Patdex - ein Ansatz zur wissensbasierten und inkrementellen Verbesserung von Ähnlichkeitsbewertungen in der fallbasierten Diagnostik*

Stefan Weß
Fachbereich Informatik – SFB 314
Universität Kaiserslautern
W-6750 Kaiserslautern
email: wess@informatik.uni-kl.de

Zusammenfassung

Die Leistungsfähigkeit fallbasierter Ansätze in der Diagnostik hängt stark von der Wahl eines für die jeweilige Anwendungsdomäne geeigneten Ähnlichkeitsbegriffs ab. In dieser Arbeit wird ein Ansatz zur fallbasierten Diagnose technischer Systeme entwickelt, der auf einem dynamischen Ähnlichkeitsmaß aufbaut. Das verwendete Ähnlichkeitsmaß kann ausgehend von einer näherungsweisen Beschreibung durch ein Lernverfahren inkrementell an den in der Anwendungsdomäne vorliegenden Ähnlichkeitsbegriff angepaßt werden. Die Bestimmung der Ähnlichkeit kann zudem durch das in diagnostischen Anwendungsdomänen vorhandene Wissen unterstützt werden.

1 Einführung

Menschliche Diagnoseexperten verwenden zum Problemlösen unterschiedliche Wissensquellen und Schlußfolgerungsmechanismen. In der Expertensystemforschung spiegelt sich diese Erkenntnis in der Realisierung von regel-, modell-[1] und fallbasierten Expertensystemen wider (vgl. [Pup90]). Während regel- und modellbasierte Systeme bereits seit längerem Gegenstand intensiver Forschung sind, finden fallbasierte Techniken zur Diagnose erst in jüngerer Zeit verstärkte Beachtung. Dies ist insofern verwunderlich, da für viele Anwendungen die Betrachtung von Fallbeispielen eine natürliche Art des diagnostischen Problemlösens darstellt. Ziel des fallbasierten Problemlösens ist es, ein aktuell vorliegendes Problem direkt auf der Basis von bereits bekannten Fallbeispielen zu lösen. Für die fallbasierte Diagnostik bedeutet dies, daß ein vorliegendes Diagnoseproblem gelöst wird, indem man die (bekannte) Diagnose eines *ähnlichen Problems* auf die aktuelle Situation überträgt. Fallbeispiele können wir in diesem Kontext als „*Protokolle des realen Problemlöseverhaltens von Diagnoseexperten*“ auffassen. Fallbasiertes Schließen als Problemlöseparadigma hat seine Wurzeln im Bereich der Kognitionspsychologie; die grundlegenden Arbeiten in diesem Bereich wurden von R.C. Schank bereits 1982 veröffentlicht (als Überblick vgl. auch [AWBS+92]). Fallbasierte Anwendungen im Bereich der Diagnostik wurden aber erst verhältnismäßig spät entwickelt, z.B. Casey [Kot88], Patdex/1 [AKM+89], Creek [Aam91] und CcC+ [PG91]. In dieser Arbeit wollen wir fallbasierte Ansätze zur Diagnose technischer Systeme aus dem speziellen Blickwinkel der im Bereich der Mustererkennung entwickelten *nearest–neighbour* Klassifikation (vgl. [Das90]) betrachten. Diese simplifizierende Sichtweise auf bestehende Ansätze zur fallbasierten Diagnose ermöglicht es uns, auf der einen Seite Defizite konkret aufzuzeigen und auf der anderen Seite entsprechende Erweiterungsmöglichkeiten zur

*erscheinen in: Puppe, Günter, Expertensysteme XPS-93, 42-55, Springer Verlag 1993. Die hier vorgestellte Arbeit wurde zum Teil gefördert durch die Deutsche Forschungsgemeinschaft, SFB 314: “Künstliche Intelligenz - Wissensbasierte Systeme“, Projekt X9 - CAbPlan.

[1] Wir verwenden im folgenden den Begriff *Modell* immer im Sinne eines qualitativen technischen Modells des zu diagnostizierenden Systems und nicht im Sinne eines konzeptionellen Modells der Wissensakquisition.

Behebung dieser Defizite vorzuschlagen. Der in der Arbeit vorgestellte Grundalgorithmus zur fallbasierten Diagnostik wird dazu schrittweise so erweitert, daß er die beschriebenen Anforderungen erfüllt. Am Beispiel des PATDEX-Systems wird anschließend exemplarisch ein System vorgestellt, welches den in der Arbeit verfolgten Ansatz realisiert. Dies wird ergänzt durch eine kurze Beschreibung von einigen über diese Arbeit hinausgehehende Aspekte der Architektur des Gesamtsystems.

2 Fallbasierte Diagnostik

In der fallbasierten Diagnostik können wir grob zwei Anwendungsszenarien unterscheiden:

Szenario 1: Fälle als Basisrepräsentation In einigen Domänen sind Fallbeispiele ein gängiger und natürlicher Formalismus für die Repräsentation von diagnostischem Problemlösewissen. Fallbeispiele werden in diesen Bereichen mangels der Kenntnis anderer Wissensquellen oder aus Effizienzgründen eingesetzt. Dazu ein Beispiel: Bei der Herstellung von technischen Produkten, für die komplexe Prozesse mit vielen Parametern notwendig sind (z.B. Chipherstellung, Produktion von Kunststoffstoßstangen), ist in der Industrie die Fehlerbehebung (Diagnostik) auf der Basis von sog. *Schichtbüchern*, also protokollierten Fallsammlungen, eine gängige Praxis. Fallbeispiele ersetzen hier nichtvorhandenes Wissen über die genauen (tiefen) Zusammenhänge in einer Domäne.

Szenario 2: Ausnahmefälle In Domänen, in denen bereits abstrakte Modelle und ausgereifte heuristische Vorgehensweisen existieren, verwenden Experten Fallbeispiele, um Fehlersituationen zu analysieren, die in ihrem Modell nicht auftreten können (z.B. Verkleben von Schaltern, Fehler bei sehr tiefen Außentemperaturen), oder um Abweichungen von der üblichen heuristischen Vorgehensweise explizit zu repräsentieren. Fallbeispiele ergänzen in diesen Situationen bereits vorhandenes Wissen und ermöglichen die direkte Diagnose von Ausnahmesituationen bzw. leiten den heuristischen Problemlöseprozeß in diesen Situationen.

Innerhalb dieser beiden Szenarios müssen wir wiederum zwischen den beiden Teilaufgaben der Diagnostik differenzieren: In diagnostischen Aufgabenstellungen soll einerseits ausgehend von einer Menge von bekannten Symptomen die für diese Symptomatik verantwortliche Ursache identifiziert werden *(Aufgabe 1: Klassifikation)*. Andererseits soll die im allgemeinen unvollständige Informationssituation durch die Erhebung weiterer Meßwerte vervollständigt werden *(Aufgabe 2: Testerhebung)*. Ein fallbasiertes Diagnosesystem *sollte beide* ineinander verzahnte Aspekte einer Diagnoseaufgabe berücksichtigen. Fallbeispiele können daher in der Diagnostik mit zwei unterschiedlichen Zielen eingesetzt werden:

Ziel 1: Erweitern der Klassifizierungsfähigkeit Durch entsprechende Fallbeispiele *(Klassifikationsfälle)* können neue Fehlersituationen korrekt diagnostiziert werden, d.h. Unterstützung der Klassifikation.

Ziel 2: Abkürzen des Problemlösungsweges Durch Fallbeispiele *(Strategiefälle)* können alternative Problemlösungswege aufgezeigt und mögliche Sackgassen vermieden werden, d.h. Unterstützung der Testerhebung.

Entscheidend für die Trennung von Testerhebungs- und Klassifikationswissen in Klassifikations- und Strategiefälle ist dabei die Überlegung, jedes der beiden Teilprobleme der Diagnostik durch fallbasiertes Schließen *unabhängig voneinander* verbessern zu können. Wir werden uns im folgenden primär auf die Klassifikation, d.h. die Zuordnung einer Diagnose zu einer gegebenen Symptomatik, konzentrieren. Viele der im Rahmen dieser Arbeit vorgestellten Überlegungen zur Klassifikation lassen sich jedoch ohne Änderungen auf *Strategiefälle* und damit auf die Steuerung der Testerhebung übertragen.

2.1 Ähnlichkeit und Ähnlichkeitsmaße

Aus einer abstrakten Sicht ist der Begriff der *Nützlichkeit* für fallbasiertes Schließen zentral. Gesucht wird ein Fallbeispiel, welches im Kontext der aktuellen Problemstellung *nützlich* für die Lösung ist. Das a posteriori Kriterium Nützlichkeit wird, mangels anderer Wissensquellen, im fallbasierten Schließen auf den Begriff der Ähnlichkeit reduziert. Diese Vorgehensweise liegt in der Hoffnung begründet, daß die Ähnlichkeit der Problemstellungen die Nützlichkeit für die Problemlösung impliziert.

