Presentation at the 5th International Workshop on Qualitative Reasoning, QR-91
Austin, Texas, May 19-22, 1991

A Theory of Model Simplification and Abstraction for Diagnosis

Draft (4/19/91)

Please, Do not Distribute! Comments welcome.

Peter Struss
Siemens AG
ZFE IS INF22
Otto-Hahn-Ring 6
D-8000 Muenchen 83
Germany

ph.: (..49) 89 636 2414
fax: (..49) 89 636 42284
e-mail: struss@ztivax.siemens.com

Abstract

Recent work on model-based diagnosis has mainly focused on problems of the diagnostic procedure, tackling the task "*Determine diagnostic candidates given the system description and the set of observations*". In this paper, we address the modeling problem, i.e. "*How to structure the system description, i.e. our knowledge about the structure and behavior of a technical system, appropriately for diagnostic purposes*?". Although considered as an essential requirement from the very beginning of research in model-based diagnosis, the actual use of multiple models at different levels of granularity, abstraction, coverage and expressiveness, and in particular the use of qualitative models in diagnostic systems is very limited. The main concern of this paper is to investigate the nature and impact of abstraction, simplification, and approximation of models and clarify their distinctions. For this purpose, we develop a theoretical framework including, as a prerequisite, a formalization of what we mean by the term "model". The paper outlines the role of these concepts in consistency-based diagnosis by considering them as instances of more general relations between models.

1 Introduction

Already at the origin of research on model-based reasoning, there was a strong understanding that the models required to solve significant problems should be highly structured, with its elements reflecting different aspects and purposes, various levels of structural and descriptive granularity and, in particular, involving qualitative models (see, for instance [Davis 82]). However, an analysis of theories and systems developed for model-based diagnosis, the most prominent sub-field in model-based reasoning, shows that most approaches are focused on problems of the diagnostic procedure, presuming there exists a simple and unique way of modeling the device or avoiding to specify its nature. Some work has been done on exploiting **fault models** ([Hamscher 90], [Struss 88b], [de Kleer-Williams 89], [Struss-Dressler 89]), **hierarchy** ([Davis 84], [Hamscher 90], [Struss 88a, 89b]) and different **views** on component models ([Davis 84], [Struss 88a,b], [Hamscher 90]), aiming, in particular, at reducing the complexity of reasoning about non-trivial artifacts. However, some of the most powerful means for this purpose, namely reasoning with **abstractions, simplifications and approximations**, has not been deeply investigated and exploited so far, and moreover, we lack a coherent theory which addresses these issues and which fits the formalisms of model-based diagnosis.

Representing knowledge about physical systems at an adequate and intuitive level of detail is a concern of qualitative physics, and recently the problems of (automated) abstraction and simplification of models and of selecting models appropriate for a given task have become one focus of research in this area (see e.g. [Falkenhainer-Forbus 90], [Weld 90], [Weld-Addanki 90]) . But, on the one hand, this work has been performed for other tasks than diagnosis (e.g. for answering queries), and, on the other hand, there is still no uniform understanding of terms like abstraction and approximation which often results in a phenomenological characterization and in an arbitrary, sometimes even interchangeable use of such concepts. This due to the lack of an agreed formalization which, more importantly, results in limitations of a rigorous analysis of the creation and use of such model transformations and their impact, for instance, on model-based diagnosis. This paper provides a theoretical basis for achieving this goal and offers definitions for the important concepts which we feel to be sufficiently general, but still to capture the important distinctions. Although the theory is motivated by problems of model-based diagnosis, it is of interest for model-based reasoning in general and, in particular, important for the use of qualitative models.

Overview

Section 3 briefly summarizes the background of model-based diagnosis in general, and of our diagnostic framework, DP, in particular. In section 4, we define and discuss the

concepts of abstraction, simplification, and approximation of models. For this purpose, we first have to define the term model. This is done by formalizing the widely used view of a model being specified by relations among system variables. We will also show in this section that intermittent behavior can be covered to some extend and that the treatment of data interpretation can be reflected in relational models. In the last section, abstraction and simplification of relational models are considered as instances of more general logical relations among models. Based on this, we outline their impact on model-based diagnosis and on the development of diagnostic strategies for the selective use of multiple models.

To illustrate the problems we want to tackle, we will start with an example from a case study, THYC, an instance of GDE$^+$ ([Struss-Dressler 89]) that diagnoses thyristors in bridge converters ([Struss 89b]).

2 A Motivating Example

A thyristor is a semi-conductor with anode, A, cathode, C, and gate, G (Fig. 1a) which operates as a (directed) switch: it works in two states, either conducting current in a specified direction with almost zero resistance, or blocking current like a resistor with almost infinite resistance, as is indicated by the characteristic curve in Fig 1b.

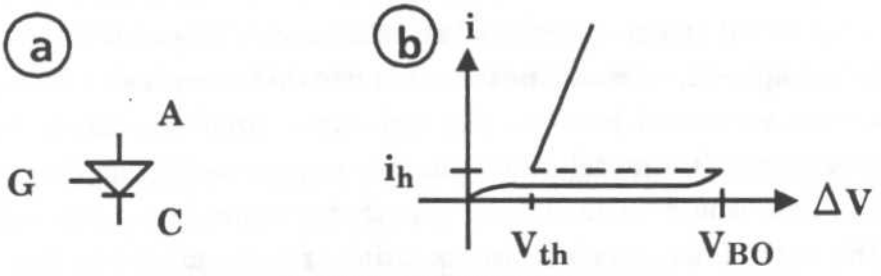

Figure 1 The thyristor

The transition from the OFF state to ON is controlled by the gate; if it receives a pulse the thyristor "fires", provided the voltage drop exceeds a threshold, V_{th}. This state ends when the current through the thyristor drops below the holding current, i_h.

This is the first description of a thyristor one probably gets out of a textbook, and this is the model (let us call it TH_{SWITCH}) that is used in THYC. It turned out that - without extensive measurements - the system suspected each single component, even though the information enabled a human expert to pinpoint the faulty thyristor. The situation changed when the system was provided with models about the possible faults a thyristor might exhibit: PUNCTURED, i. e. acting like a wire, or BLOCKING like an open switch (Deviations of the thresholds from the nominal ones might also be considered). THYC concludes that if none of these fault models, $TH_{PUNCTURED}$ and $TH_{BLOCKING}$, is consistent with the observations, the thyristor must be ok; this works fine for a class of common devices and failures.

However, in some circuits and situations, the thyristor is fired when the voltage drop **exceeds the breakover voltage**, V_{BO}, as is indicated by the characteristics in Fig. 1b. From this perspective, TH_{SWITCH} is only an approximation of this extended model, TH_{THRESH}, under which some conclusions of TH_{SWITCH} are no longer valid. Note also, there are additional faults to be considered: e. g. a decrease in V_{BO} with respect to its nominal value. Still, there is a more detailed model of the thyristor. Ungated firing may also be caused by a **voltage drop changing** too fast (independent of its magnitude), giving rise to another improvement of the model, $TH_{dV/dt}$.

And yet, the models considered so far are confined to a purely electrical view on the component, ignoring, for instance, **thermal** properties which also might influence the electrical behavior (For instance, the leakage current is temperature-dependent, and there may even be thermal triggering). Considering state transitions to occur instantaneously is yet another simplification; dropping it requires treatment of **temporal information**, such as delay time, rise time, etc. Finally, let us merely mention that a different kind of analysis might require looking in more detail at the **structure** of a thyristor, using two coupled transistors as an **analogue**, etc.

The point we want to make here is that, even if we are able to develop a detailed model, that accounts for all aspects mentioned and that covers all possible situations we might encounter in diagnosis, we would **not want to use this complex universal model at all times**, because we do not have to. The majority of problems can be solved using the simplified versions of the model, and it would be unnecessarily complex or even infeasible if all the details would be included (Note that a more detailed model may not only increase the cost of inferences but also potentially require more variables to be measured which may be expensive to obtain).We would rather want a diagnostic system to mimic a human expert whose skills include choosing the right level of detail and simplifying the problem in an appropriate way. This requires representing the various chunks of the model separately, enabling the diagnostic system to focus on the relevant parts, and combining the results obtained from the use of different models.

In the following section, we briefly summarize our background for addressing these issues - the principles of consistency-based diagnosis and our diagnostic framework, DP.

3 Model-Based Diagnosis Background

3.1 Consistency-Based Diagnosis

The consistency-based approach is oriented towards an assignment of behavioral modes (correct or faulty) to the constituents (components) of the artifact which is consistent with the system description (SD), i.e. the model, and the observations (OBS). A diagnosis is then given by a set of faulty components, $\Delta \subseteq COMPS$ such that

$$SD \cup OBS \cup \bigcup_{C \in \Delta} FAULTY(C) \cup \bigcup_{C \in COMPS \setminus \Delta} CORRECT(C)$$

is consistent ([Reiter 87], [de Kleer-Mackworth-Reiter 90]). Finding possible diagnoses is strongly driven by exploiting known *conflicts*. A conflict is a set of mode assignments, $\cup mode_{k_i}(C_i)$, to a number of components, C_1, ..., $C_n \in COMPS$, that leads to an inconsistency with SD ∪ OBS:

$SD \cup OBS \cup \cup mode_{k_i}(C_i) \vdash \bot$,

which is detected mainly through the derivation of contradictory values of one parameter. The main focus of work in this area and the subject where considerable progress has been achieved concerns the problem: *Determine diagnostic candidates given* **the** *system description, SD, and* **the** *set of observations, OBS*. In practice, this turns out to be too restricted.

3.2 DP-Diagnosis as a Process

The argument for multiple models in the descussion of the thyristor example implies that there is a **set of** system descriptions instead of a single one. Furthermore, we do not only face the problem of **selecting one** SD appropriate for a given talk (which is the concern of [Falkenhainer-Forbus 91]). Rather, the system is required to **exploit several models** (in parallel or sequentially), raising the question

- How and under what conditions do results obtained from one model carry over to diagnostic reasoning with another model?

Moreover, different models in use may not be complementary, but conflicting, some may be wrong, posing even harder problems such as

- How can the system deal with wrong information derived from an inappropriately simplified model? How can it handle contradictory parts of the model?

