

# Simulierte Pannen

**Neue rechnergestützte Verfahren zur qualitativen Analyse des Fehlerverhaltens von technischem Gerät bilden die Basis für automatisch erstellte Diagnosesysteme.**

Heiko Milde und Bernd Neumann  
Labor für Künstliche Intelligenz  
FB Informatik, Universität, Hamburg

## Diagnoseforschung in der Informatik

Verfahren zur Fehlersuche in komplexem technischen Gerät sind nicht nur Untersuchungsgegenstand von klassischen Ingenieurdisziplinen wie Maschinenbau und Elektrotechnik sondern auch - seit den frühen 80er Jahren - Thema der Informatik, dort insbesondere der Künstlichen Intelligenz (KI). Gemeinsamer Ausgangspunkt ist das Problem, von Messungen an einem technischen System auf physikalische Eigenschaften der Systemkomponenten zurückzuschliessen, insbesondere von Fehlverhalten auf mögliche Fehlerursachen. Bei zunehmender Komplexität technischer Systeme spielen dabei systematische, wissenschaftlich fundierte Diagnoseverfahren eine immer wichtigere Rolle und ersetzen oder ergänzen die Rolle menschlicher Diagnoseexperten.

Die Herausforderung, menschliche Kompetenz maschinell nachzubilden, brachte die KI ins Spiel. Wie funktionieren die Schlussfolgerungsvorgänge eines menschlichen Experten, und wie kann man sie auf einem Rechner implementieren? Als eine erste Antwort entstand die Expertensystemtechnologie. Erfahrungswissen menschlicher Experten wird dabei in Regelform kodiert und von einer "Inferenzmaschine" zur Lösung konkreter Probleme systematisch ausgewertet. Neben zahlreichen erfolgreichen Anwendungen dieser Technologie gab es allerdings auch Enttäuschungen: Regelbasierte Expertensysteme verfügten meist nur über phänomenologisches Wissen, mit dem Messungen und Beobachtungen assoziativ mit Fehlerursachen in Verbindung gebracht wurden. Fehlten einschlägige Erfahrungen, so konnte das Expertensystem nicht helfen - trotz zuweilen einfacher technisch-physikalischer Zusammenhänge, wie etwa in der Kfz-Elektrik der damaligen Zeit.

Als eine vielversprechende Alternative zur regelbasierten Diagnose entstand daraufhin die "modellbasierte Diagnose", zunächst als konzeptueller Ansatz der KI, später auch von den Ingenieurwissenschaften aufgegriffen. Hierbei stützt sich die Diagnoseleistung auf ein computerinternes Modell des technischen Gerätes, mit dem das Verhalten und Zusammenspiel von Komponenten für Diagnosezwecke nachgebildet wird. Dadurch wird es prinzipiell möglich, auch potentielle Fehlersituationen zu diagnostizieren, über die noch keine praktischen Erfahrungen vorliegen. Dabei ist bemerkenswert, dass Komponenten auch als fehlerhaft erkannt werden können, wenn man die genaue Form ihres Fehlverhaltens gar nicht kennt. Modellbasierte Diagnose verspricht interessante wirtschaftlichen Vorteile:

- Diagnosesystemen können bereits in der Entwicklungsphase unabhängig von Betriebserfahrungen erstellt werden.
- Aufgrund der austauschbaren Komponentenmodelle sind Diagnosesysteme in höherem Masse wiederverwendbar.

Bei den Untersuchungen zur modellbasierten Diagnose rückten KI-Forscher zwei Aspekte in den Mittelpunkt:

- Wie kann man qualitative Angaben, sei es bei einer Messung oder Beobachtung ("die Lampe brennt weniger hell als normal"), sei es bei einer Ursachenbeschreibung ("die Versorgungsspannung ist zu niedrig"), systematisch verarbeiten?
- Wie kann man das Gesamtverhalten eines Systems aus dem (qualitativen) Verhalten seiner Komponenten berechnen?

Die mit diesen Fragen angesprochene Forschungsrichtung wird "Qualitatives Schliessen" genannt und beschäftigt heute zahlreiche Forschungsgruppen in aller Welt. Ein charakteristisches Problem beim qualitativen Schliessen ist die Zunahme von Unsicherheit bei der Verknüpfung qualitativer Werte. Bildet man beispielsweise die Differenz zweier Grössen, für die jeweils der qualitative Wert "positiv" bekannt ist, so ist das Ergebnis entweder "positiv", "negativ" oder "Null", also keineswegs qualitativ eindeutig. Dieser Effekt kann bei qualitativer Diagnose dazu führen, dass Fehlerursachen berechnet werden, die physikalisch gar nicht möglich sind - ein durchaus unerwünschtes Resultat. In diesem Beitrag wird u.a. über neue Verfahren berichtet, mit denen die Propagation von Unsicherheit in Grenzen gehalten werden kann.