Ein gängiger Ansatz, die Ähnlichkeit von Problemstellungen zu modellieren, ist die Verwendung von Ähnlichkeitsmaßen. Das Maß bestimmt die abstrakte Klasse der Probleme, die mit einem fallbasierten Ansatz gelöst werden können, und somit ist die Auswahl des verwendeten Ähnlichkeitsmaßes von entscheidender Bedeutung.

Definition 1 (Ähnlichkeitsmaß) *Eine Abbildung* $sim : \mathcal{M} \times \mathcal{M} \rightarrow [0,1]$ *mit* $\forall x \in \mathcal{M}$ $sim(x,x) = 1$ *(Reflexivität), heißt Ähnlichkeitsmaß.*

Die Anforderungen an ein Ähnlichkeitsmaß gemäß der Definition 1 sind denkbar gering. Insbesondere verzichten wir auf die in der Literatur oft geforderte Symmetrie, d.h. $sim(x,y) = sim(y,x)$, die in diagnostischen Aufgabenstellungen wegen der Beschränktheit der Information nicht immer gesichert werden kann. Ähnlichkeitsmaße sind – meist in der Form ihres Duals, des Abstandsmaßes[2] – Gegenstand ausgedehnter Untersuchungen in verschiedenen Disziplinen. Eine ausführliche Diskussion relevanter Maße würde aber den Rahmen dieser Arbeit sprengen[3].

Die Verwendung eines Maßes für die Ähnlichkeitsbewertung ist jedoch *nur eine* der möglichen Vorgehensweisen. Genau betrachtet benötigen wir beim fallbasierten Schließen lediglich eine *Präferenzordnung* über den Fällen in der Fallbasis. Die Art und Weise, wie eine solche Ordnung definiert wird, ist für das fallbasierte Schließen nur von untergeordneter Bedeutung. Ähnlichkeitsmaße bieten hier den Vorteil, daß sie eine sehr einfache Möglichkeit zur Definition einer solchen Ordnung realisieren. Das Ähnlichkeitsmaß reduziert jeden Fall auf einen numerischen Wert, die Ordnung über den reellen Zahlen induziert dann die Präferenzordnung über der Fallbasis.

2.2 Fallbasierte Diagnostik als nearest-neighbour Klassifikation

In dieser Arbeit stellen wir uns auf einen an [SW86] bzw. [AKA91] angelehnten Standpunkt und verstehen die fallbasierte Diagnostik als einen Spezialfall der aus der Mustererkennung bekannten *nearest-neighbour* Klassifikation. Diese Sichtweise ermöglicht erstens die Einordnung der fallbasierten Diagnostik in einen allgemeinen Bezugsrahmen, zweitens erhalten wir damit einen einfachen Basisalgorithmus, den wir entsprechend den Anforderungen von diagnostischen Aufgabenstellungen erweitern können.

Im Gegensatz zu klassischen Anwendungen des Nearest-Neighbour Ansatzes, beispielsweise in der Mustererkennung, ist die Diagnostik, insbesondere die Diagnose von technischen Systemen, ein Gebiet, in dem spezifische Strukturen, Vorgehensweisen und Abhängigkeiten erkennbar sind. Diese spezifischen Eigenschaften von diagnostischen Aufgabenstellungen sind die Ansatzpunkte für die in den folgenden Abschnitten behandelten Erweiterungen des Grundalgorithmus. Zum Verständnis der Algorithmen sind jedoch zunächst einige Definitionen notwendig:

Definition 2 (Symptom) *Ein* Symptom *ist eine meßbare Größe im technischen Sinne. Jedes Symptom* $S_i = (N_i, W_i, s_i)$ *besteht aus einem* Namen N_i, *einem* Wertebereich W_i *sowie einer* Symptomvariable s_i. *Die Symptomvariable speichert dabei den jeweils konkret gemessenen* Symptomwert $v \in W_i$. *Wurde für* S_i *noch kein Wert erhoben, so gilt* $s_i = unknown$.

Definition 3 (Situation) *Als* Situation *(Symptomatik)* Sit *bezeichnen wir die zu einem bestimmten Zeitpunkt* τ *bekannten* Symptomwerte *(Meßwerte), d.h.* $Sit \subseteq \{S_1, S_2, \ldots, S_n\}$ *mit* $s_i \neq unknown$.

[2] Zur Äquivalenz von Distanz- und Ähnlichkeitsmaßen vgl. [RW91]

[3] Für einen Vergleich und eine Einordnung bekannter Maße sei z.B. auf [Wes91] verwiesen.

Definition 4 (Fall) *Ein* Diagnosefall *(Klassifikationsfall) repräsentiert die konkrete Erfahrung eines Diagnoseexperten. Ein Fall C_k ist definiert als ein Tupel (Sit_k, D_j), wobei Sit_k die Beschreibung der* Symptomatik *und D_j die in dieser bestimmten Situation gestellte* Diagnose *ist.*

Wir schreiben im folgenden $C_k.Sit$ für die einem Fall C_k zugrundeliegende Episode und $C_k.D_j$ für die in dieser Episode gestellte Diagnose. Die Diagnose $C_k.D_j$ beschreibt dabei die Klasse, der ein Fall C_k zugeordnet wird. Die Menge $FB = \{C_1, C_2, \ldots, C_m\}$ von Diagnosefällen wollen wir Fallbasis nennen. Der folgende Algorithmus bildet den Kern unseres Ansatzes zur fallbasierten Diagnostik. Die Zuordnung einer Diagnose D_i zu einer Symptomatik Sit erfolgt hier, indem die Diagnose D_j des ähnlichsten Falles C_k bzgl. eines Ähnlichkeitsmaßes sim für die aktuelle Situation übernommen wird.

Eingabe:	Fallbasis FB Ähnlichkeitmaß sim aktuelle Situation Sit
Ausgabe:	Diagnosefall C_k

1. Suche in der Fallbasis FB den Diagnosefall C_k mit: $sim(C_k.Sit, Sit)$ ist maximal, d.h. $\forall\ C_i \in FB$ gilt $sim(C_i.Sit, Sit) \leq sim(C_k.Sit, Sit)$.
2. Gebe den Fall C_k und damit die in C_k vermerkte Diagnose $C_k.D$ als Lösung aus.

In der fallbasierten Diagnostik lassen sich grundsätzlich die folgenden Ansätze unterscheiden.

Definition 5 (Fallbasierte Diagnostik) *Die Übernahme einer Diagnose D_j für eine gegebene Symptomatik Sit auf der Basis eines gefundenen Fallbeispiels $C_k = (Sit_k, D_j)$ bezeichnen wir als* fallvergleichende Diagnostik. *Wird die Diagnose D_j eines gefundenen Fallbeispiels für eine gegebene Symptomatik Sit modifiziert, so sprechen wir von* fallmodifizierender Diagnostik.

Der vorgestellte Algorithmus realisiert (nach Def. 5) auf der Basis einer nearest–neighbour Klassifikation einen einfachen Ansatz zur *fallvergleichenden Diagnostik*. Der Algorithmus wirft jedoch eine Reihe von Fragen auf, die im Rahmen dieser Arbeit beantwortet werden sollen: Wie wird das Ähnlichkeitsmaß sim realisiert? Wie kann sim verbessert werden? Welche Fallbeispiele werden in die Fallbasis aufgenommen? Wie kann Domänenwissen den Diagnoseprozeß unterstützen?

2.3 Fallbasiertes Lernen

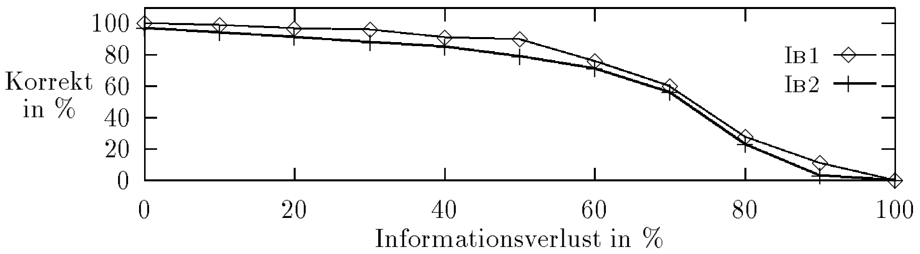
Mit dem in 2.2 beschriebenen Algorithmus ist der Begriff des Lernens direkt verknüpft. Lernen und die Anwendung des gelernten Wissens sind dabei nicht strikt voneinander getrennt, sondern eng miteinander verzahnt. Wird eine weitere Diagnoseaufgabe korrekt gelöst, so wird sie als ein neuer Fall in die Fallbasis aufgenommen. Aha [AKA91] bezeichnet diese Vorgehensweise auch als IB1–Algorithmus. Aus der Sicht des Maschinellen Lernens kann fallbasiertes Lernen daher als eine Begriffsbildungsaufgabe verstanden werden. Damit stellt sich zwangsläufig die Frage nach der Mächtigkeit fallbasierter Lernverfahren. Jantke [Jan92] hat Gemeinsamkeiten von *Induktiver Inferenz* und fallbasierten Lernverfahren theoretisch untersucht. Zusammenfassend kommt er zu dem Schluß: *"In the setting investigated, every inductive inference strategy [...] can be implemented as a case–based learning strategy [...]."* Nach Jantke ist also die (theoretische) Mächtigkeit fallbasierter Lernverfahren mit den in der Induktiven Inferenz entwickelten Verfahren gleichzusetzen.

