Design and Implementation of Diagnostic Strategies using Modal Logic

Peter Fröhlich, Wolfgang Nejdl, and Michael Schroeder

Institut für Rechnergestützte Wissensverarbeitung
Lange Laube 3, D-30159 Hannover
Tel.: ++49-511-7629710 Fax.: ++49-511-7629712
Email: {froehlich,nejdl,schroeder}@kbs.uni-hannover.de

Abstract. The ability to select suitable diagnostic assumptions and models extends the power of model-based diagnosis for complex systems and can explicitly be modeled by diagnostic strategies. Recently, Nejdl, Fröhlich and Schroeder have developed a framework, which allows to express these strategies as formulas of a meta–language. This paper presents a method for designing strategy knowledge bases as well as an efficient straightforward operational semantics for exploiting them.

1 Introduction

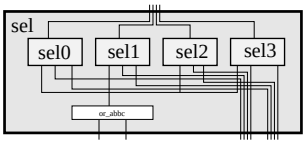
In the last years, model–based diagnosis has been extended by the introduction of the new concept of using different diagnostic assumptions (concerning number of faults, use of fault models and multiple models of the device) which can be activated during the diagnostic process. The selection of the appropriate diagnostic assumptions and system models during the diagnostic process is controlled by a set of predicates in the system description, for which Struss, Böttcher and Dressler [Str92, BD93, BD94] introduced the term *Working Hypotheses*. Diagnostic strategies are rules defining which working hypotheses should be used in a given situation during the diagnostic process. Nejdl, Fröhlich and Schroeder recently introduced a framework [NFS95] for controlling the diagnostic process by strategies expressed as sentences in a formal meta–language. Compared to previous approaches such as [BD94] this framework has the advantage of making strategic knowledge explicit and allowing the flexible specification of diagnostic strategies. The framework includes a declarative semantics for deciding whether a diagnostic process obeys the strategic knowledge specified by the strategies. In this paper we extend previous work by developing a design method and an operational semantics to efficiently handle strategic knowledge. Preference concepts, which were up to now modeled separately [DS92, DPN94, FNS94] will also be expressed by strategies. We first show how to develop a strategic knowledge base for a specific application. Then we define an operational semantics which efficiently performs the diagnostic process using the strategies provided. In the course of the paper we will discuss both deterministic and non–deterministic strategies for hierarchical circuit diagnosis. A major advantage of our method is that the strategies can be designed independently of each other, so that the

strategic knowledge can be easily extended without the need to rewrite existing strategies.

1.1 Working Hypotheses

Let $\mathcal{L}$ be a first order language with equality. We consider a system and observations described by sets of formulas $SD \subseteq \mathcal{L}$ and $OBS \subseteq \mathcal{L}$. Struss introduced the concept of working hypotheses into model–based diagnosis in order to make diagnostic assumptions explicit [Str92]. Working hypotheses are denoted by a set $WHYP$ of atoms from the language $\mathcal{L}$. They can be used to represent multiple models of the system within one system description as shown in the following example.

Example 1.1 *Use of Working Hypotheses*



A 4-bit-selector is composed of four 1-bit-selectors and an or-gate. The working hypothesis refine(sel) can be used to switch between the abstract model and the detailed model of the selector in which its subcomponents sel_i and or_abbc are visible.

In SD the behavior of the selector is modeled depending on the working hypothesis refine(sel): The rules of the abstract model contain ¬refine(sel) in their bodies, while the rules of the detailed model contain refine(sel).

To compute the diagnoses under a set of working hypotheses s, we add s and its negated complement $\neg\bar{s} := \{\neg wh | wh \notin s\}$ wrt. $WHYP$ to the system description. We do not assume a particular diagnosis definition, but we use a generic function *diag*, which implements the computation of diagnoses. This definition allows for a wide range of diagnosis concepts like minimal diagnoses [Rei87], most probable diagnoses [dKW87], preferred diagnoses [DPN94] and others.

Definition 1.1 *Let SD be a system description, OBS a set of observations and s a set of working hypotheses. Then* $diag_s(SD \cup OBS) = diag(SD \cup OBS \cup s \cup \neg\bar{s})$ *denotes the* Set Of Diagnoses Under Working Hypotheses s.

Working hypotheses are an important concept for making the current diagnostic assumptions explicit. The selection of the appropriate working hypotheses is controlled by strategies.

2 A Language for Diagnostic Strategies

In this section we extend the previously defined strategy language [NFS95] by generic deterministic and non-deterministic strategies, a careful treatment of the case where no diagnoses exist and a general condition for the termination of the process.