Im Vergleich zu rein quantitativen Methoden bietet qualitatives Schliessen völlig neue Möglichkeiten zur Entwicklung von Diagnosesystemen, also von rechnerbasierten Verfahren für die Fehlerfindung an technischem Gerät. Aufgrund der qualitativen Beschreibung reduziert sich die Zahl möglicher Fehlersituationen und der damit einhergehenden Messungen und Beobachtungen in der Regel auf eine handhabbare Menge von Fehlerfällen, die im Voraus berechnet und in ein Diagnosesystem eingebracht werden können. Damit bietet sich die Möglichkeit, den sehr aufwendigen und fehleranfälligen Prozess der manuellen Erstellung von Diagnosesystemen weitgehend zu automatisieren.

Alternativ können die für eine Diagnose erforderlichen qualitativen Schlüsse auch "online", also zur Diagnosezeit, berechnet werden. Auch hierfür sind seitens der KI innovative Verfahren entwickelt worden, die aber bisher wenig Eingang in die industrielle Praxis gefunden haben.

Dieser Beitrag berichtet über Ergebnisse eines Projektes, in dessen Rahmen die automatisierte Erstellung von Diagnosesystemen im Verbund von Forschern der Universität Hamburg, Software-Entwicklern eines jungen Hamburger Unternehmens und Partnern eines grossen in Hamburg ansässigen Industriebetriebs entscheidende Schritte vorangekommen ist. Das Projekt hatte den Namen INDIA ("Intelligente Diagnose für industrielle Anwendungen") und wurde vom Bundesministerium für Bildung und Wissenschaft im Rahmen seines Förderprogramms "Intelligente Systeme" in den Jahren 95 - 99 unterstützt. Ausser den drei Hamburger Partnern waren zwei weitere Projektteams aus Süddeutschland beteiligt. Hauptziel war die Praxiseinführung innovativer Diagnoseverfahren durch Überwindung der noch bestehenden methodischen Defizite.

Bemerkenswert war der bereits in der Projektstruktur angelegte Technologietransfer. Die Forscher der Universität analysierten die Anforderungen des Anwendungspartners und identifizierten die damit verbundenen wissenschaftliche Knackpunkte. Ihre Forschungsergebnisse wurden in Gestalt von wissenschaftlichen Methoden und prototypischen Implementierungen an das verbündete Software-Unternehmen zur Integration in ein kommerzielles Diagnosesystem

weitergegeben. Das Software-Unternehmen übernahm dann die Erprobung, Einführung und Wartung des Diagnosesystems beim Anwender.

In diesem Beitrag wird also über Forschungsergebnisse berichtet, zu denen nicht nur das INDIA-Team des Labors für Künstliche Intelligenz an der Universität Hamburg sondern letzten Endes auch kritische Projektpartner aus der Wirtschaft beigetragen haben. Für die stets konstruktive und vertrauensvolle Zusammenarbeit sei an dieser Stelle ausdrücklich gedankt.

### **Qualitative Analyse von Fehlerverhalten**

Wie gehen Menschen vor, wenn sie in komplexem technischen Gerät Fehler suchen? Zur Veranschaulichung wollen wir ein Beispiel aus dem Autofahreralltag betrachten.

Ich steige in meinen Wagen, drehe den Zündschlüssel um und schalte die Scheinwerfer ein - vor mir kein Licht, sondern Dunkelheit. Was ist passiert? Möglicherweise ist die Autobatterie zu schwach, oder der Lichtschalter ist kaputt, oder der Marder hat wieder mal die Kabel durchgebissen. Vielleicht sind auch ganz einfach die Glühlampen durchgebrannt - vier technische Fehler zur Auswahl, aber welcher ist es? Wie wär's mit Türaufmachen? Wenn bei geöffneter Tür der Innenraum beleuchtet ist, dann ist zumindest klar: Die Batterie ist okay. Gesagt, getan, Resultat: Batterie ist okay. Was mache ich jetzt? Aussteigen, ums Auto gehen und prüfen, ob die Rücklichter leuchten. Dies klärt zumindest, ob der Lichtschalter kaputt ist.

Dieses Beispiel demonstriert typisches menschliches Vorgehen, wie es nicht nur bei der Diagnose technischer Geräte, sondern auch häufig in der Medizin praktiziert wird. Passend zum beobachteten Symptom - hier das defekte Fahrlicht - wird zunächst die Menge möglicher Fehler ermittelt, und durch gezieltes Testen - erst Überprüfen der Innenbeleuchtung, dann Rücklichttest - werden Fehler nach und nach ausgeschlossen, bis schliesslich nur ein möglicher Kandidat übrig bleibt. Es wird also eine verzweigte Abfolge von Tests abgearbeitet, in der das Ergebnis eines Tests bestimmt, welcher Test als nächstes durchgeführt wird. Eine solche Testsequenz kann als Fehlersuchanleitung bezeichnet werden. Abbildung 1 zeigt den baumförmigen Aufbau einer Fehlersuchanleitung.

