3rd Int. Conference on Artificial Intelligence and Applications (AIA 03).
Benalmádena, Spain, September 2003. Edited by M. H. Hanza.
pp. 191-197, ISBN 0-88986-390-3, ACTA Press, 2003.

Model Compilation and Diagnosability of Technical Systems

Benno Stein
Paderborn University
Department of Computer Science
email: stein@upb.de

Abstract

This paper is on the automated construction of diagnosis models for complex, continuous-valued systems. Given is the following diagnosis scenario of first principles: (a) The global behavior model of the system in question can be composed from single component models, where (b) context-free models of both the normal and the faulty component behavior are known.

This situation enables one to simulate the system with respect to expected inputs along with faults, and to distill a compiled diagnosis model from the huge set of generated data with mining techniques. As well as that, this situation enables us also to define the most informative measurement points for fault isolation purposes.

The contributions of this paper are threefold: (1) It outlines the compilation approach and its realization in the domain of hydraulic engineering, (2) it extends the GDE measurement heuristics towards an optimum strategy with respect to an arbitrary observation horizon, and (3) it presents a measure to quantify the diagnosis effort for systems whose behavior can be captured by a compiled model.

Key words
diagnosis, automated modeling, simulation, data mining

1 Introduction

We present a diagnosis approach that combines the model-based paradigm with the associative (heuristic) paradigm as follows: By simulating the interesting system in various fault modes and over its typical input range a simulation database is built up. From this database a simplified rule-based behavior model is compiled where long cause-effect chains are replaced with weighted associations and which is optimized for a heuristic classification of the interesting faults. Since this process can be completely automated, the approach has the potential to combine the advantages of the model-based philosophy, such as behavior fidelity and generality, with the efficiency and robustness of a heuristic diagnosis system.

Of course, the approach must not be seen as a universal diagnosis recipe; it comes along with tight applicability conditions: All faults must be component failures, the related fault models must be known, and, typical system inputs must be given. However, there exist many situations where these conditions are fulfilled—one is presented in this paper: The automatic generation of diagnosis systems that detect abrupt component faults in hydraulic systems. Abrupt component faults cause significant deviations from steady state operations [1].

1.1 Model Compilation and its Impacts

The diagnosis of complex systems is a challenge: Following the heuristic paradigm means to capture diagnosis rules from domain experts—a road which is insecure and fault-prone, and which presupposes that expertise is available [2]. Following the model-based paradigm may be precluded for limited computational resources. Even when excellent simulation conditions are given, model-based diagnosis is still problematic: The long interaction paths between variables result in large conflict sets. Moreover, many technical systems have a feedback structure—i. e., cause-effect chains, which are the basis for an assumption-based reasoning process, cannot be easily stated.

In this connection, the compilation of a heuristic model from a model of first principles is promising strategy. Compiled models have a small computational footprint. As well as that, model compilation breaks feedback structures, and, under the assumption that all observations have already been made, an optimum measurement strategy can be developed.[1]

Based on the last consideration, we can also propose a new concept for assessing the diagnosability of a system. The key idea is to relate the information gain of increasing sets of observers to the theoretical optimum. This relation can be expressed as a concentration measure, which we call a system's *discrimination entropy*.

1.2 Relation to Existing Work

The fault detection performance of a diagnosis system depends on the adequateness of the underlying model. Model compilation is one paradigm for constructing adequate models. The model-based diagnosis paradigm, either with or without fault models, provides another possibility [3, 4]. Under the latter paradigm, the cycle of simulation and candidate discrimination is executed at runtime, while under

[1] A fact, which advised us to name our model compilation program DÉJÀVU.

the compilation paradigm it is anticipated in a preprocessing phase.

Processing a compiled diagnosis model is similar to associative diagnosis. Note, however, that the underlying model in an associative system is the result of substantial model formation considerations. By contrast, model compilation pursues a data mining strategy and aims at an automatic detection of associative knowledge [5]. The idea to derive associative knowledge from deep models was proposed amongst others in [6].

With respect to fault detection and isolation (FDI), measurement selection, and diagnosability, a lot of research has been done. A large part of this work concentrates on dynamic behavior effects, which are not covered in this paper [7, 8]. Nevertheless, since our compilation concept focuses on search space and knowledge identification aspects it can be adapted to existing FDI approaches as well.

There is also some research related to model compilation for diagnosis tasks: The work of Console et al. deals with the generation of decision trees [9]; Darwich discusses how rules can be generated for platforms where computational resources are limited [10].

