

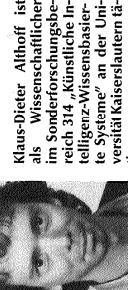
Die Lernkomponente der MOLTKE₃-Werkbank zur Diagnose technischer Systeme¹

Klaus-Dieter Althoff/Frank Maurer/Ralph Traphöner/Stefan Weiss

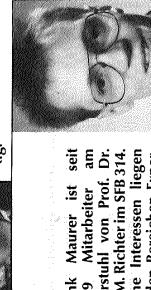
Im Rahmen des MOLTKE-Projektes² wurde am Beispiel einer komplexen, technischen Anwendungsdomäne die MOLTKE-Werkbank zur Diagnose technischer Systeme entwickelt. In ihr sind verschiedene Expertensystemtechniken der zweiten Generation integriert. Gegenstand dieses Beitrages ist die Lernkomponente der Werkbank, die sowohl die Klassifikations- als auch die Testauswahl-fähigkeiten der Diagnose- und Fallbeispiel mit Hilfe von Fallbeispielen verbessern kann. Sie besteht aus zwei Komponenten, die auf derselben Fallrepräsentation aufsetzen.

Die Teilkomponente GenRule³ lernt dabei „offline“ aus Fallbeispielen. Das Ergebnis eines Lernschrittes sind heuristische Regeln für eine MOLTKE-Wissensbasis, mit deren Hilfe das Diagnoseverfahren an das „abkürzungsfaktorierte“ diagnostische Problemlösungen⁴ eines Servicetechnikers angepasst werden kann.

Die zweite Teilkomponente PAT-DEX⁴ erweitert das Diagnoseverfahren um einen ähnlichkeitssasierten Ansatz. Hierzu werden die Fallbeispiele mit Hilfe eines fallbasierten Schlußfolgerungsmechanismus direkt interpretiert.



Klaus-Dieter Althoff ist als Wissenschaftlicher im Sonderforschungsbereich 314 „Künstliche Intelligenz-Wissensbasierter Systeme“ an der Universität Kaiserstautern tätig.



Frank Maurer ist seit 1989 Mitarbeiter am Lehrstuhl von Prof. Dr. M.W. Richter im SFB 314. Seine Interessen liegen in den Bereichen Expertensysteme und Maschinelles Lernen.



Ralph Traphöner studiert seit 1984 Informatik an der Universität Kaiserstautern und beschäftigt sich mit den Gebieten Wissens-Akquisition und Maschinelles Lernen.



Stefan Weiss ist seit Januar 1991 im SFB 314 beschäftigt. Seine Interessen liegen in den Bereichen fallbasiertes Schließen und Planung in technischen Anwendungsdomänen.

1. Motivation und Einführung

En Servicetechniker ist aufgrund seines Erfahrungswissens in der Lage, Diagnosen schneller zu stellen, falsche Diagnosen zu vermeiden, Konsequenzen aus falschen Diagnosen zu ziehen sowie das erfolglose Diagnoseverfahren bereits erlebter Situationen auf das aktuelle Problem (sinngemäß) zu übertragen. Diese Fähigkeiten beruhen dabei auf seinem intuitiven Verständnis für ähnliche Situationen beziehungsweise seinem Vermögen, verkürzte Lösungswege anwenden zu können.

Ein Beispiel aus der Domäne der CNC-Maschinen soll dies illustrieren: Eine Werkzeugmaschine wird in einem Bereich mit sehr hoher Umgebungstemperatur eingesetzt. Nehmen wir weiter an, daß bei dieser Maschine umgebungsbedingt in 90 % der Fälle ein bestimmter Teil der Hydraulik austfällt. Ein erfahrener Servicetechniker, der diese Maschine kennt, wird im allgemeinen schon nach der Feststellung der ersten Symptome entscheiden, ob der übliche

¹ Die hier dargestellte Arbeit wurde teilweise gefördert durch die Deutsche Forschungsgemeinschaft, Sonderforschungsbereiche 314 „Künstliche Intelligenz – Wissensbasierte Systeme“, Projekte X6 und X9. Das Lehrstuhl X6 wurde in Kooperation mit dem Lehrstuhl X6 für Fertigungstechnik und Qualitätssicherung (WZL, Prof. Dr.-Ing. T. Pfeiffer) der RWTH Aachen durchgeführt.

² Models, Learning und Temporal Knowledge in Expert Systems for Technical Diagnosis

³ Generator of Empirical MOLTKE Rules

⁴ Pattern Directed Expert System

- Fehler aufgetreten ist oder nicht. Erkennt er ein Teilmuster der Symptomatik des Hydraulikfehlers, so wird er auf einen Abarbeiten des kompletten Suchweges verzichten.
- Ein weiteres Szenario ist die Behandlung von bisher unbekannten Fehler-situationen: Nach der Installation und den damit verbundenen Testläufen einer CNC-Maschine wurde beim erstmaligen regulären Einsatz ein Fehler festgestellt. Nach einer langwierigen Untersuchung stellte der erneut herbeigerufene Techniker zufällig fest, daß das für den Transport aufgetragene Schmierfett einen Endschalter verklebt hatte. Diese für den Servicetechniker neue Erfahrung versetzte ihn in Zukunft in die Lage, ähnliche Fehlersituationen gezielt zu lösen.
- Die Relevanz der übrigen Symptomwerte für die Diagnose oder den fehlenden Symptomwert ist so stark, daß ein Analogieschluß zwischen der gegebenen Beispiele sollen einen ersten Eindruck hinsichtlich der Anvermittelten. Weitere Anforderungen ergeben sich aus der Integration in die MOLTKE₃-Werkbank, deren grundlegende Terminologie im folgenden Abschnitt beschrieben wird.