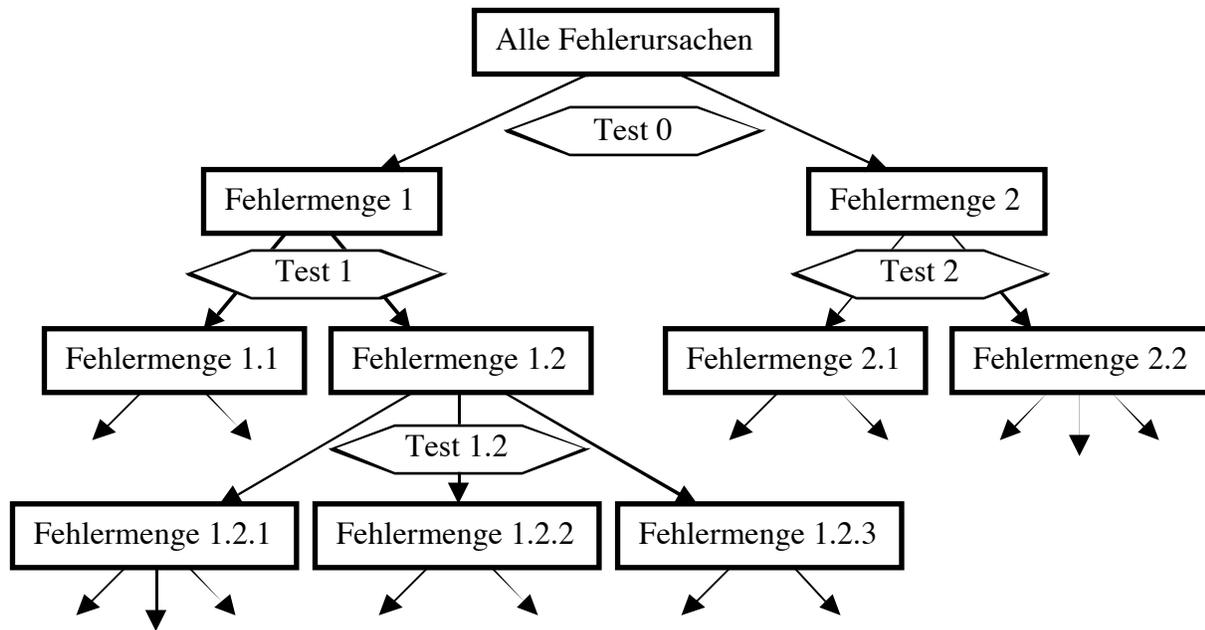


Abbildung 1: Grundsätzlicher Aufbau einer Fehlersuchanleitung zum schrittweisen Einschränken möglicher Fehlerursachen

Um wie beschrieben vorgehen zu können, müssen die Auswirkungen von Fehlern bekannt sein, z.B. dass eine defekte Autobatterie sowohl das Fahrlicht als auch die Innenbeleuchtung lahmlegt, während ein defekter Fahrlichtschalter zwar das Anschalten des Fahrlichts unmöglich macht aber keinen Einfluss auf die Innenbeleuchtung hat.

Aufgrund seiner grossen Erfahrung und seines tiefen Verständnisses der technischen Zusammenhänge im betrachteten Gerät kennt der geübte Techniker häufig derartige Fehler-Symptom-Assoziationen und kann daher auch in komplexen technischen Geräten zielsicher Fehler lokalisieren. Wie aber kann sichergestellt werden, dass ein Techniker tatsächlich für *alle* denkbaren Fehler *alle* beobachtbaren Symptome *korrekt* im Kopf hat? Wie kann garantiert werden, dass für die Fehlersuche möglichst *billige* Testsequenzen durchgeführt werden? In wie weit sind bekannte Fehler-Symptom-Assoziationen auf ein Nachfolgegerät anwendbar? Wie kann verhindert werden, dass das Diagnosewissen eines Technikers nach dessen Ausscheiden aus dem Betrieb verloren geht? Diese offenen Fragen motivieren das computerbasierte Erstellen von Fehlersuchanleitungen. Basierend auf einem Modell des technischen Gerätes werden zunächst automatisch die Auswirkungen von Fehlern bestimmt. Dann werden aus diesen Fehler-Symptom-Assoziationen Fehlersuchanweisungen erzeugt.

