appears in: Proceedings of AAAI 94, Seattle, 1994

Prediction Sharing Across Time and Contexts

Oskar Dressler and Hartmut Freitag

Siemens AG, Corporate Research & Development, Software and Engineering

Otto-Hahn-Ring 6

D-81730 Munich, Germany

{dressler, freitag}@zfe.siemens.de

Abstract

Sometimes inferences made at some specific time are valid at other times, too. In model-based diagnosis and monitoring as well as qualitative simulation inferences are often re-done although they have been performed previously. We propose a *new method for sharing predictions* done at different times, thus mutually cutting down prediction costs incurring at different times. Furthermore, we generalize the technique from 'sharing predictions across time' to 'sharing predictions across time *and* logical contexts'. Assumption-based truth maintenance is a form of sharing predictions across logical contexts. Because of the close connections to the ATMS we were able to *use* it as a means for implementation. We report empirical results on monitoring different configurations of ballast water tanks as used on offshore platforms and ships.

1. Introduction

Successfully deploying model-based diagnosis systems for complex technical devices ultimately requires an on-line coupling with the artifacts via sensors and actuators. In the field, instead of being manually activated when a malfunction occurs, as a first task an automatic on-line diagnosis system has to decide whether there actually is a diagnosis problem. Does the behavior deviate from the specified normal operation? Only then the diagnosis process will start. *A preceding monitoring phase is required.*

Once the faulty components have been identified, an integrated monitoring and diagnosis system may be allowed to switch back to monitoring mode interpreting the measurements coming from the sensors under the hypotheses that the identified components are broken. Monitoring and diagnosis may thus be interleaved.

Consider the application from (Dressler et al. 1993) depicted in figure 1. A collection of ballast tanks of various sizes is placed at different locations on a ship or offshore platform (the complete system comprises 40 tanks). Depending on load, wind and sea motion, water is pumped into or out of some of the tanks or the sea. This can be a rather time consuming process. For example, in our application on a crane ship, filling three tanks as shown in figure 1 can last up to 1.5 hours. At any time failures may occur potentially causing catastrophic damage. A broken pressure

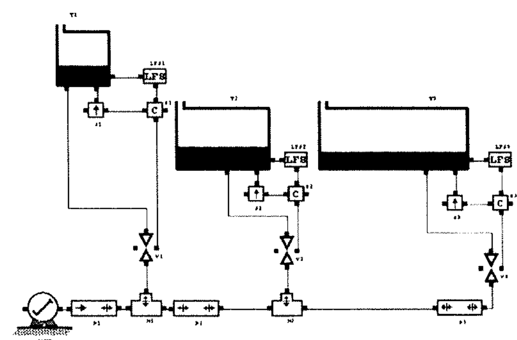

Figure 1. A ballast tank system

sensor, for instance, may enable or disable the automatic closing of a valve, thus causing an overflow or critical unbalance[1].

A stream of data coming from the system gives us values for pressures, valve status, float switch status and pump activity. This data is processed in a conventional way, such that we can assume to have derivatives for these values, too.

When observed behavior is to be classified as normal or faulty, a model is an invaluable asset. It allows predicting values for system variables, hence generating expectations about behavior when data, even if incomplete, becomes available.

But the purpose of models for monitoring and diagnosis is substantially different. While the former are only needed to *detect* malfunctions, the latter must have enough detail for *localizing* malfunctioning components. Diagnosis systems like DP (Struss 1992) and Magellan (Böttcher, Dressler 1993), however, can use multiple models of different granularity during one and the same diagnosis session. Their diagnosis process starts with coarser models that are suitable for monitoring, too.

In this paper we focus on the prediction task from the viewpoint of dependency-based diagnosis. Dependencies

1. The shipwreck of the polish ferry 'Jan Heweliusz' in January 1993 is suspected to be caused by an incorrect filling of ballast tanks.

are necessary for tracing back contradictory derivations to their origins. This allows diagnosis engines such as GDE (de Kleer, Williams 1987), GDE$^+$ (Struss, Dressler 1989), Sherlock (de Kleer, Williams 1989), (de Kleer 1991) and others to *first* identify conflicting assumption sets and *then* to generate diagnoses.

