Qualitative Model-Based Decision Tree Generation for Diagnosis in Real World Industrial Application

Heiko Milde, Lothar Hotz, Jörg Kahl, Stephanie Wessel
Laboratory for Artificial Intelligence, University of Hamburg
Vogt-Koelln-Str. 30, 22527 Hamburg, Germany
milde@kogs.informatik.uni-hamburg.de
phone: ++49 (0)40 42883-2606, fax: ++49 (0)40 42883-2572

Abstract

Computer diagnosis systems grounded on hand-crafted decision trees are wide-spread in industrial practice. Since the complexity of technical systems increases and innovation cycles get shorter, the need for systematic decision tree generation and maintenance arises. In this paper, the MAD system is introduced which generates decision trees based on a new method for qualitative electrical circuit analysis. Different resources such as design data and expert design know-how as well as diagnosis knowledge can easily be integrated into decision tree generation. Since a decision tree can be generated automatically based on a device model, the cost for providing, modifying, and maintaining diagnosis equipment can be drastically reduced and quality management of diagnosis equipment can be facilitated. Furthermore, the cost of decision-tree-based fault identification can be reduced because model-generated decision trees can be optimized. We have successfully evaluated the MAD system in cooperation with the german forklift manufacturer STILL GmbH Hamburg.

1 Introduction

More than 100.000 forklifts made by the german company STILL GmbH Hamburg are in daily use all over Europe. In order to reduce forklift downtimes, approximately 1100 STILL service workshop trucks utilize decision-tree-based computer diagnosis systems for off-line diagnosis. Due to the complexity of the electrical circuits employed in forklifts, decision trees may consist of more than 5000 objects. When forklift model ranges are modified or new model ranges are released, decision trees are manually generated or adapted by service engineers who apply detailed expert knowledge concerning faults and their effects. Obviously, this practice is costly and quality management is difficult. Furthermore, the cost of decision-tree-based fault identification is unnecessarily high because decision trees are not optimized. Hence, there is a need for computer methods to support systematic modifications and reuse as well as optimization of diagnosis systems. The introduction of new diagnosis techniques, however, raises challenges.

- First, it is essential to integrate innovative with established concepts. A total redesign of the existing diagnosis systems is unacceptable for economical reasons. In particular, for STILL, replacing decision trees was not acceptable.
- Second, it is essential to utilize available resources such as expert knowledge and available product data for diagnosis system generation. This way, the cost of diagnosis systems can be reduced and the trustworthiness of diagnosis data can be improved.

Model-based decision tree generation is a promising answer to the challenges noted above because, in principle, model-based techniques facilitate integrating available resources into the diagnosis equipment. Furthermore, grounding diagnosis systems on a model provides a systematic way for modification, reuse and optimization.

In the STILL application scenario, nodes of decision trees represent fault sets. Edges are labeled by the tests (involving measurements, observations, display values and error codes) which must be carried out to verify the corresponding child node. Although the basic concepts of model-based generation of such decision trees are already described in [Friedrich and Nejdl, 89] and [Mauss, 98], for the reader's convenience, we briefly outline the main ideas of the approach in the following.

The first step to model-based decision tree generation is to model a device. This step is supported by a component library and a device model archive (see Figure 1). Design data and knowledge from the design process (knowledge concerning intended device behavior, expected faults, available measurements) are integrated into the device modeling process. In a second step, correct and faulty device behavior is predicted automatically by evaluating the device model. The third step is to build decision trees from behavior predictions. This step is supported by a decision tree archive and a cost model for the tests which can be performed. Decision tree generation can be performed automatically or guided by service know-how, i.e. knowledge concerning preferable decision-tree topologies and fault probabilities. In order to realize these concepts, we implemented the MAD system (Modeling, Analyzing and Diagnosing) whose main parts are described in this paper.