2. Die Terminologie der MOLTKE₃-Werkbank

- Wert logisch-1 nicht erhoben. Mit anderen Worten: Er hat es nicht benötigt, um die gleiche Diagnose stellen zu können. Man kann dieses Verhalten auf zweierlei Weise interpretieren:
- Der Servicetechniker betrachtet das Symptom als für die gegebene Situation irrelevant.
- Die Relevanz der übrigen Symptomwerte für die Diagnose oder den fehlenden Symptomwert ist so stark, daß ein Analogieschluß zwischen der gegebenen Beispiele sollen einen ersten Eindruck hinsichtlich der Anvermittelten. Weitere Anforderungen ergeben sich aus der Integration in die MOLTKE₃-Werkbank, deren grundlegende Terminologie im folgenden Abschnitt beschrieben wird.

- Wert logisch-1 nicht erhoben. Mit anderen Worten: Er hat es nicht benötigt, um die gleiche Diagnose stellen zu können. Man kann dieses Verhalten auf zweierlei Weise interpretieren:
- Der Servicetechniker betrachtet das Symptom als für die gegebene Situation irrelevant.
- Die Relevanz der übrigen Symptomwerte für die Diagnose oder den fehlenden Symptomwert ist so stark, daß ein Analogieschluß zwischen der gegebenen Beispiele sollen einen ersten Eindruck hinsichtlich der Anvermittelten. Weitere Anforderungen ergeben sich aus der Integration in die MOLTKE₃-Werkbank, deren grundlegende Terminologie im folgenden Abschnitt beschrieben wird.

Im Verlauf des MOLTKE-Projektes entstanden mehrere prototypische Diagnoseanwendungen aus technischen Bereichen (CNC-Bearbeitungszentren, 3D-Meßmaschinen, Tunnelvortriebsmaschinen und Fehlersuche in heterogenen Computernetzen). Basierend auf den dabei gewonnenen Erfahrungen entwickelten wir ein einfaches Modell für technische Diagnoseanwendungen. Wir gehen davon aus, daß sich Diagnose wie folgt beschreiben läßt:

Diagnose = Klassifikation + Testauswahl.

In beiden Fällen wäre die tragische Symptomaprägung durch die restlichen Symptomwerte „partiell determiniert“. Man kann versuchen den Grad der Determinierung durch Lernen aus weiteren Fällen zu verbessern. Allgemein lernt ein Techniker auf eine doppelte Weise:

• Er „lernt“ von einem bestimmten früheren Fall, in dem er in einer konkreten Situation eine dörige Lösung

| | | |
|------------------------|-------------------|--|
| Maschinenfehlermeldung | i59 | die in dieser Reihenfolge erhobenen Symptome |
| IO-Status IN36 | logisch-1 | |
| IO-Status OUT7 | logisch-1 | |
| Ventil 5Y1 | geschaltet | |
| Ventil 5Y2 | nicht geschaltet | |
| Leitungssystem | in Ordnung | |
| Aufnahmekegel | nicht verschmutzt | |
| IO-Status IN32 | logisch-1 | |
| VO-Karte | defekt | die Diagnose |

Bild 1 - Ein „Fall“ in der MOLTKE-Wissensbasis

| | | |
|------------------------|-------------------|--|
| Maschinenfehlermeldung | i59 | die in dieser Reihenfolge erhobenen Symptome |
| IO-Status OUT7 | logisch-1 | |
| Ventil 5Y1 | geschaltet | |
| Ventil 5Y2 | nicht geschaltet | |
| Leitungssystem | in Ordnung | |
| Aufnahmekegel | nicht verschmutzt | |
| IO-Status IN32 | logisch-1 | |
| VO-Karte | defekt | die Diagnose |

Bild 2 - Ein Fall des Servicetechnikers

⁵ Aus der Wissensbasis extrahierte Fälle heißen Diagnosefälle. 6 Partielle Determiniertheit sind eine Ersatz für bedingte Wahrscheinlichkeiten mangels genauerer Information. Dabei wird der „Grad der Determiniertheit“ durch Determinationskoeffizienten (0,1) ausgedrückt, deren Wertebereich jeweils [0,1] ist.

⁶ Partielle Determiniertheit ist ein Maß für die Wahrscheinlichkeit, daß ein bestimmter Wert einer Information „richtig“ ist. „Ganz“ oder „fast ganz“ ist kein Sinn für Wahrscheinlichkeit.

