Pure and Applied Fixed-Point Logics

Stephan Kreutzer*
RWTH Aachen

1 Einleitung

In der Entwicklung der Informatik, besonders der theoretischen Informatik, haben formale Logiken eine bedeutende Rolle gespielt und als Grundlage für Abfrage- oder Spezifikationssprachen in vielen Gebieten der Informatik gedient. Ein wichtiges Beispiel für diesen Einfluß bietet die Prädikatenlogik (FO) als Basis des relationalen Kalküls im Datenbankbereich. Umgekehrt haben die in der Informatik gestellten Forderungen an Abfrage- bzw. Spezifikationssprachen wiederum die Entwicklung neuer Arten von Logiken beeinflußt. Insbesondere die in der Informatik typische Anforderung, Mengen oder Relationen durch *Rekursion* oder *Induktion* definieren zu können, führte zur Entwicklung einer ganzen Klasse von Logiken, den sogenannten Fixpunktlogiken, im Gegensatz zu den in der klassischen Modelltheorie bzw. mathematischen Logik vorherrschenden Logiken wie der Prädikatenlogik, den infinitären Logiken oder der Logik zweiter Stufe.

In Fixpunktlogiken werden rekursive Definitionen durch verschiedene Formen von Fixpunktkonstruktionen formalisiert. Logiken dieser Art findet man z.B. im Datenbankbereich, in der computerunterstützten Verifikation, sowie, eher theoretisch motiviert, in der deskriptiven Komplexitätstheorie. So unterschiedlich die Logiken in den einzelnen Bereichen auch sein mögen, die Art und Weise, in der Fixpunkte integriert werden, folgt oft einem gemeinsamen Muster.

Betrachten wir dazu zunächst eine beliebige prädikatenlogische Formel $\varphi(R, \overline{x})$ mit einer freien Relationsvariable R und k freien erststufigen Variablen $\overline{x}$. Auf einer gegebenen Struktur $\mathring{A}$ induziert eine solche Formel eine Abbildung F_φ, die eine Menge $P \subseteq A^k$ auf die Menge $\{\overline{a} : (\mathring{A}, P) \models \varphi[\overline{a}]\}$ der Tupel abbildet, die die Formel φ erfüllen, wenn deren Relationsvariable R durch die Menge P interpretiert wird. Rekursive Definitionen werden nun modelliert, indem Fixpunkte solcher Abbildungen betrachtet werden, d.h. Mengen P, für die $F_\varphi(P) = P$ gilt. Die für diese Anwendung wohl wichtigste Klasse von Fixpunkten sind die *kleinsten* Fixpunkte *monotoner* Abbildungen, also solcher Abbildungen F_φ, für die $F_\varphi(X) \subseteq F_\varphi(Y)$ für alle Mengen $X \subseteq Y$ gilt.

Kleinste Fixpunkte werden gewöhnlich wie folgt in Logiken integriert: Ist die betrachtete Ausgangsformel $\varphi(R, \overline{x})$ positiv in R, d.h. kommt R im Bereich einer geraden Anzahl von Negationen vor, so ist die durch φ induzierte Abbildung monoton. Monotone Abbildungen besitzen immer einen eindeutig bestimmten kleinsten Fixpunkt, der mit $\mathbf{lfp}(F_\varphi)$ bezeichnet wird und als Schnitt aller Fixpunkte der Abbildung definiert ist, d.h. es gilt $\mathbf{lfp}(F_\varphi) := \bigcap\{P : F_\varphi(P) = P\}$. Dies bildet die Basis der *kleinsten Fixpunktlogik*.

* Stephan Kreutzer, LuFG Mathematische Grundlagen der Informatik, RWTH Aachen, 52066 Aachen, Email: kreutzer@i7.informatik.rwth-aachen.de, Homepage: www-mgi.informatik.rwth-aachen.de/∼kreutzer

1.1 Definition. *Die* kleinste Fixpunktlogik (LFP) *ist definiert als die Erweiterung der Prädikatenlogik um Formeln ψ der Gestalt $[\mathbf{lfp}_{R,\overline{x}}\, \varphi(R, \overline{x})](\overline{t})$, wobei $\overline{t}$ ein Tupel von Termen und φ eine* LFP-*Formel ist, die positiv in R sein muß. Sei $\mathring{A}$ eine Struktur mit Interpretationen für die freien Variablen in $\overline{t}$ und den frei in φ, aber nicht in $\overline{x}$ vorkommenden Variablen. Die Formel ψ gilt in $\mathring{A}$, wenn $\overline{t}$ im kleinsten Fixpunkt der durch φ definierten Abbildung F_φ enthalten ist.*