In der Diagnostik entspricht ein zu lernender Begriff einer bestimmten Diagnose D_j. Im Gegensatz zu induktiven Lernverfahren, bei denen ein gelernter Begriff intensional durch eine entsprechende Formel repräsentiert wird, beschreiben fallbasierte Verfahren Begriffe als Tupel (sim, FB) (vgl. [Ric92]), d.h. durch ein Ähnlichkeitsmaß sim und eine Menge von Fallbeispielen FB. Werden unterschiedliche Diagnosen gelernt, so wird jede Diagnose D_j durch ein *spezifisches Tupel* (sim_j, FB_j) repräsentiert. Das fallbasierte System (sim_j, FB_j) ist dann ein Klassifikator für die Diagnose D_j bzw. für die Diagnose $\neg D_j$. Wollen wir die Klassifikationsfähigkeit eines fallbasierten Systems verbessern, so müssen wir FB oder sim verändern.

Definition 6 (Fallbasiertes Lernen) *Die Änderung der Fallbasis FB bzw. die Anpassung des Ähnlichkeitsmaßes sim bezeichnen wir als* fallbasiertes Lernen.

2.3.1 Änderung der Fallbasis

Die Anzahl der in der Fallbasis zu speichernden Fallbeispiele kann minimiert werden, indem nur „*typische*" Beispiele aufgenommen werden. Einen sehr einfachen Ansatz realisiert der IB2 genannte Algorithmus nach Aha [AKA91]. Fallbeispiele werden im Gegensatz zum IB1-Algorithmus, der alle gelösten Fälle speichert, dort nur dann in die Fallbasis FB aufgenommen, falls sie mit der bereits bestehenden Fallbasis und dem definierten Ähnlichkeitsmaß sim falsch klassifiziert werden. Für eine Beispielfallbasis haben wir die Auswirkungen eines solchen Ansatzes untersucht[4]. Zunächst wurden beide Algorithmen mit einer Menge von 100 Fallbeispielen trainiert. Der IB1 Algorithmus nahm jedes der präsentierten Fallbeispiele in seine Fallbasis auf. Der IB2-Algorithmus reduzierte die Menge der präsentierten Fallbeispiele auf eine Fallbasis von 49 Fallbeispielen. Im Anschluß daran haben wir beide Algorithmen gegen die ursprünglich gegebenen 100 Fallbeispiele getestet und dabei die Menge der pro Fallbeispiel bekannten Symptome schrittweise vermindert. Im folgenden Diagramm haben wir die Klassifikationsgenauigkeit der beiden Algorithmen in Abhängigkeit von der Vollständigkeit der präsentierten Information aufgetragen. Ein Informationsverlust von 70% bedeutet dabei: Die Klassifikation eines Fallbeispiels C_k basiert auf nur 30% der in der Situationsbeschreibung $C_k.Sit$ verzeichneten Symptome[5]. Beachtenswert ist dabei, daß obwohl IB2 die verwendete Beispielfallbasis auf weniger als die Hälfte der ursprünglichen Fallbeispiele reduzierte, die Klassifikationsgenauigkeit der beiden Algorithmen nur geringfügig differiert. Hierbei ist aber zu beachten, daß IB2 nur eine einfache Form eines Entscheidungskriteriums für die Aufnahme von Fällen in die Fallbasis realisiert. Durch die Entwicklung von anderen Kriterien zur Auswahl (vgl. z.B. auch [AKA91]) können unter Umständen noch bessere Ergebnisse erzielt werden.



Eine wichtige Aufgabe in der fallbasierten Diagnostik ist es, zu entscheiden, welche Fallbeispiele in die Fallbasis aufgenommen werden und welche nicht. Da sim und FB nicht unabhängig voneinander sind, bestimmt der im jeweiligen System verwendete Ähnlichkeitsbegriff sim, ob die Aufnahme eines weiteren Fallbeispiel in FB zu einer qualitativen Verbesserung des Gesamtsystems führt oder nicht.

2.3.2 Änderung des Ähnlichkeitsmaßes

Ist der Ähnlichkeitsbegriff für eine Domäne bekannt, so können wir ein entsprechendes Maß angeben. Dieses Ähnlichkeitsmaß ist dann eine „compilierte" Darstellung unseres Wissens über die entsprechende Domäne. Wir vertreten die These, daß der Aufwand und das nötige Wissen zur Konstruktion eines *„guten"* Ähnlichkeitsmaßes vergleichbar ist mit dem Aufwand zur Erstellung eines konventionellen Problemlösers. Nur wenn der entsprechende Problembereich vollständig

[4] Alle empirischen Untersuchungen in dieser Arbeit wurden mit dem in Abschnitt 3.4 definierten Ähnlichkeitsmaß durchgeführt. Als Beispielfallbasis diente eine im Rahmen des MOLTKE-Projektes erstellte Fallbasis mit 100 Fällen aus dem Bereich der Diagnose von CNC-Werkzeugmaschinen.

[5] Die Symptome wurden in der Reihenfolge präsentiert in der sie im Fallbeispiel verzeichnet sind. Dabei wurde die Zahl der präsentierten Symptome abgerundet, d.h. z.B. 90 % Information bei insgesamt 4 Symptomen sind nur 3 bekannte Symptomwerte. Wird aufgerundet, so sind die erzielten Ergebnisse deutlich besser.

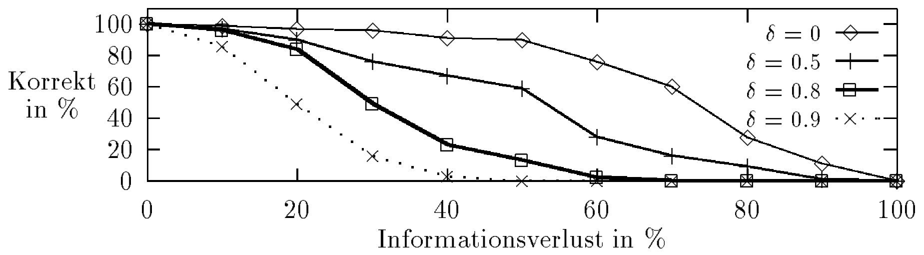
durchdrungen worden ist, kann die Ähnlichkeit zwischen diagnostischen Problemstellungen korrekt beurteilt werden. Wobei sich dann allerdings die Frage nach der Notwendigkeit eines fallbasierten Ansatzes stellt, da das vorhandene Wissen auch zum Entwurf eines konventionellen Diagnosesystems eingesetzt werden könnte. Im allgemeinen werden wir in der fallbasierten Diagnostik daher das für eine Domäne geeignete Ähnlichkeitsmaß nur näherungsweise bestimmen können (vgl. Szenario 1).

Betrachten wir den Ähnlichkeitsbegriff im Kontext des Problemlösens, so ist festzustellen, daß die Ähnlichkeit von Problemstellungen ein *a posteriori Kriterium* ist, denn erst nachdem wir versucht haben, das Problem auf der Basis des gewählten Ähnlichkeitsbegriffes zu lösen, können wir beurteilen, ob der eingeschlagene Weg zu einem Erfolg geführt hat. Dies läßt einen Rückschluß auf die Güte des zur Lösungsfindung verwendeten Ähnlichkeitsbegriffes zu. Diese *a posteriori* Information ist aber zum Zeitpunkt der Problemstellung, also *a priori*, im allgemeinen nicht direkt erhältlich. Die Suche nach einem geeigneten Ähnlichkeitsbegriff kann also als ein *Lernprozeß* aufgefaßt werden (vgl. [RW91]). Das Ziel des Lernprozesses ist es, den zur Problemlösung verwendeten Ähnlichkeitsbegriff schrittweise an den durch die konkrete Anwendung vorgegebenen Ähnlichkeitsbegriff anzunähern.