We use qualitative models for both, diagnosis and monitoring:

- For consistency-based diagnosis engines they prove to be especially useful; more detailed models become obsolete, when a qualitative abstraction of them has been refuted (Struss 1992). There often is no need to explore further details.
- For monitoring only significant deviations from normal operation are of interest. Using a qualitative model for the normal mode one can capture the complete set of good behaviors instead of just a single one.

When no discrepancies between observed and expected behavior are detected, the empty diagnosis is computed meaning that every component is working correctly. With this in mind, we can view monitoring as *'diagnosis without discrepancies'*.

For on-line coupling prediction is necessary at the rate of incoming data. Speed is of prime interest. Allowing a fault to go undetected potentially leads to catastrophe. Ideally, the consistency check, i.e. prediction, should be carried out at the sampling rate of the sensors, say every 10 seconds.

Suppose we are monitoring the process of filling the tanks. Incoming real values are first mapped to their qualitative abstractions, and then fed into the prediction machinery. This we can afford to do every, say two minutes, depending on the cost for running the model. Using qualitative models, most of the time nothing changes in qualitative terms, since different real values are mapped to the same qualitative value. Therefore, we end up making more or less the *same* predictions every two minutes while we would like a higher sampling rate and do prediction with *new* values only.

We propose a new technique for caching and generalizing previously made predictions. It allows carrying over predictions from one time to another. An inference made at a specific time in the past "generalizes" to the same inference made at *all* possible times. There is no need to ever make an inference twice. Consequently, prediction can be much faster when relevant previous inferences have been made. Due to the use of qualitative models this happens all of the time. Intuitively, only when a monitored variable changes its qualitative value, *new* predictions have to be made. In the ballast tanks application (Dressler et al. 1993), the sampling rate during the monitoring phase went down from 2 minutes to 2 seconds !

In the next section we introduce the key concept of *'prediction sharing across time'*. Section 3 generalizes our results by combination with another popular concept called *'prediction sharing across contexts'*, commonly known as ATMS (de Kleer 1986). In section 4 delayed consequences of inferences are considered and built into the framework. Finally, in section 5, we discuss empirical results we obtained for different configurations of ballast tanks.

2. Prediction Sharing Across Time

2.1. Temporally Generic Formulae

The system description *SD* is temporally generic in the sense that it describes behavior independent of the specific time at which, for example, a filling process takes place. For example, the model for the normal mode of a valve looks like follows:

$ok\,(valve) \rightarrow\ valve.status = valve.cmd$
$ok\,(valve) \rightarrow\ valve.[i_1] = -\, valve.[i_2]$
$ok\,(valve) \rightarrow\ (valve.status = close \rightarrow valve.[i_1] = valve.[i_2] = 0\,)$
$ok\,(valve) \rightarrow\ (\,valve.[i_1] \neq 0 \rightarrow valve.status = valve.cmd = open\,)$

The meaning of the variables is: *valve.cmd*: control input, *valve.status*: valve's state output, *valve.i1*: flow into the left/top end, *valve.i2*: flow into the right/bottom end. Square brackets [.] indicate qualitative variables. The complete models can be found in (Dressler et al. 1993).

In general, inferences made from such descriptions have the form

$$\alpha_1\,(t) \wedge \ldots \wedge \alpha_2\,(t) \rightarrow \beta\,(\Delta\,(t)\,)$$

where α_i and β are propositional atoms, temporal index t refers to some specific time and $\Delta\,(t)$ denotes another temporal index. We call Δ a *delay function*, but allow for negative delays. Therefore, we may draw conclusions about past system variable values, too.

A component c operating in mode $m(c)$ is assumed to exhibit the same behavior at every specific time index given that the same values are fed into it. This includes that the delay function is also independent of the specific time. Assuming linear time, we can depict the situation in figure 2.