2 Model Compilation in Hydraulic Engineering

Hydrostatic drives provide advantageous dynamic properties and represent a major driving concept for industrial applications. They consist of several types of hydraulic building blocks: Cylinders, which transform hydraulic energy into mechanical energy, various forms of valves, which control flow and pressure of the hydraulic medium, and service components such as pumps, tanks, and pipes, which provide and distribute the necessary pressure p and flow Q. Figure 1 shows two medium-sized examples of circuits we are dealing with.

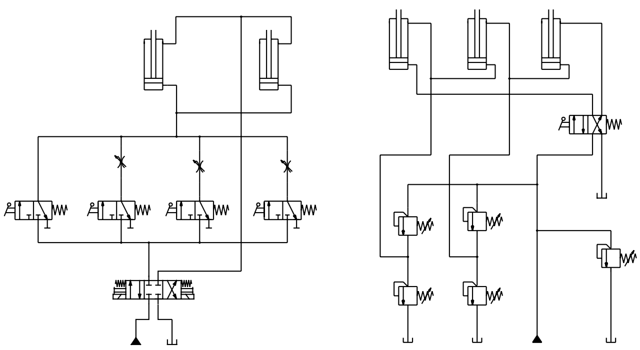

Figure 1. Two hydraulic circuit examples.

2.1 Component Faults and Fault Models

A prerequisite for applying model compilation for diagnosis purposes is that components are defined with respect to both their normal and their faulty behavior. In the following, such a fault model is stated exemplary for the check valve.

Typical check valve faults include jamming, leaking, or a broken spring. These faults affect the resistance characteristic of the valve in first place. Let p_1 and p_2 be the pressure values at the two valve connections, let q be the flow through the valve, and let R be its hydraulic resistance. Then, the pressure drop at a turbulent flow is

$$\Delta p = R \cdot q^2, \quad \text{where} \quad \Delta p := p_1 - p_2.$$

The resistance law is given in Table 1 for both the normal and the faulty behavior. If the valve is operating in its range of control, the fractions are well defined and $\Delta p > p_0$.

Normal behavior	Faulty resistance behavior
$R = \dfrac{m^2 \cdot \Delta p}{(\Delta p - p_0)^2}$	$R = \dfrac{m^2 \cdot \Delta p}{(\Delta p - p_0 \cdot (1 + \varepsilon_{\text{valve}}))^2}$

Table 1. Resistance law of a working and a faulty check valve operating in its control range. The deviation coefficient $\varepsilon_{\text{valve}}$ is a state quantity, which is modeled as a continuous random variable.

Other fault models relate to slipping cylinders due to interior or exterior leaking, incorrect clearance or sticking throttle valves, directional valves with defect solenoid and contaminated lands, and pumps showing a decrease in performance.

For all fault models, a deviation coefficient ε is modeled as a continuous random variable which defines the distribution of the fault seriousness.

2.2 Construction of a Compiled Model

We construct a compiled model in five steps. Within the first step a simulation data base $\mathcal{C}$ is built, which then is successively abstracted towards a real-valued symptom data base $\mathcal{C}_\Delta$, a symbolic interval data base $\mathcal{C}_I$, an observer data base $\mathcal{C}_O$, and, finally, a rule data base $\mathcal{C}_R$, which represents the heuristic diagnosis model.

2.3 Simulation

Behavior models of hydraulic systems are hybrid discrete-event/continuous-time models [11]. I. e., the trajectories of the state variables can be considered as piecewise continuous segments, which are called phases. The discrete state variables such as valve positions, relays, and switches are constant within a phase, and in between the phases one or more of them changes its value, leading to another mode of the system. The continuous variables such as pressures, flows, velocities, positions, which are the target of our learning process, follow continuous curves. In the following the set of continuous variables is denoted by Z.

The quasi-stationary values of the continuous state variables are in the role of symptoms, since abrupt faults may cause their significant change. Our working hypothesis is that between the continuous input variables and several of the continuous state variables a monotonic characteristic can be assumed—as long as a single phase is considered. The sampling procedure reflects this hypothesis as follows.

Let an initial state vector $\mathbf{x}_0$, a vector of input functions $\mathbf{u}(t)$, and some point in time t be given by the user. Then, during simulation, samples of the vector of state trajectories are drawn at those points in time τ, $\tau \leq t$, where a discrete event is imminent. With respect to the compilation process each sample is completed by a number π designating its phase, the input function values at time τ, and the currently enabled component faults $\mathbf{d}$. E. g., under the single-fault assumption a fault simulation vector $\mathbf{c}$ is of the following form:

$$\mathbf{c}(\pi, \mathbf{u}, \mathbf{d}) := (\pi, u_1, \ldots, u_m, x_1, \ldots, x_{|Z|}, d, \varepsilon_{\mathrm{d}})$$

The entirety of normal and fault simulation vectors forms the simulation data base $\mathcal{C}$.