2.1 Syntax of the language

Diagnostic strategies control the diagnostic process by specifying which working hypotheses to use in a given situation. The state of the diagnostic process is represented by the current set of possible diagnoses. Thus the specification of a diagnostic strategy consists of

- properties characterizing the current situation
- working hypotheses which are suitable for handling that situation

Example 2.1 *Using the technique described in example 1.1, we can represent multiple models of a system within a single system description. Suppose we have three system models M_0, M_1 and M_2, where M_0 is active by default. The working hypotheses force_M1 and force_M2 can be used to switch to M_1 and M_2 respectively. The following rule could be used to guide this selection:*

> *If all diagnoses satisfy a certain formula C which states that model M_0 is not appropriate, then the diagnostic process can continue either using model M_1 or model M_2, i.e. either the working hypothesis force_M1 or the hypothesis force_M2 can be activated.*

To capture such strategies we define modalities to specify properties of the current diagnoses as well as to propose working hypotheses

Modalities $\Box$ and $\Diamond$. The preconditions of diagnostic strategies are statements about the current set of possible diagnoses. The atomic statements in these conditions are denoted by S5 modal operators:

$\Box p$**:** p is true under all diagnoses from $diag_s$.
$\Diamond p$**:** p is true under at least one diagnosis from $diag_s$.

Modalities $\blacksquare$ and $\blacklozenge$. Strategy formulas specify which working hypotheses should be assumed in a given situation. This is achieved by the modalities:

$\blacksquare\Box wh$**:** wh is a *necessary* working hypotheses in the current situation, i.e. in all successor states of the current diagnostic state we must assume wh.
$\blacklozenge\Box wh$**:** wh is a *possible* working hypotheses in the current situation, i.e. there is a successor state of the current diagnostic state in which wh holds.

Based on properties of the current state of the diagnostic process the strategies propose new working hypotheses. They can be written as a rules $C \rightarrow H$, where C characterizes the current diagnostic state and H the immediate successor states. The head H has one level of modalities $\blacklozenge$ and $\blacksquare$, the body C has none.

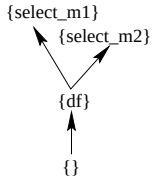
Definition 2.1 *Let us first define the* Depth *of a formula. If $L \in \mathcal{L}$, then $\Box L$ and $\Diamond L$ are formulas of depth 0. Let F, G be both formulas of depth n, then $\Box F$, $\Diamond F$, $\neg F$, $F \wedge G$, $F \vee G$ are formulas of depth n, $\blacksquare F$ and $\blacklozenge F$ are formulas of depth $n+1$. A* One–step Strategy *(in this paper simply called* Strategy*) is a formula of the form $C \rightarrow H$, where C is a formula of depth 0 and H is a formula of depth 1.*

2.2 Representation of a Diagnostic Process

The aim of strategies is to control the diagnostic process by proposing suitable working hypotheses. The diagnostic process itself can be non–deterministic, because more than one set of hypotheses can be proposed for a given situation. The *State* of the diagnostic process is characterized by the current set of working hypotheses. For representing diagnostic processes we use the notion of a *State Transition System*:

Definition 2.2 *Let S be a finite set of states, $t \in S$ an initial state and $\rightarrow \subseteq S \times S$ a transition relation. Then $(S, \rightarrow, t)$ is called a* State Transition System.

Consider the following diagnostic process. We start the diagnosis with a simple model of a device (M_0) and the single–fault–assumption[1]. It turns out that no single–fault diagnosis exists, thus we allow the assumption of double faults. Again no diagnoses are found. Now we consider the simple model of the device as too abstract and we propose the activation of one of the more detailed models M_1 or M_2. In both of these models we would find single–fault diagnoses.



The transition system for representing this process has the states $\emptyset$ (initial state, no working hypotheses), *df* (then allow double faults), *force_M1* and *force_M2* (select one of the more detailed models).

2.3 Designing Strategies

The diagnostic process is represented by a transition system. We use strategies to influence the shape of this transition system. Let us first show how to define strategies in order to obtain a certain transition system.

Deterministic Strategies. Often we want to assume one specific hypothesis in a given situation. For example, if we have found that a component C is definitely faulty, we activate the refined model for it to see which of its subcomponents caused the fault. We need a strategy which proposes a state transition leading to a state in which *refine*(C) holds:

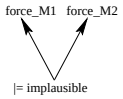
$$\Box ab(C) \rightarrow \blacklozenge \Box refine(C) \wedge \blacksquare \Box refine(C)$$

The formula describes exactly the transition system wrt. the hypothesis *refine(C)*. Other strategies specify transition systems wrt. other hypotheses. These partial transition systems are then combined by the algorithm which computes the diagnostic process. The $\blacklozenge$–operator is necessary in this formula. If we had only used the $\blacksquare$–operator in the conclusion of this formula, the formula would have been satisfied also if there were no successor of the current state. The quantification over "all successor states" would then be trivially satisfied.

[1] In section 3 we show how these assumptions can be modeled by strategies.

Non–Deterministic Strategies. Sometimes there are several possibilities for continuing the diagnostic process in a given situation. Consider the generic model selection strategy introduced in example 2.1. Again we can first develop a transition system and then describe it by a strategy.

$$\Box \mathit{implausible} \rightarrow \\ \quad \blacklozenge\Box \mathit{force\_M1} \wedge \blacklozenge\Box \mathit{force\_M2} \\ \quad \wedge\blacksquare(\Box \mathit{force\_M1} \not\leftrightarrow \Box \mathit{force\_M2})$$



In this strategy *implausible* is a predicate in the system description, which indicates that there is no plausible diagnosis under the current system model, e.g. implausible may hold if there more than two faults. The strategy proposes two possibilities for continuing the diagnostic process: Either assume M_1 or assume M_2 but do not assume both models at the same time. We will give a more specific account on model selection in section 3.

