Qualitative Analysis of Electrical Circuits for Computer-based Diagnostic Decision Tree Generation

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

Qualitative reasoning about fault effects in electrical circuits has reached a level of achievement which allows it to be used for real world industrial applications. Dealing with electrical circuits of forklifts, our work focuses on automated diagnostic decision tree generation grounded on model-based predictions of correct and faulty device behavior. In this paper, we discuss requirements for electrical circuit analysis arising from this task. According to these requirements, we developed a new method for Qualitative electrical Network Analysis (QNA) which is the main subject of this paper. QNA's qualitative calculus allows reasoning about actual parameter values and deviations from reference values. The calculus is specifically designed to avoid spurious solutions. To facilitate adequate device modeling, QNA supports that electrical design experts guide the modeling process, bringing in their knowledge concerning intended device behavior and negligible physical phenomena. We successfully evaluated QNA in our application domain.

1 Introduction

Qualitative reasoning about fault effects in electrical circuits has reached a level of achievement which allows it to be used for real world industrial applications. For instance, the FLAME system (Pugh and Snooke 1996) performs failure mode and effects analysis (FMEA) as well as sneak circuit analysis in the automotive domain. The connectivity method (Struss et al. 1995) is employed for automated FMEA and diagnosis guidelines generation for mechatronic car subsystems. The qualitative SPS method (Mauss 1998) is the basis for diagnostic decision tree generation which is also our task. Focusing on forklifts made by the german company STILL GmbH Hamburg, we developed the MAD (Modeling, Analyzing, Diagnosing) system that generates diagnostic decision trees from model-based predictions of correct and faulty device behavior. The basic concepts of MAD are described in (Guckenbiehl et al. 1999). The main subject of this paper is MAD's Qualitative electrical Network Analysis (QNA).

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, 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 numbers 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 basis for automated decision tree generation if heterogeneous electrical systems are investigated.
2. **Steady state behavior prediction suffices.** Frequently, if 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 consist of a large number of components. To assure accurate device modeling, expert knowledge concerning intended device behavior as well as know-how related to negligible physical effects should guide the modeling process. This reflects the insight that the design of technical systems and of appropriate innovative diagnosis systems is inseparable.
4. **Dealing with slight parameter deviations and changes in circuit structures is essential.** In heterogeneous electrical circuits, frequently, different operation modes result in different circuit structures. Faults may slightly modify component behavior or may even change device structures. Hence, different symptoms, such as slight deviations of parameter values and 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.

Although these requirements arise from our specific task in the forklift application scenario, they seem to be relevant for a variety of different diagnostic tasks such as FMEA for instance.

In principle, the FLAME system, the qualitative SPS method, and the Connectivity method are promising but they do not fulfill all of the requirements enumerated above. In particular, these approaches cannot deal with slight parameter deviations because their qualitative parameter representations are too simple. That is, in principle, qualitative values describe parameters in terms such as *positive, zero,* and *negative* and only **actual values** are represented. Parameter **deviations** from reference values are not explicitly described. Furthermore, some of these methods produce spurious behavior predictions.

Since reasoning about parameter deviations is fundamental, in (Milde et al. 1997) we preliminary introduced the SDSP method in order to demonstrate that, in principle, qualitative reasoning about **deviations** in electrical circuits is possible. In this paper, we present QNA, a new method for qualitative electrical circuit analysis. QNA allows reasoning about **actual values and deviations** from reference values. The underlying qualitative calculus is specifically designed to improve the accuracy of behavior predictions. Using QNA, device modeling is guided by expert knowledge concerning intended device behavior and negligible physical phenomena. Section 2 briefly describes modeling electrical devices in QNA. In Section 3, model-based computation of behavior predictions is described. We have successfully evaluated QNA in the application domain what is sketched in Section 4.

2 Device Modeling

2.1 COMEDI

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

- Using COMEDI's model builder, one can create component models based on QNA'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.

A simplified COMEDI model of a forklift frontlight and backlight circuit is shown in Figure 1. To facilitate adequate selection of library models, component behavior is described in a colloquial language that should be similar to the 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 component models described in the following section. These component models allow automated behavior prediction.