Auch die Auswirkungen eines solchen Ansatzes hat Jantke [Jan92] untersucht. Er zeigt, daß im Bereich der Induktiven Inferenz die Klasse, der mit einem *statischen Maß lernbaren Probleme* kleiner ist als die Klasse der mit einem *dynamischen Maß lernbaren Probleme*. Jantke stellt in [Jan92] unter anderem fest: *"Using fixed totally defined similarity measures is much too restrictive. [...] Maximal learning power can be achieved by learning possibly partial recursive similarity measures. There is no way to gain more power."* Die Voraussetzung für die theoretisch gleiche Mächtigkeit von fallbasierten und in der Induktiven Inferenz entwickelten Verfahren ist also, daß nicht nur die Fallbasis FB, sondern auch das Ähnlichkeitsmaß sim im Laufe des Lernprozesses verbessert werden.

2.4 Vermeidung von Fehldiagnosen

Wie wir in den letzten Abschnitten gesehen haben, können wir fallbasiertes Lernen als Begriffsbildungsaufgabe auffassen. Ähnlich wie induktive Schlüsse sind fallbasierte Schlüsse damit unsicher. Fehldiagnosen lassen sich im allgemeinen daher nicht verhindern. Wir können aber versuchen, eine „*minimale Korrektheit*“ der gefundenen Lösungen zu sichern. Betrachten wir dazu nochmals den Grundalgorithmus. Dieser liefert für $FB \neq \emptyset$ immer ein Ergebnis, da jeweils der Fall mit der maximalen Ähnlichkeit ausgegeben wird. Damit wird auch in Situationen, die das System noch gar nicht richtig diagnostizieren kann (da z.B. eine entsprechende Fehlersituation noch nie vorgelegen hat), eine falsche Fehlerhypothese ausgegeben. Für viele Anwendungen mag eine solche „*beste Näherung*“ wichtig und interessant sein, in realen Diagnoseanwendungen verursacht jedoch jede Fehlklassifikation Kosten in der Form, daß der Maschinenbediener die vom System vorgeschlagene Diagnose erst verifizieren muß, was unter Umständen sehr aufwendig sein kann. Dieser Aufwand kann vermieden werden, indem für jede Diagnose D_j eine *minimale Ähnlichkeit* zur aktuellen Symptomatik gefordert wird, bevor die entsprechende Diagnose vom System als Lösungskandidat in Betracht gezogen wird. Bei dem beschriebenen Grundalgorithmus kann dieses Ziel durch die Einführung von Schwellwerten erreicht werden. Dieser Schwellwert, nennen wir ihn δ, mit $0 \leq \delta \leq 1$, sollte in Abhängigkeit vom Kostenrisiko einer Fehldiagnose D_j bestimmt werden. Einfach zu verifizierende Fehlerhypothesen erhalten ein niedriges δ_j und können so vom System sehr schnell vorgeschlagen werden; schwierig zu verifizierende Diagnosen mit hohen δ_j werden nicht voreilig vom System als mögliche Lösung präsentiert. Wird beim Fallvergleich kein Fall mit einer den Schwellwert übersteigenden Ähnlichkeit gefunden, so kann das System nun entweder weitere Meßwerte anfordern oder mit *Keine Lösung* abbrechen. Der Schwellwert δ_j bietet so eine einfache Möglichkeit, Wissen über das Kostenrisko von Fehldiagnosen zu formulieren und im Diagnoseprozeß entsprechend zu berücksichtigen. Die Definition eines Falles wird damit um einen für jede Diagnose D_j spezifischen Schwellwert δ_j erweitert, d.h. $C_k = (Sit_k, D_j, \delta_j)$. Für die so definierte *„Kompetenzeinschätzung“* des fallbasierten Diagnosesystems muß aber ein Preis bezahlt werden. Für bereits bekannte Diagnosen D_j sinkt die Klassifikationsfähigkeit des modifizierten Algorithmus mit abnehmender Information stark ab, da die ermittelte Ähnlichkeit dann unter den definierten Schwellwert δ fällt. Wir haben in dem folgenden Diagramm erneut in Abhängigkeit von der Vollständigkeit der präsentierten Information die Korrektheit der Klassifikation für die betrachteten Algorithmen aufgetragen. Sind nur relativ wenige Symptomwerte bekannt, so sinkt beim modifizierten Algorithmus $\delta_j > 0$ die Zahl der korrekten Klassifikationen stärker als beim Algorithmus aus 2.2 ohne Schwellwert $\delta_j = 0$. Je höher δ gewählt wurde, um so stärker ist der entsprechende „Einbruch“ in der Klassifikationsfähigkeit.



Bei einem zunehmendem Anteil von dem System noch unbekannten Diagnosen D_l in der Testmenge, d.h. in der Trainingsphase, ist das Verhalten der betrachteten Algorithmen jedoch gerade invertiert. Bei den Algorithmen mit Schwellwert bleibt die Anzahl der Fehlklassifikationen nahezu konstant, während sie bei $\delta = 0$ überproportional anwächst. Ist der Lernprozeß beendet, d.h. alle D_j sind dem System bereits bekannt, so können wir also zugunsten einer verbesserten Klassifikationsfähigkeit, auf δ_j verzichten. In der Trainingsphase vermeidet δ_j jedoch unnötige Kosten in Form von Fehldiagnosen, indem vor der Ausgabe der Diagnose an den Benutzer vom System noch weitere Meßwerte angefordert werden.

3 Ein Ansatz zur fallbasierten Diagnostik

In den folgenden Abschnitten wollen wir einen Ansatz zur fallbasierten Diagnose technischer Systeme vorstellen. Wir werden dazu aufzeigen, welche Wissensquellen den Fallvergleich unterstützen können und wie die Relevanz eines Symptoms S_i für eine Diagnose D_j bestimmt werden kann. Anschließend beschreiben wir ein für die Diagnostik geeignetes Ähnlichkeitsmaß und zeigen wie wir dieses bei einer Fehlklassifikation verbessern können. Zum Abschluß stellen wir einen Algorithmus zur fallbasierten Diagnostik vor, der alle in dieser Arbeit beschriebenen Erweiterungen des Grundalgorithmus realisiert.

3.1 Wissensbasierte Unterstützung des Fallvergleichs

Zunächst soll jedoch gezeigt werden, welches Wissen zur Unterstützung von Ähnlichkeitsbewertungen in technischen Anwendungen zur Verfügung steht und wie dieses Wissen für den Fallvergleich entsprechend genutzt werden kann. Wir verfolgen also die Strategie, das *a posteriori* Kriterium Ähnlichkeit durch den Einsatz von *a priori* vorhandenem Wissen anzunähern.

3.1.1 Ähnlichkeit zwischen Symptomwerten

Bei der Berechnung der Ähnlichkeit zwischen einem gespeicherten Fall und einer gegebenen Symptomatik wird oft lediglich die Übereinstimmung $v_i = v_k$ bzw. nicht Übereinstimmung $v_i \neq v_k$ von Symptomwerten betrachtet. In der technischen Diagnostik sind aber feinere Differenzierungen wünschenswert. So unterscheidet sich beispielsweise *Öldruck sehr hoch* von *Öldruck sehr niedrig* erheblich. Die Differenz zwischen *Öldruck hoch* und *Öldruck sehr hoch* ist jedoch wesentlich geringer. Um dies in einem fallbasierten Diagnosesystem ausdrücken zu können, ist es nötig, die Ähnlichkeit von Symptomwerten $v_i, v_k \in W_j$ durch ein spezielles Ähnlichkeitsmaß ω_j zu beschreiben. Dies kann erreicht werden, indem für jedes Symptom S_i bei der Definition des entsprechenden Wertebereichs W_j das für den Vergleich von Elementen aus diesem Wertebereich jeweils zu verwendende lokale Ähnlichkeitsmaß $\omega_j(v_i, v_k)$ mit $v_i, v_k \in W_j$, angegeben wird. Unabhängig davon können wir bereits Klassen von möglichen Ähnlichkeitsmaßen für die verschiedenen Merkmalstypen vordefinieren, z.B. *numerisch, string, symbolisch binär, symbolisch mehrstufig.* Die einzelnen Maße $\omega_j(v_i, v_k)$ müssen dann aber unter Umständen noch an die Anforderungen eines konkret gegebenen Wertebereichs W_j in der betrachteten Domäne angepaßt werden. Darüberhinaus sollte es möglich sein, das System um domänenspezifische bzw. benutzerdefinierte Funktionen zur lokalen Ähnlichkeitsberechnung zu erweitern.

3.1.2 Kausale Zusammenhänge zwischen Symptomwerten

In technischen Diagnosedomänen existieren eine Vielzahl von oft einfach zu modellierenden, kausalen Zusammenhängen zwischen einzelnen Symptomwerten, z.B. *Falls das Licht brennt, so ist elektrische Spannung vorhanden.* Zusammenhänge dieser Art lassen sich sehr einfach in Form von Regeln der Art $s_1(v_1) \wedge s_2(v_2) \wedge \ldots s_n(v_n) \rightarrow s_k(v_k)$ formulieren und zur Expansion einer gegebenen Symptomatik ausnutzen. Die Informationsbasis des Fallvergleichs kann so ohne die Durchführung weiterer Tests verbessert werden. Kausales Wissen dieser Art kann vom Experten manuell eingegeben oder aus einem qualitativen Modell bzw. induktiv abgeleitet werden.