Am Labor für Künstliche Intelligenz (LKI) der Universität Hamburg wurde im Rahmen des INDIA-Projektes die Diagnose-Software MAD (Modellierung, Analyse, Diagnose) entwickelt, die modellgestützt Fehlersuchanleitungen für elektrische Schaltungen erzeugt. Wesentliches Diagnose-Know-How ist daher nicht länger personengebunden und liegt nicht länger in Form von Fehler-Symptom-Assoziationen bzw. Fehlersuchanleitungen vor, sondern ist in Schaltungsmodellen repräsentiert. Innovative Algorithmen ermitteln Fehler-Symptom-

Beziehungen, die vollständig und korrekt bezüglich des zugrundeliegenden Modells sind. Zudem kann die Optimalität der erzeugten Fehlersuchanleitungen bezüglich bestimmter Kostenkriterien garantiert werden.

MAD verfügt über eine CAD-ähnliche Oberfläche, mit deren Hilfe Diagnosemodelle elektrischer Schaltungen leicht eingegeben werden können. Dies wird durch umfangreiche Modellbibliotheken unterstützt, in denen die Modelle von Komponenten, Baugruppen oder bereits modellierten Systemen zur Verfügung stehen. Das Anpassen eines Schaltungsmodells an eine weiterentwickelte Version ist deswegen meist nur mit wenig Aufwand verbunden.

Eine Schaltung wird wie folgt modelliert:

1. Zunächst wird die Schaltungsstruktur eingegeben, indem - ähnlich wie bei CAD - Komponenten (z.B. Batterien, elektrische Leitungen, Schalter und Glühlampen) am Bildschirm zusammengefügt werden. Die für die Diagnose zur Verfügung stehenden Messungen werden durch symbolische Messgeräte dargestellt.
2. Der zum Durchführen einer Messung notwendige Aufwand kann in Form einer Zahl eingegeben werden, die als Kosteneinheit zu verstehen ist. Durch die Angabe von Messkosten ist es später möglich, automatisch kostenoptimierte Fehlersuchanleitungen zu erzeugen, bei denen aufwendige Messungen soweit wie möglich hinausgezögert werden.
3. Für die Diagnose kann das Gerät in verschiedene Betriebszustände versetzt werden. Ein Betriebszustand wird im wesentlichen dadurch modelliert, dass die damit verbundenen Schalter-, Pedal- und Reglerstellungen sowie eingeprägte Strom- und Spannungswerte angegeben werden. Kosten für Betriebszustandseinstellungen können ebenfalls benannt werden. Sie werden später beim Erzeugen von kostenoptimierten Fehlersuchanweisungen berücksichtigt.
4. Alle Fehler, die von der zu erzeugenden Fehlersuchanleitung gefunden werden sollen, werden explizit modelliert. Hierfür gibt man für die einzelnen Komponenten an, welches fehlerhafte Verhalten berücksichtigt werden soll. So kann z.B. für eine Glühlampe neben ihrem korrekten Funktionieren auch "Glühlampe durchgebrannt" berücksichtigt werden. Elektrische Leitungen können z.B. gebrochen und Batterien zu schwach oder leer sein. Das gleichzeitige Auftreten verschiedener Komponentenfehler kann ebenfalls modelliert werden.

Wenn die Modellierung der zu diagnostizierenden Schaltung abgeschlossen ist, kann die Verhaltensvorhersage erfolgen. Für jeden im Modell berücksichtigten Fehler und Mehrfachfehler sowie für alle Betriebszustände werden die sich daraus ergebenden Messungen berechnet. Das Ergebnis ist eine umfassende Tabelle mit Fehler-Symptom-Assoziationen.

MAD ermittelt die Auswirkungen von Fehlern rein qualitativ, es werden also keine Zahlenwerte miteinander verrechnet. Das hierzu verwendete Verfahren ist ein am LKI entwickelter Schlussfolgerungskalkül, der mit ingenieurmässigem Schliessen vergleichbar ist. Wenn z.B. der Lichtschalter defekt ist, so dass er keinen elektrischen Strom zulässt, ermittelt MAD zunächst, dass im gesamten dazugehörigen Stromkreis kein Strom fließt und daher auch in keiner beteiligten Komponente, auch nicht in den Glühlampen. Zudem "weiss" MAD, dass eine Glühlampe nur leuchtet, wenn durch sie ein ausreichender Strom fließt. Auf diese Weise entsteht

die Schlussfolgerung: Ein defekter Fahrlichtschalter führt dazu, dass das Fahrlicht nicht leuchtet. Dieses Ergebnis kommt zustande, ohne dass hierfür Zahlenwerte miteinander verrechnet werden.

Qualitatives Schliessen über das Verhalten elektrischer Schaltungen ist keinesfalls trivial, da Standardmethoden der Künstlichen Intelligenz wie z.B. das systematische Propagieren von Beschränkungen (Constraint Propagation) im allgemeinen nicht erfolgreich angewendet werden können. MAD löst dieses Problem, indem das Schaltungsmodell in eine Baumstruktur aus Ersatzschaltungen transformiert wird und so die in der Schaltungsstruktur implizit enthaltene Informationen für die Analyse verfügbar gemacht werden. Einzelheiten von MADs qualitativem Schlussfolgerungsmechanismus werden in Abbildung 2 beschrieben.