Eine Diagnose wird durch eine Menge von Symptomwerten charakterisiert. Formal werden Diagnosen über Diagnosempfehlungen beschrieben. Die Verfeinerungshierarchie der Diagnosen wird mit Hilfe eines Graphen, dem Kontextgraphen, modelliert. Kontexte repräsentieren dabei Crob-, Zwischen- oder Enddiagnosen. Einem Kontext werden zweilei. Arten von Regeln zugeordnet: Reihenfolge- und Abkürzungswahlen. Die Bedingungssteile dieser Regeln sind Diagnoseformeln. Als Aktionen werden im Falle einer „feuernden“ Reihenfolge regeln ein Test angestossen und im Falle einer Abkürzungswahl ein Symptom mit einem Wert belegt. Dabei repräsentieren Abkürzungswahlen Zusammenhänge zwischen Symptomen, beispielsweise „Wenn das Licht brennt, dann ist der Schalter geschlossen“. Die Strategien des Experten kann durch die Reihenfolgezeilen dargestellt werden. Diese ordnen einer Situation den als nächstes durchzuführenden Test zu.

4.1. GenRule

Zur Verbesserung einer gegebenen MOLTKE₃-Wissensbasis mit Hilfe empirischer Diagnosefälle orientiert sich GenRule an dem „abkürzungswahlorientierten“ diagnostischen Prozess erfahrener Servicetechniker. Dazu lernt GenRule „Abkürzungen“ aus Analogien zwischen bereits in die Wissensbasis integrierten Diagnosefällen und neu präsentierten Fällen [6], [11]. Diese Abkürzungen werden in die MOLTKE-Shell als partielle, das heißt heuristische, Aktivierungsrègeln implementiert. Durch das Anwenden dieser Regeln kann das Diagnoseverfahren beträchtlich abgekürzt und durch die Fokussierung der Diagnosestrategie auf das Feuern der Abkürzungswahlen darüber hinaus auch transparenter gestaltet werden. Zudem können durch die Komponenten von GenRule Inkonsistenzen in der Wissensbasis aufgedeckt werden.

4. Beschreibung der Lernkomponente

Grundlegend für die MOLTKE₃-Wissensakquisition = Wissensmodellierung + Wissenserhebung.

Zur Modellierung technischer Diagnoseproblematiken wird dabei das in § 2 skizzerte Modell verwendet. Im Sinne von KADS [8] ist somit zum einen ein Designmodell vorhanden, zum anderen existieren Verfahren zur Generierung eines Implementierungsmodells. Im einzelnen sind dies Lernverfahren sowie Verfahren aus dem Bereich des qualitativen Schließens beziehungsweise der Unterstützung manueller Wissenssakquisition.

Die eingesetzten Lernverfahren haben also die Funktion, die Generierung eines Implementierungsmodells für ein Konkretes technisches System, auf der Basis eines Designmodells für die Diagnose technischer Systeme, möglichst weitgehend zu unterstützen. Dies ist durch die in einem Determinationsfaktor akkumulierte statistische Information, überschreitet der Determinationsfaktor einen gegebenen Schwellwert (zum Beispiel 0,9), so wird die Regel als „Abkürzungswiegel“ in die Shell integriert. Unterschiedet er dagegen einen zweiten Schwellwert (zum Beispiel 0,1), so wird die Regel als Redundanzwiegel integriert¹². Letzteres ist damit gleichzusetzen.

• Das GenRule-System lebt von Analogien zwischen neu präsentierten Diagnosefällen und bereits in die Wissensbasis integrierten Diagnosefällen¹⁰. Diese analogen Schlüsse

Gerechtfertigt wird eine derartige Regel durch die in einem Determinationsfaktor akkumulierte statistische Information. Überschreitet der Determinationsfaktor einen gegebenen Schwellwert (zum Beispiel 0,9), so wird die Regel als „Abkürzungswiegel“ in die Shell integriert. Unterschiedet er dagegen einen zweiten Schwellwert (zum Beispiel 0,1), so wird die Regel als Redundanzwiegel integriert¹². Letzteres ist damit gleichzusetzen.

• Die später eingeführten „Redundanzregeln“ sind lediglich eine „Spielart“ der Abkürzungswahlen.

• Sie auch Bild 2

• Ein Diagnosefall ist ein Protokoll eines Diagnoseablaufs, basierend auf der aktiven Wissensbasis des Expertensystems, ein Fall besteht aus einer Situation und der gestellten Diagnose, d.h. Diagnosefälle und „pade“ sind syntaktisch gleich (siehe auch Bild 1).

zen, daß das entsprechende Symptom für die Klassifikation in dem jeweiligen Diagnosefall nicht mehr verwendet wird.

Ist der Diagnosefall „echt länger“ als der Diagnosepad, so wird dies als ein Hinweis auf eine Inkonsistenz in der Wissensbasis interpretiert, falls eine hinreichende statistische Rechtfertigung vorliegt (Schwellwert). Die Konsequenz ist dann die Generierung einer Reihefolge, welche die Erfassung des beziehungsweise der fehlenden Symptome erzwingt.