The above design method allows to define strategies independently without having to care about interference between different strategies. We will use it to define a complete set of strategies for circuit diagnosis (section 3) and describe how the independence can be preserved during the diagnostic process (section 4).

2.4 Consistency of Transition Systems

In this section we formalize what it means for a transition system to be *consistent* with a diagnostic problem and a set of strategies, i.e. when the way the diagnostic process proceeds is consistent with what the strategies propose. Later, in section 4 we will define how to compute such a consistent transition system. A transition system is just an encoding of the working hypotheses used in each step of the diagnostic process and is influenced by the diagnoses found and the strategies given. To understand the declarative semantics, consider a given transition system $(S, \rightarrow, t)$. We check whether this transition system is a valid solution to the diagnostic problem given by $SD \cup OBS$ and a set of strategies $STRAT$. We define when a state s of a transition system $(S, \rightarrow, t)$ satisfies a formula $F \in STRAT$, which we will denote by $(S, \rightarrow, t) \models_s F$. Then we call a transition system *consistent*, iff all its states satisfy all formulas from $STRAT$, i.e. $\forall s \in S. \forall F \in STRAT. (S, \rightarrow, t) \models_s F$. Next we define the operator $\models_s$.

Modalities $\Box$ *and* $\Diamond$. First consider a formula F without modal operators. A state s satisfies $\Diamond F$ if there is a diagnosis corresponding to state s under which F holds, and a state satisfies $\Box F$, if F holds under all diagnoses in state s. Additionally, we consider the case when there are no diagnoses, because the inconsistency of certain assumptions with the current situation should not lead to termination of the diagnostic process but rather to a change of assumptions. The absence of diagnoses under a given set of literals is indicated by the literal $ab(SD)$, intuitively indicating that the system description is not suited for the current set of assumptions. Let $\mathcal{D} = diag_s(SD \cup OBS)$, then we define:

$(S, \rightarrow, t) \models_s \Diamond F$, iff $\mathcal{D} \neq \emptyset$ and there exists $D \in \mathcal{D}$, such that $SD \cup OBS \cup s \cup \neg\bar{s} \cup D \models F$
or $\mathcal{D} = \emptyset$ and $s \cup \neg\bar{s} \cup \{ab(SD)\} \models F$.
$(S, \rightarrow, t) \models_s \Box F$, iff $\mathcal{D} \neq \emptyset$ and for all $D \in \mathcal{D}$: $SD \cup OBS \cup s \cup \neg\bar{s} \cup D \models F$
or $\mathcal{D} = \emptyset$ and $s \cup \neg\bar{s} \cup \{ab(SD)\} \models F$.

Modalities ■ *and* ⧫. Now let F be a formula of depth 0. The semantics for the operators $\blacksquare F$ and $\blacklozenge F$ is given by the transitions. A state s satisfies $\blacksquare F$, if the formula F holds in all successor states. Similarly, a state satisfies $\blacklozenge F$, if the formula F holds in at least one successor state.

$(S, \rightarrow, t) \models_s \blacklozenge F$, iff $\exists s' \in S$ such that $s \rightarrow s'$ and $(S, \rightarrow, t) \models_{s'} F$.
$(S, \rightarrow, t) \models_s \blacksquare F$, iff $\forall s' \in S$: from $s \rightarrow s'$ follows $(S, \rightarrow, t) \models_{s'} F$.

For the logical connectives $\wedge$, $\vee$, $\neg$, etc. the $\models_s$ operator is defined as usual. We will discuss the computation of a consistent transition system in section 4.

2.5 Results of the Diagnostic Process

The aim of the diagnostic process could be to identify one unique diagnosis. In general this would be a too restrictive criterion for terminating the diagnostic process because we might not have enough knowledge to discriminate among all the diagnoses. So we define that the diagnostic process terminates in a state where we assumed exactly all possible and necessary hypotheses.

This corresponds to a loop in the transition system as depicted on the right. If such a state yields diagnoses we cannot reach a more preferred state by applying another strategy.

Definition 2.3 *Stable State*
Let s be a state in the consistent state transition system $(S, \rightarrow, t)$ and STRAT a set of strategy formulas. The state s is stable *wrt. $(S, \rightarrow, t)$, iff*

1. $diag_s(SD, OBS) \neq \emptyset$
2. $s = \{wh \mid (S, \rightarrow, t) \models_s \blacklozenge\Box wh \wedge \blacksquare\Box wh\}$

It is called weakly stable, *if $s \subseteq \{wh \mid (S, \rightarrow, t) \models_s \blacklozenge\Box wh \wedge \blacksquare\Box wh\}$*

The first condition states that $SD \cup OBS \cup s$ is consistent and the second condition is a fixpoint condition: s is already the set of all possible and necessary working hypotheses. The result of the diagnostic process is given by the diagnoses corresponding to the stable states and weakly stable states, respectively.