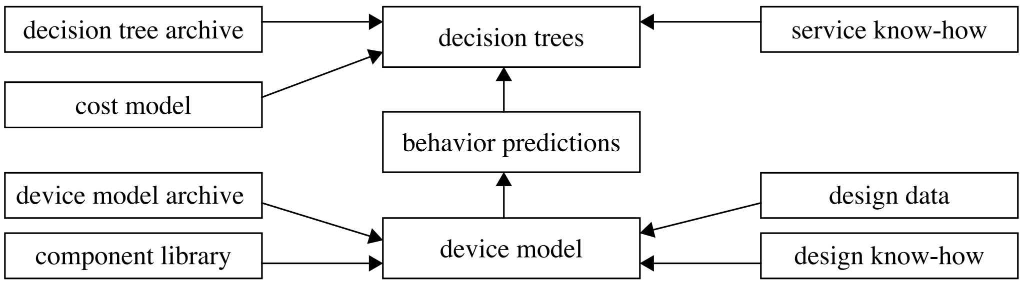

Figure 1: Basic concepts of model-based decision tree generation

In principle, model-based techniques provide a systematic way for predicting the behavior of electrical devices, including faulty behavior. However, since adequate modeling of **heterogeneous** electrical circuits is still a challenge we developed a new method for model-based qualitative network analysis which allows accurate device behavior prediction. Section 2 presents device modeling in MAD. In Section 3, model-based behavior prediction is described. Section 4 briefly outlines the decision tree generation. The evaluation of the MAD system described in Section 5 was performed in cooperation with the STILL GmbH Hamburg.

2 Device modeling

In our application, model-based approaches have to deal with electrical circuits of the automotive domain. These circuits usually consist of components that show a variety of different behavior types, such as analog, digital, static, dynamic, linear, nonlinear and software-controlled behavior. Considering model-based generation of diagnostic decision trees in the forklift application scenario, we could identify the following requirements for reasoning about fault effects in electrical circuits.

1. **Qualitative modeling is essential.** Considering model-based decision tree generation, for

all fault models of the device model, device behavior has to be predicted. Thus, for the sake of tractability, the number of fault models has to be limited. However, in heterogeneous circuits, the number of component faults is unlimited because, if faults occur, analog parameters such as resistances may have any value. Hence, describing faults by exact figures would be highly inappropriate. However, a single qualitative fault model can represent a certain component fault class consisting of an infinite number of different faults. Thus, qualitative network analysis is the essential basis for automated decision tree generation if heterogeneous electrical systems are investigated.

2. **Steady state behavior prediction suffices.** If STILL service workshops apply decision-tree-based diagnosis equipment, only steady state diagnosis is performed. Therefore, only steady state behavior of physical components has to be represented in component models. In particular, an explicit representation of temporal dependencies is not necessary.

3. **Integration of expert knowledge is essential.** Adequate device models are fundamental for accurate behavior prediction and for dealing with complex circuits which consists of a large number of components. To assure accurate device modeling, expert knowledge concerning intended device behavior as well as know-how referring to ignorable physical effects should guide the modeling process. This reflects the insight that the design of modern technical systems and of appropriate innovative diagnosis systems is inseparable.

4. **Dealing with slight parameter deviations and handling changes in circuit structures is essential.** Faults may slightly modify component behavior or may even change device structures. Hence, heterogeneous symptoms, such as slight deviations of parameter values or total loss of functionality may occur. Thus, to assure accurate fault modeling and symptom predicting in different operation modes, reasoning about deviations from reference values as well as reasoning about actual parameter values is fundamental.

5. **Spurious behavior predictions have to be avoided.** If a decision tree is based on spurious behavior predictions, certain faults may not be distinguishable in the decision tree although, in practice, these faults can be easily discriminated. As another point, possibly, decision trees obviously grounded on spurious behavior predictions will not be accepted by service technicians at all. Hence, avoiding spurious behavior predictions is essential.

Established methods for qualitative electrical circuit analysis such as the FLAME system [Pugh and Snooke, 96], the qualitative SPS method [Mauss and Neumann, 96], and the Connectivity method [Struss et al., 1995] are promising but they do not fulfill all of the requirements enumerated above. In particular, these approaches cannot deal with slight parameter deviations and some of these methods generate spurious behavior predictions. Thus, according to the requirements enumerated above, we developed a new method for qualitative electrical circuit analysis described in the following.

2.1 COMEDI

COMEDI (COmponent Modeling EDItor), the user interface of MAD facilitates the integration of expert know-how into device models. That is, in COMEDI, expert knowledge concerning intended device behavior and know-how referring to ignorable physical effects can guide the modeling process as summarized in the following.

- Using COMEDI's model builder, one can create component models based on MAD's internal standard components and qualitative values described in Section 2.2 and 2.3. Due to space limitations, we do not elaborate on the model builder.