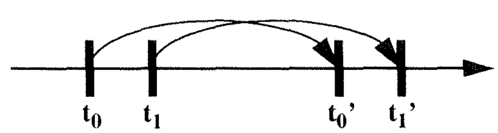

Figure 2. A time-independent delay-function

$$\forall x.\ [t_1 = t_0 + x \rightarrow \Delta\,(t_0) + x = \Delta\,(t_0 + x\,) = \Delta\,(t_1)] \wedge$$
$$[\,[\,\alpha_1(t_0) \wedge \ldots \wedge \alpha_n\,(t_0) \rightarrow \beta(\Delta(t_0))]$$
$$\rightarrow [\,\alpha_1(t_0 + x) \wedge \ldots \wedge \alpha_n\,(t_0 + x) \rightarrow \beta(\Delta(t_0 + x))\,]]$$

More generally, assuming arbitrary delay functions Δ_1 and Δ_2, 'independence of an inference of the specific time' is expressed as:

$$\forall\Delta_1\,\forall\Delta_2\,\forall\,t.[\,\Delta_2\,(\Delta_1\,(t)) = \Delta_1\,(\Delta_2\,(t))\,] \wedge$$
$$[\,[\alpha_1(t_0) \wedge \ldots \wedge \alpha_n\,(t_0) \rightarrow \beta(\Delta_1(t_0))\,]$$
$$\rightarrow [\alpha_1(\Delta_2(t_0)) \wedge \ldots \wedge \alpha_n\,(\Delta_2(t_0)) \rightarrow \beta(\Delta_1(\Delta_2(t_0)))\,]]$$

A formula of the form $\alpha_1\,(t) \wedge \ldots \wedge \alpha_2\,(t) \rightarrow \beta\,(\Delta\,(t)\,)$

with propositional atoms $\alpha_1, \ldots, \alpha_n$ and β indexed by time t and $\Delta(t)$ where the delay function Δ adheres to this restriction is called *temporally generic* (or t-generic). Without loss of generality we assume the system description to be a set of t-generic formulae. Please note, that we are not committed to a specific ontology of time like time points or intervals.

For the rest of section 2 and section 3 the delay function considered is identity, i.e. no delays. In section 4 we show how delay is built into the framework.

2.2. Temporal Generalization of Single Instance Inferences

From the discussion so far it is clear that some inferences made at a specific time will be valid at other times, too. Thus, there is no need to re-do them when we employ an appropriate caching scheme.

We start from statements like proposition ϕ holding at time t_i, $\phi@t_i$, called *temporally indexed statements*.

In consistency-based diagnosis we are given a set of them, usually the observations *OBS* made at certain times and the assumptions Π that the corresponding components are working in a specific mode at a certain time. The task is to check the consistency of $SD \cup OBS \cup \Pi$ where *SD* is independent of time in the sense discussed before.

In qualitative simulation we are given a set of such statements for some initial time t_0. The task there is to enumerate possible evolutions of the system from this point on. Again, we are dealing with a system description *SD* which is independent of time and observations at a specific time t_0.

When we predict a value ϕ to hold at t_i, $\phi@t_i$, the underlying support consists of a set of t-generic formulae $SD' \subseteq SD$ and a set of sentences S holding at specific times $t_1, \ldots, t_n$: $SD' \cup S \models \phi$

Since we have restricted the delay functions to identity, all of these temporal indices are identical to t_i, i.e. $t_i = t_1 = \ldots = t_n$. It follows immediately that the derivation of ϕ can be generalized from the single time index t_i to sets of time indices.

Definition: The *temporal extent of* α, $TE(\alpha)$, denotes the set $\{t_i \mid \alpha \text{ holds at } t_i\}$.

The t-generic formulae in the system description hold at all times, but propositions about observed values etc. are only available at certain times.

Definition: A set GS of non-universally holding formulae is called *ground support for* ϕ iff there exists $SD' \subseteq SD$ such that $SD' \cup GS \models \phi$.

Lemma: If GS is a ground support for ϕ then $\bigcap_{\alpha \in GS} TE(\alpha) \subseteq TE(\phi)$

This means we can generalize a derivation of ϕ at a specific time t_i to the intersection of temporal extents of ϕ's support. Whenever all the propositions in *GS* hold at some time $t_j \neq t_i$ we know *without re-deriving* ϕ that it holds at t_j, too.