3 A Strategy Knowledge Base for Circuit Diagnosis

In this section we apply the strategy language to the diagnosis of digital circuits. We model strategies such as the choice among multiple models, structural refinement, measurements and preferences. In section 5 use them to diagnose the voter circuit from [Isc85].

Multiple Views. Multiple views allow to describe the diagnosed systems emphasizing different aspects. For circuit diagnosis it is often important to consider a physical view beside a functional one, because the physical view additionally takes the layout into account [Dav84]. We want to employ the functional model by default and the physical model only if we do not obtain good diagnoses.

Strategy (1) tells us how to choose between the models using the hypotheses *force_physical* and *force_functional*. The predicate *implausible*, which in our example holds if no single or double fault diagnosis exists, indicates that the other view should be activated. To avoid more than one activation it is also checked that *force_functional* does not yet hold. Once the body of the strategy is satisfied we have to make sure that the diagnostic process continues in two directions with the functional and the physical model, respectively, as active model. Thus, we adopt either *force_functional* or *force_physical*. Once this model selection has taken place both hypotheses are kept (monotonic addition of working hypotheses) (2, 3).

$$\begin{aligned}&\Box \mathit{functional} \wedge \Box \mathit{implausible} \wedge \Box\neg \mathit{force\_functional} \rightarrow \\ &\quad \blacklozenge\Box \mathit{force\_functional} \wedge \blacklozenge\Box \mathit{force\_physical} \wedge \\ &\quad \blacksquare\Box(\mathit{force\_functional} \leftrightarrow \neg \mathit{force\_physical}) \qquad (1)\\ &\Box \mathit{force\_physical} \rightarrow \blacklozenge\Box \mathit{force\_physical} \wedge \blacksquare\Box \mathit{force\_physical} \qquad (2)\\ &\Box \mathit{force\_functional} \rightarrow \blacklozenge\Box \mathit{force\_functional} \wedge \blacksquare\Box \mathit{force\_functional} \qquad (3)\end{aligned}$$

In the system description we model the connection between the working hypotheses *force_physical* and *force_functional* and the literals *physical* and *functional*, which select the appropriate system model. The functional model is used by default when no hypothesis is active:

$$\begin{aligned}&\neg \mathit{force\_functional} \wedge \neg \mathit{force\_physical} \rightarrow \mathit{functional}\\ &\mathit{force\_functional} \rightarrow \mathit{functional},\ \mathit{force\_physical} \rightarrow \mathit{physical}\end{aligned}$$

Structural Refinement. Many authors address the use of hierarchies to reduce the complexity of diagnosis problems [Dav84, Ham91, Gen84, Moz91, BD94]. In particular, Böttcher and Dressler introduce the strategy of structural refinement which states that an abstract model of a component is refined only if it is uniquely identified as defective [BD94]. Only if all diagnoses contain a component C, it is possible and necessary to activate a detailed model of C:

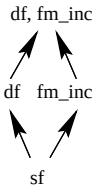
$$\forall C.(\Box \mathit{ab}(C) \wedge \mathit{refineable}(C)) \vee \Box \mathit{refine}(C) \rightarrow \blacklozenge\Box \mathit{refine}(C) \wedge \blacksquare\Box \mathit{refine}(C) \qquad (4)$$

In the system description the rules describing the abstract model are active when $\mathit{refine}(C)$ is false and the rules describing the detailed model are activated if $\mathit{refine}(C)$ is true. This variant of using hierarchies is very efficient since the refinement of the model is postponed until the erroneous components are identified.

Preference Relations among Diagnoses. Preferences state that diagnoses with certain properties are better than others with other properties [DS92, DPN94, FNS94]. Frequently used preferences are for example the single fault assumption or "physical negation" [SD89], i.e. the assumption that the known fault models of the components are complete. To use preferences efficiently, the preferred property is activated by default and is relaxed only if there are no diagnoses which have the intended property. We can use $ab(SD)$ to detect if there are any diagnoses. $ab(SD)$ holds iff $diag_s(SD \cup OBS) = \emptyset$ (see sec. 2.4).

For a given number n of faults the system description contains a predicate nf that holds iff at most n faults are assumed:

$$\bigvee_{i,j=1,\, i \neq j}^{n+1} C_i = C_j \leftarrow nf, \bigwedge_{i=1}^{n+1} ab(C_i)$$



The preference relation on the left states that by default we are only interested in single faults (*sf*). If there are no diagnoses under the single–fault assumption we allow either diagnoses with double faults (*df*) or incompleteness of the fault models (*fm_inc*). If there are still no diagnoses under one of these relaxed hypotheses, we allow double fault diagnoses and incompleteness of fault models at the same time.