- In COMEDI, predefined component models can be taken from a library. Unlike some other qualitative methods, for each library component, COMEDI users can choose from alternative behavior models that represent different physical phenomena. For instance, the library contains two battery models, one ignoring the internal resistance of the battery whereas the other model explicitly represents the internal resistance.

To facilitate adequate selection of library models, component behavior is described in colloquial language that should be similar to engineer's thinking about component behavior. For instance, a library model of a battery is called "*idealized-battery*". The behavior is described as *"Battery modeled as idealized voltage source, no internal resistance."* Note that, due to the informal character of these behavior models, they cannot be utilized for automated behavior prediction. Internally, COMEDI models are represented by formalized standard components described in the following section. These components, in principle, allow automated behavior prediction.

In COMEDI, each behavior model represents a single ok behavior mode and it may show one or even more corresponding fault modes. Exemplarily, a behavior model of a battery is shown in Figure 2. It consists of two behavior modes, *ok: idealized-battery* and *fault: battery-voltage-low*.

COMEDI users perform the following steps to model a certain operation mode of a device. First, in order to determine the model structure, COMEDI users assemble icons representing components. Second, for each component, an adequate behavior model is selected. Third, for each behavior model determined in the previous step, a behavior mode (correct or faulty) is selected. A simplified COMEDI model of a forklift frontlight and backlight circuit is shown in Figure 2.

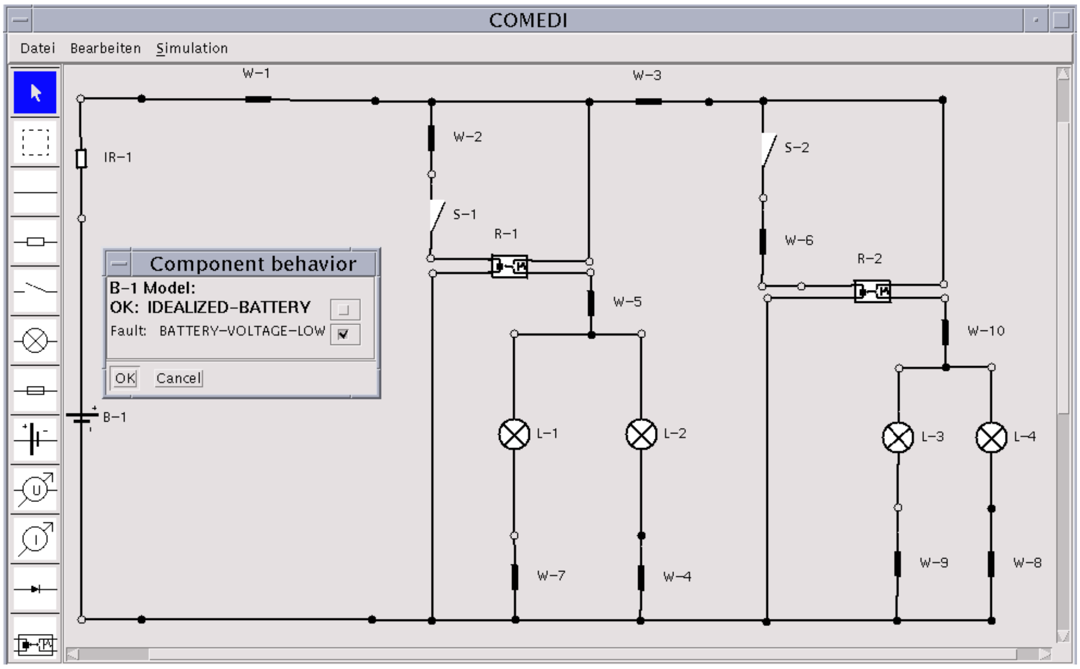

Figure 2: Forklift frontlight and backlight circuit in COMEDI and behavior modes of a battery behavior model

For modeling devices, in COMEDI, component models can be easily combined because of their local internal behavior descriptions (no-function-in-structure principle, [de Kleer and Brown, 84]) presented in the following subsections.

2.2 Standard components

Internally, COMEDI models are mapped to formalized standard components showing well defined and idealized behavior. MAD provides four different standard components, i.e. idealized voltage sources, consumers, conductors and barriers. The behavior of idealized voltage sources is well-known from electrical engineering. Consumers are passive and their current/voltage characteristic is monotonous, i.e. they show positive resistances. Idealized conductors do not allow any voltage drop. Thus, they do not show any resistance at all. Idealized barriers do not allow any current, that is, their resistance is infinite. Standard components can be connected in combinations of series, parallel, star and delta groupings. This simple internal representation of electrical circuits is sufficient for the following reasons.