2.3. Symbolic Computation of Temporal Extents

For derived formulae ϕ which do not occur in temporally indexed statements, like e.g. $\phi@t_{13}$, the temporal extent can be computed *symbolically* by considering *all* ground support sets for ϕ, $GS(\phi)$.

Lemma: Let no explicit temporal statements about ϕ be available. Then

$$TE(\phi) = \bigcup_{S \in GS(\phi)} \bigcap_{\alpha \in S} TE(\alpha).$$

If explicit temporal statements about ϕ are available, we have to add these times.

Lemma: If explicit temporal statements about ϕ at times $t_1, \ldots, t_n$ are available, then

$$TE(\phi) = \bigcup_{S \in GS(\phi)} \bigcap_{\alpha \in S} TE(\alpha) \cup \{t_1, \ldots, t_n\}.$$

For the propositions α that may occur in the ground support of derived formulae we introduce symbols TE_α to represent $TE(\alpha)$. These propositions α are exactly the propositions for which we have temporally indexed statements. The symbols TE_α are called *temporal base symbols*. In model-based diagnosis this means we are creating these symbols for the observable values and for the modes of components. In qualitative simulation the qualitative values of the initial state are treated in this way.

Using these symbols each atom is labelled with a *unique* symbolic representation of its temporal extent.

Definition: *Temporal Label*

A set of symbol sets, $\{\{TE_{\alpha_{11}}, \ldots, TE_{\alpha_{1n}}\}, \ldots, \{TE_{\alpha_{m1}}, \ldots, TE_{\alpha_{mk}}\}\}$, is called *temporal label* of ϕ, $TL(\phi)$, iff

[*Correctness*]

Each set $\{TE_{\alpha_{i1}}, \ldots, TE_{\alpha_{ij}}\}$ is a ground support for ϕ.

[*Completeness*]

If S is a ground support for ϕ, then there exists $\{TE_{\alpha_{i1}}, \ldots, TE_{\alpha_{ij}}\}$ in $TL(\phi)$ such that $\{TE_{\alpha_{i1}}, \ldots, TE_{\alpha_{ij}}\} \subseteq S$.

[*Minimality*]

For no i and j, $i \neq j$, $\{TE_{\alpha_{i1}}, \ldots, TE_{\alpha_{ik}}\}$ is a subset of $\{TE_{\alpha_{j1}}, \ldots, TE_{\alpha_{jm}}\}$.

[*Consistency*]

For no i, $\{TE_{\alpha_{i1}}, \ldots, TE_{\alpha_{ik}}\}$ is a ground support for $\perp$.

2.4. Implementation

The similarities to logical labels as used in the ATMS (de Kleer 1986) are apparent and our implementation makes use of this fact. Simply defining the newly introduced symbols TE_α to be assumptions (in ATMS terminology) suffices. The ATMS will then compute the temporal labels as defined. The relation to the ATMS is very close as we shall see in the next section.

3. Prediction Sharing Across Time and Contexts

In section 2 we have seen how from statements such as e.g. $\alpha@t_1$ and $\alpha@t_2$ the temporal information is factored out and handled separately from the proposition α's content: we compute α's temporal label. In systems like TCP (Williams 1986), HEART (Joubel, Raiman 1990), EEP (Guckenbiehl 1991) and TARMS (Holtzblatt et al. 1991) the above statements would be handled as two separate entities. This not only prevents these systems from sharing predictions across time as described in section 2. When these approaches are combined with assumption-based truth maintenance for the purpose of dependency-recording, they are hit by a multiplied exponential blowup: since the two statements are two separate entities, both of them have their own ATMS-label.

Our approach allows for a smooth integration with the ATMS. Actually, we have *used* the ATMS to implement it. After a brief review of the ATMS, we sketch how 'prediction sharing across time' is done. Then we extend the scheme to cover the usual prediction sharing across *logical* ATMS contexts.