Als Beispiel für eine kleinste Fixpunktformel betrachten wir folgende Formel

$$\varphi(R, x) := x \in \Sigma \vee \exists u, v \in R\,[x = \text{“}(u \cdot v)\text{”} \vee x = \text{“}(u \cup v)\text{”} \vee x = \text{“}(u)^*\text{”}].$$

Bezeichnet Σ ein nicht-leeres Alphabet, so definiert diese Formel über einer geeigneten Struktur die Worte x, die entweder aus einem Buchstaben des Alphabets Σ bestehen oder aber die Form $(u \cdot v)$, $(u \cup v)$ oder $(u)^*$ besitzen, wobei u und v zwei Worte aus R sind. Der kleinste Fixpunkt der durch diese Formel induzierten Abbildung F_φ, d.h. die kleinste Menge M, die unter F_φ abgeschlossen ist, ist die Menge der regulären Ausdrücke über dem Alphabet Σ. Das heißt, die Formel $[\mathbf{lfp}_{R,x}\, \varphi(R, x)](x)$ definiert die regulären Ausdrücke über Σ.

Neben der oben angegebenen Definition kleinster Fixpunkte als Schnitt aller Fixpunkte einer monotonen Abbildung F gibt es noch eine andere, induktive Definition. Dazu betrachten wir die folgende transfinite Sequenz $(R^\alpha)_{\alpha \in \mathrm{Ord}}$ von Mengen :

$$\begin{aligned} R^0 &:= \varnothing \\ R^{\alpha+1} &:= F(R^\alpha) \\ R^\lambda &:= \bigcup_{\beta < \lambda} R^\beta \qquad \text{für Limesordinale } \lambda \end{aligned} \tag{1}$$

Da die Abbildung F monoton ist, ist diese Sequenz aufsteigend, d.h. es gilt $R^\alpha \subseteq R^{\alpha+1}$ für alle α. Daher muß es ein Ordinal α geben, so daß $R^\alpha = R^\beta$ für alle $\beta > \alpha$, d.h. die Sequenz erreicht einen Fixpunkt $R^\infty := R^\alpha$. Aus dem Satz von Knaster und Tarski folgt, daß der kleinste Fixpunkt einer monotonen Abbildung genau der induktive Fixpunkt obiger Sequenz ist, d.h. $\mathbf{lfp}(F) = R^\infty$. Somit sind beide Definitionen äquivalent und können als Basis der kleinsten Fixpunktlogik verwendet werden. In Anwendungen der Informatik ist die induktive Definition oft intuitiver, da sie dem schrittweisen Aufbau von Mengen durch Rekursion bzw. Induktion folgt. Betrachten wir das Beispiel der Definition regulärer Ausdrücke in LFP, so entsprechen die einzelnen Stufen der Fixpunktinduktion dem Aufbau komplexer regulärer Ausdrücke aus einfachen. Des weiteren demonstriert dieses Beispiel eine Fixpunktinduktion, deren Fixpunkt nicht nach endlich vielen Schritten, sondern erst nach dem ersten Limesschritt ω erreicht wird.

Neben den kleinsten wurden noch verschiedene andere Arten von Fixpunktkonstruktionen im Zusammenhang mit Fixpunktlogiken untersucht,[1] besonders inflationäre Fixpunkte, die ein zentrales Thema meiner Arbeit sind. Zu ihrer Definition betrachten wir noch einmal die induktive Definition kleinster Fixpunkte von positiven Formeln $\varphi(R, \overline{x})$ wie in (1). Da

[1] Siehe [EF99] für eine ausführliche Darstellung von Fixpunktlogiken auf endlichen Strukturen bzw. [DG02] für eine Übersicht, die auch unendliche Strukturen behandelt. Eine exzellente und auch heute noch sehr empfehlenswerte Quelle ist [Mo74].

$\varphi(R, \overline{x})$ positiv in R ist, F_φ somit monoton, und daher $R^\alpha \subseteq R^{\alpha+1}$, ändert sich die Fixpunktinduktion nicht, wenn die zweite Regel zu $R^{\alpha+1} := R^\alpha \cup F_\varphi(R^\alpha)$ umgeschrieben wird, d.h. eine Induktionsstufe jeweils explizit mit der nächsten vereinigt wird. Offensichtlich gilt durch diese Modifikation immer $R^\alpha \subseteq R^{\alpha+1}$, unabhängig von der Formel φ. Somit ist diese Sequenz aufsteigend und erreicht eine Stufe R^α, ab der $R^\alpha = R^\beta$ für alle $\beta > \alpha$. Diese Menge wird als der *inflationäre Fixpunkt* von φ bezeichnet. Im allgemeinen ist er kein Fixpunkt der Abbildung F_φ, sondern der Abbildung $I_\varphi(R) := R \cup F_\varphi(R)$.

