. TrenDx (Haimowitz & Kohane 1993) is a monitoring system which uses a semi-quantitative representation of behavior and attempts to fit data to this behavior representation. Since TrenDx uses pre-defined behavior templates no refinement can be performed. PRET (Bradley & Stolle 1996) automatically constructs a precise ODE model of a physical system. PRET focuses on system identification and not on monitoring.

Furthermore, our monitoring method is directly applicable to fault diagnosis in dynamic systems. Fault hypotheses can be proposed for monitoring based on initial weak information such as the signs of discrepancies between observations and predictions, by using existing methods such as (de Kleer & Williams 1987; Ng 1991). Automatic model-building methods can select relevant model-fragments from a background knowledge base to express initially weak knowledge about a fault as an SQDE (Crawford, Farquhar, & Kuipers 1990; Rickel & Porter 1994). The observation stream is then used to refine or refute each proposed model.

Acknowledgments

This work has taken place in the Qualitative Reasoning Group at the Artificial Intelligence Laboratory, The University of Texas at Austin. Research of the Qualitative Reasoning Group is supported in part by NSF grants IRI-9504138 and CDA 9617327, by NASA grant NAG 9-898, and by the Texas Advanced Research Program under grants no. 003658-242 and 003658-347. Bernhard Rinner is supported by the Austrian Science Fund under grant number J1429-MAT.

References

Bradley, E., and Stolle, R. 1996. Automatic Construction of Accurate Models of Physical Systems. *Annals of Mathematics of Artificial Intelligence* 17:1–28.

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

de Kleer, J., and Williams, B. C. 1987. Diagnosing Multiple Faults. *Artificial Intelligence* 32:97–130.

Dvorak, D., and Kuipers, B. 1991. Process Monitoring and Diagnosis: A Model-Based Approach. *IEEE Expert* 5(3):67–74.

Haimowitz, I. J., and Kohane, I. S. 1993. Automated Trend Detection with Alternate Temporal Hypotheses. In *Proceedings of the 13th International Conference on Artificial Intelligence*, 146–151.

Iwasaki, Y.; Farquhar, A.; Saraswat, V.; Bobrow, D.; and Gupta, V. 1995. Modeling Time in Hybrid Systems: How Fast Is "Instantaneous"? In *Proceedings of the Ninth International Workshop on Qualitative Reasoning.*

Kay, H., and Ungar, L. H. 1993. Deriving Monotonic Function Envelopes from Observations. In *Proceedings of the Seventh International Workshop on Qualitative Reasoning about Physical Systems*, 117–123.

Kay, H. 1996. *Refining Imprecise Models and Their Behaviors.* Ph.D. Dissertation, University of Texas at Austin.

Kay, H. 1997. Robust identification using semiquantitative methods. In *IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes.*

Kay, H. 1998. SQSIM: A Simulator for Imprecise ODE Models. *Computers and Chemical Engineering* 23(1):27–46.

Kuipers, B. 1994. *Qualitative Reasoning: Modeling and Simulation with Incomplete Knowledge.* MIT Press.

Mosterman, P. J.; ; Zhao, F.; and Biswas, G. 1998. An Ontology for Transitions in Physical Dynamic Systems. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence*, 219–224.

Ng, H. T. 1991. Model-Based, Multiple-Fault Diagnosis of Dynamic, Continuous Physical Devices. *IEEE Expert* 6(6):38–43.

Nishida, T., and Doshita, S. 1987. Reasoning about Discontinuous Change. In *Proceedings of the Sixth National Conference on Artificial Intelligence*, 643–648. Seattle, Washington.

Rickel, J., and Porter, B. 1994. Automated Modeling for Answering Prediction Questions: Selecting the Time Scale and System Boundary. In *Proceedings of the Twelfth National Conference on Artificial Intelligence*, 1191–1198. Seattle, Washington: AAAI.