2.4 Symptom Identification

For each fault simulation vector $\mathbf{c}(\pi, \mathbf{u}, \mathbf{d}) \in \mathcal{C}$ the deviations of its state variables to the faultless simulation vector $\mathbf{c}(\pi, \mathbf{u})$ with same $\mathbf{u}$ in the same phase π is computed.

The computation is based on a special operator "$\ominus$", which distinguishes between effort variables and flow variables. The former are undirected, and a difference between two values of this type is computed straightforwardly. The latter contain directional information, and their difference computation distinguishes between seven cases.

Result of this step is the symptom data base of $\ominus$-deviations, $\mathcal{C}_\Delta$, which contains symptom vectors of the form $(\pi, \mathbf{u}, \delta_1, \ldots, \delta_{|Z|}, d)$.

2.5 Interval Formation

The symptom vectors in $\mathcal{C}_\Delta$ are generalized by mapping the deviations $\delta_z^{(1)}, \ldots, \delta_z^{(|\mathcal{C}|)}$, $\delta_z^{(i)} \in \mathbf{R}$, $z \in Z$, onto p intervals $\mathbf{I}_z^{(1)}, \ldots, \mathbf{I}_z^{(p)}$, $\mathbf{I}_z^{(j)} \subset \mathbf{R}$, with $\bigcup_j \mathbf{I}_z^{(j)} = \mathbf{R}$. This is an optimization task where, on the one hand, the loss of discrimination information shall be kept minimum (the larger p the better), while on the other hand, constraints of measuring devices and human operators must be obeyed (the smaller p the better).

This interval formation can be compared to a discretization method that maps a cardinal domain onto an ordinal domain in order to make a classification approach applicable [12, 13]. Such methods are distinguished with respect to locality, supervision, and interdependency. We realized a method that is global, since the interval formation is applied to the entire range of a variable; it is supervised, since it exploits classification knowledge (the faults within an interval); however, it does not consider dependencies between variables.

With this abstraction step the domain of real numbers is replaced by a symbolic, say, propositional-logical representation: For each state variable $z \in Z$ a new domain I_z is introduced. I_z is the union of *interval names* ι_z, which map in a one-to-one manner onto the real-valued intervals $\mathbf{I}_z \subset \mathbf{R}$. The symbolic interval database that develops from $\mathcal{C}_\Delta$ by interval formation is denoted with $\mathcal{C}_I$ and contains symbolic symptom vectors of the form $(\pi, \mathbf{u}, \iota_1, \ldots, \iota_{|Z|}, d)$. Note that the number of simulation vectors has not been reduced, say, $|\mathcal{C}_I| = |\mathcal{C}|$.

2.6 Measurement Selection

By means of simulation, values are computed for all variables in Z. In fact, restricted to a handful of measuring devices or sensors, only a small subset O of Z can be observed at the system. Measurement selection means to determine the most informative variables in Z—or, speaking technically, to place a set of $|O| = k$ observers such that as much faults as possible can be classified. In the sequel, the variables in O are called observers.

O is determined by analyzing for each phase π and for each variable $z \in Z$ the correlations between the symbolic intervals I_z and the set of component faults D. The analysis combines considerations from statistics and information theory [14, 15].

- *Observer Dependency.* Observers that depend on each other correlate in their diagnosis information and must be excluded from further examination.

  Because of the multivariate rule generation approach, the dependency analysis here can be restricted to the bivariate case. Since the observers' domains are nominally scaled, the contingency coefficient of Pearson is used. It relies on the χ^2 contingency which measures the association between two variables in a two-way table. Table 2 shows an example.

	$\delta_{p_x} < 20$	$\delta_{p_x} \geq 20$	$\sum$
$\delta_{q_y} < 1.5$	$\boxed{a}$ $\boxed{b}$ $\boxed{c}$	$\boxed{a}$ $\boxed{d}$	5
$\delta_{q_y} \geq 1.5$	$\boxed{d}$	$\boxed{b}$ $\boxed{c}$	3
$\sum$	4	4	8

Table 2. The table shows for observed deviations at pressure p_x and flow q_y the distribution of four component faults $\boxed{a}$, $\boxed{b}$, $\boxed{c}$, $\boxed{d}$. Note that each variable is associated with two symbolic intervals.