We are not simply talking about models that are absolutely wrong or inconsistent. The problem is that they suffice in some situations but are inappropriate for other cases, i.e. the design of the models was (consciously or unconsciously) based on some assumption about their context or the problem to be solved. Such modeling assumptions (and, in particular, simplifying assumptions) are only one instance of a number of **diagnostic assumptions** involved in each diagnostic process, such as assumptions about independence and non-intermittency of faults, completeness of knowledge about possible faults, and the system structure being unchanged. The problem is that they are present only in an implicit, hardwired form, and, hence, cannot be subject to reasoning and be retracted. We have developed an approach to handling such diagnostic assumptions and a system, DP (for "Diagnosis as a Process"), that, accordingly, can serve as a basis for using multiple and simplified models ([Struss 88a], [Struss 89a], [Beschta et al. 90]).

DP introduces another element to the theory: the **set of diagnostic hypotheses**, DHYP, which represent working hypotheses that guide and focus the problem solving process unless they are recognized to be inadequate and dropped. This includes simplifying assumptions and, when working with multiple models, modeling assumptions. A diagnosis is now defined as the union of a set of faulty components, Δ_{COMPS}, **and a set of retracted diagnostic hypotheses**, Δ_{DHYP}, such that

$$SD \cup OBS \cup DHYP \setminus \Delta_{DHYP} \cup \neg\Delta_{DHYP} \cup \bigcup_{C \in \Delta_{COMPS}} FAULTY(C) \cup \bigcup_{C \in COMPS \setminus \Delta_{COMPS}} CORRECT(C)$$

is consistent, where $\neg\Delta_{DHYP} := \{\neg dhyp \mid dhyp \in \Delta_{DHYP}\}$. In other words, we search for a mode assignment to constituents that is consistent with SD ∪ OBS **under certain assumptions**. It is crucial that DP does not generate all possible diagnoses, but only those specified by a **focus of suspicion,** which can be used to treat sets of diagnostic assumptions as defaults.

We want to emphasize that, on the one hand, existing consistency-based systems can be regarded as instances of DP. On the other hand, it is easy to realize that assigning TRUE or FALSE to diagnostic hypotheses can be viewed as an analogy to mode assignments to constituents. This provides us with the basis for the implementation of DP, since we can apply an existing diagnostic engine, in our case GDE$^+$ ([Struss-Dressler 89]), to debug the diagnostic hypotheses as well as the device.

4 Relational Models

Models are a representation of our knowledge about the real-world behavior of real-world systems or processes that can be used in order to derive a more complete description of an actual behavior given some partial information. For instance, we know the device input and want to infer the output, or, for some initial conditions, we want to determine the subsequent behavior. We do not necessarily require the derived information to be complete and unambiguous. Ruling out some possibilities may be of sufficient value.

4.1 Models and Behavioral Modes

We take the view that

- a system is composed of some behavioral constituents (such as components or processes),
- the system's behavior is established by the behaviors of its constituents,
- the behavior of some constituent, C, can be specified in terms of local variables $v_i \in VARS(C)$, i. e. by a tuple, $\underline{v}_C = (v_1, v_2, ..., v_k)$.

(Locality includes that, conceptually, two different constituents do not have variables in common; connectivity is established by stating equality of variable values in SD). If

$DOM(v_i)$ denotes a possible domain of v_i, then

$$DOM(\underline{v}_c) := DOM(v_1) \times DOM(v_2) \times ... \times DOM(v_k)$$

is a space of (theoretically) possible behaviors, and a certain mode of behavior is given by some relation $R \subseteq DOM(\underline{v}_c)$.

This means, the physical condition of the constituent (e. g. a thyristor being correct or punctured) restricts the physically possible behaviors to some subset of $DOM(\underline{v}_c)$. In whatever situation (e. g. given by particular test vectors) we inspect the constituent, the observed (or inferred) value of $\underline{v}_c$ lies within R. Additionally, the environment of the device (e. g. the outside temperature) may impose further restrictions on what we encounter in reality. The set of situations which are physically possible due to the actual **physical condition** of a **constituent** and, perhaps, due to **environmental conditions** will be denoted SIT.

If, in a particular situation, $s \in SIT$, $\underline{v}_c$ has the value $\underline{v}_o \in DOM(\underline{v}_c)$, we will write

$$Val(s, \underline{v}_c, \underline{v}_o).$$

It is not required that $\underline{v}_o$ be unique; even with $DOM(\underline{v}_c)$ unchanged, there may be one or more other values, $\underline{v'}_o \in DOM(\underline{v}_c)$, such that $Val(s, \underline{v}_c, \underline{v'}_o)$ holds for the same situation, s (This is one reason for using the Val-Predicate instead of simply writing $\underline{v}_c = \underline{v}_o$). This allows us not only to use different domains for $\underline{v}_c$, but also to accept, for instance, different measurements for $\underline{v}_c$ that are not considered to establish a contradiction (see also subsections 4.3 through 4.5).

In some domains, one value may may be implied by another one. For instance, if $DOM(\underline{v}_c)$ consists of intervals, then if some interval, int, is a value in a situation, s, so is any interval that contains int:

$$\forall s \in SIT \quad Val(s, \underline{v}_c, int) \wedge int \subseteq int' \Rightarrow Val(s, \underline{v}_c, int').$$

However, to keep things simple, we will consider domains first that do not contain such dependencies among values. We will call them **irreducible**. In section 4.4 the theory will be extended to cover also reducible domains, in particular for the treatment of representational transformations.

We assume that a constituent has a number of distinct possible behavioral modes, due to different physical conditions. A correct (unbroken) wire in a circuit exhibits a particular behavioral mode, and a (permanently) broken wire has another one. A wire that potentially switches between these two conditions gives rise to a third kind of physical condition and establishes a behavioral mode that is distinguished from the two other modes.

For principled reasons, our knowledge about the behavioral modes of a constituent is limited:

- globally: we may be unable to enumerate the set of behavioral modes, because we cannot anticipate all possible physical conditions of a constituent (in particular, the faults)
- locally: we may be unable to exactly describe the relation characterizing a particular behavioral mode, for instance, due to incomplete knowledge about the physical principles.

Even if we consider the second restriction irrelevant for the application we have in mind (i. e. we pretend to be able to explicitly and precisely associate a behavioral mode with some relation, R) it is, again for fundamental reasons, **impossible to positively verify a particular behavioral mode** to be present. Firstly, in most cases, R and/or DOM($\underline{v}_c$) will be infinite, and, hence, we cannot exhaustively check the space of possible tuples. Secondly, even if we (in the finite case) detected in experiments all tuples of R and none outside R, we can still not guarantee that future observations will not include a tuple out of R's complement which would consequently invalidate the respective behavioral mode.

This is what we are normally capable of: **falsifying the presence of a behavioral mode** based on an observation or an inference that is definitly inconsistent with this mode. For this purpose, we are not required to have an explicitly given precise relation for this mode. We only have to use a relation that is **guaranteed to include** the unknown ideal relation.

Positively identifying the presence of a particular behavioral mode can only be done by ruling out all other modes. But in order to do so, we have to enumerate and model all other modes, which was stated above to be, in principle, impossible.

To summarize:

- We can **rule out** the presence of behavioral modes based on assumptions of the **quality of the single models** (i. e. ignoring the local restrictions of modeling).
- We can **positively identify** a behavioral mode if we additionally assume we have **complete knowledge** about the **set of modes** (i. e. ignoring the global restrictions of modeling).

The first issue formulates the principle of consistency-based diagnosis and suggests that it is the natural approach to model-based diagnosis. The second issue explains why "pure" consistency-based systems (like GDE) never infer the innocence of a constituent, and shows that, if other approaches do so, this is based on some global assumption about the possible behavioral modes, either explicitly (like GDE$^+$ or DP) or implicitly.

There may be different suitable sets of descriptive variables for behavioral modes, i. e. $\underline{v}_c$ might vary, and there may be different useful domains for a particular $\underline{v}_c$, i. e. DOM($\underline{v}_c$)

varies (for instance, when quantitative and qualitative values are considered). This will be discussed in subsection 4.3 through 4.5.

We emphasize that, by adopting the relational representation, we do **not restrict** our capabilities of dealing with dynamic systems. It is general enough to cover not only episode-based temporal representations, but also continuous systems; for instance, DOM($\underline{v}_c$) can consist of histories, trees of qualitative behaviors, or even continuous functions. This justifies the choice of the term "behavioral constituent" instead of "component", because the former subsumes processes and abstract functional entities as well as physical objects. It is a myth that model-based diagnosis in general and ATMS-based diagnosis in particular are limited to static devices or to component-oriented modeling. (Theoretically, even causal models might be represented by relations, if variables are split into several ones reflecting different causal roles, even though this may not be the most elegant and intuitive way).

In order to establish logical links between models we cannot simply consider a model to **be** a relation. Intuitively speaking, a model is the claim (or the guess) that a certain relation, R, covers all behaviors permitted by the current physical condition in any situation, s∈SIT, we might encounter (Fig. 4.1).

Definition 4.1 (Model)

A relation R⊆DOM($\underline{v}_c$) specifies **a model** of a constituent C by

M(C, R) ⇔

$\forall \underline{v}_0 \in DOM(\underline{v}_c)$ (($\exists s \in SIT$ $Val(s, \underline{v}_c, \underline{v}_0)$) ⇒ $\underline{v}_0 \in R$).

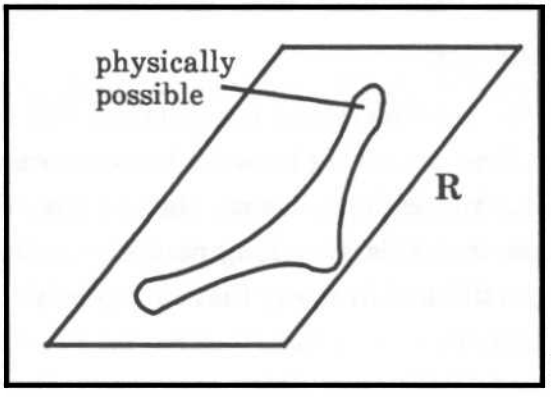

Fig. 4.1 A model covers the physically possible values

Definition 4.2 (Strong Model)

A relation R⊆DOM($\underline{v}_c$) specifies a **strong model** of C by

B(C, R) ⇔

$\forall \underline{v}_0 \in DOM(\underline{v}_c)$ (($\exists s \in SIT$ $Val(s, \underline{v}_c, \underline{v}_0)$) ⇔ $\underline{v}_0 \in R$).