4.1.2. Organisation der Diagnosefälle

Die Diagnosefälle werden im Fallgedächtnis von GenRule organisiert. Der Aufbau der Speicherstruktur orientiert sich an den Arbeiten von Koldohar [12], stellt aber in seiner konkreten Problemlösung angepaßte Realisierungen einer Neuentwicklung dar. Grundidee ist die Integration neuen Wissens (hier: Erfahrungswissen in Form von Diagnosefällen) in bereits bekanntes und die dafür erforderliche automatische Reorganisation der Struktur. Folgende Teilaufgaben von GenRule werden durch das Gedächtnis besonders unterstützt:

- die Partitionierung der Fälle gemäß ihrer Diagnose,
- das Auffinden von Diagnosefällen, die bestimmte, vorgegebene Situationen enthalten (zur Berechnung von Determinationsfaktoren),
- Auffinden von Diagnosefällen und -pläden, die einander ähnlich sind,
- die Aktualisierung der generierten Abkürzungs-, Redundanz- und Reihenfolgeregeln,
- die Aktualisierung der PATDEX₂-zugehörigen Fälle.

4.2. PATDEX₂

PATDEX₂ versucht zu einer gegebenen aktuellen Situation in seiner Fallbasis einen möglichst ähnlichen Fall zu finden. Ist die Ähnlichkeit des ermittelten Falles zur aktuellen Situation ausreichend, so übernimmt es die Diagnose des gefundenen Falles. Andernfalls wird versucht, die Beschreibung der Situation durch die Erhebung eines weiteren Symptoms so zu verbessem, daß anschließend eine Diagnose möglich ist. Zur Lösung eines vorliegenden Diagnoseproblems stellt PATDEX₂ Hypothesen auf und bietet den Benutzer an, die zu verifizieren beziehungsweise

weise gegebenenfalls zu korrigieren. Dieses Feedback aus der realen Welt verhindert PATDEX, dann zur Verbesserung seines diagnostischen Verfahrens. Die Wiederholung einmal gemachter Fehler ist somit weitgehend ausgeschlossen.

5. Kooperation zwischen PATDEX₃, GenRule und der MOLTKE₃-Diagnoseshell

Zum Aufbau der Klassifikationsfähigkeit eines Diagnosesystems ist wichtig, alle (interessanten) Fehler zu identifizieren und in Form von Kontexten im System zu modellieren. Kontexte werden dabei durch Mengen von Symptomwerten, die die Kontextvorbedingungen der Inklusionsbeziehungen der Vorbedingungen wird, dann der Kontextgraph aufgebaut. Die Testauswahlmöglichkeit wird dadurch realisiert, daß auf der Menge der für einen Kontext interessanten Symptome¹³ eine Ordnung definiert wird. Diese Ordnung kann sowohl explizit durch Reihenfolgeregeln angegeben werden als auch implizit über Abkürzungswahlen. Letztere resultiert dann durch die Konkretisierung einer Metastratégie (zum Beispiel „feuere so viele Abkürzungswahlen wie möglich“) in einer anwendbaren

Situation ist ein entscheidender Punkt der fallbasierten Wissensverarbeitung. So ist die Feststellung „Glühbirne nicht defekt“ für eine mögliche Diagnose „Sicherung defekt“ durchaus relevant. Die Feststellung „Innenraumtemperatur = 25 Grad“ ist in diesem Zusammenhang aber wahrscheinlich absolut irrelevant. Es sind aber Situationen denkbar, in denen die Bewertung gerade umgekehrt erfolgen muß, die durch die damit verbundene Erweiterung der Vorbedingungen, die Erfassung zusätzlicher Symptome, Akquirings- und Redundanzregeln bewirken genauso das Gegen teil. Der Benutzer wird allerdings bei jeder Enddiagnose darüber informiert, daß Symptomwerte unrischer abgeleitet beziehungsweise überflüssig gehalten wurden.

Die Testauswahlfähigkeiten der Shell werden ebenfalls von allen Regeln beeinflußt. Reihenfolge- und Redundanzregeln haben dabei allerdings lediglich das Hinzufügen beziehungsweise Streichen von Tests zur Folge. Akquiringsregeln bewirken darüber hinaus die Orientierung der Shell an dem abkürzungswahlorientierten diagnostischen Problemlosen“ der Servicetechniker durch die Konkretisierung der bereits oben erwähnten Metastratégie. PATDEX₂ erweitert die Klassifikationsfähigkeit der Shell durch das Einbeziehen von Ähnlichkeiten der aktuellen Situation zu bereits bekannten Fällen. Stehen ausreichend Fälle zur Verfügung, so ist das Testauswahlverfahren von PATDEX₂ eine fallbasierte Alternative zu den bislang beschriebenen Strategien.

¹³ Siehe zum Beispiel [7].

¹⁴ Dies ist die einzige Möglichkeit, die in den Vorbereitungen möglicher Nachfolgekontakte innerhalb des Graphen verbleiben.

3. Aufgaben der Lernkomponente

werden in heuristische Regeln kompiliert.

- Das PATDEX₃-System führt fallbasiertes Schließen direkt auf einer Fallbasis durch. Dies ist sowohl zur Behandlung von Aushahnfehlern, als auch zur Wissensakquisitionsunterstützung wichtig (direktes Interpretieren von Regeln, Akquisition weiterer Diagnosefälle).

Das Hauptunterscheidungsmerkmal der beiden Systeme ist, daß GenRule analoges Schließen „offline“ zur Generierung von Regeln einsetzt, während PATDEX₂ es „online“ als eigentlichen Problemlösemechanismus verwendet.

3. Aufgaben der Lernkomponente

Grundlegend für die MOLTKE₃-Wissensakquisition = Wissensmodellierung + Wissenserhebung.