Bei der Entwicklung einer qualitativen Schaltungsanalyse-Software besteht eine besondere wissenschaftliche Herausforderung darin, nicht nur das blosses Funktionieren bzw. Nichtfunktionieren einer Komponente vorhersagen zu können, sondern auch leichte Parameterabweichungen korrekt zu beschreiben. Wenn z.B. die Batterie etwas zu schwach ist, sollte die Software ermitteln können, dass das Fahrlicht weniger hell ist als normal. MAD ermöglicht den systematischen Umgang mit derartigen fehlerbedingten leichten Verhaltensänderungen, indem Auswirkungen von Abweichungen relativ zum Normalverhalten beschrieben und berechnet werden können.

Eine zweite Herausforderung stellen integrierte Bausteine mit nichtlinearem Verhalten dar, die häufig mithilfe von Input-Output-Tabellen beschrieben werden. Schaltungen, bei denen einfache Bauelemente mit integrierten Bausteinen kombiniert werden, spielen in der industriellen Praxis bekanntermassen eine immer wichtigere Rolle. MAD unterstützt die Modellierung derartiger Schaltungen durch die Möglichkeit, Komponentenverhalten tabellarisch zu definieren. Zur Fehleranalyse wurde ein neues Verfahren entwickelt, das die Analyse linearer Netze mit eingebetteten tabellarisch beschriebenen Komponenten ermöglicht.

----- (Beginn separater Kasten)

**Modellierung des Stromlaufplans einer Anlasserschaltung.** Der vereinfachte Stromlaufplan einer Anlasserschaltung besteht aus einer Batterie, dem eigentlichen Anlasser, elektrischen Leitungen, dem Zündschlossschalter und einem Relais. Wenn der Zündschlossschalter geschlossen ist, fliesst Strom durch die Relaispule, wodurch ein Magnetfeld entsteht, das den Relaischalter schliesst, weshalb der Anlasser aktiviert wird. MAD stellt eine Oberfläche zur Verfügung, mit der die einzelnen Komponenten am Bildschirm zu einem Schaltungsmodell zusammengefügt werden können, was automatisch eine MAD-interne Modellierung des Schaltungsverhaltens nach sich zieht.

**Interne Modellierung.** MADs interne Schaltungsmodelle bestehen aus Netzwerkmodellen ergänzt durch Tabellenmodelle. Ein Netzwerk ist aus passiven und aktiven Elementarkomponentenmodellen mit fest vorgegebenen Verhaltensrelationen aufgebaut. Tabellenmodelle repräsentieren weitere Parameterabhängigkeiten. Im Beispiel ist dies die Relaisstabelle, die die Abhängigkeit des Relaischalters ( $R_s$ ) vom Spulenstrom ( $I_{sp}$ ) darstellt. Elektrische Parameter werden qualitativ beschrieben, z.B.  $I_{sp}$ =positiv (pos). Um Fehler und Symptome adequat darstellen zu können, wird sowohl das aktuelle als auch das

Referenzverhalten der Schaltung sowie die Abweichung zwischen beiden modelliert. Dieses Vorgehen mag auf den ersten Blick redundant erscheinen, ist es jedoch nicht, was durch die qualitative Parameterrepräsentation begründet ist. Um die qualitativen Werte von Strömen und Spannungen zu berechnen, wird zunächst das Netzwerk unabhängig von der Relais-tabelle betrachtet.

**Netzwerktransformation.** Das Netzwerk wird in einen Baum transformiert, der die Elementartopologien (Serien-, Parallel-, Stern- und Dreieckschaltungen), aus denen das Netzwerk aufgebaut ist, explizit repräsentiert. Der Knoten S2 z.B. zeigt an, dass die passiven Komponenten A und R<sub>s</sub> eine Serienschaltung bilden. Anhand des Baumes werden Relationen instantiiert, die für die Elementartopologien gelten, z.B. eine Spannungsteilerrelation für die Serienschaltung. Die Netzwerktransformation bietet verschiedene Vorteile: 1. Lokales Propagieren von qualitativen Werten ist einfach und effizient möglich, weil keine zyklischen Parameterabhängigkeiten vorliegen. 2. Da nur eine begrenzte Anzahl von verschiedenen Elementartopologietypen auftritt, können dazugehörige Relationen "manuell" ausgewählt, definiert und optimiert werden, so dass physikalisch nicht gerechtfertigte Ergebnisse vermieden werden.