Note that any superset of a relation that specifies a model also specifies a model:

Lemma 4.1

$$M(C, R) \wedge R \subseteq R' \Rightarrow M(C, R').$$

In particular,

$$B(C, R) \wedge R \subseteq R' \Rightarrow M(C, R').$$

(Actually, even DOM($\underline{v}_c$) is a model). In consistency-based diagnosis, Definition 4.1 suffices to conclude from observing $\underline{v}_0 \notin R$ in one situation that the behavioral mode which is meant to be covered by M(C, R) is not present. In contrast, strong models describe exactly what is possible given a certain physical state of C. The idea is that they serve as "ideal" models of the physically possible **behavioral modes** we are aware of (this is why they are denoted B(C, R)), even though we may be unable to explicitly describe R and, hence, have to cover this mode by (weak) models or approximate it.

We require that important distinctions between different (physical) behavioral modes can be expressed by $\underline{v}_c$ and DOM($\underline{v}_c$); they may overlap, but they are exclusive if and only if they differ by at least one value: Let $R, R' \subseteq DOM(\underline{v}_c)$,

$$R \neq R' \Leftrightarrow (B(C, R) \Rightarrow \neg B(C, R')).$$

The idea that there is a set of strong models that correspond to a number of known, distinguishable, and, in a sense, stable behavioral modes appears to be inadequatly restrictive and to prevent us from handling behavioral modes of unknown or unstable type. However, we provide a general mechanism for dealing with incomplete knowledge of behavioral modes (as exemplified in section 3.2). Moreover, the concepts introduced above allow us to model intermittent behavior to a certain degree.

4.2 Intermittent Behavior

By intermittent behavior, we understand the behavior of a constituent whose physical condition changes over time, switching between two or more conditions that correspond to elementary behavioral modes in the sense stated above (i. e. it is not an arbitrary behavior). If we assume that this switching occurs only **between** different observed situations, SIT can be partitioned in a way that each partition covers one of the "stable" behavioral modes $B(C, R_i)$:

$$SIT = \cup SIT_i$$

$$\wedge \forall i \ (\forall \underline{v}_0 \in DOM(\underline{v}_c) \ (s \in SIT_i \wedge Val(s, \underline{v}_c, \underline{v}_0) \Rightarrow \underline{v}_0 \in R_i)).$$

A model for such a behavior is specified by the union of the single relations:

$$M(C, \cup R_i).$$

Whether this union provides us with a strong model depends on the nature of the intermittency. If the switching between different physical conditions is controlled by some "hidden parameter" that is not part of the model but physically linked to modelled

variables or parameters (e. g. a fault occurs if a threshold is exceeded by the temperature, which depends on, say, the speed) we have only a weak model. $B(C, \cup R_i)$ holds only if intermittency is random from the constituent's point of view, i. e. governed by a principle that acts independently of the constituent's physics, including absolute randomness (Fig. 4.2).

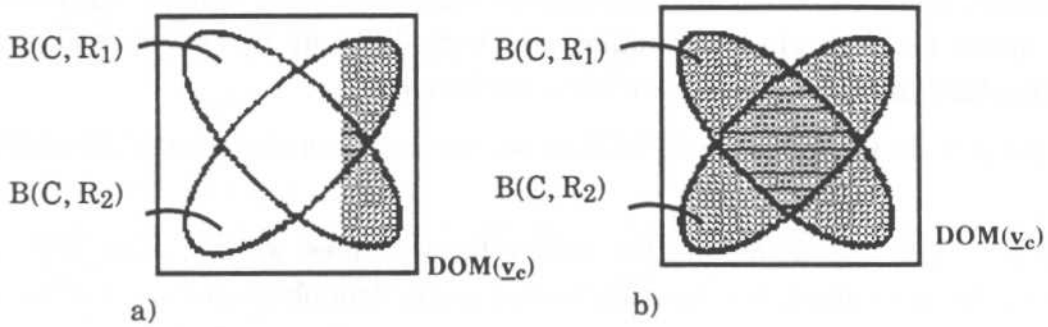

Figure 4.2 Intermittent behavior (shaded)
a) controlled by a "hidden parameter" b) random

In any case, if $M(C, R_i)$ are models of some behavioral modes, we can derive a model for a mode intermittent between them by $M(C, \cup R_i)$. This treatment of intermittency differs from that in [Raiman-de Kleer-Saraswat 90] under two aspects:

- It allows to **positively** model intermittent behavior and to distinguish different intermittent behaviors, whereas the latter exploits **non-intermittency** (by postulating a functional dependency of output values on input values).
- It is more **general** due to the use of relational models, rather than input - output models.

4.3 Multiple Representations, Abstraction, Simplification and Approximation

4.3.1 Multiple Models

For different purposes different representations and models of constituents' behaviors may be useful or even necessary. In our relational framework, this may involve the use of

- different variables, $\underline{v}_c$, and/or
- different domains, $DOM(\underline{v}_c)$, for the same variables, and/or
- different relations for defining a model while $\underline{v}_c$ and $DOM(\underline{v}_c)$ remain fixed.

A shift between planar coordinates and polar coordinates

$(x, y) \mapsto (\psi, \rho)$

is an example for the first case, whereas interchanging integers and binary numbers illustrates the second. In each example, either representation captures the same information. But in many cases, the introduction of another representation or model is motivated by the goal of making things simpler by ignoring details to some degree,

rather than representing the same details differently. The thyristor example in section 1.2 extensively illustrates this point.

The use of abstraction, simplification, and approximation reflects this objective and has become one major topic in recent work in qualitative physics. But so far, the three terms have often been used without explicit definition and somewhat arbitrarily. We propose to make a clear distinction between them, since we argue that these different operations, although often accompanying each other, are very different in nature, and, more importantly, have different effects in a model-based system.

In order to motivate the following definitions, we present some examples of simplifying models:

Example 1: The use of qualitative values for variables which, ultimately, are considered to be real-valued. For instance, we map the real numbers onto the four-valued space of signs:

$q: \mathbb{R} \rightarrow Q_3 = \{-, 0, +\} = \{(-\infty, 0), [0, 0], (0, \infty)\}$.

Example 2: The characterization of continuously differentiable functions, $f \in C^1(\mathbb{R})$, by global features, such as "monotonically increasing", "periodic", "linear", etc.

Example 3: Describing the dynamics of objects by its mass, height, and velocity, (m, h, v), or, alternatively, by its energy, $E = m \cdot h - \frac{1}{2} mv^2$.

Example 4: Generating a behavior description of a composite device in terms of its input-output values only, i. e. ignoring internal variables.

Example 5: Using the switch model, TH_{SWITCH}, for the thyristor instead of TH_{THRESH}, thus ignoring the the potential firing by exceeding the breakover voltage as well as thermal triggering.

Example 6: Replacing the sections of the characteristic curve of the thyristor by pieces of linear functions.

In each example, the shift aims at making models simpler, eliminates some details, and potentially collapses previously distinguishable cases into a single one. However, in the first four examples, this is done by a general change in the representation which then leads to a new model; this is what we call **abstraction**. In contrast, example 6 basically changes the relation that defines the model based on an unchanged representation. Example 5 comprises both: while ignoring some variables, such as temperature, (and, thus, changing the representation), the shift also modifies the modeling relation by eliminating some tuples (where transition to ON occurs without a gate pulse). We consider these two examples to involve **simplification**, where example 6 illustrates a special case: **approximation**.

Hence, **abstraction is a relation between representations** (inducing a relation between models), whilst **simplification (including approximation as a special case) is a relation between models** based on the same representation.

We use our formal framework to give precise definitions for these terms. But first, for the sake of a compact notation, for an arbitrary mapping

$m: A \rightarrow B$,

we introduce the inverse (set-valued) mapping

$m^{-1}: B \rightarrow P(A)$

(P(A) : power set of A) by

$m^{-1}(b) := \{a \in A \mid m(a) = b\}$,

i. e. the preimage of b consists of all tuples that are mapped onto b. Furthermore, we expand m and m^{-1} to power sets : Let $A' \subseteq A$ and $B' \subseteq B$

$m(A') := \bigcup_{a \in A'} \{m(a)\}$ and

$m^{-1}(B') := \bigcup_{b \in B'} m^{-1}(b)$.

4.3.2 Representational Transformations

Definition 4.3 (Representation)

A representation for modeling is a pair $(\underline{v}_c, DOM(\underline{v}_c))$.

A transformation that turns one representation into another one should preserve the Val predicate in the following sense:

(i) If Val holds for a value in the original representation, then it also holds for the transformed value. For instance, if we map real numbers to signs then Val(s, $\underline{v}_c$, 0.5) implies Val(s, $\underline{v}_c$, +).

(ii) If Val holds for a transformed value, this must be the case for one of its preimages in the original representation: Val(s, $\underline{v}_c$, +) is only true if there exists some positive real number $r \in \mathbf{R}$ such that Val(s, $\underline{v}_c$, r).

Intuitively. these properties express that none of the representations covers situations that cannot be represented by the other one, although different kinds of distinctions may be captured. This is formalized by the following definition.

Definition 4.4 (Representational Transformation)

Let $(\underline{v}_c, DOM(\underline{v}_c))$ and $(\underline{v}'_c, DOM'(\underline{v}'_c))$ be two (irreducible) representations. A mapping

$\tau_0: DOM(\underline{v}_c) \rightarrow DOM'(\underline{v}'_c)$

is a representational transformation, iff

$\forall \underline{v}'_0 \in \tau_0(DOM(\underline{v}_c)) \quad \forall s \in SIT$

$Val(s, \underline{v}_c, \underline{v}'_o) \Leftrightarrow \exists \underline{v}_0 \in DOM(\underline{v}_c) \, (Val(s, \underline{v}_c, \underline{v}_o) \wedge \tau_0(\underline{v}_0) = \underline{v}'_0)$.

We will use the notation

$DOM(\underline{v}_c) \sim_\tau DOM'(\underline{v}'_c)$ (under τ_0)

if there exists a representational transformation

τ_0: $DOM(\underline{v}_c) \rightarrow DOM'(\underline{v}'_c)$,

and

$DOM(\underline{v}_c) \simeq_\tau DOM'(\underline{v}'_c)$

if τ_0 is surjective, i. e. every value in $DOM(\underline{v}_c)$ has a preimage under τ_0:

$\tau_0(DOM(\underline{v}_c)) = DOM'(\underline{v}'_c)$.

The important property of representational transformations is that they preserve models as stated by the following theorem.