3.1. Prediction Sharing Across Logical Contexts

The language of the ATMS (de Kleer 1986) consists of propositional horn clauses called *justifications*

$$\alpha_1 \wedge \ldots \wedge \alpha_n \rightarrow \beta.$$

A distinguished subset *ASSM* of the occurring propositional atoms *PROP* is called *assumptions*: $ASSM \subseteq PROP$. The set of atoms derivable from a set of assumptions (*environment*) E is called *(logical) context* of E and denoted by $cxt(E)$. All environments which allow deriving the constant $\perp$ are considered inconsistent.

Reasoning in *multiple contexts* then can be characterized as considering all consistent contexts $cxt(E)$ of all subsets $E \subseteq ASSM$ of the given assumptions. All propositions are labelled with the complete set of minimal (w.r.t. set inclusion) consistent environments from which they are derivable. I.e. for a proposition p its *(logical) label* is defined as

$$LL(p) = \{ E \subseteq ASSM |\ E \text{ consistent} \wedge p \in cxt(E) \wedge \forall E' \subset E\ p \notin cxt(E') \}$$

Justifications are used to record the inferences as performed by a problem solver, in our case a predictive engine. The label of a proposition is computed by propagating labels in the network of justifications using basic set operations. By caching inferences as justifications an inference is done once for some context and the results are shared by contexts characterized by superset environments, thus avoiding expensive re-computations.

The labels the ATMS must compute can grow big and hamper larger applications. Focusing on *interesting* contexts (Dressler, Farquhar 1990) avoids this problem while maintaining the essential properties of assumption-based truth maintenance.

3.2. The ATMS as a Mechanism for Maintaining Temporal Labels

As usual we use the ATMS for recording the inferences made by the problem solver, i.e. the predictive engine. When the antecedents $\alpha_1,...,\alpha_n$ simultaneously hold at some time t_i, the predictive engine will conclude that β holds, too, given the t-generic formula

$$\alpha_1(t) \wedge \ldots \wedge \alpha_n(t) \rightarrow \beta(t)$$

in SD. A justification $\alpha_1 \wedge \ldots \wedge \alpha_n \rightarrow \beta$ is then submitted to the ATMS, and for temporal base symbols ATMS assumptions are created. The following theorem shows that this suffices to compute temporal labels.

Theorem: Let *TBS* be the set of temporal base symbols, $TBS\text{-}ASSM \subseteq ASSM$ be the subset of ATMS assumptions corresponding to temporal base symbols, and Ψ: $TBS\text{-}ASSM \rightarrow TBS$ be the bijective mapping that associates assumptions with their corresponding symbols. Then

$$TL(\phi) = \{ \{ e' |\ e \in E \wedge e' = \Psi(e) \} \mid E \in LL(\phi) \}.$$

All that remains to be done is to record temporally indexed statements. To this end, we create symbols $EXT - TE_{\alpha_i}$ that denote the enumeration of times where α_i holds. Each temporal index t_i actually occurring in an observation like $X{=}15@t_i$ is treated as an assumption, too. Then justifying $EXT - TE_{\alpha_i}$ by temporal indices at which α_i holds like e.g.

$$t_{17} \rightarrow EXT - TE_{\alpha_i} \text{ and } t_{143} \rightarrow EXT - TE_{\alpha_i}$$

guarantees that the logical label of $EXT - TE_{\alpha_i}$ enumerates the appropriate times:

$$LL(EXT - TE_{\alpha_i}) = \{ \{t_{17}\}, \{t_{143}\}, \ldots \}.$$

Please note that the possibly large number of assumptions for times t_i does not cause an exponential growth of label sizes. $LL(EXT - TE_{\alpha_i})$ grows linearly and from $EXT - TE_{\alpha_i}$ no further propagation is possible.

Querying the system about the temporal extent of an atom ϕ proceeds in two stages. First, a lookup of ϕ's temporal label is done. Then the union of intersections of the enumerated temporal extents $EXT - TE_{\alpha_i}$ gives the answer. In a similar way queries about a specific time are processed.