4.1. Generierung von Abkürzungs-, Redundanz- und Reihenfolgeregeln

Grundlegend für GenRule ist der Vergleich von neu präsentierten Diagnosefällen mit „ähnlichen“ Diagnosefällen. Ein Diagnosepfad ist dabei dem Diagnosefall ähnelich, wenn er die gleiche Diagnose hat und zudem alle Symptomwerte des Falles beinhaltet. Ein solcher Diagnosepfad ist dem Diagnosefall umso „ähnlicher“, je „kürzer“ er ist. Ist der Diagnosepfad „echt kürzer“ als der Pfad, so wird dies als Abkürzung interpretiert und eine entsprechende Regel generiert. Die linke Seite einer derartigen Regel wird dabei auf der Grundlage der Situation des vorgelegten Diagnosefalles gebildet. Die rechte Seite ist sozusagen der jeweilige „gelehrte“ Symptomwert des ähnlichsten Diagnosepfades.

Gerechtfertigt wird eine derartige Regel durch die in einem Determinationsfaktor akkumulierte statistische Information. Überschreitet der Determinationsfaktor einen gegebenen Schwellwert (zum Beispiel 0,9), so wird die Regel als „Abkürzungswiegel“ in die Shell integriert. Unterschiedet er dagegen einen zweiten Schwellwert (zum Beispiel 0,1), so wird die Regel als Redundanzwiegel integriert¹². Letzteres ist damit gleichzusetzen.

• Die später eingeführten „Redundanzregeln“ sind lediglich eine „Spielart“ der Abkürzungswahlen.

• Sie auch Bild 2

• Ein Diagnosefall ist ein Protokoll eines Diagnoseablaufs, basierend auf der aktiven Wissensbasis des Expertensystems, ein Fall besteht aus einer Situation und der gestellten Diagnose, d.h. Diagnosefälle und „pade“ sind syntaktisch gleich (siehe auch Bild 1).

¹³ Siehe § 1 zur Motivation und Einführung

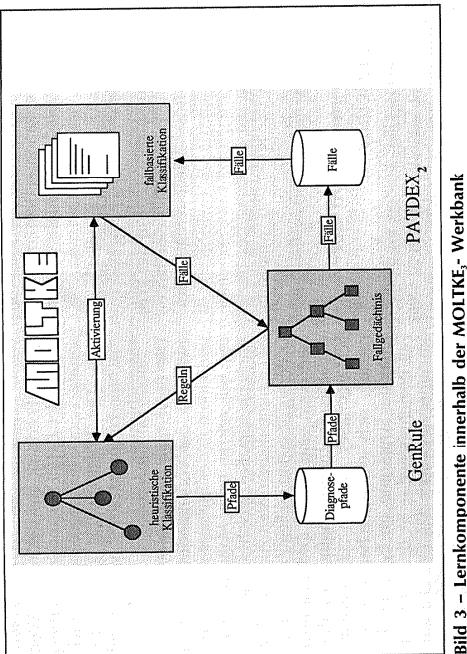
¹⁴ Redundanzregeln sind somit einer speziell interpretierte Klasse von Abkürzungswiegeln.

5.1. Verbesserung der Wissensbasis

Eine Wissensbasis (WB) kann als schwache Hintergrundtheorie angesehen werden, das heißt, sie ist redundant, unvollständig und/oder inkonsistent. Die Redundanz äußert sich dabei durch zu viele Diagnosefade in der WB bezüglich Weise durch Plade, die an sich zu lang sind. Unvollständigkeit der WB dagegen bedeutet, daß Diagnosefade fehlen beziehungsweise zu kurz sind, um aus widersprüchlichen Situationen herausweise Diagnosen ihrer Fälle. Es ist nun die Aufgabe der Lernkomponente, die WB mit Hilfe empirischer Diagnosefälle zu verbessern. Für die angesprochenen drei Grundproblematiken bieten sich folgende Lösungsansätze:

- **Redundanz:** Diagnosefade werden durch die Generierung heuristischer Regeln (Abkürzungsregeln, Redundanzregeln) verhindert. Bei seiner Analyse der WB gibt GenRule eine Warnmeldung aus, sobald es Diagnosefade findet, die durch das Diagnoseverfahren nicht mehr „verwendet“ werden können.
- **Unvollständigkeit:** GenRule, daß Diagnosefade zu kurz sind, so generiert es Reihenfolgegelenke, die die Erhebung der „fehlenden“ Symptome im Rahmen des Diagnoseverfahrens erzwingen. Diagnosefälle, für die GenRule „keinen ähnlichen“ Diagnosefad findet, werden an die PATDEX₂-Teilkomponente weitergereicht, die die Shell um ähnlichkeitsbasierte Klassifikation auf der Grundlage der weitergereichten Fälle erweitert.
- **Inkonsistenz:** GenRule berücksichtigt explizit die Tatsache, daß zu einer Diagnose häufig mehrere Fehler korrespondieren¹⁵. PATDEX₂ kann widersprüchliche Fälle bearbeiten und nach Ähnlichkeitsskriterien anordnen. Lösungen auf Basis derartiger Informationen haben dann eher einen Hinweischarakter.