Theorem 4.2

The image of a model under a representational transformation is a model: Let $R \subseteq DOM(\underline{v}_c)$ and $R' \subseteq DOM'(\underline{v}'_c)$. If

$DOM(\underline{v}_c) \sim_\tau DOM'(\underline{v}'_c)$ under τ_0

then

$M(C,R) \Rightarrow M(C, \tau_0(R))$ and

$B(C,R) \Rightarrow B(C, \tau_0(R))$.

The inverse transformation also preserves models:

$M(C,R') \Rightarrow M(C, \tau_0^{-1}(R' \cap \tau_0(DOM(\underline{v}_c))))$.

If τ_0 is surjective,

$DOM(\underline{v}_c) \simeq_\tau DOM'(\underline{v}'_c)$,

then the last statement simplifies to

$M(C,R') \Rightarrow M(C, \tau_0^{-1}(R'))$.

The proof for this theorem for irreducible domains is obtained by proving its generalization in subsection 4.4. It is easy to see that representational transformations are compositional in the following sense:

Lemma 4.3

$DOM(\underline{v}_c) \simeq_\tau DOM'(\underline{v}'_c) \wedge DOM'(\underline{v}'_c) \sim_\tau DOM''(\underline{v}''_c)$

$\Rightarrow DOM(\underline{v}_c) \sim_\tau DOM''(\underline{v}''_c)$.

4.3.3 Abstraction

We consider the following properties as features defining abstraction transformations:

- The abstract representation is "grounded" in the lower level, i.e. each abstract unit (value) is the image of at least one lower level instance.
- Abstraction really eliminates some distinctions: different lower level instances are mapped onto one abstract unit.

Definition 4.5 (Abstraction)

A representational transformation

α_0: DOM($\underline{v}_c$) → DOM'($\underline{v}'_c$)

is a (representational) abstraction, iff it is

- surjective and
- and not injective, i. e $\exists x, y \ (x \neq y \wedge \alpha_0(x) = \alpha_0(y))$.

($\underline{v}'_c$, DOM'($\underline{v}'_c$)) is called an abstraction of ($\underline{v}_c$, DOM($\underline{v}_c$)), for short

DOM($\underline{v}_c$) $\sim_\alpha$ DOM'($\underline{v}'_c$),

if there exists an abstraction

α_0: DOM($\underline{v}_c$) → DOM'($\underline{v}'_c$).

Of course, Definition 4.4 is rather weak, since it is satisfied as soon as a single image has two preimages under α_0. It does not capture criteria for useful abstractions which can only be discussed with respect to the properties of DOM($\underline{v}_c$) (e. g. topologies, metrics, etc.) and the task. But even at this general level, two ways to obtain abstractions can be distinguished:
In example 1, only the domain of the variables involved was changed, as opposed to example 3, where a new variable substituted three others.

Definition 4.6 (Domain Abstraction)

An abstraction

α_0 : DOM($\underline{v}_c$) → DOM'($\underline{v}_c$) = DOM'(v_1) × ... × DOM'(v_k)

is a domain abstraction iff

$\alpha_0 = (\alpha_1, ..., \alpha_k)$

with α_i : DOM(v_i) → DOM'(v_i) for $i > 0$,

(where achieving $\underline{v}_c = \underline{v}'_c$ may require permutation of variables). Otherwise, α_0 is a **variable abstraction.**

Domain abstraction covers, for instance, qualitative abstraction (example 1), but can also be used for temporal abstraction (example 2). Usually, we expect variable abstraction to decrease the number of variables (as in example 3). However, this is dependent on superficial features, such as whether a tuple-valued variable is expanded into its elements (e. g. writing ($x = x_0$, $y = y_0$, ...) instead of ($p = (x_0, y_0)$, ...)), and not a necessary condition for an abstraction.

Note that example 4 (structural aggregation) does not necessarily lead to an abstraction, although the number of variables is decreased: if the internal variables are completely determined by the global input-output (in particular, if there is no "memory" or energy storage that points back prior to the situation, s), the mapping between the representations is injective.

We can now describe how representational abstractions induce model abstraction.

Definition 4.7 (Model Abstraction)

Let $R \subseteq DOM(\underline{v}_c)$ and $R' \subseteq DOM'(\underline{v}'_c)$. M(C, R') (or B(C, R')) is a model abstraction of M(C, R) (or B(C, R), respectively), also denoted by the symbol

$M(C, R) \sim_\alpha M(C, R')$,

iff there exists a representational abstraction

$\alpha_0 : DOM(\underline{v}_c) \rightarrow DOM'(\underline{v}'_c)$ and

$R' = \alpha_0(R)$.

(Fig. 4.3)

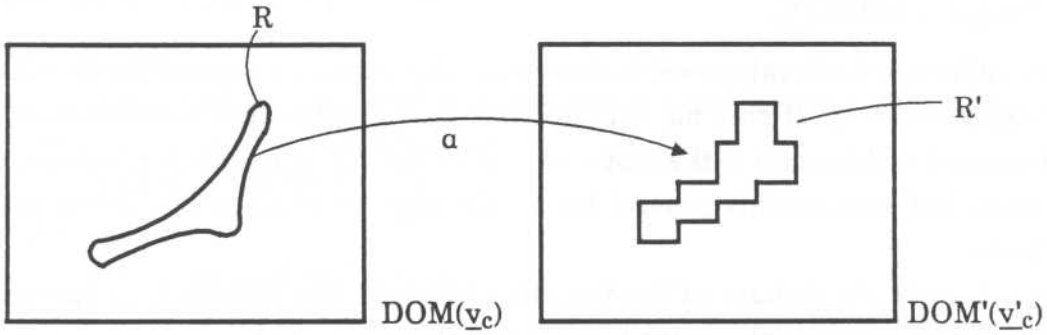

Figure 4.3 Model abstraction

Remark $\sim_\alpha$ is a transitive, anti-symmetric, irreflexive relation.

From Theorem 4.2, it follows directly how models on the abstract level can be constructed and that they form strong models only if they are the image of a strong model.

Corollary 4.4

Let $DOM(\underline{v}_c) \sim_\alpha DOM'(\underline{v}'_c)$ and $R \subseteq DOM(\underline{v}_c)$. Then

$M(C, R) \Rightarrow M(C, \alpha_0(R))$,

$B(C, R) \Rightarrow B(C, \alpha_0(R))$, and

$M(C, R') \Rightarrow M(C, \alpha_0^{-1}(R'))$

holds.

Remark Abstraction may render models or behavioral modes indistinguishable; for instance, it may be the case that

$R_1 \neq R_2 \wedge B(C, R_1) \wedge B(C, R_2) \wedge \alpha_0(R_1) = \alpha_0(R_2) = R'$,

and, hence,

$(B(C, R_1) \sim_\alpha B(C, R')) \wedge (B(C, R_2) \sim_\alpha B(C, R'))$.

Note also that a strong model may also be obtained as an abstraction of a weak model:

$M(C, R_1) \sim_\alpha B(C, R')$,

namely if α_0 eliminates the differences between its relation and that of a strong model.

Also note that, in general, mapping an abstract strong model to the concrete level yields only a weak model:

$B(C, R') \Rightarrow M(C, \alpha_0^{-1}(R'))$;

$B(C, R')$ does not imply $B(C, \alpha_0^{-1}(R'))$.

We see that **abstraction**, being a representational transformation, and its inverse **preserves models.** Under abstraction, the discriminating power of models (w.r.t. behavioral modes) may be lost, and under the inverse mapping, a strong model will often loose its strength; but still, the images of models are models.

4.3.4 Simplification

In examples 5 and 6, this is different: in both cases, some "surgery" has been applied to the model that cannot guarantee that the result is still a model. In one case, those tuples were removed from the defining relation that allowed firing of the thyristor without a gate pulse (i. e. tuples (OFF→ON, pulse=0)). In fact, things are simpler now, since we can conclude pulse=1 whenever the transition OFF→ON occurs, but also things may go wrong, e. g. if the thyristor is triggered by dV/dt. Note that the representation (tuples of transitions and gate pulses) has not been changed, but within the same representation the model has been simplified, at the risk of no longer being a model. This motivates the following definition.

Definition 4.8 (Simplification)

A simplification is a transitive, irreflexive, and antisymmetric relation

$\sigma_0 \subseteq P(DOM(\underline{v}_c)) \times P(DOM(\underline{v}_c))$,

such that a simplification criterion $C_{\sigma_0}(R, R')$ is satisfied for all $(R, R') \in \sigma_0$.

We use the notations

$R \sim_\sigma R'$ and $M(C, R) \sim_\sigma M(C, R')$

if R' is a simplification of R under some σ_0.

Unfortunately, it seems hard to be more specific about the magic simplification criterion at this level of analysis. In many cases it will be handcrafted models as replacements for others rather than relations generated from others according to some general principle. Example 5 gives a flavor of such a principle, but also illustrates the difficulties. One could imagine the respective procedure to be composed of two steps: the first one is variable abstraction, namely elimination of temperature (and dV/dt etc.), followed by a simplification that eliminates tuples from the relation for which there is no "cause" in the abstracted representation, in particular, tuples containing (OFF→ON, pulse=0) which lost their justification because thermal triggering can no longer be described in the abstract representation. From this perspective, simplification is far from being a merely syntactical transformation but involves some causal analysis.

A different view, which is potentially more appropriate for generalization, is the following. The motivation for the ultimate simplification is that temperature and other variables remain in a range where they do not "disturb" the principal behavior of the thyristor. Hence, in example 5, first simplification is performed by shrinking the original relation to the nominal value (or range) of temperature etc. Thus, thermal triggering etc. is eliminated. The simplified model is then projected to the sparse representation (which is an abstraction if we started from nominal ranges rather than single values).

This kind of simplification, reduction of a model-defining relation to the nominal values or normal operation ranges, should be rather widespread and covers cases like ignoring friction in mechanical systems etc. Based on our definiton of models it provides us with a way to clearly specify the semantics of a simplification.

This is important, because it gives us the chance to reason about conditions, under which a simplification yields a model. Remember that, unless the simplification is the generation of a weaker model, it is not guaranteed that the simplified relation, R', still covers the behavioral mode (Fig. 4.4).