Lemma:

$$\phi@t_i \text{ iff } \{t_i\} \in \bigcup_{tenv \in TL(\phi)} \bigcap_{TE_{\alpha_i} \in tenv} LL(EXT - TE_{\alpha_i})$$

Please, note that if temporally indexed statements about ϕ are available, then $\{TE_\phi\}$ is an element of $TL(\phi)$.

3.3. Combining Prediction Sharing Across Time with Assumption-based Truth Maintenance

Logical and temporal contexts are orthogonal concepts in the sense that they ought to be combinable without restriction. For example, we might want to state that α holds at t_{17} *but only under (logical) assumptions A* and *B*. There are two principal entry points for this type of statements with logical context qualifications. On the level of temporally indexed statements we need to capture conditions like the one above. On the level of recorded inferences we must be able to express that

$$\alpha_1(t) \wedge \ldots \wedge \alpha_n(t) \rightarrow \beta(t)$$

holds regardless of the time t as before, *but only under (logical) assumptions*, say A and B.

Temporally indexed statements can be qualified with logical context information by using justifications like

$A \wedge B \wedge t_{17} \rightarrow EXT-TE_{\alpha_i}$ instead of

$t_{17} \rightarrow EXT-TE_{\alpha_i}$.

Consequently, the logical label $LL(EXT-TE_{\alpha_i})$ is $\{\{t_{17}, A, B\}, \ldots\}$. Each environment contains exactly one temporal index assumption while the rest of its assumptions provides the desired logical context. Note, that the usual minimization and consistency maintenance done by the ATMS takes care of redundant and inconsistent information in $LL(EXT-TE_{\alpha_i})$.

On the level of recorded inferences the solution is equally simple. Logical assumptions, say A and B, are added to the antecedents:

$$A \wedge B \wedge \alpha_1 \wedge \ldots \wedge \alpha_n \rightarrow \beta$$

The temporal labels then are relative to the logical context.

Definition: *Temporal Label under Assumptions*

A set of symbol sets, $\{\{TE_{\alpha_{11}}, \ldots, TE_{\alpha_{1n}}\} \ldots, \{TE_{\alpha_{m1}}, \ldots, TE_{\alpha_{mk}}\}\}$, is called *temporal label* of ϕ under logical assumptions Θ, $TL(\phi, \Theta)$ iff

[*Correctness*]

Each set $\{TE_{\alpha_{i1}}, \ldots, TE_{\alpha_{ij}}\} \cup \Theta$ is a ground support for ϕ.

[*Completeness*]

If $S \cup \Theta$ is a ground support for ϕ with temporal base symbols S, then there exists $\{TE_{\alpha_{i1}}, \ldots, TE_{\alpha_{ij}}\}$ in $TL(\phi)$ such that $\{TE_{\alpha_{i1}}, \ldots, TE_{\alpha_{ij}}\} \subseteq S$.

[*Minimality*]

For no i and j, $i \neq j$, $\{TE_{\alpha_{i1}}, \ldots, TE_{\alpha_{ik}}\}$ is a subset of $\{TE_{\alpha_{j1}}, \ldots, TE_{\alpha_{jm}}\}$.

[*Consistency*]

For no i, $\{TE_{\alpha_{i1}}, \ldots, TE_{\alpha_{ik}}\} \cup \Theta$ is a ground support for $\bot$.

Given this relative notion of temporal label the theorem from 3.2 changes, too.

Theorem: Let *TBS* be the set of temporal base symbols, *TBS-ASSM* $\subseteq$ *ASSM* the subset of ATMS assumptions corresponding to temporal base symbols, Ψ: *TBS-ASSM* $\rightarrow$ *TBS* the bijective mapping that associates assumptions with their corresponding symbols and Θ a set of logical assumptions. Then

$$TL(\phi, \Theta) = \{\{e' \mid e \in E \cap TBS\text{-}ASSM \wedge e' = \Psi(e)\} \mid E \in LL(\phi) \wedge (E \setminus TBS\text{-}ASSM) \subseteq \Theta\}$$

The two stage approach to answering queries remains. The evaluation, however, is done relative to the logical context specified as part of the query.