The system description captures the default assumption of single faults by the rule $\neg df \wedge \neg tf \rightarrow sf$. The strategy of relaxing the single-fault property (5) checks (using $ab(SD)$) if no diagnoses exist and if neither df nor fm_inc hold. In this case it is possible to adopt either fm_inc or df, but not both at the same time. Finally, double faults together with the assumption of incomplete fault models are allowed only if there are no double fault diagnoses and no single–fault diagnoses with incomplete fault modes (6).

$$\Box ab(SD) \wedge \Box\neg df \wedge \Box\neg fm\_inc \rightarrow \blacklozenge\Box fm\_inc \wedge \blacklozenge\Box df \wedge \blacksquare(\Box df \not\leftrightarrow fm\_inc) \quad (5)$$

$$\Box ab(SD) \wedge (\Box df \vee \Box fm\_inc) \rightarrow \blacklozenge\Box(df \wedge fm\_inc) \wedge \blacksquare\Box(df \wedge fm\_inc) \quad (6)$$

These kind of strategies show the desired behavior discussed and implemented in [FNS94]. First, the diagnosis system tries to find diagnoses under the most preferred set of properties (in our case diagnoses with only single faults). Only if this is not possible (i.e. $ab(SD)$) is true), these properties are exchanged with the next most preferred set of properties (in our case either double faults or the assumption of "fault mode incomplete") and so on. Whenever a diagnosis strategy not related to these preferences is executed (e.g. refinement, multiple views etc), diagnosis in this changed state again starts by trying to find diagnoses corresponding to the most preferred set of properties.[2]

[2] This is the result of making the preference property assumptions not monotonic.

Measurements. De Kleer, Raiman and Shirley view diagnosis as an incremental task involving the three phases of generating explanations, choosing actions differentiating among them and performing these actions [dKRS91]. Our framework allows to incorporate these phases into the diagnostic process. Strategy (7) proposes a point X of the circuit for measurement if there are two diagnoses predicting different values for X. As measurements are expensive, we want to apply the strategy only if all refinements are already done which is checked in the first line.

$$\forall C.(\Box \mathit{refineable}(C) \wedge \Diamond \mathit{ab}(C) \rightarrow \Box \mathit{refine}(C)) \wedge$$
$$\forall X.\Box \mathit{point}(X) \wedge \Diamond \mathit{val}(X,0) \wedge \Diamond \mathit{val}(X,1) \rightarrow \blacklozenge \Box \mathit{propose}(X) \wedge \blacksquare \Box \mathit{propose}(X) \quad (7)$$

The second phase of choosing the right action is carried out by procedural attachment in the system description. The (procedural) predicate *best_meas(X)* is true for a measurement point X, which is optimal according to some specification (for example minimum entropy). It need only be evaluated for the measurement points X proposed by strategy (8).

$$\forall X.\Box \mathit{propose}(X) \wedge \Box \mathit{best\_meas}(X) \rightarrow \blacklozenge \Box \mathit{measure}(X) \wedge \blacksquare \Box \mathit{measure}(X) \quad (8)$$

In the system description the hypothesis $\mathit{measure}(X)$ causes the specific measurement of X to be executed, which is also done by procedural attachment. The modification of the system description due to the insertion of the measured value has to be reflected by a change of the diagnostic state. This is achieved by using the hypothesis *measure(X)* in a monotonic way (strategy (9)). Another (weak) monotonicity axiom is used for *propose*, which is active until the next consistent state is reached, in which the measurement is carried out, i.e. as long as $\mathit{ab}(SD)$ holds (strategy (10)).

$$\forall X.\Box \mathit{measure}(X) \rightarrow \blacklozenge \Box \mathit{measure}(X) \wedge \blacksquare \Box \mathit{measure}(X) \quad (9)$$
$$\forall X.\Box \mathit{ab}(SD) \wedge \Box \mathit{propose}(X) \rightarrow \blacklozenge \Box \mathit{propose}(X) \wedge \blacksquare \Box \mathit{propose}(X) \quad (10)$$

4 Operational Semantics

The strategies presented in this paper were designed by describing transition systems. These strategies have the important property that they are satisfied by exactly one transition system. Thus, the meaning of these strategies is completely determined by the semantics of the strategy language. Now the question arises, whether every transition system can be defined by a strategy. The answer is positive. We will define the *Characteristic Formula* of a transition system in this section. All the strategies in this paper are equivalent to characteristic formulas. We further present a method which combines the transition systems defined by a set of strategies. Our method will have the property that it maximizes the chance that a consistent diagnostic process is found. Finally the method is exploited by a simple iterative algorithm.