- A small number of qualitative standard components suffices, because, often, different physical components show similar electrical behavior, i.e. their current/voltage characteristics differ only slightly. Qualitative versions of these current/voltage characteristics are frequently identical.

- MAD's standard components are deliberately selected so that important behavior classes of the application domain can be represented adequately.

- An explicit representation of temporal dependencies is not necessary because MAD focuses on steady state behavior analysis.

Due to analogies between electrics, mechanics and hydraulics, the internal MAD representation is, in principle, also adequate for other technical domains.

2.3 Qualitative representation of physical parameters and variables

In MAD, for each physical parameter type actual values and deviations from reference values are explicitly represented because, as stated above, reasoning about these values is essential. Additionally, reference values are also explicitly represented because, if qualitative values are considered, reference values are not redundant. Moreover, in [Milde et al., 99] we demonstrate that MAD's threefold parameter representation is essential for accurate behavior predictions. In the following, MAD's qualitative parameter representation is described in detail.

For each parameter type, MAD's qualitative representation consists of three attributes, i.e. actual value, reference value and deviation value. For each of these attributes, MAD provides a specific set of qualitative interval-based values. Table 1, 2, and 3 show attributes and corresponding qualitative value sets of resistances, currents, and voltages (abbreviations in brackets). The semantics of the qualitative values should be obvious.

attributes	qualitative values
actual value (act)	zero (0), positive (pos), positive-infinite (pos-inf)
reference value (ref)	zero (0), positive (pos), positive-infinite (pos-inf)
deviation value (dev)	negative-infinite (neg-inf), negative (neg), zero (0), positive (pos), positive-infinite (pos-inf)

Table 1: Qualitative representation of resistances

Note that in MAD's internal models of electrical devices, infinite current values may occur because MAD provides idealized voltage sources and idealized conductors (zero resistances) as standard components.

attributes	qualitative values
actual value (act)	negative-infinite, negative, zero, positive, positive-infinite
reference value (ref)	negative-infinite, negative, zero, positive, positive-infinite
deviation value (dev)	negative-infinite, negative, zero, positive, positive-infinite

Table 2: Qualitative representation of currents

MAD's set of standard components does not include idealized current sources. Thus, in MAD's internal device models, voltages show certain limits and voltage values beyond these limits can be considered as *impossible* values.

attributes	qualitative values
actual value (act)	negative-impossible, negative-maximum, negative-between, zero, positive-between, positive-maximum, positive-impossible
reference value (ref)	negative-impossible, negative-maximum, negative-between, zero, positive-between, positive-maximum, positive-impossible
deviation value (dev)	negative, zero, positive

Table 3: Qualitative representation of voltages

Due to the MAD's explicit representation of voltage limits, in principle, dealing with logical circuits is possible. For instance, logical values (*low, high)* can be mapped to MAD's voltage values *zero* and *positive-maximum*. Furthermore, MAD's qualitative voltage representation allows to handle electrical devices showing more than only one source. In particular, the representation of impossible voltage values paves the way to define a qualitative version of the superposition principle well-known from electrical engineering. Dealing with logical values as well as handling multiple sources is the basis for dealing with hybrid systems consisting of both analog and digital subsystems.

3 Automated behavior prediction

In order to compute qualitative values, local propagation methods have been investigated [Struss, 90]. Since detailed studies proved that local propagation in electrical networks is not successful, we follow a different approach first presented by [Mauss and Neumann, 96]. That is, networks are transformed into trees which explicitly represent the network structures. Exploiting these structure trees, qualitative device behavior can in fact be computed by local propagation. Unlike other qualitative methods such as the FLAMES system, the qualitative SPS method, and the Connectivity method, MAD offers certain features to improve the accuracy of qualitative behavior predictions. In the following, these features are summarized. A more detailed presentation of MAD's automated behavior prediction can be found in [Milde et al., 99].

3.1 Complex qualitative operators

Rather than relying on qualitative versions of basic arithmetics, MAD computes qualitative attribute values by a set of qualitative operators which are qualitative versions of quantitative equations. In the following, we outline how MAD's qualitative operators are defined by exemplarily regarding a parallel grouping of two resistances. We demonstrate that MAD's utilization of qualitative versions of equations is fundamental for accurate device behavior prediction.