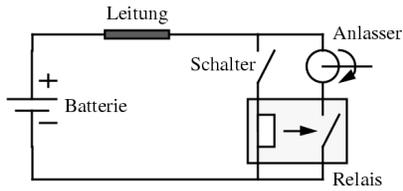
**Lokale Propagation qualitativer Werte.** Das lokale Propagieren von qualitativen Parameterwerten in MAD profitiert massgeblich von der relationalen Beschreibung von Komponenten und Elementartopologien:

1. Lokales Wertepropagieren ist mittels des Join-Operators der Relationenalgebra sehr effizient implementierbar, denn der Join-Operator wird z.B. durch PROLOG oder SQL direkt unterstützt.
2. Als Ergebnis des lokalen Propagierens qualitativer Werte durch Relationen-Joins entsteht eine Netzwerkrelation, deren Tupel mögliche Kombinationen qualitativer Parameterwerte beschreiben. Abhängigkeiten zwischen qualitativen Werten verschiedener Parameter werden daher "automatisch" ermittelt.

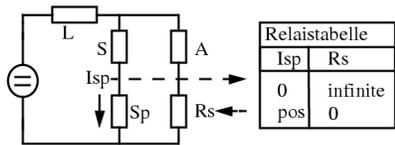
**Relationales Verschmelzen von Netzwerk und Tabelle.** Da auch Tabellenmodelle relational repräsentiert sind, können Netzwerk und Tabelle einfach unter Verwendung des Join-Operators verschmolzen werden. Als Ergebnis entsteht eine Schaltungsrelation, deren Tupel mögliche Kombinationen qualitativer Parameterwerte im Schaltungsmodell beschreiben.

(Text möglichst neben Abbildung 2 setzen)

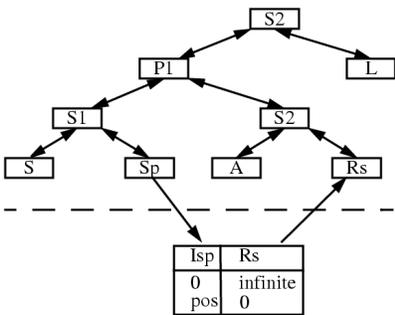
## Modellierung des Stromlaufplans einer Anlasserschaltung



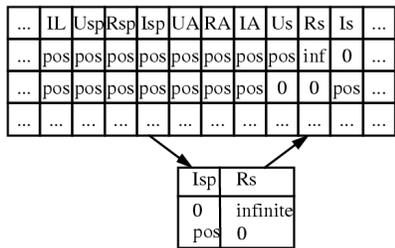
interne Modellierung



Netzwerktransformation



lokale Propagation qualitativer Werte



relationales Verschmelzen von Netzwerk und Tabelle

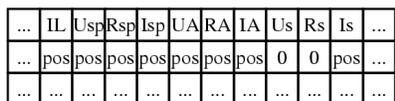


Abbildung 2:  
Verarbeitungsschritte zur  
Erstellung einer qualitativen  
Verhaltenstabelle

----- (Ende separater Kasten)

### **Automatisches Erzeugen von Fehlersuchanleitungen**

Eine automatisch erzeugte Fehler-Symptom-Tabelle enthält die Ausgangsdaten zur Erzeugung einer Fehlersuchanleitung. Wie oben beschrieben handelt es sich bei einer Fehlersuchanleitung um eine verzweigte Testsequenz, bei der der Ausgang eines Tests bestimmt, welcher Test als nächstes erfolgen soll. Das Durchführen der einzelnen Tests schränkt die Menge aller möglichen Fehler schrittweise ein, bis nur noch ein Kandidat übrig bleibt. Die möglichen Messungen und damit die auszuwertenden Symptome sind im allgemeinen redundant angelegt, so dass verschiedene Testsequenzen zu einer eindeutigen Fehlerbestimmung führen können. Hier bietet sich die Möglichkeit für eine Kostenminimierung. Ziel ist es also, genau die Fehlersuchanleitung zu generieren, deren Abarbeitung mit minimalem durchschnittlichen Aufwand verbunden ist.

Offensichtlich sind die Kosten eines Tests ein wichtiges Auswahlkriterium. Aber auch die Diskriminierungsfähigkeit eines Tests spielt eine bedeutende Rolle. Zur Erläuterung betrachten wir Tests genauer. Ein Test ist eine Messung oder Beobachtung, die in einem speziellen Betriebszustand des zu diagnostizierenden Gerätes durchgeführt wird. Aufgrund der qualitativen Betrachtungsweise haben Messungen und somit auch Tests eine endliche Menge von verschiedenen Resultaten. Ein Test kann z.B. die beiden möglichen Resultate "Spannung okay" und "Spannung nicht okay" aufweisen und ist damit zweiwertig, oder er zeigt z.B. an, dass eine Spannung "zu hoch", "okay" oder "zu niedrig" ist und ist dann dreiwertig. Ein Test erzeugt also aus einer Fehlermenge entsprechend seiner Wertigkeit zwei, drei oder noch mehr Teilmengen, in denen sich jeweils alle Fehler befinden, die zum zugehörigen Testresultat führen.