In COMEDI, each behavior model represents a single ok behavior mode, and it may include one or more fault modes. Exemplarily, a behavior model of a battery is shown in Figure 1. It consists of two behavior modes, *ok: idealized-battery* and *fault: battery-voltage-low*.

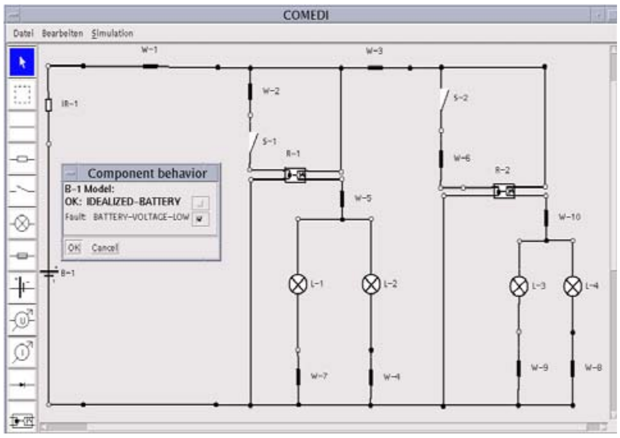

Figure 1: Forklift frontlight and backlight circuit in COMEDI and behavior modes of a battery behavior model

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.

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 1984)) presented in the following subsections.

2.2 Standard Components

Internally, COMEDI models are mapped to formalized standard component models showing well-defined and idealized behavior. QNA 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.
- QNA'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 we focus on steady state behavior analysis.

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

2.3 Qualitative Representation of Physical Variables and Parameters

Due to the standard components described in Section 2.2, in QNA, only three different parameter types have to be represented, i.e. current, voltage, and resistance. For each parameter type, actual values and deviations are explicitly represented because, as stated in the introduction, reasoning about these values is essential.

In (Malik and Struss 1996), it is demonstrated that, in principle, reasoning about actual values and deviations is possible without considering reference values. Thus, at first sight, representing reference values does not seem to be necessary. Furthermore, if quantitative parameter values were used, reference values would be redundant (*reference value = actual value - deviation*).

Nevertheless, in QNA, reference values are explicitly represented because, if qualitative values are considered, reference values are not redundant. This can be demonstrated by exemplarily considering a certain parameter, assuming that its actual, reference, and deviation value are all known as *positive*. Representing actual and deviation values only, the corresponding reference value can be computed as *positive - positive = (negative* or *zero* or *positive).* Hence, if qualitative values are considered, explicit representation of knowledge concerning actual values, deviations, and reference values is more precise than representing actual values and deviations only. Moreover, in Section 3, we demonstrate that QNA's threefold parameter representation is essential for accurate behavior predictions. In the following, QNA's qualitative parameter representation is described in detail.

For each parameter type, QNA's qualitative representation consists of three attributes, i.e. actual value, reference value and deviation value. For each of these attributes, QNA 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.

Unlike in (Malik and Struss 1996), QNA represents infinite deviations. This is reasonable because QNA also represents infinite actual and reference values and $x = infinite - a \Rightarrow x = infinite \quad \forall a \in \Re$ holds (Struss 1990).

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 QNA's internal models of electrical devices, infinite current values may occur because QNA 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

QNA's set of standard components does not include idealized current sources. Thus, in QNA's internal device models, voltages show certain limits and voltage values beyond these limits can be considered as *impossible* values (see Table 3). Due to QNA's explicit representation of voltage limits, in principle, dealing with logical circuits is possible. For instance, logical values (*low, high)* can be mapped to QNA's voltage values *zero* and *positive-maximum*. Furthermore, QNA'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.

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

Table 3: Qualitative representation of voltages

3 Automated Behavior Prediction

In this section, QNA's computation of qualitative values of parameter attributes is described. In order to compute qualitative values, local propagation methods have been investigated (Struss 1990). Since detailed studies proved that local propagation in electrical networks is not successful, we follow (Mauss and Neumann 1996). That is, networks are transformed into trees which explicitly represent the network structure. Exploiting these structure trees, qualitative device behavior can in fact be computed by local propagation.