3.1.3 Pathologische Symptomwerte

In einem technischen System deuten gewisse Meßwerte auf ein Fehlerverhalten hin, z.B. *Spannung zu hoch*, während andere Meßwerte durchaus ihren Ursprung in einem Normalverhalten des technischen Systems haben können, z.B. *Relais geschaltet.* Meßwerte, die eindeutig auf ein Fehlverhalten des technischen Systems zurückzuführen sind, bezeichnen wir als *pathologische*[6] Symptomwerte. Eine Fehlerdiagnose ist für eine aktuelle Symptomatik um so eher geeignet, je mehr dieser *beobachteten pathologischen* Symptomwerte sie erklären kann. Diese spezielle Eigenschaft von diagnostischen Anwendungen muß bei einem Fallvergleich berücksichtigt werden. Wir können dies erreichen, indem wir Fälle bevorzugen, die *pathologische* Meßwerte erklären können, d.h. die diese Meßwerte auch in der eigenen Situationsbeschreibung Sit enthalten. In der fallbasierten Diagnostik ermöglicht die Überprüfung auf die Übereinstimmung von pathologischen Meßwerten damit eine Einschätzung der Nützlichkeit von gefundenen, ähnlichen Fallbeispielen für die aktuelle Diagnosesituation. Pathologische Symptomwerte können entweder vom Experten deklariert oder durch Simulation eines qualitativen Modells ermittelt werden. Dabei ist jedoch zu beachten, daß im allgemeinen nicht für alle Symptome in allen Situationen eindeutig entschieden werden kann, ob ein Symptomwert einen pathologischen Charakter hat oder nicht. Ein Relais kann sich beispielsweise zufällig in einem korrekten Schaltzustand befinden, obwohl es eigentlich defekt ist.

3.1.4 Defaultwerte für Symptome

Wird ein technisches System von einem Servicetechniker diagnostiziert, so hat dieser im allgemeinen eine Vorstellung davon, welche Meßwerte zunächst erhoben werden sollten und welche nicht. Einzelne Messungen bezieht ein Experte erst dann in seine Überlegungen ein, falls eine gewisse Evidenz für einen bestimmten Fehler vorliegt. Diese Beobachtung liegt dem Konzept von Defaultwerten für Symptome zugrunde. Symptome mit Defaultwert erhalten, solange keine genaue Meßung vorgenommen wurde, den im Default angegebenen Meßwert. Beim Fallvergleich werden diese Symptome dann so behandelt, als wäre eine entsprechende Meßung bereits durchgeführt worden. Damit werden Fallbeipiele, die Symptome mit einem dem Default entsprechenden Symptomwert in ihrer Situationsbeschreibung enthalten, in ihrer Ähnlichkeit zur aktuellen Situation verstärkt. Die Ähnlichkeit von Fallbeispielen, in denen explizit ein dem Defaultwert widersprechender Meßwert gefordert wird, wird jedoch nicht verändert. Verglichen mit dem im letzten Abschnitt eingeführten pathologischen Meßwerten entsprechen Defaultwerte eher dem Normalverhalten eines technischen Systems. Typische Defaults sind etwa *Leitungen dicht* oder auch *Netzspannung vorhanden.* Diese Meßwerte sind für eine grobe Bestimmung der Fehlerursache, im Gegensatz zu pathologischen Meßwerten, im allgemeinen zunächst wenig informativ.

[6] in [Pup90] werden solche Symptomwerte als *abnormal* bezeichnet

Im Verlauf des Diagnoseprozesses nimmt aber ihre Bedeutung für die Diskriminierung von Diagnosen zu. Ist ein Symptom S_i mit Default für die Diskriminierung zwischen konkurierenden Diagnosen D_j und D_l von Bedeutung, so wird S_i im Laufe des Diagnoseprozesses erhoben. Wird dann ein dem Defaultwert widersprechender Meßwert festgestellt, so müssen alle Fallbeispiele, in denen der entsprechende Default angenommen wurde, erneut aufgegriffen und die Ähnlichkeit neu bewertet werden. Die Defaults werden somit zurückgenommen und die von der Änderung betroffenen Fallbeispiele scheiden damit möglicherweise als potentielle Lösungskandidaten aus. Diese Vorgehensweise besitzt den Vorteil, daß ein fallbasiertes Diagnosesystem bereits nach der Eingabe von nur wenigen Symptomen potentielle Lösungskandidaten präsentieren kann, die dann gezielt weiter überprüft werden können.

3.2 Das ratio model nach Tversky

In der Diagnostik stehen, im Gegensatz zu üblichen Anwendungen von Ähnlichkeitsmaßen, in denen vorwiegend kardinale bzw. ordinale Daten in Form von *vollständigen* Merkmalsvektoren verarbeitet werden, hauptsächlich nominale (symbolische) Daten im Mittelpunkt. Für die Diagnostik benötigen wir daher ein Maß, das symbolische, unter Umständen voneinander abhängige Merkmale, sowie nur partiell definierte Merkmalsvektoren verarbeiten kann. Für diagnostische Aufgabenstellungen ist das folgende, *ratio model* genannte Modell nach Tversky [Tve77] besonders geeignet. Das *ratio model* geht von einer Beschreibung der Merkmale der zu vergleichenden Objekte aus. Ein Merkmal hat dabei die Form: *Attribut* hat *Wert*. Auf die Diagnose übertragen ist also *Meßpunkt* hat *Meßwert* ein Merkmal im Sinne des *ratio models*. Seien A und B die Beschreibungen der entsprechenden Objekte a und b, so ist das *ratio model* definiert als:

$$\mathrm{sim(a,b)} = \frac{\mathrm{f\ (A \cap B)}}{\mathrm{f\ (A \cap B)} + \alpha\, \mathrm{f\ (A - B)} + \beta\, \mathrm{f\ (B - A)}} \qquad \alpha, \beta \geq 0$$

Dieses Modell ist genaugenommen kein konkretes Maß, sondern ein Schema für die Definition einer ganzen Schar von konkreten Ähnlichkeitsmaßen. Durch die Wahl von α, β und f können eine Vielzahl von unterschiedlichen Maßen realisiert werden. Beispiel: Bei $\alpha = \beta = 1, f :=$ *Kardinalität der betrachteten Mengen*, entspricht die Ähnlichkeit der Anzahl gemeinsamer Merkmale geteilt durch die Anzahl aller Merkmale. Durch eine geeignete Instantiierung von f können wir das Modell leicht an Anforderungen einer speziellen Domäne anpassen. Insbesondere bietet die Funktion f einen Ansatzpunkt, das im letzten Abschnitt beschriebene Wissen bei der Berechnung der Ähnlichkeit zu berücksichtigen.

3.3 Relevanz von Symptomen für eine Diagnose

Wenn wir Fallbeispiele als „*Protokolle des realen Problemlöseverhaltens von Diagnoseexperten*“ auffassen, so stellt sich das Problem, daß die durch die Beobachtung eines Experten gewonnenen Fälle möglicherweise inkonsistente, unvollständige oder auch überflüssige Informationen enthalten, d.h. wir erhalten nicht genau die für die Klassifikation relevanten Symptome. Damit die Klassifikationsfähigkeit eines fallbasierten Systems durch derartig *verrauschte* Informationen nicht beeinträchtigt wird, ist es für den Fallvergleich nötig, die Relevanz eines Symptoms S_i für eine bestimmte Diagnose D_j zu ermitteln. Der Fallvergleich findet dann nur auf der Basis dieser relevanten Symptome statt. Für Tversky [Tve77] ist die Bestimung von relevanten Eigenschaften eine Vorbedingung für die Anwendung eines Ähnlichkeitsmaßes. Er schreibt dazu: *“When faced with a particular task (e.g. identification or similartiy assessment) we extract and compile from our data base a limited list of relevant features on the basis of which we perform the required task.“* In der fallbasierten Diagnostik kann Modellwissen Anhaltspunkte für die Relevanz von Symptomen für eine bestimmte Diagnose liefern. Ein solcher Ansatz entspricht dem in [JWM92] als *Goal-Driven Similarity Assessment* bezeichneten Vorgehen. Auf der anderen Seite können wir versuchen, die Relevanz eines Symptoms für eine Diagnose inkrementell anzunähern, d.h. Häufigkeit impliziert Relevanz. Ist das benötigte Hintergrundwissen vollständig vorhanden, so bietet sich die erste Strategie an. In allen anderen Situationen empfiehlt sich eine gemischte Strategie.