Lemma: $\phi@t_i$ under assumptions θ iff

$$\{t_i\} \cup S \in \bigcup_{tenv \in TL(\phi, \theta)} \bigcap_{TE_{\alpha_i} \in tenv} LL(EXT-TE_{\alpha_i}) \wedge S \subseteq \theta$$

4. Delayed Consequences

Delayed consequences are required to model a component such as a valve which receives e.g. an 'open' command and then changes to state 'open' *after some time*. Generally, in qualitative simulation (Kuipers 1986) the interstate behavior, i.e. P- and I-transitions, requires delay. An inference with delayed consequent

$$\alpha_1(t) \wedge \ldots \wedge \alpha_2(t) \rightarrow \beta(\Delta(t))$$

is handled specially. No direct translation into a justification is possible. Instead, for the atom β in the delayed consequent a temporal base symbol TE_β is introduced and handled as before: an assumption is created and also a symbol $EXT\text{-}TE_\beta$ to denote the extensional description of times and logical contexts where β holds. A simple demon mechanism guarantees that, whenever the conjunction of α_i holds at some t_j under assumptions Θ, $\{\Delta(t_j)\} \cup \Theta$ is recorded by $EXT\text{-}TE_\beta$, i.e. an assumption $\Delta(t_j)$ and the justification

$\Theta_1 \wedge \ldots \wedge \Theta_n \wedge \Delta(t_j) \rightarrow EXT\text{-}TE_\beta$ with $\Theta_1, \ldots, \Theta_n \in \Theta$

are created.

5. Empirical Results

We experimented with a variety of ballast tank configurations to provide evidence that we have actually met our goal of reducing prediction costs when relevant inferences have been made previously. In the figures below we show run time and number of necessary new predictive inferences (y-axis) as they develop over time (x-axis). As a measure for new predictive inferences we have chosen the increment of the number of justifications submitted to the system. This, however, can only be an approximation of the really necessary efforts since a single justification may cause a huge amount of label propagation. In the figures the dashed curve shows the new justifications while the solid line indicates prediction time.

The correlation between run time and number of necessary new predictions is apparent. All our experiments on different configurations of ballast tanks show the same pattern: In the beginning the prediction cost (run time) is substantial. No previous predictions have been cached and every possible derivation has to be done explicitly. Later on *when* a number of *variables change their qualitative value*, prediction cost increases but does not reach the initial cost. Without prediction sharing run time is in the range of the initial cost all of the time !

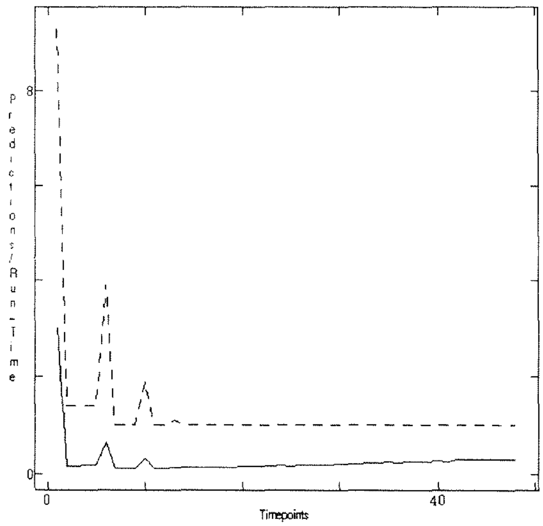

Figure 3. Monitoring the 3-Tank system from figure 1, considering 10 different test vectors occuring at most 10 times. Prediction time for first timepoint: 3.04s, average for additional timepoints: 0.21s