Da reale empirische Falldaten nicht fehlerfrei sind, ergibt sich eine zusätzliche Aufgabe für die Lernkomponente, nämlich die vorgelegten Fälle möglichst weitgehend zu filtern. Die beiden grundsätzlichen Hilfsmittel, die die Lernkomponente hierzu verwendet, sind die häufigkeitsorientierte Rechteckigung der generierten Regeln und die

Bild 3 - Lernkomponente innerhalb der MOLTKE₃-Werkbank

probabilistische Klassifikation auf der Basis von Ähnlichkeiten. So ignoriert GenRule zum Beispiel redundante Diagnosefade, solange diese nicht zu häufig auftreten. Inkonsistente Fälle werden von PATDEX₂ verarbeitet, ob derartige Änderungen Konsequenzen für die an PATDEX₂ weitergereichten Fälle haben, das heißt ob zusätzliche Fälle weitergereicht beziehungsweise wieder zurückgenommen werden müssen. Ein Überblick über die Lernkomponente bietet Bild 3.

6. Bewertung und Diskussion

Diskussion

Die Diagnoseshell ist das Ergebnis einer mehrjährigen Kooperation mit einem anerkannten Ingenieurinstitut und erfüllt die Anforderungen, die seitens des Instituts gestellt wurden ([17], [15]). In diese Kooperation flossen auch theoretische Vorarbeiten im Bereich der Modellierung diagnostischen Problemlosen mit ein ([22]), die für den praktischen Einsatz vorausgesetzt waren. Aufbauend auf den Ergebnissen der Kooperation wurde die MOLTKE₃-Werkbank entwickelt [5].

5.2. Überblick über die Lernkomponente innerhalb der Werkbank

Die Grundlage der Kooperation zwischen PATDEX₂ und GenRule bildet eine gemeinsame Fallrepräsentation sowie ein gemeinsames Fällgedächtnis. Während GenRule immer auf der Fallgesamtheit operiert, verwendet PATDEX₂ lediglich die Fälle, für die entweder kein ähnlicher Diagnosefad existiert, oder noch kein Regenerierungsverfahren durchgeführt wurde.

Die in der WB vorhandenen Diagnosefade können automatisch durch eine Teilkomponente von GenRule, den KnowledgeBaseExpander (KBE), ermittelt werden.

Für die Lernkomponente diente die ursprüngliche Anwendung (Diagnose) als CNC-Bearbeitungszentren; siehe [4] als Medium zur Integration verschiedener Wissensakquisitionssysteme und Lernverfahren. Der Aspekt der Zweckhaftigkeit der eingesetzten Lernverfahren war dabei von Anfang an wesentlich. Dies

führte zu den spezialisierten Teilkomponenten GenRule und PATDEX₂, und deren vollständiger Eingliederung in die Werkbank. Insbesondere, die Integration der Lernergebnisse von GenRule zusammen mit denen der fall- und modellbasierten¹⁶ beziehungswise maßnahmen Wissensakquisition ist ein ausschlaggebendes Merkmal für die Attraktivität dieses Ansatzes. So unterstreichen zum Beispiel die in [20] vorgestellten Untersuchungsergebnisse und Integrationsvorschläge auf eindrucksvolle Weise die Qualität der MOLTKE₃-Werkbank, die somit den State-of-the-Art im Bereich Wissensintegration darstellt, mit der Einschränkung auf technische Diagnoseproblematisken.

Nach Kodrattoff [11] handelt es sich bei GenRules Analogien um „gute“ Analogien mit „voller“ Komplexität. Die Komplexität der Analogie ist nicht „irreal“, da die Kausalität zwischen Symptomwerten (Situationen) und Diagnose eng mit dem definierten Ähnlichkeitsbegriff verwohnt ist. Ähnlichkeit und Kausalität können auch nicht „einfach“ kombiniert werden und dann auf jedes beliebige Ziel, das heißt präsentierten Diagnosefall, angewendet werden. Den Allgemeinheitsgrad von Diagnosefällen und -fadern ist gleich. Übergeneralisierungen sind mit GenRule nur eingeschränkt möglich. Insbesondere ist die Behandlung von Abkürzungs- und Redundanzen durch die Diagnoseshell relativ unkritisch, da die Klassifikationsfähigkeit des Systems davon unberührt bleibt.

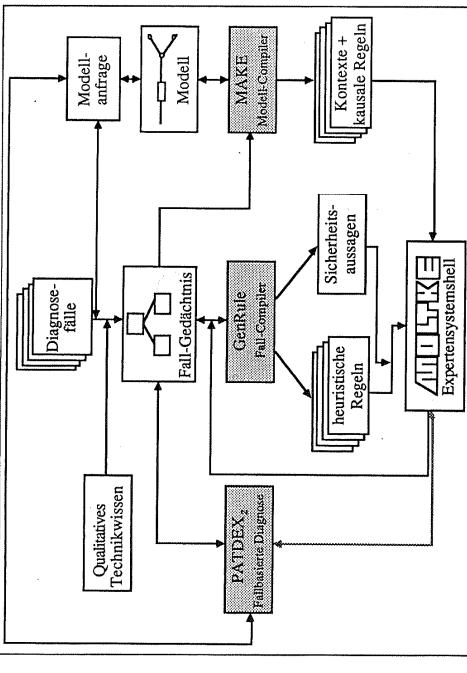
Der analogiebasierte Inferenzmechanismus von PATDEX₂ kann mit Blick auf die Klassifikation als „transformational analogy“ eingedeutet werden, mit

sicht auf die Testselektion allerdings als „Derivation Analogie“ ([9], [10]). Darüber hinaus bestehen Parallelen zum „Memory-Based Reasoning“-Ansatz nach Stanfill und Waltz [18].