The compensation resistance of a parallel grouping of two resistances *R1* and *R2* can be computed by applying the equation *Rp = (R1 * R2) / (R1 + R2)*. In MAD, qualitative values of *Rp* are computed by applying the qualitative operator QRp_act/ref, which is a qualitative versions of *Rp = (R1 * R2) / (R1 + R2)*. This operator computes the actual (reference) values of the compensation resistance *Rp* from actual (reference) values of *R1* and *R2*. QRp_act/ref is defined by applying the corresponding quantitative equation to the intervals represented by the qualitative actual (reference) values of *R1* and *R2*. That is, for the definition of QRp_act/ref, sign arithmetic is performed and certain limits are calculated. Table 4 presents the definition of QRp_act/ref.

R1_act I R2_act R1_ref I R2_ref	0	pos	pos-inf
0	0	0	0
pos	0	pos	pos
pos-inf	0	pos	pos-inf

Table 4: QRp_act/ref, computation of actual values and reference values of compensation resistance *Rp*

Note that, qualitative actual and reference values of *Rp* cannot be derived by applying qualitative basic arithmetics. In particular, evaluation of *Rp = (R1 * R2) / (R1 + R2)* by stepwise applying qualitative basic arithmetics is impossible because qualitative multiplication is undefined if *R1* and *R2* show the qualitative actual value *zero* and *positive-infinite*, respectively (see shaded cells in Table 4). Therefore, MAD's definitions of qualitative operators provide a way for accurate computation of qualitative values. In the following, we briefly summarize some of MAD's features and properties which are described more detailed in [Milde et al., 99].

3.2 Further features and properties of MAD's calculus

In order to avoid spurious predictions, attribute values of currents and voltages are computed twice. For example, considering a parallel grouping of two resistances *R1* and *R2*, the current *I1* through resistance *R1* is computed by applying the well-known current divider rule. Additionally, *I1* is calculated by applying Ohm's law.

Qualitative deviation values are computed from actual and reference values. Additionally, output deviation values are inferred from input deviation values, assuming that parameter dependencies are monotonous. It can be shown that MAD's exploitation of properties of monotony avoids spurious deviation predictions.

MAD's qualitative calculus is based on about 100 qualitative operators represented by a set of tables comprising more than 30.000 entries. These tables had to be generated by computer in order to secure reliability. A limited number of operators suffices because MAD's internal representation of electrical circuits offers a limited number of standard components and elementary network structures.

If certain assumptions such as the single fault assumption hold, MAD's qualitative calculus based on local propagation is sound and complete. In [Milde et al., 99], these assumptions are enumerated and the proof of soundness and completeness is sketched.

If circuit structures show nested star and delta groupings, MAD's computation of qualitative values described so far may be unsound. In order to overcome this deficiency, in addition to local propagation of qualitative values, MAD globally analyses network structures and structure trees to eliminate spurious predictions.

Some electrical components show internal dependencies. That is, their behavior depends on certain current or voltage values. For instance, a relay switch is closed only if there is current through the corresponding relay coil. MAD's dealing with these components is similar to the FLAME system. That is, behavior models show model conditions which are tested after behavior prediction. If model conditions are violated, alternative behavior models are instantiated and behavior prediction is restarted.

3.3 Generation of fault-symptom tables

In order to generate decision trees, behavior predictions are performed for all operation modes of the device for which diagnosis support is required. That is, for all combinations of different ok behavior, faults, and fault combinations, all symptoms (measurements, observations, error codes, display values) are computed which are in principle available for diagnosis. The output of the prediction step is model-based diagnosis knowledge in form of an extensive table of fault-symptom associations. This table is the basis for decision tree generation.

4 Decision tree generation

In order to facilitate the integration of diagnosis expert know-how as well as existing diagnosis data into automatic decision tree generation, MAD offers three different possibilities to generate decision trees. First, based on fault-symptom tables, decision trees can be created automatically. Second, decision trees from archives can be reused. Third, in order to permit manual adaptation and modification of decision trees, MAD offers basic editing operations, such as moving a certain fault from one fault set to another and recomputing the corresponding tests. In the following, automated decision tree generation is presented in more detail. One can choose from the following criteria to guide decision tree generation.