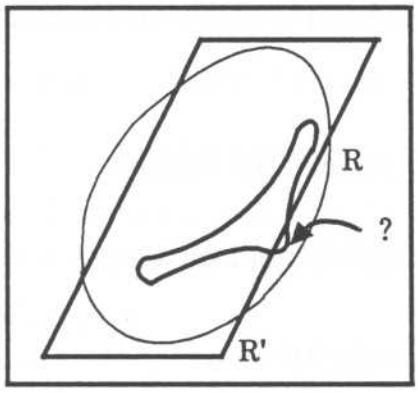

Figure 4.4. R' as a simplification of R

Lemma 4.5

Let $R, R' \subseteq DOM(\underline{v}_c)$.

$$(M(C, R) \wedge (\forall s \in SIT\ Val(s, \underline{v}_c, \underline{v}_o) \Rightarrow \underline{v}_0 \notin R \backslash R')) \Rightarrow M(C, R').$$

Intuitively, this lemma (which is general, but of particular interest for $R \sim_\sigma R'$) simply states that the modified relation still defines a model, if the parts of R it truncates do not correspond to encountered situations. The condition of lemma 4.5 can be seen as the assumption that the physical condition of C and/or **environmental conditions** restrict the possible physical situations accordingly. This idea is of particular importance if we simplify a strong model, B(C, R), because it claims to cover exactly all physical situations, and, hence, $R \backslash R' = \varnothing$ seems to be a necessary consequence. But in this case, the condition

in lemma 4.2 formulates the expectation that environmental factors reduce SIT to some SIT'⊆SIT that excludes the dangerous cases. Remember, in consistency-based diagnosis, the danger comes from falsely falsifying a behavioral mode. In our framework, observing some $\underline{v}_0 \in R \backslash R'$ gives us the choice of negating the model **or the condition**, which can be given a clear semantics. We will continue this discussion in section 5.

4.3.5 Approximation

There remains a special class of simplifications to be discussed which is represented by example 6. Here, one model is replaced with the goal of, though making the model simpler (piecewise linear), defining it rather **close to** the original one. Speaking about closeness imposes a condition on the nature of the representation: it must be a metric space, i.e. we can define some suitable distance, d_R, on $P(DOM(\underline{v}_c))$:

$$d_R: \quad P(DOM(\underline{v}_c)) \times P(DOM(\underline{v}_c)) \rightarrow \mathbf{R}$$

with $d_R(R, R') = d_R(R', R) \geqq 0$,

$d_R(R, R') = 0 \Leftrightarrow R = R'$,

and $d_R(R, R') + d_R(R', R'') \leq d_R(R', R'')$.

Definition 4.9 (Approximation)

A simplification, $\sigma(\varepsilon_0)$, is an ε_0-approximation, iff

- d_R is a distance defined on $P(DOM(\underline{v}_c))$, and
- $\exists \varepsilon_0 \in \mathbf{R} \quad (C_{\sigma(\varepsilon_0)}(R, R') \Leftrightarrow d_R(R, R') \leqq \varepsilon_0 \wedge C'_\sigma(R, R'))$.

This means, the simplification criterion contains, among other criteria (e. g. linearity), a restriction of the deviation from the approximated model. This links the problem to a lot of standard approximation techniques.

We will write

$$R \sim_\varepsilon R' \quad \text{and} \quad M(C, R) \sim_\varepsilon M(C, R')$$

if R' is an ε_0-approximation of R for some ε_0. Note that $\sim_\varepsilon$ is transitive, but, usually, not w.r.t. a fixed ε_0.

For the special case that all $DOM(v_i) = \mathbf{R}^n$, we can use a distance d on $\mathbf{R}^n$ in order to define d_R. For instance, we might define

$$d_x: \mathbf{R}^n \times P(\mathbf{R}^n) \rightarrow \mathbf{R}$$

by $d_x(x', R) := \min_{x \in R}(d(x', x))$,

and $d_R: P(\mathbf{R}^n) \times P(\mathbf{R}^n) \rightarrow \mathbf{R}$

by $d_R(R, R') := \max\left(\max_{x' \in R'}(d_x(x', R)), \max_{x \in R}(d_x(x, R'))\right)$.

We conclude this subsection by summarizing that we consider

- abstraction as a special transformation of representations (that induces a

transformation of models)

- and simplification as a transformation of models within the same representation.

The former preserves model properties while the latter may violate them, but in a way that can be described. Approximations are special cases of simplifications that allow measuring the proximity of models.

4.4 Reducible Domains

We will now drop the restriction that domains be irreducible, i.e. we now allow that one value is implied by another one (Readers who are not interested in reducible domains and in the proof of theorem 4.2 may skip this subsection). A justification for this extension can already be obtained from the simple sign algebra used in example 1 in the preceding subsection. One might want to replace

$Q_3 = \{-, 0, +\} = \{(-\infty, 0), [0, 0], (0, \infty)\}$

by $Q_4 = \{-, 0, +, ?\} = \{(-\infty, 0), [0, 0], (0, \infty), (-\infty, \infty)\}$

(where $? = (-\infty, \infty)$ is the unrestricted value), because this allows, for instance, a complete definition of addition (see [Struss 89c]). In this case,

$Val(s, \underline{v}_c, +) \Rightarrow Val(s, \underline{v}_c, ?)$

would be a consequence. This motivates the following definitions.

Definition 4.10 (Irreducible Domain)

An **implicant** of some $\underline{v}_0 \in DOM(\underline{v}_c)$ is a value that implies $\underline{v}_0$ in all situations. $IMP(\underline{v}_0)$ denotes the set of implicants of $\underline{v}_0$:

$$IMP(\underline{v}_0) := \{\underline{v}'_0 \in DOM(\underline{v}_c) \mid \forall s \in SIT \;\; Val(s, \underline{v}_c, \underline{v}'_0) \Rightarrow Val(s, \underline{v}_c, \underline{v}_o)\}.$$

A value is called **reducible** iff $IMP(\underline{v}_0) \neq \{\underline{v}_0\}$.

The **value hull** $V(R)$ of some relation $R \subseteq DOM(\underline{v}_c)$ is defined by

$$V(R) := \{\underline{v}_0 \in DOM(\underline{v}_c) \mid IMP(\underline{v}_0) \cap R \neq \emptyset\}$$
$$= \{\underline{v}_0 \in DOM(\underline{v}_c) \mid \exists \underline{v}'_0 \in R \;\; \forall s \in SIT \;\; Val(s, \underline{v}_c, \underline{v}'_0) \Rightarrow Val(s, \underline{v}_c, \underline{v}_0)\}.$$

R is called **closed** w.r.t. Val iff $R = V(R)$.

A domain $DOM(\underline{v}_c)$ is called **irreducible** if all subsets are closed w.r.t. Val:

$\forall R \subseteq DOM(\underline{v}_c) \;\; V(R) = R$.

It is because of reducible values that Definition 4.1 of a relational model is inappropriate for relations R which are not closed w.r.t. Val. This can be mended in the following way.

Definition 4.1' (Models for Reducible Domains)

A **closed** relation $R \subseteq DOM(\underline{v}_c)$ specifies a model of a constituent C by

$$M(C, R) \Leftrightarrow$$
$$\forall \underline{v}_0 \in DOM(\underline{v}_c) \;\; ((\exists s \in SIT \;\; Val(s, \underline{v}_c, \underline{v}_0)) \Rightarrow \underline{v}_0 \in R)$$

We also have to alter Definition 4.4 of a representational transformation, because the justification (ii) presented there is too strong for reducible domains. The target domain

may contain reducible values that are not image of any value of the original domain. In the Q_4 example, Val (s, $\underline{v}_c$, ?) is true even though it is not the image of any real number under the sign transformation. The appropriate condition is

(ii') If Val holds for a value in the target representation, then it also holds for one of the preimages of one of its implicants.

The following definition also distinguishes between the two conditions in order to obtain a refined version of Theorem 4.2.

Definition 4.4' (Representational Transformation)

Let $(\underline{v}_c, DOM(\underline{v}_c))$ and $(\underline{v}'_c, DOM'(\underline{v}'_c))$ be two representations. A mapping

τ_0: $DOM(\underline{v}_c) \rightarrow DOM'(\underline{v}'_c)$

is **Val-preserving** iff

$\forall \underline{v}_0 \in DOM(\underline{v}_c) \quad \forall s \in SIT$

$Val(s, \underline{v}_c, \underline{v}_0) \Rightarrow Val(s, \underline{v}_c, \tau_0(\underline{v}_0))$.

It is **Val-grounding** iff

$\forall \underline{v}'_0 \in DOM'(\underline{v}'_c) \quad \forall s \in SIT$

$Val(s, \underline{v}_c, \underline{v}'_0) \Rightarrow \exists \underline{v}_0 \in DOM(\underline{v}_c) \, (Val(s, \underline{v}_c, \underline{v}_0) \wedge \tau_0(\underline{v}_0) \in \mathbf{IMP}(\underline{v}'_0))$.

It is a representational transformation, iff it is Val-preserving and Val-grounding.

With these modified definitions, Theorem 4.2 holds also for the reducible case (for irreducible domains, Definitions 4.1 and 4.4 follow from Definitions 4.1' and 4.4', respectively), if modified appropriately.

Theorem 4.2'

Let $R \subseteq DOM(\underline{v}'_c)$, $R' \subseteq DOM'(\underline{v}'_c)$, and

τ_0: $DOM(\underline{v}_c) \rightarrow DOM'(\underline{v}'_c)$.

If τ_0 is **Val-grounding** then the (hull of the) **image of a model is a model**:

$M(C, R) \Rightarrow M(C, V(\tau_0(R)))$.

If τ_0 is **Val-preserving**, then the inverse transformation also preserves models:

$M(C,R') \Rightarrow M(C, V(\tau_0^{-1}(R' \cap \tau_0(DOM(\underline{v}_c)))))$.

If τ_0 is a **representational transformation** then the (hull of the) **image of a strong model is a strong model**:

$B(C, R) \Rightarrow B(C, V(\tau_0(R)))$.

Proof

(i) "$M(C, R) \Rightarrow M(C,V(\tau_0(R)))$".

Let $\underline{v}'_0 \in DOM'(\underline{v}'_c)$ and $s \in SIT$ such that $Val(s, \underline{v}'_c, \underline{v}'_0)$ holds. We have to show that $\underline{v}'_0 \in V(\tau_0(R))$ also holds under the condition $M(C, R)$.

From $Val(s, \underline{v}'_c, \underline{v}'_0)$ and Definition 4.4' (Val-grounding) we obtain

$\exists \underline{v}_0 \in DOM(\underline{v}_c) \; Val(s, \underline{v}_c, \underline{v}_0) \wedge \tau_0(\underline{v}_0) \in IMP(\underline{v}'_0)$.