Grundsätzlich gilt für die Bewertung von GenRule und PATDEX₂, das gleiche wie für die Bewertung der gesamten Lernkomponente. Sie können sinnvoll eingesetzt werden, wenn sie nach dem Gesamtzusammenhang herausgezogen werden. Charakteristisch für beide Komponenten ist, daß sie eingesetzten Verfahren Einfügen, ins Fällgedächtnis, Zeitspanne und Speicherkomplexität beim Testselektion beeinflussen. Die Klassifikationsfähigkeit einer WB dagegen wird entweder „erweitert“ (ähnlichkeitssbasierter Weise) durch Generierung zusätzlicher Reihenfolgegelenk)

¹⁵ ModelBased Automated knowledge Extrator: hierzu siehe (5) beziehungsweise (16).

¹⁶ Dadurch können „systematische Inkonsistenzen“ behoben werden.

Bild 4 - Der Aufbau der MOLTKE₃-Werkbank

oder nur „bedingt verkürzt“, indem hierfür explizit Abkürzungs- und Redundanzenzeichen repräsentiert werden. Bei der Ausgabe einer Enddiagnose wird dann auf unsicher abgeleitete beziehungsweise für überflüssig erachtete Symptome hingewiesen. Diese Aufteilung resultiert zu einem aus Anforderungen der zugrunde liegenden Domäne, zum anderen ist sie die Konsequenz aus dem Vorhandensein von MAKE¹⁹ (Bild 4), das auf der Basis von Maschinennplänen bereits eine MOLTKE-WERkerzeugen kann.

In ihrer konkreten Zielführung sind die Ansätze von GenRule und PATDEX₂ neu sowohl für das Verfeinern von WBs als auch für den Bereich des fallbasierten Schließens. Insbesondere konzentrieren sich existierende Systeme meist auf den Klassifikationsaspekt, während zum Beispiel in PATDEX₂, sowohl die Klassifikation als „transformational analogy“ eingeordnet werden, mit Sicht auf die Testselektion allerdings als „Derivation Analogie“ ([9], [10]). Darüber hinaus bestehen Parallelen zum „Memory-Based Reasoning“-Ansatz nach Stanfill und Waltz [18].

Grundsätzlich gilt für die Bewertung von GenRule und PATDEX₂, das gleiche wie für die Bewertung der gesamten Lernkomponente. Sie können sinnvoll eingesetzt werden, wenn sie nach dem Gesamtzusammenhang herausgezogen werden. Charakteristisch für beide Komponenten ist, daß sie eingesetzten Verfahren Einfügen, ins Fällgedächtnis, Zeitspanne und Speicherkomplexität beim Testselektion beeinflussen. Die Klassifikationsfähigkeit einer WB dagegen wird entweder „erweitert“ (ähnlichkeitssbasierter Weise) durch Generierung zusätzlicher Reihenfolgegelenk)

¹⁹ ModelBased Automated knowledge Extrator: hierzu siehe (5) beziehungsweise (16).

²⁰ Die von technischen Systemen allerdings meistens erfüllt wird

²¹ Die Implementierungssprache ist Smalltalk-80. Implemetierungsmedium ist die Werkbank auf fast allen Workstations und größeren PCs verfügbar.

²² Darunter fallen zum Beispiel Z sammenhängende „Kabel schalten Ventil“ etc., diese wiederum steuern die Hydraulik, etc.

MOLTKE₃-Werkbank

Danksagung

Unser Dank gilt in besonderem Maße Herrn Prof. Dr. M. M. Richter, sowie der ML-Arbeitsgruppe in Kaiserslautern, für die unzähligen, engagierten Diskussionen, die die Lernkomponente in der hier beschriebenen Art und Weise erst möglich machten.

Literatur

- [1] Althoff, K.-D.: Eine fallbasierte Lernkomponente als integrierter Betandteil der MOLTKE₃-Werkbank zur Diagnose technischer Systeme. 1991 (in Vorbereitung)
- [2] Althoff, K.-D./De la Ossa, A./Maurer, F./Stadler, M./Weiß, S.: Adaptive Learning in the Domain of Technical Diagnosis, in: Proc. Workshop on Adaptive Learning, FAW Ulm, 1989
- [3] Althoff, K.-D./De la Ossa, A./Maurer, F./Stadler, M./Weiß, S.: Case-Based Reasoning for Real World Applications, interner Bericht, Universität Kaiserslautern, 1990
- [4] Althoff, K.-D./Faupel, B./Kockskämper, S./Traphöner, R./Wernicke, W.: Knowledge Acquisition in the Domain of CNC Machining Centers: the MOLTKE Approach, in: Proc. EKAW 1989, S. 180-195