- *Observer Information.* At heart, the considerations presented here generalize the idea of hypothetical measurements; the idea goes back on the work of Forbus and de Kleer who try to estimate the measuring cost hidden in a particular diagnosis situation.

Reasoning by hypothetical measurements means to evaluate for all $z \in Z$ how an observed deviation δ_z would reduce the set of possible diagnosis D. For instance, assuming that $D = \{\boxed{a}, \ldots, \boxed{h}\}$ and that we are given the simulation results shown in Table 2, a measurement of q_y resulting in the symptom "$\delta_{q_y} \geq 1.5$" complies only with the component faults $\boxed{b}$, $\boxed{c}$, and $\boxed{d}$. However, the measurement could also result in the symptom "$\delta_{q_y} < 1.5$" where the component faults $\boxed{a}$,...,$\boxed{d}$ come into question.

With respect to the database $\mathcal{C}_I$ and a given phase π, let $\kappa(z, \iota) \subseteq D$ designate the set of diagnoses that comply with symptom (z, ι). Related to the example, $\kappa(q_y, \text{"}\geq 1.5\text{"}) = \{\boxed{b}, \boxed{c}, \boxed{d}\}$. If one presumes that all diagnoses (component faults) in the set D occur equally likely, then the probability that a particular symptom (z, ι) will occur can be estimated by $|\kappa(z, \iota)| / \sum_{\tau \in I_z} |\kappa(z, \tau)|$, the fraction of diagnoses that comply with the symptom.

If we also knew the measurement effort to discriminate amongst the remaining diagnoses $\kappa(z, \iota)$, the most informative observer in Z could be determined. Here, the simplifying assumption is made that the diagnoses D are equally distributed over the $|I_z| = r$ intervals in I_z, $z \in Z$. Henceforth, $\log_r \kappa(z, \iota)$ defines a lower bound for the number of measurements that are necessary to isolate each of the faults in $\kappa(z, \iota)$.

The considerations are comprised in Equation (1), which estimates the discrimination effort to identify a component fault from D using observer z, when given the diagnosis situation described by the interval database $\mathcal{C}_I$.

$$e(z) = \sum_{\iota \in I_z} \frac{|\kappa(z, \iota)|}{\sum_{\tau \in I_z} |\kappa(z, \tau)|} \cdot log_r |\kappa(z, \iota)|, \qquad (1)$$

where $r = |I_z|$. The minimization of Equation (1) over all $z \in Z$ is used as a heuristic to determine the most informative observers $O \subset Z$. If a-priori probabilities $P(d)$ for the faults $d \in D$ are known, they can be integrated in the likelihood estimator of Equation (1).

Equation (1) resembles the formula of Forbus and de Kleer; it differs with respect to the identity $\sum_{\tau \in I_z} |\kappa(z, \tau)| = |D|$, which does not hold in our compilation situation.

Let $O \subset Z$ be the set of selected observers. The database that emerges from the symbolic interval database $\mathcal{C}_I$ by eliminating all variables in $Z \setminus O$ is called observer database $\mathcal{C}_O$; it is much smaller than $\mathcal{C}_I$. However, its number of elements is unchanged, i. e., $|\mathcal{C}_O| = |\mathcal{C}|$.

2.7 Rule Generation

Within the rule generation step reliable diagnosis rules are extracted from the observer database $\mathcal{C}_O$. The rules have a propositional-logical semantics and are of the form

$$\mathbf{r} = \iota_{o_1} \wedge \ldots \wedge \iota_{o_k} \rightarrow d,$$

where $\iota_{o_i} \in I_{o_i}, o_i \in O, d \in D$, and $k \leq |O|$. $O \subset Z$ is the set of the chosen observers; the symbols of a rule form a subset of a single vector $\mathbf{c} \in \mathcal{C}_O$. The left and right sides of the rule are called premise and conclusion respectively.

The semantics of such a rule $\mathbf{r}$ is defined by means of two propositional-logical truth assignment functions, $\alpha : \bigcup_{z \in Z} I_z \rightarrow \{0, 1\}$ and $\beta : D \rightarrow \{0, 1\}$. For some constraint variable $z \in Z$, let $\mathbf{I}$ be the real-valued interval associated with the interval symbol ι, and let δ_z be a symptom. Then α and β are defined as follows.

$$\alpha(\iota) = \begin{cases} 1 & \Leftrightarrow \ \delta_z \in \mathbf{I} \\ 0 & \text{otherwise.} \end{cases} \qquad \beta(d) = \begin{cases} 1 & \Leftrightarrow \ \text{fault is } d. \\ 0 & \text{otherwise.} \end{cases}$$

A truth assignment function α matches a rule $\mathbf{r}$ if its premise, $\mathbf{r}^-$, becomes true under α. If also the rule conclusion becomes true under β, then $\mathbf{r}$ is called positive.