If M(C,R) holds, this implies (by Definition 4.1')

$\exists \underline{v}_0 \in R \;\; \tau_0(\underline{v}_0) \in IMP(\underline{v}'_0)$,

and, hence,

$\tau_0(R) \cap IMP(\underline{v}'_0) \neq \emptyset$.

This means $\underline{v}'_0 \in V(\tau_0(R))$, completing the proof of $M(C, V(\tau_0(R)))$.

(ii) "$M(C,R') \Rightarrow M(C, R)$, where $R = V(\tau_0^{-1}(R' \cap \tau_0(DOM(\underline{v}_c))))$".

This part of the theorem will be proved by reductio ad absurdum. We assume

$M(C,R') \wedge \neg M(C, R)$.

$\neg M(C, R) \Rightarrow \exists \underline{v}_0 \in DOM(\underline{v}_c) \;\; \exists s \in SIT \;\; Val(s, \underline{v}_c, \underline{v}_0) \wedge \underline{v}_0 \notin R$.

If τ_0 is Val-preserving, this implies

$\exists \underline{v}_0 \in DOM(\underline{v}_c) \quad \exists s \in SIT \;\; Val(s, \underline{v}'_c, \tau_0(\underline{v}_0)) \wedge \tau_0(\underline{v}_0) \notin \tau_0(R)$,

and

$\exists \underline{v}'_0 \in DOM'(\underline{v}'_c) \;\; \exists s \in SIT \;\; Val(s, \underline{v}'_c, \underline{v}'_0) \wedge \underline{v}'_0 \notin R'$.

This contradicts M(C,R'), and, hence, M(C, R) holds.

(iii) "$B(C,R) \Rightarrow B(C, V(\tau_0(R)))$".

Let $\underline{v}'_0 \in V(\tau_0(R))$; we must show there exists a situation s with $Val(s, \underline{v}'_c, \underline{v}'_0)$, given B(C, R). $\underline{v}'_0 \in V(\tau_0(R))$ implies

$\exists \underline{v}_0 \in R \;\; \tau_0(\underline{v}_0) \in IMP(\underline{v}'_0)$.

B(C,R) yields

$\exists \underline{v}_0 \in DOM(\underline{v}_c) \; (\exists s \in SIT \;\; Val(s, \underline{v}_c, \underline{v}_0)) \wedge \tau_0(\underline{v}_0) \in IMP(\underline{v}'_0)$.

Definition 4.4' (Val-preserving) establishes

$\exists \underline{v}_0 \in DOM(\underline{v}_c) \;\; (\exists s \in SIT \;\; Val(s, \underline{v}'_c, \tau_0(\underline{v}_0)) \wedge \tau_0(\underline{v}_0) \in IMP(\underline{v}'_0)$,

and finally,

$\exists s \in SIT \;\; Val(s, \underline{v}'_c, \underline{v}'_0)$

completing the proof of $B(C, V(\tau_0 CR)))$.

4.5 Data Interpretation

Definition 4.1 may seem a bit strong in that it requires that the defining relation really **contains** all possible values. Often, we might want to reflect aspects like measurement precision or the degree of commitment to a certain inferred or observed value (e. g. by using fuzzy numbers) and rather ask whether or not this is considered **conflicting** with the relation. In particular, this is true if approximate models are used. For instance, when using a linear approximation, we will want to weaken the criterion for what establishes a discrepancy.

There are different ways to incorporate this aspect in our framework. One is to explicitly specify interpretations of values as mappings

ϕ: $DOM(\underline{v}_c) \rightarrow P(DOM'(\underline{v}_c))$

that associate each value, $\underline{v}_0$, with the set of values, $\phi(\underline{v}_0)$, it is consistent with, or, rather, which are considered not to establish a discrepancy with $\underline{v}_0$. For real numbers, this interpretation could be given by a sphere with a radius of precision or an environment defined by order of magnitude relations. It could also be specified by a threshold for a fuzzy membership function.

The interpretation can be global or specific for a certain (class of) models. It may also vary for a given relation, R, thus defining a family of models based on R, rather than a single one. This is captured by the following variation of Definition 4.1.

Definition 4.1" (Model)

Let $R \subseteq DOM'(\underline{v}_c)$. R and an interpretation

ϕ: $DOM(\underline{v}_c) \rightarrow P(DOM'(\underline{v}_c))$

specify a model by

$M(C, R, \phi) \Leftrightarrow$

$\forall \underline{v}_0 \in DOM(\underline{v}_c)$ ($(\exists s \in SIT \quad Val(s, \underline{v}_c, \underline{v}_o)) \Rightarrow \phi(\underline{v}_0) \cap R \neq \emptyset$).

Note, however, that, if $DOM(\underline{v}_c) = DOM'(\underline{v}_c)$, we can achieve the same effect by a modification of the relation R, namely by adding to R the fringe of values compatible with R under ϕ:

$R_\phi := \{\underline{v}_0 \in DOM(\underline{v}_c) \mid \phi(\underline{v}_0) \cap R \neq \emptyset\}$.

In this case, $\underline{v}_0 \in \phi(\underline{v}_0)$ is a natural requirement. Then Definitions 4.1 and 4.1" are equivalent:

$M(C, R_\phi) \Leftrightarrow M(C, R, \phi)$.

This is why we will continue using Definition 4.1 in the following. Varying the strength of the interpretation while keeping R fixed yields weaker or stronger models.

Definition 4.10 (Weaker Interpretation)

An interpretation ϕ_1 is weaker than another one, ϕ_2, iff

$\forall \underline{v}_0 \in DOM(\underline{v}_c) \quad \phi_2(\underline{v}_0) \subseteq \phi_1(\underline{v}_0)$.

In other words, a weaker interpretation results in a model that is more "careful" in generating discrepancies.

Lemma 4.6

Let $R \subseteq DOM'(\underline{v}_c)$ and ϕ_1, ϕ_2 be two interpretations. If ϕ_1 is weaker than ϕ_2, then the same holds for the induced models:

$M(C, R, \phi_2) \Rightarrow M(C, R, \phi_1)$.

5 Diagnosis with Abstract and Simplified Models

In this section, the utility of the modeling theory for consistency-based diagnosis will be discussed. Because this is not the focus of this paper, we can only outline some

fundamental aspects of the impact of abstract and simplified models on the diagnostic procedure. A more thorough analysis and a technical discussion of diagnosis with multiple models are presented in [Struss 91].

The theory of model-based diagnosis we use is independent of the particular modeling formalism (and, hence, we simply denote models by M, M', etc.). It only requires that the models can be seen as propositions and that logical connectors can be established between them. Our relational formalism forms one example. We assume there exists a set of **explicit models** which can be used directly for prediction of constituent behavior, e.g. relational models defined by some (extensionally or intensionally) specified relation. However, normally, a concept like FAULTY(C) will not be an explicit model, but rather be specified by a choice between different possible fault modes which might be represented by explicit models. In order to include such "implicit models", we can generalize the concept of a model by the following recursive definition.

Definition 5.1 (Generalized Model)

A (generalized) model is

- an explicit model, or
- a disjunction of (generalized) models, or
- a conjunction of (generalized) models, or
- the antecedent of a view or of a simplification that is a (generalized) model.

The relations mentioned in the last item of this definition will be defined in the following subsection.

5.1 Model Relations

Definition 5.2 (View)

A model M' is a view of another model, M if

$M \Rightarrow M'$.

In other words, M' is a necessary condition for M to hold.

Examples

1. From Theorem 4.2 and Corollary 4.4, it follows directly that representational transformations in general and model **abstraction** in particular lead to a view:

   $M(C, R) \sim_a M(C, R') \Rightarrow (M(C, R) \Rightarrow M(C, R'))$.

2. Also the incremental **simulation of dynamic systems** can be covered by the concept of a view: Let us regard an n-step behavior, B_n, to be described by a tuple $\underline{v}_c = (s_1, s_2, ..., s_n)$ where s_1 denotes the initial state, and each s_i is a variable representing the state consecutive to s_{i-1} for $i > 1$. A particular behavior

   $B_{n_o} = (s1_o, s2_o, ..., sn_o)$

is interpreted as a conjunction of n states,

$s_1 = s_{1_0} \wedge s_2 = s_{2_0} \wedge \ldots \wedge s_n = s_{n_0}$,

An n-step simulation, S_{n_0}, is a disjunction of possible n-step behaviors,

$B_{n_1} \vee B_{n_2} \vee \ldots \vee B_{n_m}$

represented by $S_{n_0} = \{B_{n_i}\}$

Furthermore, we call a simulation algorithm **monotonic** iff it has the property that obtaining an n+1-step simulation from an n-step simulation involves only two operations:

- some B_{n_i} in S_n may be deleted because they are detected to be inconsistent in step n+1
- the other B_{n_i} in S_n are extended by (alternative) successor states.

Most simulation systems will have this property; for instance, QSIM does. In this case we can conclude:

Lemma 5.1

If an n-step simulation of a monotonic simulation system is regarded as a model, $M_n = S_n$, it is a view of an n+1-step simulation:

$M_{n+1} \Rightarrow M_n$.

(Note that mapping a n+1-step simulation to a n-step simulation (by "cutting off" the last state in each path) can be regarded as an example for a representational transformation which is Val-preserving but not Val-grounding. The deeper reason for this is that the difference between the two levels is not a mere representational shift, but incorporates the inferential power of the simulation algorithm).

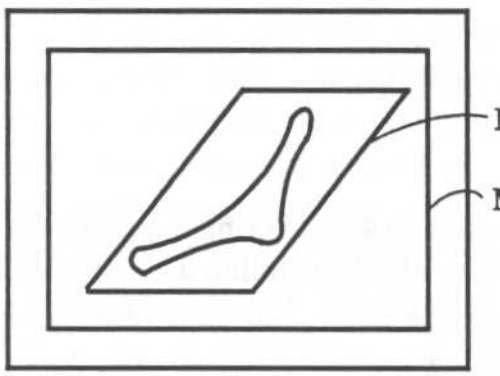

Figure 5.1 M' is a view of M

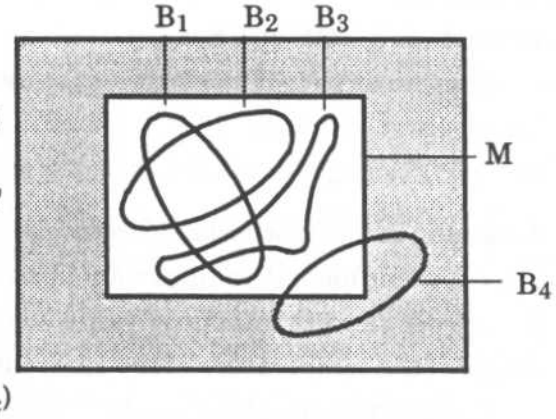

Figure 5.2 $\{B_1, B_2, B_3\}$ is a choice for M

Definition 5.3 (Choice)

A set of models, $\{M_i\}$, is a choice for a model, M, iff

$M \Rightarrow M_1 \vee M_2 \vee \ldots \vee M_n$.