3.3.1 Ein Ansatz zur Beschreibung

Die Relevanz eines Symptoms S_i für eine Diagnose D_j kann durch die Angabe eines spezifischen Wichtungsfaktor w_{ij} formuliert werden. Je stärker der entsprechende Wichtungsfaktor für ein Symptom S_i in einem Fall C_k ist, umso höher wird bei einer Übereinstimmung der Symptomwerte in Fall und aktueller Situation der entsprechende Ähnlichkeitswert. Bei widersprüchlichen Symptomwerten wird der Ähnlichkeitswert des Falles C_k dann auch stärker verringert als bei einem kleineren Gewicht. Zur Repräsentation der Wichtungsfaktoren w_{ij} definieren wir eine $n \times m$ Relevanzmatrix $\mathcal{R}$, wobei n die Anzahl der im System definierten Symptome S_i und m die Anzahl der unterschiedlichen Diagnosen D_j ist. Der Spaltenvektor $\vec{d_j} = (w_{1j}, w_{2j}, \ldots, w_{nj})$ beschreibt dabei die Relevanz der Symptome S_i unter der Voraussetzung, daß die Diagnose D_j vorliegt. In Tverskys *ratio model* haben wir, falls f entsprechend gewählt wird, durch die Einführung einer Relevanzmatrix $\mathcal{R} = [w_{ij}]$ für jede im System repräsentierte Diagnose D_j ein eigenes Ähnlichkeitsmaß sim_j definiert. Das Ähnlichkeitsmaß ist dann eine compilierte Repräsentation des Wissens, welche Informationen für die Klassifikation von D_j relevant sind. Das Ähnlichkeitsmaß sim_j für Fallbeispiele mit der Diagnose D_j wird dabei durch den entsprechenden Spaltenvektor $\vec{d_j}$ charakterisiert.

3.3.2 Bestimmung der Relevanz

Wird ein wissensbasiertes Verfahren wie *Goal-Driven Similarity Assessment* [JWM92] verwendet, so kann mit dem Ergebnis dieses Verfahrens die Relevanzmatrix initialisiert werden. Ist dies nicht möglich, so muß versucht werden, die w_{ij} durch ein Lernverfahren zu bestimmen. Wird ein neuer Fall C_l mit einer bisher unbekannten Diagnose D_l in die Fallbasis aufgenommen, so wird der Spaltenvektor $\vec{d_l}$ zunächst mit den relativen Häufigkeiten der Symptome $S_i \in C_l.Sit$ in FB initialisiert. Durch das in Abschnitt 3.5 näher beschriebene Lernverfahren werden dann die Wichtungsfaktoren w_{il} iterativ angepaßt. Dies kann im Laufe der Anwendung oder im Rahmen einer gesonderten Trainingsphase erfolgen. Im Laufe des fallbasierten Diagnoseprozesses ist somit für jede Diagnose D_j eine von alternativen Diagnosen D_i unabhängige Verbesserung des Ähnlichkeitsbegriffs möglich. Verrauschte Daten können so schrittweise ausgefiltert werden. Da die Relevanz eines Symptoms für eine Diagnose, bedingt durch die Verwendung eines Ähnlichkeitsmaßes, in beiden Ansätzen durch die Angabe von w_{ij} repräsentiert wird, ist die Kombination beider Ansätze möglich. Nichtvorhandenes Wissen kann so durch eine entsprechende Menge an Erfahrung ausgeglichen werden.

3.4 Ein dynamisches Ähnlichkeitsmaß

Das hier verwendete Ähnlichkeitsmaß ist direkt dem *ratio model* nach Tversky entlehnt. Wir haben es jedoch um die vorgestellten Konzepte wie der lokalen Ähnlichkeitsmaße $\omega_i(v_j, v_k)$ für Symptomwerte, um Default- und pathologische Symptomwerte sowie die unterschiedlichen Relevanzen w_{ij} von Symptomen S_i für Diagnosen D_j erweitert. Durch diese Erweiterungen haben wir die in Tverskys Modell nicht näher spezifizierte Funktion f für Diagnoseaufgaben konkretisiert. Für die folgende Definition des Ähnlichkeitsmaßes sei gegeben:

Symptomatik Sit und ein Fallbeispiel $C_k := (Sit_k, D_j, \delta_j)$
w_{ij} sei die Gewichtung des Symptoms S_i für alle Fälle mit der Diagnose D_j
ω_i sei das für alle Meßwerte $v_l \in W_i$ des Symptoms S_i definierte lokale Ähnlichkeitsmaß
$v_i^{C_k}$ sei der in Sit_k und v_i^{Sit} sei der in Sit vorliegende Meßwert für das Symptom S_i

$$\begin{array}{lcll} E & := & \{S_i \mid S_i \in Sit_k \cap Sit\} & \textit{übereinstimmende Meßpunkte} \\ U & := & \{S_i \mid S_i \in Sit_k \wedge S_i \notin Sit\} & \textit{unbekannte Meßwerte} \\ A & := & \{S_i \mid S_i \notin Sit_k \wedge S_i \in Sit \wedge s_i \; ist \; pathologisch\} & \textit{zusätzliche Meßwerte} \end{array}$$

Sollen unterschiedliche Relevanzen w_{ij} und die Ähnlichkeit zwischen Symptomwerten $\omega_i(v_i^{C_k}, v_i^{Sit})$ bei der Ähnlichkeitsbewertung berücksichtigt werden, so müssen wir das Maß um diese Faktoren erweitern. Die Entscheidung ob ein Symptomwert in Situation Sit und dem gegebenen Fall

$C_k.Sit$ übereinstimmt oder nicht, erfolgt dann über das jeweilige lokale Ähnlichkeitsmaß ω_i.

$$\mathcal{E} := \sum_{S_i \in E} w_{ij}\,\omega_i(v_i^{C_k}, v_i^{Sit}) \quad \mathcal{W} := \sum_{S_i \in E} w_{ij}\,(1 - \omega_i(v_i^{C_k}, v_i^{Sit})) \quad \mathcal{U} := \sum_{S_i \in U} w_{ij} \quad \mathcal{A} := |A|$$

Das verwendete Schema für Ähnlichkeitsmaße hat unter Berücksichtigung der gerade definierten, gewichteten Merkmalsmengen die folgende Form:

$$sim(C_k.Sit, Sit) = \frac{\alpha\,\mathcal{E}}{\alpha\,\mathcal{E} + \beta\,\mathcal{W} + \gamma\,\mathcal{U} + \eta\,\mathcal{A}} \quad \alpha, \beta, \eta, \gamma \geq 0$$

Ein wesentlicher Vorteil des hier vorgestellten Maßes, der sich auch in unseren empirischen Untersuchungen bestätigt hat, ist die im Vergleich zu anderen Maßen große Robustheit gegenüber Informationsverlust (vgl. auch Diagramm in Abschnitt 2.4). Durch die Wahl der Parameter $\alpha, \beta, \eta, \gamma$ können wir das Diagnoseverhalten des Systems in gewissen Grenzen steuern. Für die Diagnostik hat sich in unseren Experimenten eine eher *pessimistische* Vorgehensweise mit $\alpha = 1, \beta = 2, \gamma = 0.5, \eta = 1$ bewährt. Unterschiede in den Meßwerten werden so relativ stark bewertet, während noch fehlende Symptomwerte nicht überbewertet werden.

3.5 Anpassung des Ähnlichkeitsmaßes

Wird ein Fall als mögliche Lösung vorgeschlagen, so muß nach Abschnitt 3.4 gelten:

$$sim(C_k.Sit, Sit) > \delta_k, \text{ d.h. } sim(C_k.Sit, Sit) = \delta_k + \Delta \text{ mit } \Delta > 0$$

Ist aber eine Fehlklassifikation eingetreten, so muß eigentlich $sim(C_k.Sit_k) \leq \delta$ gelten (vgl. Gleichungen in 1). Dies kann erreicht werden, indem die Gewichtsfaktoren der Symptome des Falles C_k (d.h. das Ähnlichkeitsmaß für die Diagnose D_j) entsprechend verändert werden. Hierzu sind folgende Überlegungen notwendig: Die aktuelle Situation Sit und der Fall C_k besitzen eine Reihe von gemeinsamen Symptomwerten, sowie eine Reihe von unterscheidenden Symptomwerten. Tritt nun eine Fehlklassifikation auf, so bedeutet dies:

- Die Gewichtung $\mathcal{E}$ der *übereinstimmenden* Symptomwerte aus E war zu hoch.
- Die Gewichtung $\mathcal{W}$,$\mathcal{U}$ der *unterscheidenden* Symptomwerte aus $E \cup U$ war zu niedrig .