- **Minimization of average diagnosis cost.** Automated decision tree generation uses the well-known A*-algorithm [Hart et al., 68] to select the tests minimizing the average diagnosis cost according to a cost model (see Figure 1).

- **Grouping by observations, error codes, display values.** Decision trees are generated such that subsets of faults correspond to a prespecified symptom. For instance, all faults are grouped together which cause the frontlights not to shine correctly.

- **Grouping by aggregate structure.** If the aggregate structure of the device is known, decision trees can be generated such that subsets of faults correspond to the same physical component. For instance, faults occurring on a certain board may be grouped together.

Figure 3 shows a decision tree for the forklift frontlight and backlight circuit. This decision tree was generated automatically, guided by the criterion *minimization of average diagnosis cost*. Our model-based prediction and automated decision tree generation guarantee, that decision trees are correct and complete with respect to the underlying device model. All faults considered in the device model occur in the generated decision tree, and tests are selected correctly to discriminate fault sets. This holds even if decision trees are modified manually. Furthermore, average fault identification cost is minimal within the constraints imposed by a prespecified decision tree structure.

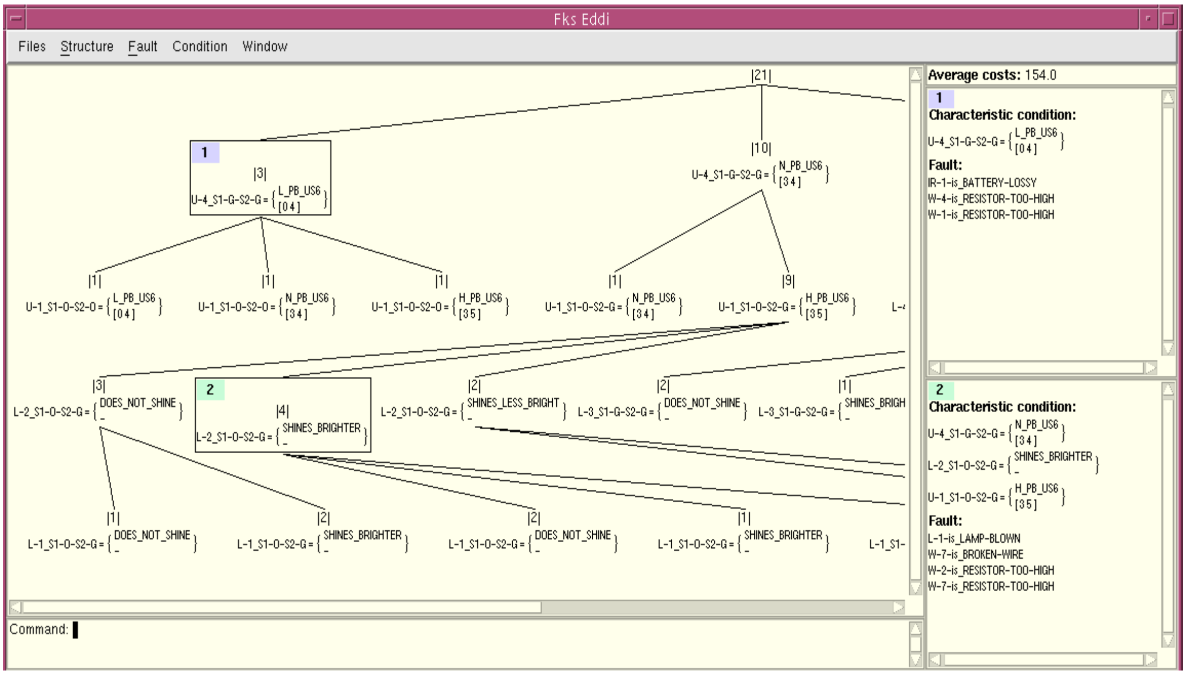

Figure 3: Decision tree for forklift frontlight and backlight circuit

5 Conclusions

Investigating our application domain, we figured out that the following challenges arise when innovative diagnosis techniques are applied to industrial real world applications. First, innovative diagnosis concepts have to be incorporated into existing diagnosis equipment. Replacing existing diagnosis systems is not acceptable. Second, expert design knowledge and diagnosis know-how as well as existing product data and diagnosis strategies have to be integrated into the process of diagnosis system generation. Developing MAD, we paid massive tribute to these challenges. In particular, MAD's decision trees are integrated into existing STILL diagnosis systems. As another point, libraries and archives allow extensive reuse of existing resources such as design data, device models and fault trees. Furthermore, MAD provides multiple possibilities for design engineers and diagnosis experts to guide automated decision tree generation.