Example

If model abstraction collapses a number of (strong) models into a single one this can be expressed by a choice, e.g.

$$B(C, R) \Rightarrow \bigvee_{B(C,R_i) \sim_\alpha B(C, R)} B(C, R_i).$$

Note that this means essentially expressing the claim that the set $\{B(C, R_i)\}$ really contains models for all (physically) possible behaviors covered by R (Fig. 5.2).

Handling simplification of models is one important task which has already been discussed for relational models in section 4.3. As the examples in section 4.3 illustrate, it is often not possible to guarantee that the result of the simplification is still a model under all circumstances. Using the simplification nevertheless, is based on the **assumption** that the potential deviations from the real behavior mode do not occur in the case we are looking at. Remember that such diagnostic assumptions were introduced in the DP system as the set DHYP.

Definition 5.4 (Simplification)

M' is a simplification of M, if

$$\exists \{dhyp_i\} \subseteq DHYP \qquad M_1 \wedge \bigwedge_i dhyp_i \Rightarrow M'.$$

Examples

1. Simplification may be obtained by modifying the relation that specifies a model. In this case, Lemma 4.5 provides us with a way to specify the simplifying assumption:

Corollary 5.3

Let $R, R' \subseteq DOM(\underline{v}_c)$ and

$$dhyp \Rightarrow (\forall s \in SIT\ Val(s, \underline{v}_c, \underline{v}_o) \Rightarrow \underline{v}_o \notin R \backslash R').$$

Then

$$M(C, R) \wedge dhyp \Rightarrow M(C, R').$$

2. As stated in the discussion of example 5 in section 4.3 (ignoring thermal influences in the thyristor model) an important special form of manipulating the relation directly is its projection to the nominal value or range of a certain variable, such as the normal range of the temperature of the environment, e. g.

   $R' = R \cap [temp_{low}, temp_{high}] \times DOM(v_2) \times \ldots \times DOM(v_n)$.

   In this case, dhyp represents the assumption that we encounter only situations with normal temperature conditions.

3. Global assumptions about sets of models and behavior modes establish another type of simplifying assumptions. Assuming the completeness of the set of modelled behavioral modes is a typical example. We can make it explicit by introducing a

choice for a simplification, M', of a model, M, rather than for M itself:

$M \wedge dhyp \Rightarrow M'$

$M' \Rightarrow M_1 \vee M_2 \vee ... \vee M_n$.

It is easy to see that Definition 5.4 captures the transitivity property of simplification.

Remark 5.4

If M" is a simplification of M': $M' \wedge dhyp' \Rightarrow M''$,

and M' is a simplification of M: $M \wedge dhyp \Rightarrow M'$,

then M" is a simplification of M: $M'' \wedge dhyp'' \Rightarrow M$

with $dhyp'' = dhyp' \wedge dhyp$.

Moreover, the approximation property carries over to views:

Remark 5.5

$(M \wedge dhyp \Rightarrow M' \wedge M' \Rightarrow M'')$

$\Rightarrow \quad M \wedge dhyp \Rightarrow M''$.

In [Struss 91], more basic relations are introduced in order to structure the model set. Thus we turn a simple list of behavioral modes into a graph which contains (generalized) models as nodes and labelled arcs defined by the model relations.

5.2 The Model Graph

We illustrate this by returning to the initial example. Fig. 5.3 shows a possible model graph for the thyristor. σ-arcs correspond to simplifications, v-arcs are views, and choices are marked with "c". The graph indicates, for instance, that under simplifications σ_1, σ_2, σ_3, the model TH_{SWITCH} is a valid model for the correct behavior, and we have to consider $TH_{BLOCKING}$ and $TH_{PUNCTURED}$ as the only possible faults. Graph nodes with bold labels are the (ideal) behavioral modes which are considered to be only implicit models to be checked by views and/or simplifications. As a technical remark, we state that these ideal behavioral modes define the places where assumptions are introduced (namely that the respective mode is the actual one) which are then recorded by the ATMS and propagated via view and simplification links. Only simplification links add further assumptions (which may be either unspecified ATMS assumptions or nodes ultimately justified by explicit simplification conditions as specified by Lemma 4.5). For instance, TH_{SWITCH} might be labelled by the assumption set $\{th_{CORRECT}, \sigma_1, \sigma_2, \sigma_3\}$. This allows us to use ATMS-based focusing techniques as described in [Dressler-Farquhar 90] for guiding the focus for prediction by simplification assumptions. In order to banish potential objections about too many assumptions floating around, we finally mention that rather than positively representing the presence of simplification assumptions we encode them as the **absence of exceptions** from the standard case, and treat them as defaults in the style of [Dressler-Struss 91].

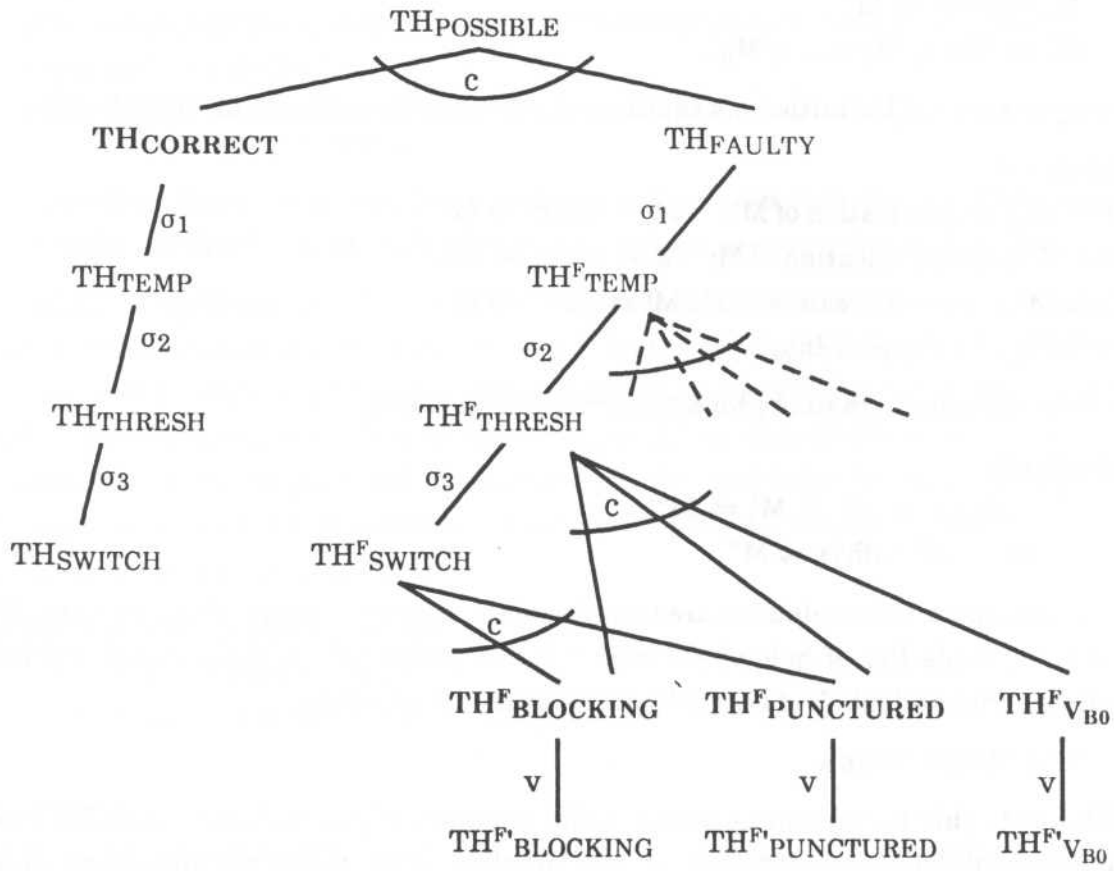

Figure 5.3 A model graph for the thyristor

5.3 Diagnosis with Multiple Models

The model graph is used to guide the selection and instantiation of models (and also their deactivation) in the course of the diagnostic process. Basically, we start at the leaves; views and simplifications are to be used first in order to save costs in prediction. We climb up in the model graph if there is evidence that a revision of modeling assumptions and/or a refinement of models is required. The system description, SD, is no longer fixed for the entire diagnostic process, but may change.

SD can (and for practical purposes has to) be decomposed into different knowledge sources. In a first step, we can identify that SD comprises (at least) knowledge about

- the **domains of the variables** used to define models (e.g. what constitutes a discrepancy) (V-DOMAIN)
- the **constituents** in the application domain (LIBRARY),
- the **structure** of of the device to be diagnosed (e.g. identify variables shared among constituents) (D-STRUCTURE).

(Again, we refer to [Struss 91] for a more detailed description of SD's content, in particular for relational models). Furthermore, the **LIBRARY** contains

- the **model graphs** for the constituents (M-GRAPH), i.e. inferences such as
  $M_1 \wedge \text{dhyp} \Rightarrow M_1'$,
- the set of **model definitions** for the explicit models (M-DEF), in the relational case according to Definition 4.1 or its variants:

$$M_1 \Leftrightarrow$$
$$\forall \underline{v}_0 \in DOM(\underline{v}_c) \quad ((\exists s \in SIT \quad Val(s, \underline{v}_c, \underline{v}_0)) \Rightarrow \underline{v}_0 \in R) .$$

Hence, we assume that, at each stage of the diagnostic process, SD is given by

$$SD = \text{V-DOMAIN} \cup \text{D-STRUCTURE} \cup \text{M-GRAPH} \cup \text{M-DEF}_{ACT}$$
$$= SD_{CORE} \cup \text{M-DEF}_{ACT} ,$$

where the set of active models, $\text{M-DEF}_{ACT} \subseteq \text{M-DEF}$, normally contains only a small subset of model definitions at a time. For the sake of simplicity, we assume that SD_{CORE} is always complete, and remains stable (actually, we need to consider only subsets of it, as well, dependent on the active models).