4.1 Characteristic Formulas

Given a transition system $(S, \rightarrow, t)$, we can systematically define a formula F_t, which completely characterizes $(S, \rightarrow, t)$:

Definition 4.1 *Let $(S, \rightarrow, t)$ be a transition system. Then F_t is called* Characteristic Formula*, where t is the initial state of $(S, \rightarrow, t)$. In general the characteristic formula of a state s is defined recursively by*

$$F_s = G_s \wedge \bigwedge_{s \rightarrow s'} \blacklozenge F_{s'} \wedge \blacksquare \bigvee_{s \rightarrow s'} F_{s'}$$
$$G_s = \Box \bigwedge \{wh \mid wh \in s\} \cup \{\neg wh \mid wh \in (\bigcup S) \backslash s\}$$

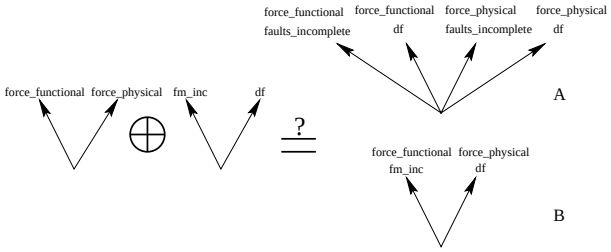
The first conjunct G_s of the formula F_s *fully* characterizes the current state s, the second conjunct manifests the existence of the successor states and the third conjunct states that there are no other successors.

Example 4.1 *For the strategy structural refinement in section 3, the head of the strategy is a characteristic formula for the transition system with the only transition from the empty set to the state $\{refine(C)\}$.*

4.2 Combining Strategies

When we consider more than one strategy formula we have to solve the problem of combining the proposed transitions. Suppose we have two strategy formulas $C_1 \rightarrow H_1$ and $C_2 \rightarrow H_2$ and the current state of the process satisfies both C_1 and C_2. How do we combine the transitions proposed by H_1 and H_2?

Example 4.2 *Recall the strategies* (6) *of preferring single fault diagnoses over double faults and incomplete fault modes and the strategy* (1) *of activating the physical model. Assume the bodies of* (6) *and* (1) *are satisfied in the current state and we have to perform transitions to satisfy the heads $\blacklozenge\Box fm\_inc \wedge \blacklozenge\Box all\_df \wedge \blacksquare(\Box all\_df \not\leftrightarrow fm\_inc)$ and $\blacklozenge\Box force\_functional \wedge \blacklozenge\Box force\_physical \wedge \blacksquare\Box(force\_functional \leftrightarrow \neg force\_physical)$, respectively. There are several transition systems satisfying the conjunct of these two heads:*



Solution B is not desired as it includes only two of four possible combinations of the working hypotheses.

Formally the independence of two strategies proposing working hypotheses wh_1 and wh_2 means that looking at a state in which wh_1 is active, we cannot derive the truth value of wh_2 in that state.

Definition 4.2 *Independence of Strategies*
Let F be a set of strategies. Let $(S, \rightarrow, t)$ be a consistent transition system wrt. F. The state $s \in S$ satisfies Independence of Strategies*, iff there is no transition system $(S_1, \rightarrow_1, t_1)$ consistent with F such that*

1. *for all working hypotheses $wh_1, wh_2 \in S \cup \neg S$ such that $(S_1, \rightarrow_1, t_1) \models_s \blacksquare\Box wh_1 \rightarrow wh_2$ we have $(S, \rightarrow, t) \models_s \blacksquare\Box wh_1 \rightarrow wh_2$*
2. *there are working hypotheses $wh_1, wh_2 \in S \cup \neg S$ such that $(S, \rightarrow, t) \models_s \blacksquare\Box wh_1 \rightarrow wh_2$ but $(S_1, \rightarrow_1, t_1) \not\models_s \blacksquare\Box wh_1 \rightarrow wh_2$*

The transition system $(S, \rightarrow, t)$ satisfies independence of strategies iff every state $s \in S$ satisfies independence of strategies. Treating strategies as independent has several advantages: When writing down strategies we explicitly specify dependencies among certain hypotheses. Independence to other hypotheses need not to be specified. This is important in a case where a strategy formula is added to a large set of existing formulas. Furthermore, assuming independence maximizes our chance to find a solution in the case of non–determinism. If this transition system does not lead to a stable state, there will be no other system leading to a consistent state.

In the following we will show that there is only one transition system that satisfies independence of strategies for a given set of strategies. Thus the semantics is completely specified and can be computed efficiently. In order to combine the transition systems defined by the heads of two strategies while preserving independence, we simply combine the successor states in all possible ways. We call this operation the *State Product*.

Definition 4.3 *Given two sets of states S_1 and S_2 the* State Product $S_1 \otimes S_2$ *is defined as $\{s_1 \cup s_2 \mid s_1 \in S_1, s_2 \in S_2\}$.*

When constructing the successor transitions for a given state during the diagnostic process, we instantiate the strategies (quantification over components) and collect the heads H_i of the strategies $C_i \rightarrow H_i$, whose conditions C_i are satisfied. We construct the transition systems corresponding to the heads $\{H_i\}$. Then we combine them by applying the following theorem:

Theorem 4.1 *Let $H_1, H_2, \ldots H_n$ be characteristic formulas of depth 1 which have no working hypotheses in common and let $(S_1, \rightarrow_1, t_1), (S_2, \rightarrow_2, t_2), \ldots (S_n, \rightarrow_n, t_n)$ be the corresponding transition systems. The following transition system $(S, \rightarrow, t)$ satisfies independence of strategies:*

$$S = \emptyset \cup \bigotimes_{i=1}^{n} (S_i - \{\emptyset\}), \quad \rightarrow = \{(\emptyset, s) \mid \ s \in S\}, \quad t = \emptyset$$

The theorem 4.1 describes how to compute the successor states of a given state under the assumption of independent strategies. Iterative application of this theorem yields a straightforward method for computing a transition system satisfying a given set of strategies. In a given state s starting with $\emptyset$ we have to execute the following steps:

1. Compute the diagnoses and corresponding system models under state s.
2. Instantiate the body of the strategies using the current diagnoses/models. Collect the heads of the satisfied strategies.
3. Construct a transition system for each head.
4. Combine the resulting transition systems using state product.