QNA's structure trees explicitly represent four different elementary network topologies, i.e. series, parallel, star and delta groupings. Structure trees show two different types of nodes, i.e. resistance nodes and equation nodes. Resistance nodes represent (compensation) resistances and the corresponding currents and voltages. Equation nodes represent behavior of elementary network topologies. That is, these nodes represent equations that hold between parameters of adjacent resistance nodes (see Figure 2). These equations, in principle, allow local propagation. The following simple example outlines the concepts of network transformation and local propagation in structure trees.

Figure 2 shows a subnetwork consisting of two resistances *R1* and *R2*. The network is transformed into a structure tree which explicitly represents that *R1* and *R2* are grouped in parallel. That is, both resistances, their compensation resistance *Rp*, and the corresponding currents and voltages are represented in resistance nodes. In the equation node, there are physical dependencies that hold in this particular parallel grouping. Due to space limitations, in Figure 2, we do not present all parallel grouping equations utilized by QNA.

The physical dependencies represented in the equation node in Figure 2 allow local propagation in the structure tree. First, the compensation resistance *Rp* is determined by bottom-up propagation. That is, given *R1* and *R2*, *Rp* can be computed by applying the equation $Rp = (R1 * R2) / (R1 + R2)$. Second, current and voltage values are determined by top-down propagation. For instance, given *Ip, I1* can be computed by applying the well-known current divider rule $I1 = R2 / (R1 + R2) * Ip$. At first sight, it might be surprising that, in Figure 2, *I1* can also be computed by applying $I1 = Up / R1$. In Section 3.3, we outline QNA's alternative computations of qualitative attribute values.

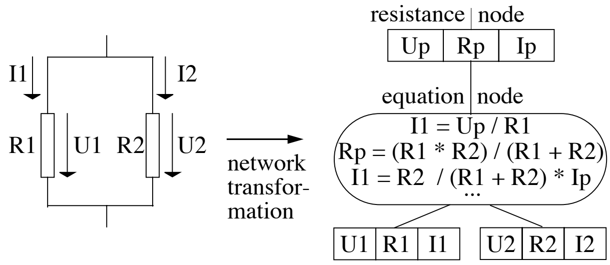

Figure 2: Parallel grouping network and corresponding structure tree

Unlike other approaches such as the FLAMES system, the qualitative SPS method, and the Connectivity method, QNA offers certain features to improve the accuracy of qualitative behavior predictions. In the following, these features are summarized. In Section 3.4, we enumerate preconditions which secure that QNA's qualitative calculus is sound and complete. Additionally, the proof of these properties is sketched. Note that definitions of soundness and completeness are taken from (Struss 1990).

3.1 Complex Qualitative Operators

Rather than relying on qualitative versions of basic arithmetic operators (+, -, *, /), QNA computes qualitative attribute values by a set of qualitative operators which are qualitative versions of quantitative equations represented in equation nodes of structure trees. QNA's qualitative calculus is based on about 100 qualitative operators represented by a set of tables. A limited number of operators suffices because QNA's internal representation of electrical circuits offers a limited number of standard components and elementary network structures. In the following, we outline how QNA's qualitative operators are defined by exemplarily considering the parallel grouping example (see Figure 2). We demonstrate that QNA's utilization of qualitative versions of equations is fundamental for accurate device behavior prediction.

As noted above, 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 QNA, the qualitative attribute values of compensation resistance *Rp* are calculated by applying four qualitative operators, i.e. QRp_act, QRp_ref, QRp_dev1, and QRp_dev2. In this section, we elaborate on QRp_act and QRp_ref. QRp_dev1 and QRp_dev2 are described in Section 3.2.

QRp_act is a qualitative versions of $Rp = (R1 * R2) / (R1 + R2)$. This operator computes the **actual values** of the compensation resistance *Rp* from actual values of *R1* and *R2*. QRp_act is defined by applying the corresponding quantitative equation to the intervals represented by the qualitative actual values of *R1* and *R2*. That is, for the definition of QRp_act, interval arithmetics is performed and certain limits are calculated. Table 4 presents the definition of QRp_act.