Organizing the use of constituent models in the diagnostic process according to the principle stated above is based on a monotonicity property whose ultimate foundation is captured by the following theorems.

Theorem 5.1

Let M, and M' be models of one constituent C, and M-DEF(M) and M-DEF(M') be the respective model definitions.

if Δ is a diagnosis for $OBS \cup SD_{CORE} \cup \text{M-DEF}_{ACT} \cup \{\text{M-DEF(M)}\}$
and M'(C) is a view of M(C),
then Δ is a diagnosis for $OBS \cup SD_{CORE} \cup \text{M-DEF}_{ACT} \cup \{\text{M-DEF(M')}\}$.

Basically, this theorem says that when working with a view of a model, we do not loose a diagnosis we would obtain when using the original model; or, stated differently, that switching to this more powerful model is really a step of refinement. This provides the ultimate justification for applying models, which are gained by (qualitative) abstraction or which model only particular physical aspects, in order to cut down the space of possible diagnoses before further investigation with more fine-grained, but also more costly models.

Of course, when using simplified models, this kind of monotonicity will be restricted, as indicated by the following theorem.

Theorem 5.2

Let M, and M' be models of one constituent C, and M-DEF(M) and M-DEF(M') be the respective model definitions.

if Δ is a diagnosis for $OBS \cup SD_{CORE} \cup M\text{-}DEF_{ACT} \cup \{M\text{-}DEF(M)\}$,
and $M'(C)$ is a simplification of $M(C)$: $M \wedge dhyp \Rightarrow M'$,
then Δ is a diagnosis for $OBS \cup SD_{CORE} \cup M\text{-}DEF_{ACT} \cup \{M\text{-}DEF(M')\}$,
or $dhyp \in \Delta$.

This theorem formulates what is in accordance with our intuition, namely that by using a simplified model, the system will infer all diagnoses that can be obtained from the original one and do not contain the retraction of the underlying simplifying assumption. Part of the diagnosis space that is based on $\neg dhyp$ may be invisible; however, it can be regained in DP, since the diagnostic assumptions are made explicit and kept in dependencies.

We illustrate this process by going back to the thyristor example and its model graph in Fig. 5.3. Assume we diagnose a thyristor stand-alone and start by activating the correct mode only while keeping all simplifying assumptions:

$M\text{-}DEF_{ACT1} = \{M\text{-}DEF(TH_{SWITCH})\}$.

If the predictions based on $SD_1 = SD_{CORE} \cup M\text{-}DEF_{ACT1}$ are inconsistent with the observations, $\{th_{CORRECT}, \sigma_1, \sigma_2, \sigma_3\}$ is a conflict, and, under a focus of suspicion that does not contain diagnoses involving any of σ_1, σ_2, σ_3 (the diagnostic hypotheses currently taken for granted), $\Delta = \{th_{CORRECT}\}$ is the only diagnosis, and the thyristor is considered faulty.

If now fault models are activated by the system, while still keeping the simplifications σ_1, σ_2, σ_3, we have

$M\text{-}DEF_{ACT2}$
$= \{M\text{-}DEF(TH_{SWITCH}), M\text{-}DEF(TH^{F'}{}_{BLOCKING}), M\text{-}DEF(TH^{F'}{}_{PUNCTURED})\}$.

Let us assume that the two fault models are also contradicting the observations. This invalidates $TH^{F}{}_{BLOCKING}$ and $TH^{F}{}_{PUNCTURED}$ and, hence, $TH^{F}{}_{SWITCH}$. Under the simplifying assumptions σ_1, σ_2, σ_3, now both $TH_{CORRECT}$ and TH_{FAULTY} are refuted, and so is $TH_{POSSIBLE}$, which is considered to be a fact. This inconsistency triggers a change in the focus of suspicion, since it can only be resolved by retracting simplifying assumptions, in our case (at least) σ_3. Permitting the occurrence of the respective modeling assumption in diagnoses extends the space of diagnoses again and activates new models:

$M\text{-}DEF_{ACT3} = \{M\text{-}DEF(TH_{THRESH}), M\text{-}DEF(TH^{F'}{}_{BLOCKING}),$
$M\text{-}DEF(TH^{F'}{}_{PUNCTURED}), M\text{-}DEF(TH^{F'}{}_{V_{B0}})\}$.

If is $TH^{F'}{}_{V_{B0}}$ is also inconsistent with the observations while TH_{THRESH} is not, the thyristor is considered correct, under the simplifying assumptions σ_1, σ_2.

6 Summary

We presented a theory of transformation, abstraction and simplification of relational models that allows us to prove logical links between different models which in turn lay the foundation for their well-understood utility in a diagnostic framework. In particular, we are able to characterize classes of mappings between different representations that preserve model properties. Furthermore, explicit conditions can be described for simplified versions of a model to be still appropriate.

The concepts for structuring models we developed, together with the capabilities of the DP framework enables us to chunk our knowledge about a system's behavior in such a way that we can obtain results by instantiating and using only a portion of the entire model. Because modeling assumptions can be represented explicitly, the system is able to reason about them and has a basis for a controlled navigation through the model graph. We are convinced that progress in developing a theory of diagnosis with multiple, abstract and simplified models will be a major step towards a general theory of diagnosis and crucial for expanding the range of real applications of model-based diagnosis.

Acknowledgements I would like to thank Toni Beschta, Danny Bobrow, Johan de Kleer, Oskar Dressler, Hartmut Freitag, Gerhard Friedrich, Georg Gottlob, Walter Hamscher, Wolfgang Nejdl, Olivier Raiman, Brian Williams, and several reviewers for discussions and comments on earlier versions. This work was supported in part by BMFT (ITW 8506 E4, ITW 9001 A9) and has been partially undertaken in Esprit Project ARTIST (P5143) which involves the following partners: Cise, Cepsa, Delphi, Heriot-Watt University, Labein and Siemens. The author wishes to acknowledge the contribution of all members of the project team whilst taking full responsibility for the views expressed in this paper. The work is partly supported by the C.E.C. under the ESPRIT program.

References

[Beschta et al. 90]
Beschta, A., Dressler, O., Freitag, H., and Struss, P., *A Model-based Approach to Fault Localization in Power Delivery Networks*, Siemens Technical Report INF 2 ARM-15-D-90, 1990 (In German)

[Davis 82]
Davis, R., *Expert Systems: Where Are We? And Where Do We Go From Here?*, The AI Magazine, Spring 1982.

[Davis 84]
Davis, R., *Diagnostic Reasoning Based on Structure and Behavior.* Artificial Intelligence, 24(1):347-410, 1984.

[de Kleer et al. 90]
de Kleer, J., Mackworth, A., and Reiter, R., *Characterizing Diagnoses.* In: Proceedings of the AAAI 90

[de Kleer-Williams 87]
de Kleer, J., and Williams, B. C., *Diagnosing Multiple Faults.* In: Artificial Intelligence, 32(1):97-130, April 1987

[de Kleer-Williams 89]
de Kleer, J., and Williams, B. C., *Diagnosis with Behavioral Modes.* In: Proc. 11th Int. Joint Conf. on Artificial Intelligence, pages 1324-1330, Detroit, MI, 1989

[de Kleer-Williams 90]
de Kleer, J.and Williams, B. C., *Focusing the Diagnosis Engine*, Xerox PARC, 1990

[Dressler 88]
Dressler, O., *An Extended Basic ATMS.* In: Proceedings of the Second Workshop on Non-Monotonic Reasoning, Springer 1988

[Dressler 90]
Dressler, O., *Computing Diagnoses as Coherent Assumption Sets.* In: G. Gottlob, W. Nejdl (eds), Expert Systems in Engineering, Heidelberg, 1990

[Dressler-Farquhar 90]
Dressler, O. , Farquhar, A., *Putting the Problem Solver Back in the Driver's Seat: Contextual Control of the ATMS.* In: J.P.Martins (ed.). Proceedings of the ECAI 90 Truth Maintenance Workshop

[Dressler-Struss 91]
Dressler, O. , Struss, P., *Back to Defaults: Computing Diagnoses as Coherent Assumption Sets.* Working Paper, Munich, 1991

[Falkenhainer-Forbus 90]
Falkenhainer, B. and Forbus, K. D., *Compositional Modeling of of Physical Systems.* 4th International Workshop on Qualitative Physics, Lugano, Switzerland, 1990

[Friedrich et al. 90]
Friedrich, G., Gottlob, G., and Nejdl, W., *Physical Impossibility Instead of Fault Models.* In: Proceedings of the AAAI 90

[Hamscher 90]
Hamscher, W. C., *Diagnosing Devices with Hierarchic Structure and Known Component Failure Modes.* In: Proc. 6th IEEE Conf. on A. I. Applications, Santa Barbara, CA, March 1990

[Struss 88a]
Struss, P., *A Framework for Model-based Diagnosis.* Technical Report INF 2 ARM-10-88, Siemens, Munich, 1988

[Struss 88b]
Struss, P., *Extensions to ATMS-based Diagnosis.* In: J. S. Gero, editor, Artificial Intelligence in Engineering: Diagnosis and Learning, pages 3-27. Elsevier and Computational Mechanics Publications, Amsterdam and Boston, August 1988

[Struss 89a]
Struss, P., *Diagnosis as a Process.* First International Workshop on Model-Based Diagnosis, Paris, 1989

[Struss 89b]
Struss, P., *Model-Based Diagnosis - Progress and Problems.* In: Proc. 3rd Intern. GI Congress "Knowledge-Based Systems", Munich 1989

[Struss 89c]
Struss, P., *Problems of Interval-Based Qualitative Reasoning - Revised Version.* In: Weld, D. and de Kleer, J. (eds.), Readings in Qualitative Reasoning about Physical Systems, San Mateo, 1989

[Struss 91]
Struss, P., *What's in SD? - Towards a Theory of Model-based Diagnosis.* Working Paper, Munich, 1991

[Struss-Dressler 89]
Struss, P., Dressler, O., *"Physical Negation" - Integrating Fault Models into the General Diagnostic Engine,* Proceedings IJCAI-89

[Weld 90]
Weld, D.S., *Approximation Reformulations.* In: Proceedings AAAI-90

[Weld-Addanki 90]
Weld, D.S. and Addanki, S., *Task-Driven Model Abstraction.* 4th International Workshop on Qualitative Physics, Lugano, Switzerland, 1990