Tests mit höherer Wertigkeit können offensichtlich zu einer stärkeren Einschränkung der Fehlermöglichkeiten und damit zu einer günstigeren Fehlersuchanleitung führen. Betrachten wir z.B. eine Fehlermenge mit 100 Fehlern. Test 1 möge die Fehlermenge in zwei Teilmengen à 50 Fehler teilen, Test 2 zerlege die gleiche Fehlermenge in zehn Teilmengen mit jeweils 10 Fehlern. Das Durchführen von Test 1 im Rahmen der Fehlersuche reduziert also die Menge der möglichen Fehler von 100 auf 50, Test 2 dagegen lässt nur 10 Fehler übrig.

Als zweites Beispiel betrachten wir wieder eine Menge von 100 möglichen Fehlern. Es mögen wieder zwei Tests zur Verfügung stehen, beide sind diemals zweiwertig und erzeugen daher jeweils zwei Teilmengen. Test 1 spaltet die 100 Fehler in zwei Mengen à 50 Fehler und reduziert deswegen die Menge möglicher Fehler unabhängig vom Resultat von 100 auf 50. Test 2 hingegen erzeugt eine Teilmenge mit 99 Fehlern sowie eine mit nur einem Fehler und führt daher in 99 von 100 Fällen zu fast keiner Verringerung der Fehleranzahl. Eine detaillierte Betrachtung zeigt, dass prinzipiell die Tests zu bevorzugen sind, die Fehlermengen in möglichst gleichmächtige Teilmengen zerlegen.

Zusammenfassend lässt sich sagen: Ein Test ist bei der Fehlersuche genau dann zu favorisieren, wenn er 1. wenig Kosten verursacht, 2. eine hohe Wertigkeit hat, und 3. möglichst gleichmächtige Fehlerteilmengen erzeugt. In der Regel konkurrieren diese drei Kriterien. Dadurch wird das Erstellen einer kostenminimierenden Fehlersuchanleitung zu einem Optimierungsproblem, das schon bei einer geringen Anzahl von Fehlern und Tests ohne Computerunterstützung nicht systematisch gelöst werden kann.

Die Informatik verfügt über ein grosses Repertoire von Verfahren, mit denen Optimierungsprobleme dieser Art behandelt werden können. Ein naheliegender Ansatz könnte darin bestehen, alle möglichen Fehlersuchanleitungen erschöpfend zu generieren und dann die kostengünstigste auszuwählen. Überraschenderweise ist die Anzahl der dabei zu berechnenden Testsequenzen schon bei einer recht kleinen Fehlerzahl und bei nur wenigen zur Verfügung stehenden Tests so gross, dass auch leistungsstärkste Computer dies nicht in akzeptabler Zeit schaffen. Im Falle unseres Anwendungspartners im INDIA-Projekt ging es um die Diskriminierung von mehreren Hundert Fehlermöglichkeiten in Transportfahrzeugen. Hier hätte eine erschöpfende Suche zu Rechenzeiten von mehreren Jahren geführt.

Ein intelligenterer Ansatz besteht darin, das Problem als klassisches Suchproblem zu interpretieren, als Suche nach der kostengünstigsten Fehlersuchanleitung im Raum aller Testsequenzen. Die Informatik und insbesondere die Künstliche Intelligenz haben hierfür Methoden entwickelt, mit denen solche Suchaufgaben handhabbar gemacht werden können - unabhängig vom speziellen Anwendungsproblem. Die Grundidee ist dabei immer, den Suchraum keinesfalls vollständig zu erkunden, sondern schrittweise und erfolgsorientiert zu durchforsten, bis das Ziel erreicht ist. Je zielgerichteter die Suche fortschreitet, desto früher wird das Ziel gefunden, und desto besser ist der Algorithmus. Bezogen auf die Suche nach der idealen Testsequenz bedeutet dies, dass beginnend bei der leeren Testsequenz nur besonders vielversprechende Testfolgen um weitere, zielorientiert ausgewählte Tests ergänzt werden. Dies wird im folgenden näher erläutert.

Für jede Testsequenz kann eine Kostenbewertung ermittelt werden, indem die Testsequenz für jeden Fehler der Fehler-Symptom-Tabelle abgearbeitet wird. Durch die Verzweigungen in der Sequenz werden dabei verschiedene Tests für verschiedene Fehler durchgeführt. Die Kosten eines einzelnen Fehlers sind die Summe der dazugehörigen Testkosten, und die Kosten der Testsequenz sind die Summe aller Einzelfehlerkosten. Der Such-Algorithmus ist nun ganz einfach: Es wird jeweils die günstigste Testsequenz um den günstigsten Test erweitert, der zur Verfügung steht. Es wird also jeweils die günstigste neue Sequenz erzeugt. Dann wird geprüft, ob diese Testsequenz eine Fehlersuchanleitung darstellt. Falls ja, ist man fertig, sonst wird die nächste Sequenz erzeugt. Es wird also nicht gezielt nach einer Fehlersuchanleitung gesucht, sondern es werden immer längere, möglichst günstige Testsequenzen erzeugt. Das skizzierte Suchverfahren verfolgt eine best-first Strategie und basiert auf dem bekannten Suchalgorithmus von Dijkstra. Es garantiert, dass die zuerst gefundene Fehlersuchanleitung optimal ist.