Note that the inference direction of the above rules is reverse to the cause-effect computations when simulating a behavior model: We now ask for symptoms and deduce faults, and—as opposed to the simulation situation—this inference process must not be definite. Perhaps there is a definite mapping from symptoms to faults in the original simulation database $\mathcal{C}$. Even so, it is very likely that the rigorously simplified observer database $\mathcal{C}_O$ encodes ambiguities. I. e., rules with the same premise (symbolic intervals) that are associated with different faults.

To cope with this form of uncertain knowledge we forget about a strictly logical interpretation and characterize each rule $\mathbf{r}$ by its confidence, c, and its support, s:

$$c(\mathbf{r}) = \frac{h(\mathbf{r})}{h(\mathbf{r}^-)} \qquad \text{and} \qquad s(\mathbf{r}) = \frac{h(\mathbf{r})}{|\mathcal{C}_O|},$$

where $h(\mathbf{r})$ denotes the frequency of $\mathbf{r}$ in $\mathcal{C}_O$, while $h(\mathbf{r}^-)$ denotes the frequency of the rule's premise in $\mathcal{C}_O$.

Rule generation is realized with data mining methods and yields the rule database $\mathcal{C}_R$. In particular, we employ strategies with respect to confidence-thresholds and subsumption handling to avoid computational overhead. Nevertheless, rule generation still is a combinatorial problem. Note that in the data mining jargon, rules of the described form are called "association rules" [17].

2.8 DÉJÀVU: Model Application and Results

Model application means to process the rules in $\mathcal{C}_R$ in the context of observed symptoms. It requires an operational semantics for the rules' confidence and support values. The classics amongst the rule-based systems that employs rules with confidences is MYCIN [18]. MYCIN's underlying computation scheme is designed for the accounting of a handful of rules—it fails in our setting where confidences of 10-100 rules predicting the same diagnosis candidate $d \in D$ must be accounted.

We developed the better suited formula below, which computes for each fault $d \in D$ its confidence in "$\beta(d) = 1$" given a rule database $\mathcal{C}_R$ and a truth assignment α. The formula consists of two parts: (1) A base term, where the impact of a positive rule with maximum confidence cannot be weakened and, (2) an update term, where the confidences of the positive rules are weighted with all matching rules.

$$c(\text{"}\beta(d) = 1\text{"}) = c(\mathbf{r}^*) + \big(1 - c(\mathbf{r}^*)\big) \cdot \frac{1}{|\mathcal{R}^-|} \sum_{\mathbf{r} \in \mathcal{R}} c(\mathbf{r}),$$

where $\mathcal{R}^- \subset \mathcal{C}_R$ comprises the matching rules, $\mathcal{R} \subset \mathcal{C}_R$ comprises the positive rules, and $\mathbf{r}^*$ denotes a positive rule of maximum confidence.

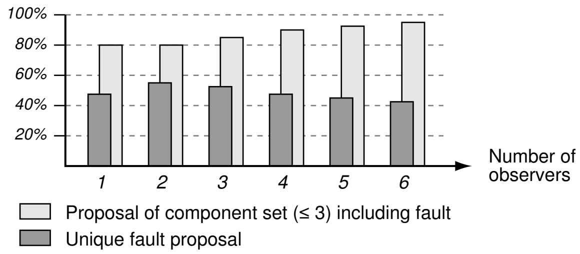

Figure 2. Classified faults depending on the number of observers. Dark bars indicate a unique proposal of the faulty component, light bars a multiple prediction ($\leq$ 3 components including fault).

The outlined model construction process as well as the rule inference have been realized within the diagnosis program DÉJÀVU [19]. For simulation purposes, DÉJÀVU employs the FLUIDSIM simulation engine [20]. Note that the automatic simulation and recording of a large number of operating scenarios implies demanding problems on its own, which cannot be discussed in this place.

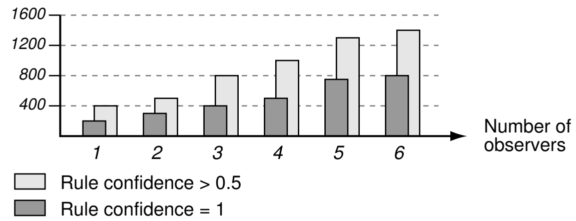

Figure 3. Number of generated rules depending on the number of observers. Dark bars indicate rules with a confidence value of 1, light bars stand for confidence values > 0.5.