Wir suchen nun eine Änderung der Parameter $\mathcal{E} - z, \mathcal{W} + x$ und $\mathcal{U} + y$, mit $x, y, z \geq 0$, so daß dieser Fall C_k gerade noch abgelehnt wird, d.h. es gilt dann $sim(C_k.Sit, Sit) = \delta$:

$$\frac{\alpha\,\mathcal{E}}{\alpha\,\mathcal{E} + \beta\,\mathcal{W} + \gamma\,\mathcal{U} + \eta\,\mathcal{A}} = \delta + \Delta \qquad \frac{\alpha\,(\mathcal{E} - \mathrm{z})}{\alpha\,(\mathcal{E} - \mathrm{z}) + \beta\,(\mathcal{W} + \mathrm{x}) + \gamma\,(\mathcal{U} + \mathrm{y}) + \eta\,\mathcal{A}} = \delta \tag{1}$$

Die Werte für x, y, z sind nun in Abhängigkeit vom Fehler Δ zu bestimmen. Damit erhalten wir:

$$\frac{\alpha\,\mathcal{E}}{\alpha\,\mathcal{E} + \beta\,\mathcal{W} + \gamma\,\mathcal{U} + \eta\,\mathcal{A}} - \Delta = \frac{\alpha\,(\mathcal{E} - \mathrm{z})}{\alpha\,(\mathcal{E} - \mathrm{z}) + \beta\,(\mathcal{W} + \mathrm{x}) + \gamma\,(\mathcal{U} + \mathrm{y}) + \eta\,\mathcal{A}} \tag{2}$$

Die Zu- und Abnahme des Symptomgewichts ist dabei *proportional* im Falle der Übereinstimung und *antiproportional* zum bestehenden Symptomgewicht im Falle der Unterscheidung, d.h.

- Übereinstimmende Symptome aus E mit hohem Gewicht werden stark vermindert.
  Sie haben einen hohen Beitrag zur Fehlklassifikation geleistet.
- Übereinstimmende Symptome aus E mit niedrigem Gewicht werden nur schwach vermindert.
- Unterscheidende Symptome aus $E \cup U$ mit niedrigem Gewicht erhalten einen großen Gewichtszuwachs.
  Sie wurden bei der Ähnlichkeitsbeurteilung, die zur Fehlklassifikation führte, nicht berücksichtigt.
- Unterscheidende Symptome aus $E \cup U$ mit hohem Gewicht erhalten einen niedrigen Gewichtszuwachs.

Die nötigen Gewichtsänderungen lassen sich nun aus Gleichung 2 bestimmen. Die genaue Formel ist hier jedoch von eher untergeordnetem Interesse. Bei Bedarf kann sie in [Wes91] nachgeschlagen werden. Entscheidend ist hier die Tatsache, daß, ausgehend von einer Fehlklassifikation, der Fehler Δ bestimmt und die Parameter w_{ij} für das Ähnlichkeitsmaß sim_j einer spezifischen Diagnose D_j entsprechend geändert werden können.

3.6 Steuerung des Testerhebungsprozesses

Während wir uns bisher lediglich auf *einen Durchlauf* der Klassifikationskomponente eines fallbasierten Diagnosesystems konzentriert haben, betrachten wir im weiteren Verlauf den gesamten Prozeß des diagnostischen Problemlösens, d.h. die Schleife: *Wiederhole Klassifikation und Testerhebung bis eine Diagnose möglich ist.* Aus dieser Sicht betrachtet ist es sinnvoll, einem fallbasierten Diagnosesystem einen Abbruch dieser Schleife zu ermöglichen, falls sich in der Fallbasis kein geeigneter Lösungskandidat findet. Das System würde ansonsten immer versuchen, schrittweise *alle möglichen Meßwerte* zu erheben. Zu diesem Zweck erweitern wir die Falldefinition um einen weiteren Schwellwert $0 \leq \epsilon < \delta \leq 1$. Der Schwellwert ϵ hat dabei die folgende Semantik: Für einen Fall $C_k = (Sit, D, \delta, \epsilon)$ werden nur dann weitere Meßwerte erhoben, falls $sim(C_k.Sit, Sit) \geq \epsilon$ gilt.

4 Das PATDEX System

Das PATDEX–System (vgl. [AKM+89, Wes91, AW91, RW91]) ist ein auf der Basis des vorgestellten Ansatzes in SMALLTALK-80 implementiertes fallbasiertes Diagnosesystem. Es ist in die MOLTKE Werkbank zur Diagnose technischer Systeme [AMWT92] integriert und kann mit den in der Werkbank vorhandenen Komponenten kooperieren [AMTW91].

Eingabe: aktuelle Symptomatik Sit
Ausgabe: Diagnose D oder Fail
Seiteneffekte geänderte Fallbasis FB, geändertes Ähnlichkeitsmaß sim

1. Bestimme alle aus der gegebenen Symptomatik Sit ableitbaren Symptomwerte und pathologische bzw. Defaultwerte. Erweitere Sit entsprechend.
2. Suche den in der Fallbasis zur aktuellen Symptomatik ähnlichsten Fall C_k, d.h. $\forall\ C_i \in FB\ sim(C_k.Sit, Sit) \geq sim(C_i.Sit, Sit)$.
3. Ist der Fall C_k nicht *minimal* ähnlich, d.h. $sim(C_k.Sit, Sit) < \epsilon_k$, dann **STOP** mit *Keine Diagnose möglich.*
4. Ist der Fall C_k ausreichend ähnlich, d.h. $sim(C_k.Sit, Sit) > \delta_k$, dann gebe die Diagnose $C_k.D$ als Lösungsvorschlag an den Benutzer aus.
   (a) Ist die Diagnose $C_k.D$ korrekt, dann nehme gegebenenfalls den neuen Fall $C := (Sit, C_k.D, C_k.\delta, C_k.\epsilon)$ in die Fallbasis FB auf. **STOP**
   (b) Ist die Diagnose $C_k.D$ nicht korrekt, so ändere das Ähnlichkeitsmaß sim, so daß $sim(C_k.Sit, Sit) = \delta_k$ gilt. Lösche (temporär) alle Fälle $C_i \in FB$ mit $C_i.D = D_k$ aus der Fallbasis FB. Weiter bei Schritt 5.
5. Ansonsten bestimme das nächste zu erhebende Symptom S_i mit noch unbekanntem Symptomwert.
6. Frage den Benutzer nach dem Meßwert für Symptom S_i und erweitere Sit um den entsprechenden Symptomwert, d.h. $Sit := Sit \cup S_i$. Weiter bei Schritt 1.

Das PATDEX-System kann sowohl stand-alone, als auch im Rahmen der MOLTKE-Werkbank eingesetzt werden. Damit unterstützt es *beide* im Abschnitt 2 angegebenen Szenarios zum Einsatz fallbasierter Systeme in der Diagnostik. In vielen Aspekten geht die konkrete Implementierung über den in dieser Arbeit vorgestellten Rahmen hinaus. Eine Auswahl dieser Aspekte soll im folgenden kurz vorgestellt werden:

Strategiefälle: PATDEX unterstützt beide Teilaufgaben der Diagnostik, d.h. sowohl die fallbasierte Klassifikation, als auch die fallbasierte Unterstützung der Testerhebung durch entsprechende *Strategiefälle* (vgl. Abschnitt 2). Das Konzept der *Strategiefälle* wird im Rahmen des PATDEX-Systems verwendet, um heuristisches Strategiewissen eines Serviceexperten in der Form von Fallbeispielen zu repräsentieren. Ein Strategiefall ist in PATDEX von der Form: $C_{Strat} := (Sit, S_i)$, d.h. er steuert im Diagnoseprozeß das nächste zu erhebende Symptom S_i.

Lösungsanpassung: In Kombination mit einer modellbasierten Komponente zur Lösungsanpassung MoCas [PWW92] kann mit Patdex auch ein *fallmodifizierender Ansatz* zur Diagnostik realisiert werden (vgl. Def. 5). MoCas kann auf der Basis des Maschinenmodells gefundene Diagnosen anpassen und wird daher dann eingesetzt, wenn durch einen *fallvergleichenden Ansatz* keine zufriedenstellende Diagnose gefunden wird. MoCas setzt auf das im Rahmen des Moltke-Projektes entwickelte Make System [Reh91] (modellbasierte Generierung von Moltke-Wissensbasen) auf. Der Einsatz von MoCas ist hierdurch auf komponentenorientierte, hierarchisch und statisch modellierbare technische Systeme mit zentraler Steuerung beschränkt.

Retrieval: Basis des Patdex Systems ist die objektorientierte Datenbank GemStone zur Speicherung und Verwaltung der Falldaten [Öc92]. Um eine effektive Suche nach ähnlichen Fallbeispiele zu ermöglichen, werden aufbauend auf der Datenbank k-d Bäume als Zugriffstruktur eingesetzt [FBF77]. Die Suche nach ähnlichen Fallbeispielen, basierend auf dem vorgestellten Ähnlichkeitsmaß, kann in Abhängigkeit von der Anzahl n der vorhandenen Fälle mit einem durchschnittlichen Aufwand der Größenordnung $O(\log n)$ erfolgen.