The method must be recursively applied to every generated state.

5 An Example

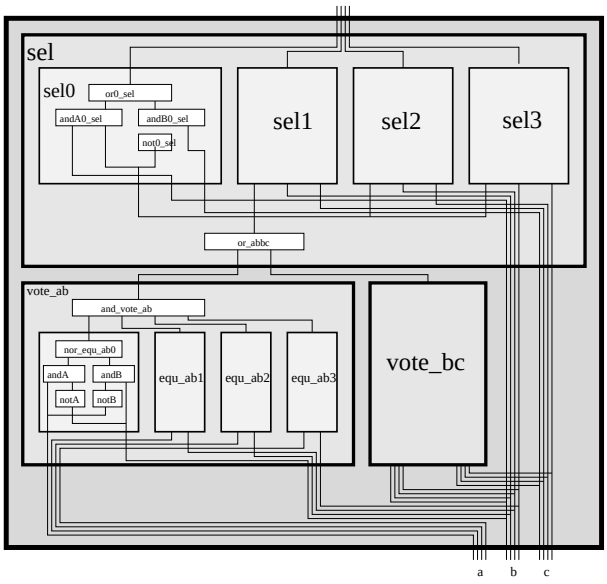


Fig. 1. Voter

A voter (see Fig. 1) has three 4-bit-inputs a, b, c. It outputs b if $(a = b) \vee (b = c)$ and otherwise c. The equality check is realized by the components *vote_ab* and *vote_bc*. Both are composed of an and-gate and 4 comparators *equ_xy*, which serve as inputs for the and-gate. A comparator *equ_xy* compares 2 bits by realizing the boolean function $x\bar{y} \vee \bar{x}y$ and thus consists of 2 not- and 2 and-gates and a

nor-gate. The select component in turn contains 4 one-bit-selectors sel_i which are controlled by the or-gate *or_abbc_sel*. If it is high, selector sel_i lets b_i pass, otherwise c_i is passed. This is realized by 2 and-gates, an or- and a not-gate.

In the process depicted in Fig. 2 the three input words are all 0000 and the

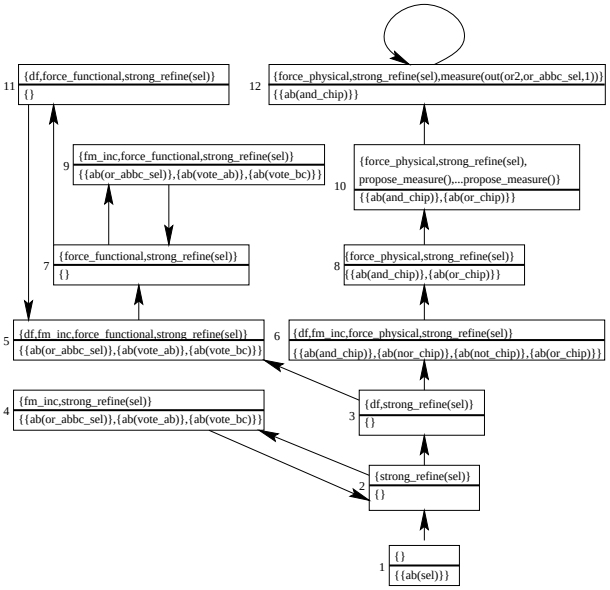


Fig. 2. A diagnostic process.

output is observed to be 1111. The top level diagnosis uniquely identifies *sel* as abnormal (1), but the following refinement does not lead to any diagnoses (2). So the single-fault-assumption is relaxed and two successor states are created, allowing double faults and incomplete fault models, respectively. With incomplete fault modes some diagnoses are found (4). Since the hypothesis of incomplete fault modes is not monotonic, we have to drop it again. The consequence is a loop between two states, in which only the state with incomplete fault modes is consistent. By definition 2.3 we have reached a weakly stable state. The search for double faults (3) in the other branch is not successful. Two strategies apply in this situation. In all successor states we have to allow incompleteness of fault models in addition to double faults (section 3). Furthermore we have to branch between physical and functional model as proposed by the multiple views strategy (section 3). Two successor states are generated:

– With double faults and incomplete fault modes three diagnoses are found (5). The search for more preferred diagnoses first leads to no diagnoses (7). Allowing double faults does not help (11), while dropping the completeness

of fault modes assumption yields three single faults (9), so that this state is again weakly stable.
- Beside the computations in the functional model, we obtain diagnoses of the physical model (6). With double faults and incomplete fault modes allowed, five diagnoses are consistent with the observation. Thus in the next step the preferences are relaxed and *and_chip* and *or_chip* are valid diagnoses in the physical view (8). In order to discriminate among those two diagnoses several measurements are proposed (10). Among them the point *or_abbc_sel* is chosen and finally the state $\{force\_physical, strong\_refine(sel), measure(out(or2, or\_abbc\_sel, 1))\}$ is stable.