Using DÉJÀVU and FLUIDSIM the approach has been applied to several medium-sized hydraulic circuits (about 20-40 components) with very promising results. Figure 2 shows diagnosis hit rates depending on the number of observers; basis were more than 2000 variations of $|D| \approx 15$ different component faults, the circuits' driving processes were clearly defined. The results were achieved with automatically constructed rule databases $\mathcal{C}_R$ that have not been manually revised. Figure 3 shows average values of the rule data base sizes.

3 Model Compilation Enlarges the Observability Horizon

The abstraction from the real-valued simulation database $\mathcal{C}$ towards the symbolic interval database $\mathcal{C}_I$ provides the ground for applying information-theoretical considerations to the measurement selection. This section shows the true power of model compilation: The selection heuristic, Equation (1), can be turned into an optimum measurement strategy.

Equation (1), which estimates the effort to discriminate between several diagnoses in D when using observer $z \in Z$, has a look-ahead of 1: For each observable interval ι, the discrimination must be continued amongst the remaining set of diagnoses $\kappa(z, \iota)$. A global selection strategy would determine a set of observers $O \subset Z$ such that the overall discrimination effort is minimum.

Within the diagnosis setting of the GDE, a *globally* optimum selection strategy can only be employed, if additional hypothetical simulation runs are performed. Hypothetical simulations are initiated by hypothetical measurements. In this connection let ι be a possible outcome from a hypothetical measurement at some observer $z \in Z$. Then ι is interpreted as additional system input, and for each component in the conflict set a simulation is carried out having its state transition function disabled. Since such a symptom-driven, hypothetical simulation concept is computationally very expensive, Forbus and de Kleer do not follow this idea. Moreover, the execution of symptom-driven simulations in connection with real-valued behavior models is questionable because of the infinite hypotheses space.

Within our compiled model setting, which is encoded by $\mathcal{C}_I$, the situation is different. A large database with simulation scenarios is at our disposal that can be exploited for a global selection strategy. In this regard, we introduce the conditional probability $P_z(\iota|D)$ which specifies the probability that the symptom ι can be observed at observer $z \in Z$ under the condition that some fault from D has been occurred. We use the frequency distribution of D in the database $\mathcal{C}_I$ to estimate the probabilities $P_z(\iota|D)$:

$$P_z(\iota|D) = \frac{\left|\{d \in k(z,\iota) \mid d \in D\}\right|}{\sum_{\iota \in I_z} \left|\{d \in k(z,\iota) \mid d \in D\}\right|},$$

where $k(z, \iota)$ is the multiset counterpart of $\kappa(z, \iota)$. I. e., $k(z, \iota)$ is the set of diagnosis that comply with symptom "(z, ι)", and multiple occurrences of the same interval-fault combination are counted multiply. Related to the example in Table 2, $k(q_7,$"< 1.5"$) = \{\boxed{a}, \boxed{a}, \boxed{b}, \boxed{c}, \boxed{d}\}$, and $P_{q_7}($"$<$ 1.5"$|\{\boxed{a}, \boxed{b}\}) = 3/4$.

Now Equation (1) can be extended to exploit a-priori knowledge about the diagnoses D amongst which the observer $z \in Z$ shall discriminate:

$$e(z, D) = \sum_{\iota \in I_z} P_z(\iota|D) \cdot log_r|D \cap \kappa(z, \iota)|, \qquad (2)$$

where $r = |I_z|$. The minimization of Equation (2) over Z yields the most informative observer for a look-ahead of 1. By a recursive application of Equation (2) to the remaining sets of diagnoses $D \cap \kappa(z,\iota)$, we can enlarge the observation horizon—until a unique fault classification is achieved. Each recursion step corresponds to a new observation.

Given a number of observations allowed, k, we define the *discrimination effort* for a system as the number of observations that must additionally be made to discriminate between all diagnoses. Clearly, this makes sense only if the k observations are optimum with respect to the expected information gain. The following definition fulfills the demanded; it provides a lower bound for the expected number of additional observations.