5 Diskussion

Obwohl im letzten Abschnitt Aspekte einer konkreten Implementierung angesprochen wurden, haben die in dieser Arbeit vorgestellten Überlegungen einen eher allgemeinen Charakter. Bei einigen Punkten, z.B. Anpassung des Ähnlichkeitsmaßes, sind aber durchaus alternative Vorgehensweisen denkbar.

Wird Patdex in (noch) unstrukturierten Anwendungsdomänen eingesetzt in denen die Ähnlichkeit von Problemstellungen nur näherungsweise bestimmt werden kann *(erstes Szenario)*, so kann Patdex in der Anwendungsphase dynamisch um vorhandenes Hintergrundwissen erweitert werden. Die Einschränkung des Ansatzes auf technische Diagnosedomänen liegt in der Tatsache begründet, daß entsprechendes Wissen zur Unterstützung des Fallvergleichs in diesen Anwendungsdomänen meistens sehr leicht zugänglich ist. Das dynamische Ähnlichkeitsmaß von Patdex ermöglicht dann eine schrittweise Annäherung an den durch die Anwendungsdomäne vorgegebenen Ähnlichkeitsbegriff.

Wird ein fallbasierter Ansatz im Kontext eines Gesamtsystems, welches unterschiedliche Problemlöser integriert, gesehen *(zweites Szenario)*, so relativiert sich unter diesem Gesichtspunkt die Frage nach dem nötigen Aufwand zur Wissensakquisition, da das für eine wissensbasierte Unterstützung von Patdex benötigte Wissen im Rahmen eines Gesamtsystems bereits vorhanden ist. Ein solcher Ansatz zur Integraton von regel-, modell- und fallbasiertem Wissen wird z.B. in der Moltke–Werkbank [AMTW91, AMWT92] zur Diagnose technischer Systeme verfolgt.

Trotz der nicht in allen Situationen gesicherten diagnostischen Kompetenz von fallbasierten Ansätzen zeigen positive Reaktionen von Anwendern, daß ein integrierter Ansatz, wie er im Patdex System realisiert wird, für die Lösung von diagnostischen Aufgabenstellungen in technischen Domänen sinnvoll ist. Der Einsatz von Patdex bietet sich dort an, wo diagnostisches Wissen bereits in der Form vorliegt, in der es das System direkt verarbeiten kann, also in der Form von Diagnosefällen. Man denke beispielsweise an alle Formen von technischen Prüfständen. Die anfallenden Prüfprotokolle können direkt als Basis für ein fallbasiertes Diagnosesystem wie Patdex dienen.

Danksagung
Ich danke M.M. Richter, K.P. Jantke, K.D. Althoff, G. Derwand, D. Janetzko, F. Maurer, K. Nökel, J. Paulokat, Th. Schult und M. Stadler, sowie der gesamten Arbeitsgruppe in Kaiserslautern für viele Diskussionen und hilfreiche Anmerkungen zu früheren Versionen dieser Arbeit.

Literatur

[Aam91] Agnar Aamodt. *A Knowledge-Intensive, Integrated Approach to Problem Solving and Sustained Learning.* Dissertation, University of Trondheim, 1991.

[AKA91] David W. Aha, Dennis Kibler und Marc K. Albert. *Instance-Based Learning Algorithms. Machine Learning*, 6:37–66, 1991. March 1991.

[AKM+89] Klaus-Dieter Althoff, Sabine Kockskämper, Frank Maurer, Michael Stadler und Stefan Wess. Ein System zur fallbasierten Wissensverarbeitung in technischen Diagnosesituationen. In ÖGAI, Hrsg, *Proceedings 5. Österreichische Artificial-Intelligence Tagung*, Seiten 65–70. Springer-Verlag, 1989.

[AMTW91] Klaus-Dieter Althoff, Frank Maurer, Ralph Traphöner und Stefan Wess. *Die Lernkomponente der* MOLTKE-*Werkbank zur Diagnose technischer Systeme. KI - Künstliche Intelligenz*, (1), 1991.

[AMWT92] Klaus-Dieter Althoff, Frank Maurer, Stefan Wess und Ralf Traphöner. MOLTKE - an integrated workbensch for fault diagnosis in engineering systems. In S. Hashemi, J.P. Marciano und G. Gouarderes, Hrsg, *Proc. 4th international conference Artificial Intelligence & Expert Systems Applications (EXPERSYS-92)*, Paris, Oktober 1992. i.i.t.t international.

[AW91] Klaus-Dieter Althoff und Stefan Wess. Case-Based Knowledge Acquisition, Learning and Problem Solving for Diagnostic Real World Tasks. In Duncan Smeed, Hrsg, *Proceedings of the 5th European Knowledge Acquisition Workshop EKAW'91.* GMD-Bericht, 1991.

[AWBS+92] K.-D. Althoff, S. Wess, B. Bartsch-Spörl, D. Janetzko, F. Maurer und A. Voss. *Fallbasiertes Schliessen in Expertensystemen: Welche Rolle spielen Fälle für wissensbasierte Systeme? KI – Künstliche Intelligenz*, (4), Dezember 1992.

[Das90] Belur Dasarathy. *Nearest Neighbor Norms: NN Pattern Classification Techniques.* IEEE Computer Society Press, 1990.

[FBF77] J.H. Friedman, J.L. Bentley und R.A. Finkel. *An algorithm for finding best matches in logarithmic expected time. ACM Trans. math. Software*, 3:209–226, 1977.

[Jan92] Klaus P. Jantke. Case-Based Reasoning and Inductive Inference. Gosler Report 08/92, University of Leipzig, Leipzig, Germany, 1992.

[JWM92] Dietmar Janetzko, Stefan Wess und Erica Melis. Goal-Driven Similarity Assessment. In Hans-Jürgen Ohlbach, Hrsg, *Proc. German Workshop on AI (GWAI'92).* Springer Verlag, 1992. (in Vorbereitung).

[Kot88] P. Koton. Reasoning about Evidence in Causal Explanation. In Janet L. Kolodner, Hrsg, *Proceedings Case-Based Reasoning Workshop*, Seiten 260–170, Morgan Kaufmann Publishers, 1988.

[Öc92] Hannes Öchsner. *Mehrdimensionale Zugriffspfadstrukturen für das ähnlichkeitsbasierte Retrieval von Fällen* . Diplomarbeit, Universität Kaiserslautern, Fachbereich Informatik, 1992.

[PG91] Frank Puppe und Klaus Goos. Improving Case-Based Classification with Expert Knowledge. In Thomas Christaller, Hrsg, *Proceedings of the German Workshop on Artificial Intelligence 1991*, Berlin, September 1991. Springer-Verlag.

[Pup90] Frank Puppe. *Problemlösungsmethoden in Expertensystemen.* Studienreihe Informatik. Springer Verlag, 1990.

[PWW92] Gerd Pews, Frank Weiler und Stefan Wess. Bestimmung der Ähnlichkeit in der fallbasierten Diagnose mit simultationsfähigen Maschinenmodellen. In K-D. Althoff, S. Wess, B. Bartsch-Spörl und D. Janetzko, Hrsg., *Ähnlichkeit von Fällen in Systemen des fallbasierten Schließens*, SEKI-Report, Workshop Universität Kaiserslautern, SFB 314, 25.-26. Juni 1992.

[Reh91] Robert Rehbold. *Integration modellbasierten Wissens in technische Diagnostik-Expertensysteme.* Dissertation, Fachbereich Informatik, Universität Kaiserslautern, 1991.

[Ric92] Michael M. Richter. Classification and learning of similarity measures. In *Proc. der 16. Jahrestagung der Gesellschaft für Klassifikation e.V.* Springer Verlag, 1992.

[RW91] Michael M. Richter und Stefan Wess. Similarity, Uncertainty and Case-Based Reasoning in PATDEX. In Robert S. Boyer, Hrsg, *Automated Reasoning*, Essays in Honor of Woody Bledsoe, Seiten 249–265. Kluwer Academic Publishing, 1991.

[SW86] Craig Stanfill und David Waltz. *Toward Memory-Based Reasoning. Communications of the ACM*, 29(12):1213–1229, 1986.

[Tve77] A. Tversky. *Features of Similarity. Psychological Review*, 84:327–352, 1977.

[Wes91] Stefan Wess. PATDEX/2: Ein System zum adaptiven, fallfokussierenden Lernen in technischen Diagnosesituationen. SEKI-Working Paper SWP91/01, Fachbereich Informatik, Universität Kaiserslautern, Januar 1991.