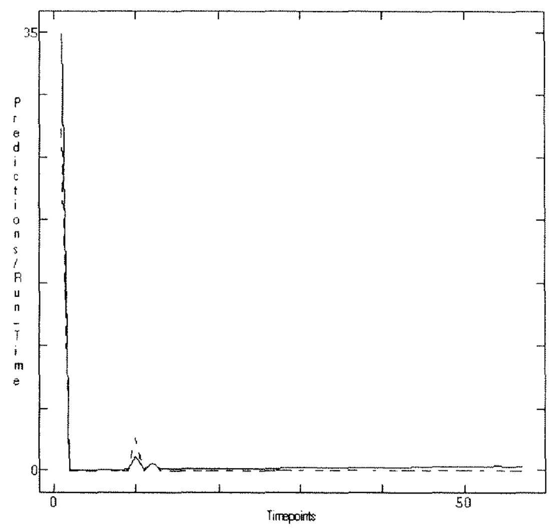

Figure 4. Monitoring a 20-Tank system, considering 10 different test vectors occuring at most 10 times. Prediction time for first timepoint: 36.3s, average for additional timepoints: 3.8s

Currently we attribute the slow, seemingly linear increase of runtime to the monotonically increasing number of justifications and assumptions. Simply handling these structures requires some time. For example, after 57 timepoints the systems maintains 12792 justifications and 405 assumptions for the 20 tanks system. This suggests that we reduce the amount of recorded past data by introducing a time window for relevant data.

Acknowledgements. We would like to thank Anton Beschta, Thomas Guckenbiehl, Wera Klein, Michael Montag and Peter Struss for comments on earlier drafts and discussions on the subject of the paper. This work has been supported by the BMFT, project BEHAVIOR, ITW 9001 A9.

References

C. Böttcher and O. Dressler. Diagnosis Process Dynamics: Holding the Diagnostic Trackhound in Leash. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*. Morgan Kaufmann Publishers, 1993.

J. de Kleer. An Assumption-based TMS. *Artificial Intelligence*, 28:127–162, 1986.

J. de Kleer. Focusing on Probable Diagnoses. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pages 842–848, Anaheim, 1991. Morgan Kaufmann Publishers.

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

J. de Kleer and B. C. Williams. Diagnosis with Behavioral Modes. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1324–1330. Morgan Kaufmann Publishers, 1989.

O. Dressler and A. Farquhar. Putting the problem solver back in the driver's seat: Contextual control over the ATMS. In M. Reinfrank J.P. Martins, editor, *Truth Maintenance Systems*, pages 1–16. Springer LNAI 515, 1990.

O. Dressler, C. Böttcher, M. Montag and A. Brinkop. Qualitative and Quantitative Models in a Model-Based Diagnosis System for Ballast Tank Systems. In *Proceedings of the International Conference on Fault Diagnosis (TOOLDIAG)*, pages 397–405, Toulouse, France, April 1993.

T. Guckenbiehl. The extended episode propagator, version 2-preliminary report. Internal Report, FhG-IITB, 1991.

L. Holtzblatt, M. Neiberg, R. Piazza, and M. Vilain. Temporal Methods: Multi-Dimensional Modeling of Sequential Circuits. In L. Console, editor, *Working Notes of the International Workshop on Principles of Diagnosis (DX)*, Technical Report RT/DI/91-10-7, pages 111–120. Dipartimento di Informatica, Universita da Torino, Italy, 1991.

C. Joubel and O. Raiman. How Time Changes Assumptions. In *9th European Conference on Artificial Intelligence (ECAI 90)*, Stockholm, 1990.

B. Kuipers. Qualitative Simulation. *Artificial Intelligence*, 29(3):289–338, 1986.

P. Struss. Diagnosis as a Process. In W. Hamscher, J. de Kleer, and L. Console, editors, *Readings in Model-based Diagnosis*. Morgan Kaufmann Publishers, 1992.

P. Struss. What's in SD? Towards a Theory of Modeling for Diagnosis. In W. Hamscher, J. de Kleer, and L. Console, editors, *Readings in Model-based Diagnosis*. Morgan Kaufmann Publishers, 1992.

P. Struss and O. Dressler. Physical Negation – Integrating Fault Models into the General Diagnostic Engine. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1318–1323. Morgan Kaufmann Publishers, 1989.

B. C. Williams. Doing Time: Putting Qualitative Reasoning on Firmer Ground. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, Philadelphia, Penn., August 1986.