Definition 1 (Expected Discrimination Effort) *Let S be a system that is characterized by an interval database $\mathcal{C}_I$. $\mathcal{C}_I$ defines the set of diagnoses D, the set of possible observers Z, the conditional probabilities P_z, and the function κ. Then, the expected discrimination effort of S with respect to a maximum number of observations $k > 0$ is defined as*

$$\hat{e}(D,k) = \begin{cases} \min\limits_{z \in Z} \left(\sum\limits_{\iota \in I_z} P_z(\iota|D) \cdot \hat{e}(D \cap \kappa(z,\iota), k-1) \right), & \text{if } k > 0 \text{ and } |D| > 1 \\ \log_r(|D|), & \text{if } k = 0 \text{ or } |D| = 1 \end{cases}$$

where I_z comprises the intervals of an observer $z, z \in Z$, $r = |I_z|$, and the function κ returns for an observer z and an interval $\iota \in I_z$ the set of complying diagnoses.

When setting $k = 1$ and employing the relative frequency instead of the conditional probability, $\hat{e}(D,k)$ becomes the original formula of Forbus and de Kleer [16].

Remarks. The definition of the expected discrimination effort implies several assumptions. (1) The set of diagnoses, D, is complete, (2) the diagnoses in D are equally distributed, and (3) the available observers, Z, are independent from each other. The presented formula uses the same resolution r for all observers but can easily be extended to allow for observer-specific resolutions r_z.

4 Quantifying a System's Diagnosability

There is the interesting question of how to assess the difficulty to diagnose a system. In the following we will present the necessary considerations and develop such measure. Starting point is the formula for the expected discrimination effort, $\hat{e}(D,k)$.

If $k = 1$ then $\hat{e}(D,k) = \min_{z \in Z} \left(\sum_{\iota \in I_z} P(\iota|D) \cdot \log_r(|\kappa(z,\iota)|) \right)$. The term $\sum_{\iota \in I_z} P(\iota|D) \cdot \log_r(|\kappa(z,\iota)|)$ becomes minimum if the diagnoses are distributed equally amongst the r intervals in I_z. This, in turn, allows us to factor out the term $\log_r(|\kappa(z,\iota)|)$, and $\hat{e}(D,1)$ simplifies to $\log_r(|\kappa(z,\iota)|) \cdot \sum_{\iota \in I_z} P(\iota|D) = \log_r(\frac{|D|}{r})$.

Repeating the same assumptions for $k = 2$ yields:

$$\begin{aligned} \hat{e}(D,k) &= \min_{z \in Z} \left(\sum_{\iota \in I_z} P(\iota|D) \cdot \hat{e}(\kappa(z,\iota),1) \right) \\ &= \hat{e}(\kappa(z,\iota),1) \\ &= \log_r(\tfrac{|\kappa(z,\iota)|}{r}) \\ &= \log_r(\tfrac{|D|}{r^2}) \end{aligned}$$

Note that the minimum number of observations totally required depends on both the observers' resolution, say, their number of intervals, r, and the number of diagnoses $|D|$. The infimum number of observations necessary to discriminate between each diagnosis is $\lfloor \log_r |D| \rfloor$. It is used to specify E^*, the accumulated ideal discrimination effort of a system as follows.

Definition 2 (Accumulated Ideal Discrimination Effort) *The accumulated ideal discrimination effort of a system S with respect to a set of diagnoses D and an observer resolution r is defined as*

$$E^*(D) := \sum_{i=1}^{\lfloor \log_r |D| \rfloor} \log_r \frac{|D|}{r^i}$$

The difference between the accumulated expected and the accumulated ideal discrimination effort can be used as a measure for the difficulty to diagnose a system. The larger this difference is the more does a faulty system behave agnostic. Note that this measure gives an estimation that is independent of the number of possible observers, thus providing a system-specific characteristic. At the best, the difference between the expected and the ideal discrimination effort is zero. Figure 4 illustrates the difference between the discrimination efforts pictorially; the accumulated difference is called discrimination entropy here.

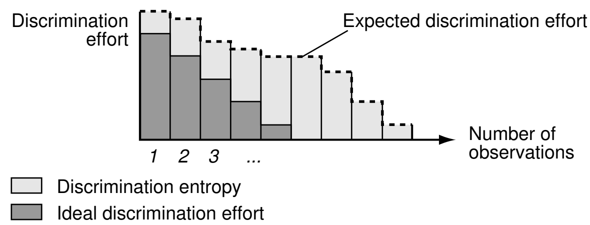

Figure 4. Discrimination entropy: The difference between the accumulated expected and the ideal discrimination effort.

Definition 3 (Discrimination Entropy) *The discrimination entropy E of a system S with respect to a set of diagnoses, D, is defined as*

$$E := \left(\sum_{k=1}^{\infty} \hat{e}(D,k) \right) - E^*(D)$$

5 Conclusions

Our work shows that the model compilation paradigm can be applied to generate working diagnosis models for complex systems such as hydraulic plants. However, the construction process is fairly involved and employs methods from learning theory, statistics, information theory, and data mining.