QRp_ref is also a qualitative versions of *Rp = (R1 * R2) / (R1 + R2)*. This operator computes the **reference values** of the compensation resistance *Rp* from reference values of *R1* and *R2*. Since actual values and reference values of resistances are described by the same set of qualitative values (see Table 1), Table 4 also presents the definition of QRp_ref.

R1_act(ref) \ R2_act(ref)	0	pos	pos-inf
0	0	0	0
pos	0	pos	pos
pos-inf	0	pos	pos-inf

Table 4: QRp_act and QRp_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 arithmetic operators 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, QNA's definitions of qualitative operators are fundamental for accurate computation of qualitative values.

3.2 Exploitation of Threefold Parameter Representation

In this subsection, we introduce the qualitative operators QRp_dev1 and QRp_dev2 that allow computation of *Rp*'s deviation values. We demonstrate that QNA's threefold parameter representation is the basis for the definitions of these operators which are both fundamental to assure accurate computation of *Rp*'s qualitative deviation values.

QRp_dev1 is a qualitative version of the equation *deviation = actual value - reference value* which holds for each parameter type. QRp_dev1 computes *Rp*'s qualitative **deviation values** from *Rp*'s qualitative actual and reference values. Thus, unlike deviation computation in (Malik and Struss 1996), QNA's computation of deviations is inseparable from computation of qualitative reference values. The definition of QRp_dev1 is presented in Table 5. If actual and reference value are both *pos-inf* the equation *deviation = actual value - reference value* cannot be applied because *infinite - infinite* is undefined. In this case the deviation value *zero* is reasonable because, any other value indicates that actual behavior is different from reference behavior. Considering *0 < pos < pos-inf*, Table 5 should be obvious ("/" means logical "or").

QP_dev2's definition is based on the assumption that none of the resistances *R1* and *R2* is *zero*. In this case, *Rp* is a monotonously increasing function of *R1* and *R2* because *dRp/dR1 > 0* and *dRp/dR2 > 0* hold. That is, an increasing (decreasing) value of *R1* or *R2* leads to an increasing (decreasing) value of *Rp*.

Rp_act \ Rp_ref	0	pos	pos-inf
0	0	neg	neg-inf
pos	pos	neg / 0 / pos	neg-inf
pos-inf	pos-inf	pos-inf	0

Table 5: QRp_dev1, computation of deviation values of compensation resistance *Rp*

QRp_dev2 qualitatively represents that *Rp* is a monotonously increasing function of *R1* and *R2*. Applying QRp_dev2, qualitative deviations of *Rp* are computed from deviations of *R1* and *R2*. The definition of QRp_dev2 is presented in Table 6. Note that Table 6 cannot be derived by stepwise evaluation of *Rp = (R1 * R2) / (R1 + R2)*. Again, QNA's exploitation of complete equations is the basis for accurate behavior prediction.

R1_dev \ R2_dev	neg-inf	neg	0	pos	pos-inf
neg-inf	neg-inf	neg	neg	neg / 0 / pos	neg / 0 / pos
neg	neg	neg	neg	neg / 0 / pos	neg / 0 / pos
0	neg	neg	0	pos	pos
pos	neg / 0 / pos	neg / 0 / pos	pos	pos	pos
pos-inf	neg / 0 / pos	neg / 0 / pos	pos	pos	pos-inf

Table 6: QRp_dev2, computation of deviation values of compensation resistance *Rp*

As described above, QRp_dev2 can only be applied if *0 < R1* and *0 < R2* hold. This applicability condition has to be fulfilled by actual and reference values. Hence, the applicability condition implies that in QNA, computation of qualitative deviations is inseparable from computation of actual and reference values. As another point, due to QRp_dev2's applicability condition, QRp_dev1 is fundamental for the **completeness** of QNA's computation of *Rp*'s deviation values. The following simple example demonstrates that QRp_dev2 is essential to secure **soundness** of QNA's computation of *Rp*'s deviation values.

QNA's computation of qualitative attribute values of the compensation resistance *Rp* of a parallel grouping of two resistances *R1* and *R2* is investigated (see Figure 3). *R1* and *R2* are represented by *R1_[act_pos, ref_pos, dev_0]* and *R2_[act_pos, ref_pos, dev_pos]*. First, applying QRp_act and QRp_ref (see Table 4), *Rp*'s qualitative actual and reference values are computed as *act_pos* and *ref_pos*. Second, applying QRp_dev1 (see Table 5), qualitative deviation values of *Rp* are inferred as *dev_(neg / 0 / pos)*. Third, utilization of QRp_dev2 (see Table 6) leads to *dev_pos*. Note that the applicability condition of

QRp_dev2 is obviously fulfilled.