Die am LKI entwickelte Diagnosesoftware MAD erzeugt kostenoptimale Fehlersuchanleitungen ebenfalls nach einer best-first Strategie, allerdings wird der bekannte A\*-Algorithmus verwendet, der eine Erweiterung des Dijkstra-Algorithmus darstellt. Beim A\*-Algorithmus wird eine Testsequenz nicht nur bezüglich ihrer Kosten bewertet, sondern auch bezüglich der noch zu erwartenden Kosten, wenn die Testsequenz zu einer vollständigen Fehlersuchanleitung

komplettiert wird. Offensichtlich muss für die Abschätzung der noch ausstehenden Kosten zusätzliches Wissen z.B. über die Struktur des Suchraums verwendet werden. Der A\*-Algorithmus ist daher in der Regel besser "informiert" als Dijkstra's Algorithmus und findet die kostengünstigste Suchanleitung schneller. Für typische Fehlersuchanleitungen unseres Anwendungspartners wurden Rechenzeiten im Minutenbereich benötigt.

## **Bilanz anwendungsorientierter Forschung**

Das INDIA-Projekt hat überzeugend gezeigt, dass Anwendungsziele und wissenschaftliche Forschung gemeinsam und mit Gewinn für alle Beteiligten in einem Verbundprojekt verfolgt werden können. Dazu haben verschiedene Gründe beigetragen. Zum einen beruht der Erfolg auf realistisch abgesteckten Projektzielen, die den Wissenschaftlern genügend Raum für eine fundierten Bearbeitung lassen, aber auch die Erfordernisse des Anwenders hinsichtlich praktischer Verwendbarkeit und Integrierbarkeit berücksichtigen. Dieser Realismus zeigt sich z.B. konkret darin, dass dem Transfer von Forschungsergebnissen in die Anwendungssysteme hinein genügend Zeit gewährt wird - im INDIA-Projekt zwischen 6 und 12 Monaten. Zudem stellt die Bearbeitung durch ein Software-Unternehmen einen notwendigen Zwischenschritt von Forschung zu Anwendung dar, indem marktorientierte Gesichtspunkte eingebracht werden.

Ein weiterer interessanter Grund für die Verträglichkeit von Forschungs- und Anwendungszielen ist das gemeinsame Interesse an generischen, also möglichst allgemeingültigen Lösungen. Dies ist für die Wissenschaft ziemlich selbstverständlich, für viele IT-Anwender allerdings erst eine in jüngster Zeit gewachsene Erkenntnis. Dahinter steht die Erfahrung, dass Software-Lösungen nur dann wirtschaftlich sind, wenn sie an veränderte Anwendungssituationen angepasst werden können, also ein gewisses Mass von Allgemeingültigkeit besitzen. Die oben beschriebene Methodik zur komponentenbasierten Verhaltensanalyse illustriert diesen Aspekt: Komponentenmodelle bilden die Basis für wissenschaftlich fundierte Verfahren zur qualitativen Verhaltensanalyse, gleichzeitig können sie vom Anwender in hohem Masse wiederverwendet werden, anders etwa als Erfahrungswissen.

Das INDIA-Projekt kann auch als ein Modell für praktizierten Technologietransfer betrachtet werden. Das hier implementierte Zusammenspiel von Forschern, Produktentwicklern und Anwendern zeigt, dass Technologietransfer in der Regel nicht einfach durch "Abholen" von Forschungsergebnissen aus den Universitäten oder durch "Anwenden" von Forschungsergebnissen in der Praxis zustandekommt, sondern einen mehrstufigen Prozess erfordert, in dem Grundlagenforschung, anwendungsorientierte Grundlagenforschung, Werkzeugentwicklung, Anwendungsentwicklung und Anwendungsintegration unterschieden werden können.

Für den Fachbereich Informatik ist das INDIA-Projekt eine von zahlreichen Aktivitäten im Bereich des Technologietransfers. Um diesen Bereich auszubauen, hat der Fachbereich den Verein HITeC (Hamburger Informatik Technologie-Center) eingerichtet, der die Hochschullehrer bei der Vernetzung mit Hamburger Unternehmen und bei der Vermittlung von Kooperationen unterstützt. Mehr Informationen zu HITeC gibt es unter [www.hitec-hh.de](http://www.hitec-hh.de).