Perhaps more interesting are the presented "by-products" of model compilation: A globally optimum measurement strategy and a diagnosability measure, called discrimination entropy. Given a database with simulation records of some—possibly unknown—system, the concept of discrimination entropy allows us to quantify the diagnosis effort that can be expected.

Especially with respect to the design of a system both considerations may be important: A perfect measurement strategy provides guidance to place sensor devices optimally; the concept of discrimination entropy can be used to construct systems with respect to their maintenance effort.

References

1. Pieter J. Mosterman and Gautam Biswas. Diagnosis of Continuous Valued Systems in Transient Operating Regions. *IEEE Transactions on Systems, Man, and Cybernetics*, 29(6):554–565, 1999.

2. F. Hayes-Roth, D. A. Waterman, and D. B. Lenat. *Building Expert Systems*. Addison-Wesley, Massachusetts, 1983.

3. Raymond Reiter. A Theory of Diagnosis from First Principles. *Artificial Intelligence*, 32(1):57–95, April 1987.

4. Peter Struß and Oskar Dressler. "Physical Negation"—Integrating Fault Models into the General Diagnostic Engine. In *Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence (IJCAI 89)*, pages 1318–1323, 1989.

5. R. Srikant and R. Agrawal. Mining Quantitative Association Rules in Large Relational Tables. In H. V. Jagadish and I. S. Mumick, editors, *Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data*, pages 1–12, Montreal, Canada, June 1996. ACM Press.

6. Luc Steels. Components of Expertise. *AI Magazine*, 11(2):28–49, 1990.

7. Paul Frank. Fault diagnosis: A survey and some new results. *Automatica: IFAC Journal*, 26(3):459–474, 1990.

8. Sriram Narasimhan, Pieter J. Mosterman, and Gautam Biswas. A Systematic Analysis of Measurement Selection Algorithms for Fault Isolation in Dynamic Systems. In *Proc. of the International Workshop on Diagnosis Principles*, pages 94–101, Cape Cod, MA, 1998.

9. L. Console, C. Picardi, and D. Theseider Dupré. Generating Temporal Decision Trees for Diagnosing Dynamic Systems. In *Proceedings of the 11th Int. Workshop on Principles of Diagnosis*, 2000.

10. Adnan Darwich. On Compiling System Descriptions into Diagnostic Rules. In *Proceedings of the 10th International Workshop on Principles of Diagnosis*, Scotland, June 1999.

11. Paul I. Barton. Modeling, Simulation, and Sensitivity Analysis of Hybrid Systems. In Peter Buneman and Sushil Jajodia, editors, *Proceedings of the IEEE International Symposium on Computer Aided Control System Design*, pages 117–122, Anchorage, Alaska, September 2000. ACM Press.

12. J. R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, San Mateo, CA, 1993.

13. J. Dougherty, R. Kohavi, and M. Sahami. Supervised and Unsupervised Discretization of Continuous Features. In A. Prieditis and S. Russell, editors, *Proceedings of the 12th International Conference on Machine Learning*, pages 194–202, Menlo Park, CA, July 1995. Morgan Kaufmann.

14. Joachim Hartung. *Statistik*. Oldenbourg, München, 1999.

15. Johan de Kleer and Brian C. Williams. Diagnosing Multiple Faults. *Artificial Intelligence*, 32:97–130, 1987.

16. Kenneth D. Forbus and Johan de Kleer. *Building Problem Solvers*. MIT Press, Cambridge, Massachusetts, 1993.

17. R. Agrawal, T. Imielinski, and A. Swami. Mining Association Rules between Sets of Items in Large Databases. In Peter Buneman and Sushil Jajodia, editors, *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, Washington D. C., May 1993. ACM Press.

18. E. H. Shortliffe. *Computer Based Medical Consultations:* MYCIN. Elsevier, New York, 1976.

19. Uwe Husemeyer. *Heuristische Diagnose mit Assoziationsregeln*. Dissertation, University of Paderborn, Dept. of Mathematics and Computer Science, 2001.

20. Benno Stein, Daniel Curatolo, and Marcus Hoffmann. Simulation in FLUIDSIM. In Helena Szczerbicka, editor, *Workshop on Simulation in Knowledge-Based Systems (SIWIS 98)*, number 61 in ASIM Notes, Bremen, Germany, April 1998. Committee 4.5 ASIM of the GI.