6 Conclusion

To cope with large-scale systems the theory of model-based diagnosis has been extended to include concepts such as multiple views [Dav84, Ham91], hierarchies [Dav84, Ham91, Gen84, Moz91, BD94], preferences [DPN94, FNS94] and measurements [dKRS91]. Struss introduced the idea of diagnosis as process [Str92], further developed by Böttcher and Dressler [BD93, BD94]. Fröhlich, Nejdl, Schroeder formalized it by defining a meta-language that allows to describe the process declaratively [NFS95]. We continued this work in two directions. First, we showed how to design strategies to cover the concepts mentioned above. Second, we developed an operational semantics and an algorithm that processes these strategies and efficiently computes the diagnostic process. We identified generic strategies to deal with monotonic and non-monotonic working hypotheses as well as deterministic and non-deterministic strategies. In particular, the combination of non-monotonic working hypotheses and non-determinism allowed us to express preferences which usually have to be treated in a separate framework [DPN94, FNS94]. We showed how to use multiple views and how to employ hierarchies by the strategy of structural refinement. We integrated measurement strategies using procedural attachment. We defined characteristic formulas and independence of strategies which lead to an efficient algorithm that covers the whole variety of strategies. The design and evaluation of these strategies was evaluated in the domain of digital circuits using the voter circuit from [Isc85].

Acknowledgement

We would like to thank BMFT for financial support and the referees for their comments.

References

[BD93] C. Böttcher and O. Dressler. Diagnosis process dynamics: Holding the diagnostic trackhound in leash. In *IJCAI93*, pages 1460–1471. Morgan Kaufmann Publishers, Inc., 1993.

[BD94] C. Böttcher and O. Dressler. A framework for controlling model-based diagnosis systems with multiple actions. *Annals of Mathematics and Artificial Intelligence*, 11(1–4), 1994.

[Dav84] R. Davis. Diagnostic reasoning based on structure and behaviour. *Artificial Intelligence*, 24:347–410, 1984.

[dKRS91] J. de Kleer, O. Raiman, and M. Shirley. One step lookahead is pretty good. In *Second International Workshop on the Principles of Diagnosis*, Milano, Italy, October 1991.

[dKW87] J. de Kleer and B. C. Williams. Diagnosing multiple faults. *Artificial Intelligence*, 32:97–130, 1987.

[DPN94] C. V. Damásio, L. M. Pereira, and W. Nejdl. Revise: An extended logic programming system for revising knowledge bases. In *KRR94*, pages 607–618, Bonn, Germany, 1994. Morgan Kaufmann Publishers, Inc.

[DS92] O. Dressler and P. Struss. Back to defaults: Characterizing and computing diagnoses as coherent assumption sets. In *ECAI92*, pages 719–723, 1992.

[FNS94] P. Fröhlich, W. Nejdl, and M. Schröder. A formal semantics for preferences and strategies in model-based diagnosis. In *5th International Workshop on Principles of Diagnosis (DX-94)*, pages 106–113, New Paltz, NY, 1994.

[Gen84] M. R. Genesereth. The use of design descriptions in automated diagnosis. *Artificial Intelligence*, 24:411–436, 1984.

[Ham91] Walter C. Hamscher. Modeling digital circuits for troubleshooting. *Artificial Intelligence*, 51(1-3):223–271, October 1991.

[Isc85] The iscas-85 benchmark archive. Accessible via anonymous ftp from ftp.mcnc.org, 1985.

[Moz91] Igor Mozetič. Hierarchical model-based diagnosis. *International Journal of Man-Machine Studies*, 35:329–362, 1991.

[NFS95] W. Nejdl, P. Fröhlich, and M. Schroeder. A formal framework for representing diagnosis strategies in model–based diagnosis systems. In *IJCAI95*, pages 1721–1727, 1995. Morgan Kaufmann Publishers, Inc.

[Rei87] Raymond Reiter. A theory of diagnosis from first principles. *Artificial Intelligence*, 32:57–95, 1987.

[SD89] P. Struss and O. Dressler. Physical negation — Integrating fault models into the general diagnostic engine. In *IJCAI89*, pages 1318–1323, Morgan Kaufmann Publishers, Inc.

[Str92] P. Struss. Diagnosis as a process. In W. Hamscher, L. Console, and J. de Kleer, editors, *Readings in Model-Based Diagnosis*, pages 408–418. Morgan Kaufmann Publishers, Inc., 1992.

This article was processed using the LaTeX macro package with LLNCS style