- [5] Althoff, K.-D./Maurer, F./Rehbold, R.: Multiple Knowledge Acquisition Strategies in MOLTKE, in: Proc. EKAW 1990, S. 21-40
- [6] Althoff, K.-D./Traphöner, R.: GenRule: Learning of Shortcut-Oriented Diagnostic Problem Solving in the MOLTKE₃-Workbench, interner Bericht, Universität Kaiserslautern, 1990
- [7] Block, H.: Automatische Klassifikation, Göttingen 1974
- [8] Breuker, J./Wielinga, B.: Model-Driven Knowledge Acquisition: Interpretation Models, Memo 87, Deliverable task A1, Esprit Project 1098, 1987
- [9] Carbonell, J.G.: Learning by analogy: formulating and generalizing plans from experience, in: Michalski, R.S., Carbonell, J.G. & Mitchell, T.M. (eds.): Machine Learning, Togio Publishing Co., Palo Alto, 1983
- [10] Carbonell, H.G.: Derivational Analogy in Problem Solving and Knowledge Acquisition, in: Michalski, R.S., Carbonell, J.G. & Mitchell, T.M. (eds.): Machine Learning, Vol II, Morgan Kaufmann, Los Altos CA, 1986
- [11] Kodratoff, Y.: Combining Similarity and Causality in Creative Analogy, in: Proc. ECAL 1990, S. 398- 403
- [12] Kolodner, J.L.: Maintaining Organization in a Dynamic Long-Term Memory, Cognitive Science (7), S. 243-280, 1983
- [13] Maurer, F.: FOMEX: ein fehlerorientiertes, modulares Expertensystem zur Diagnose von CNC- Bearbeitungszentren, Diplomarbeit, Universität Kaiserslautern, 1989
- [14] Morik, K.: Sloppy Modeling, in: (Morik, K. (d.): Knowledge Representation and Organization in Machine Learning, Springer Verlag: Berlin, Heidelberg 1989), S. 107-134
- [15] Pfeifer, T./Held, H.-J./Faupel, B.: Aufbau einer Wissensbasis für Fehlerdiagnosesysteme von Bearbeitungszentren, VDI-Z VDI-Verlag 10, 1988
- [16] Rehbold, R.: Integration modellbasierten Wissens in technische Diagnostikexpertsysteme, Dissertation, Universität Kaiserslautern, 1991 (in Vorbereitung)
- [17] Richter, M.M./Pfeifer, T./Althoff, K.-D./Faupel, B./Nökel, K./Rehbold, R.: Abschlußbericht Teilprojekt X6, Sonderforschungsbereich „Künstliche Intelligenz – Wissensbasierte Systeme“, Kaiserslautern, 1990
- [18] Stanfill, C./Waltz, D.: The memory based reasoning paradigm, Proc. DARPA Workshop on Case-Based Reasoning, Morgan Kaufmann, 1988
- [19] Traphöner, R.: Ein Konzept zur Verarbeitung von Erfahrungswissen in MOLTKE, Projektarbeit, Universität Kaiserslautern, 1990
- [20] van Someren, M.W./Zheng, L.L./Post, W.: Cases, Models or Compiled Knowledge; a Comparative Analysis and Proposed Integration, in: Proc. EKAW 1990, S. 339-355
- [21] Weiß, S.: PATDEX₂: ein System zum adaptiven, fallfokussierenden Lernen in technischen Diagnosesituationen, Diplomarbeit, Universität Kaiserslautern, 1990
- [22] Wetter, T.: Ein modallogisch beschriebenes Expertensystem, ausgeführt am Beispiel von Ohrenkrankungen, Dissertation, RWTH Aachen 1984



NEU

EDV-Verträge richtig gestaltet

Praktischer Ratgeber für Verträge nach den „Besonderen Vertragsbedingungen (BVB)“

Egbert-E. Reichel, Norbert H. Siegrist

Das Buch enthält als Arbeitsgrundlage und -hilfe:

- einen Textabdruck aller „Besonderen Vertragsbedingungen (BVB)“ mit zahlreichen zusätzlichen Erläuterungen und Anmerkungen,
- eine Zusammenfassung der wichtigen Rechte und Pflichten des Auftragnehmers und Auftraggebers (Checkliste),
- diverse in der Praxis anwendbare Formulare, Muster eines Kauf-, Miet- und Wartungsscheins sowie
- wichtige Tips und Hinweise rund um die BVB.

Ziel des Buches ist es, allen Projektverantwortlichen (Führungskräfte, Organisatoren usw.) wichtige Arbeitsgrundlagen und Entscheidungshilfen an die Hand zu geben.

(395 Seiten, August 1990, 68 DM, Format 16,5 x 23,5 cm, brosch., ISBN 3-922213-09-X, Bestell-Nummer A 8156)

FACHBUCH-SERVICE

FBO
Fachverlag für Büro- und
Organisationstechnik GmbH
Postfach 316

7570 Baden-Baden

Ich/wir bestelle(n) gegen Rechnung und zur Lieferung an die untenstehende Anschrift:

— Exemplar(e) **EDV-Verträge richtig gestaltet**
Bestell-Nr. 8156 (68 DM zzgl. Versandkosten)

() Bitte senden Sie mir Ihr Gesamtverzeichnis zu.

Firma/Abteilung

Name/Vorname

Postfach/Straße

PLZ/Ort

Datum



Unterschrift