In this example, applying QRp_dev1 in fact generates spurious solutions that can be avoided by utilization of QRp_dev2. If *Rp*'s actual and deviation values were computed such as in (Malik and Struss 1996), in this example, unsound results would be obtained.

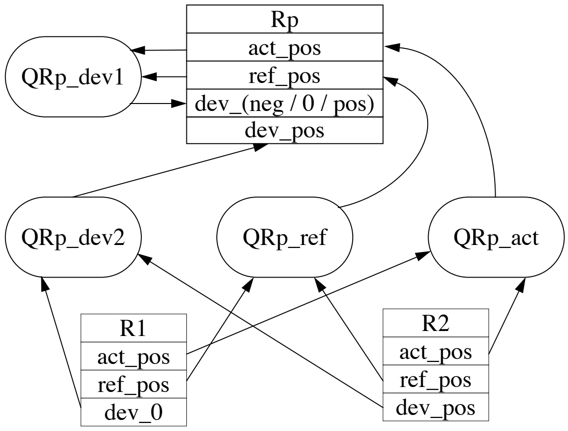

Figure 3: Computation of qualitative attribute values of parallel compensation resistance *Rp*

In this subsection, we have demonstrated that QRp_dev1 and QRp_dev2 are both necessary to assure soundness and completeness of QNA's computation of *Rp*'s qualitative deviation values. Hence, QNA's threefold qualitative parameter representation is essential for accurate behavior predictions.

3.3 Twofold Computation of Current and Voltage Values

In this subsection, the computation of qualitative attribute values of currents and voltages is considered. In QNA, each current or voltage represented in the structure tree is computed twice. First, it is derived from the current of the corresponding father resistance node. Second, it is computed from the voltage of the father node.

For example, considering the parallel grouping shown in Figure 2, the current *I1* through resistance *R1* is computed from *Ip*, the current of the father node by applying the well-known current divider rule $I1 = R2 / (R1 + R2) * Ip$. Additionally, *I1* is calculated from *Up*, the voltage of the father node by applying $I1 = Up / R1$.

If **quantitative** parameter values were considered, the alternative computations of *I1* would produce the same result because they are based on a redundant set of physical dependencies between parameters of adjacent resistance nodes. Considering **qualitative** values, it can be demonstrated that QNA's twofold computation of current and voltage values is essential to avoid spurious behavior predictions. Due to lack of space, we cannot elaborate on this topic.

3.4 Properties of the Qualitative Calculus

The qualitative calculus described so far is sound and complete if the network shows the following characteristics. First, there is only one source in the circuit. Second, the single fault assumption holds. Third, the network can be structured in combinations of series and parallel groupings of standard components. Fourth, components do not show internal dependencies, i.e. their behavior does not depend on certain current or voltage values. If these conditions are fulfilled, the proof of soundness and completeness can be sketched as follows.

Structure trees consist of combinations of subtrees such as exemplarily shown in Figure 2. These subtrees consist of two child resistance nodes and one father resistance node. Equation nodes are not significant for the explanations in this subsection. The father node is either the root node of the structure tree or it represents a series or parallel grouping such as in Figure 2. Note that propagation at the root node is different from propagation described in this paper. Thus, there are three different subtree types to be investigated, i.e. root node, series grouping, and parallel grouping.

It can be shown that for all subtree types and all **possible** combinations of qualitative attribute values of the source and of resistances, QNA's qualitative calculus computes exactly **one** qualitative value for attributes of currents, voltages and compensation resistances. That is, there are no disjunctions of qualitative values computed at all. This suffices to prove soundness and completeness because, in (Struss 1990) it is shown that local propagation methods grounded on interval-based qualitative values and interval arithmetics are complete but may be unsound. Since these methods are complete, unsound results may only occur if disjunctions of qualitative values are computed.