In order to apply model-based decision tree generation to industrial scenarios, adequate modeling and accurate behavior prediction is essential. Thus, we developed a new qualitative modeling approach that allows precise behavior predictions for the following reasons. First of all, since qualitative parameter representations describe actual values and deviations from reference values, faults and symptoms can adequately be characterized. Furthermore, MAD's internal standard components represent important behavior types of the electrical domain whereby components can be modeled accurately. As another point, exploitation of network structures and certain features to avoid spurious solutions (see Section 3) assure accurate behavior predictions.

In cooperation with the STILL GmbH Hamburg, we have evaluated the MAD system in the application scenario and found that using the modeling techniques of MAD with some extensions regarding the component model builder which allows modeling complex components such as electronic control units, more than 90% of the faults of the current hand-crafted diagnosis system can be handled successfully. The prototypical implementation allows model-based behavior prediction and automatic generation as well as manual modification of decision trees. Furthermore, we successfully integrated these decision trees into existing STILL diagnosis systems.

There is a great industrial need for computer-based decision tree generation because diagnosis equipment based on hand-crafted decision trees is wide-spread in practice. Since the MAD system grounds decision tree generation on a model, a systematic way for diagnosis system generation is provided and the following benefits arise. First, cost of diagnosis system generation, modification, and maintenance is reduced. Second, quality management is facilitated. Third, average decision-tree-based fault identification cost is reduced. Thus, the MAD system is a generic approach to bridge the gap between (some) basic AI research concepts and industrial applications. In particular, our new approach towards qualitative reasoning about faults in electrical circuits has reached a level of achievement so that it can be utilized to generate diagnosis systems employed in industry.

References

[Friedrich and Nejdl, 89] Friedrich, G., Nejdl, W.: Generating Efficient Diagnostic Decision Tree Procedures for Model-Based Reasoning Systems, in: Proc. DX-89, International Workshop on Model-based Systems, Paris, France, 1989.

[Hart et al., 68] Hart, P. E., Nilsson, N. J., Raphael, B.: A formal basis for the heuristic determination of minimum cost paths, in: IEEE Transactions on Systems Science and Cybernetics, SSC-4(2): 100 - 107, 1968.

[de Kleer and Brown, 84] de Kleer, J., Brown, J. S.: A Qualitative Physics Based on Confluences, in: AI Journal, 1984.

[Malik and Struss 96] Malik, A., Struss, P.: Diagnosis of Dynamic Systems Does Not Necessarily Require Simulation. In: Workshop Notes of the 10th International Workshop on Qualitative Reasoning QR-96, AAAI Press, pp. 127-136, 1996.

[Mauss, 98] Mauss, J.: Analyse kompositionaler Modelle durch Serien-Parallel-Stern Aggregation, DISKI 183, Dissertationen zur Künstlichen Intelligenz, 1998.

[Mauss and Neumann, 1996] Mauss, J., Neumann, B.: Qualitative Reasoning about Electrical Circuits using Series-Parallel-Star Trees, in: Proc. QR’96, 10th International Workshop on Qualitative Reasoning about Physical Systems, 1996.

[Milde et al., 99] Milde, H., Hotz, L., Kahl, J., Wessel, S.: Qualitative Analysis of Electrical Circuits for Computer-based Diagnostic Decision Tree Generation, in: Proc. QR’99, 13th International Workshop on Qualitative Reasoning about Physical Systems, and in: Proc. DX-99, 10th International Workshop on Principles of Diagnosis, 1999.

[Pugh and Snooke, 96] Pugh, d., Snooke, N.: Dynamic Analysis of Qualitative Circuits, in: Proc. of Annual Reliability and Maintainability Symposium, 37 - 42, IEEE Press, 1996.

[Struss, 90] Struss, P.: Problems of Interval-based Qualitative Reasoning, in: Readings in Qualitative Reasoning about Physical Systems, Weld, D., de Kleer, J. (Eds.), Morgan Kaufmann, 1990.

[Struss et al., 1995] Struss, P. Malik, A., Sachenbacher, M.: Qualitative Modeling is the Key, in: Proc. DX-95, 6th International Workshop on Principles of Diagnosis, 1995.