Note that the conditions enumerated at the beginning of this subsection, limit the number of possible attribute values. For instance, the single fault assumption secures that Table 6 is never evaluated with *R1_[dev_pos]* and *R2_[dev_neg]*. Due to some extensions of the calculus not summarized in this paper, correct results are computed even when multiple sources or multiple faults occur. This secures correct results even if components show internal dependencies. Anyhow, if nested star and delta groupings of standard components occur, the calculus is unsound. Hence, in QNA, in addition to the local qualitative calculus, qualitative attribute values are globally computed.

3.5 Global Computation of Qualitative Attribute Values

In addition to local propagation of qualitative values, QNA globally analyses network structures and structure trees in order to eliminate spurious predictions. In particular, a global analysis of the network structure allows to determine current directions. Knowledge about current directions can be used to eliminate some spurious predictions concerning actual and reference values of currents.

Subnetworks that behave like passive electrical double-poles can easily be identified by a global analysis of the structure tree. Due to Kirchhoff's laws, voltage drops across components and groupings of components inside a certain passive double-pole network are equal or of lower amount than the voltage drop across the two poles of the double-pole. This inequality can be exploited to eliminate some spurious predictions concerning actual and reference values of voltages. However, a detailed analysis of the soundness of QNA's calculation of qualitative attribute values still has to be performed.

3.6 Dealing with Complex Component Behavior

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. QNA's dealing with these components is similar to the FLAME system. In a first step, for each of these components, one of its alternative behavior models is instantiated. Second, qualitative attribute values of voltages and currents are computed. Third, internal model conditions are verified. If an internal model condition is violated, one component behavior model is changed and computation of qualitative attribute values is restarted. If all internal model conditions are fulfilled, the steady state behavior prediction is successful. Steady state behavior prediction fails if all possible combinations of alternative behavior models lead to violated model conditions.

4 Conclusions

To assure industrial applicability of diagnostic decision trees which are automatically generated from model-based behavior predictions we found a number of requirements for modeling electrical devices. In particular, investigating our application, we figured out that reasoning about actual values and deviations from reference values is essential to describe faults and symptoms adequately. Spurious behavior predictions have to be avoided to secure that faults can be distinguished in decision trees. Furthermore, to assure adequate device modeling, the modeling process should be guided by detailed expert knowledge concerning faulty device behavior and ignorable physical effects.

Developing QNA, we paid massive tribute to these requirements. In particular, we developed a new qualitative modeling approach that allows reasoning about actual values and deviations from reference values. Certain features, such as definitions of complex qualitative operators, explicit representation of reference values, and exploitation of a redundant set of physical dependencies allow accurate behavior predictions. Additionally, to facilitate the integration of expert know-how into the modeling process, QNA users can choose from alternative component library models that represent different physical phenomena. Furthermore, one can create component models using QNA's component model builder.

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 QNA with some extensions regarding the component model builder (see Section 2.1) which allows dealing with electronic control units, more than 90% of the faults of the current hand-crafted diagnosis system can be handled successfully. Furthermore, we integrated computer-generated diagnostic decision trees into existing STILL diagnosis systems.

Acknowledgments

This research has been supported by the Bundesministerium für Bildung, Wissenschaft, Forschung und Technologie (BMBF) under the grant 01 IN 509 D 0, INDIA - Intelligente Diagnose in der Anwendung.

References

(de Kleer and Brown 1984) de Kleer, J., Brown, J. S.: A Qualitative Physics Based on Confluences, in: AI Journal, 1984.

(Guckenbiehl et al. 1999) Guckenbiehl, T., Milde, H., Neumann, B., Struss, P.: Meeting Re-use Requirements of Real-Life Diagnosis Applications, in: Proc. XPS-99: Knowledge-Based Systems, Springer, LNAI 1570, pp. 90-100, 1999.

(Malik and Struss 1996) 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 1998) 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. 1997) Milde, H., Hotz, L., Möller, R., Neumann, B.: Resitive Networks Revisited: Exploitation of Network Structures and Qualitative Reasoning about Deviations is the Key, in: Proc. DX-97, 8th International Workshop on Principles of Diagnosis, 1997.

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

(Struss 1990) 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.