1.2 Definition. *Die* inflationäre Fixpunktlogik (IFP) *ist definiert als die Erweiterung der Prädikatenlogik um Formeln ψ der Gestalt $[\mathbf{ifp}_{R,\overline{x}}\, \varphi(R, \overline{x})](\overline{t})$, wobei $\overline{t}$ ein Tupel von Termen und φ eine beliebige* IFP*-Formel ist. Sei Å eine Struktur mit Interpretationen für die freien Variablen in $\overline{t}$ und den frei in φ, aber nicht in $\overline{x}$ vorkommenden Variablen. Die Formel ψ gilt in Å, wenn $\overline{t}$ im inflationären Fixpunkt von φ enthalten ist.*

Im Gegensatz zur kleinsten Fixpunktlogik sind hier also beliebige IFP-Formeln im Fixpunktoperator erlaubt, also auch solche Formeln $\varphi(R, \overline{x})$, die nicht positiv in R sind. Wenn φ allerdings positiv in R ist, stimmen die kleinste und die inflationäre Fixpunktinduktion überein. Daraus folgt, daß jede LFP-Formel äquivalent zu einer IFP-Formel ist, d.h. LFP $\subseteq$ IFP.

Inflationäre Fixpunkte stellen eine Möglichkeit dar, die Restriktion der kleinsten Fixpunktlogik auf positive Formeln zu überwinden – eine Einschränkung, die sich in verschiedener Hinsicht negativ auswirkt. Beispielsweise ist die Formalisierung von Eigenschaften in LFP teilweise kompliziert, da immer darauf geachtet werden muß, daß die Formeln positiv in den jeweiligen Fixpunktvariablen bleiben. Dies führt oft zu nicht-intuitiven Formeln, die sowohl schwer zu schreiben als auch zu verstehen sind. Der Wegfall dieser Beschränkung in IFP erlaubt es hingegen, den schrittweisen Aufbau von Mengen durch iteriertes Auswerten einer Abfrage, Anwenden einer Abbildung etc. in einer Formel nachzubilden – ein Vorteil, der besonders im praktischen Einsatz von auf inflationären Fixpunkten basierenden Logiken zum Tragen kommt.

Darüber hinaus ist das Konzept inflationärer Fixpunkte in gewissem Sinne "robuster", da es sich auch auf Basislogiken anwenden läßt, bei denen die Monotonie der definierten Abbildungen nicht durch einfache Konzepte wie positive Formeln sichergestellt werden kann. Während das Konzept kleinster Fixpunkte hier an ihre Grenze stößt, können inflationäre Fixpunkte auch auf solche Basislogiken aufgesetzt werden. So können zum Beispiel Fixpunktlogiken definiert werden, bei denen gewisse Formen nicht-deterministischer Auswahl von Elementen der Struktur in die Logik integriert werden. Mit solchen Möglichkeiten können leicht positive Formeln geschrieben werden, die keine monotonen Abbildungen definieren. Die Integration dieser Konzepte in IFP stellt allerdings kein Problem dar. Eine solche Logik wird in Kapitel 6 der Dissertation betrachtet.

Neben der Art der verwendeten Fixpunkte stellt die Wahl der Basislogik eine weitere Variationsmöglichkeit dar. Analog zu LFP und IFP als Erweiterung der Prädikatenlogik um kleinste bzw. inflationäre Fixpunkte, können auch andere Logiken um Fixpunktoperatoren erweitert werden. Die Wahl der Basislogik beeinflusst ganz entscheidend das Anwendungsgebiet, in welchem darauf aufbauende Fixpunktlogiken eingesetzt werden können. So werden auf der Prädikatenlogik basierende Logiken vor allem im Bereich der deskriptiven Komplexitätstheorie studiert, während im Datenbankbereich vor allem Fixpukterweiterungen von konjuktiven Abfragen betrachtet werden. Auch hier sind kleinste Fixpunkte von besonderer Bedeutung und bilden die Basis der bekannten Abfragesprache DATALOG. Ein weiteres Gebiet, in dem Fixpunktlogiken eingesetzt werden, ist der Verifikationsbereich. Hier werden Logiken als Spezifikationssprachen genutzt, um gewünschte Eigenschaften von Prozessen zu beschreiben. Beim sogenannten *Model-Checking* sollen diese Spezifikationen dann – möglichst automatisch – gegenüber einem Prozess verifiziert werden. Für diesen Einsatz ist die Prädikatenlogik viel zu komplex, weshalb hier im wesentlichen Erweiterungen einer sehr viel schwächeren Logik, der *Modallogik*, zum Einsatz kommen. Eine in diesem Gebiet wichtige und gut untersuchte Logik ist der *modale μ-Kalkül* (L_μ), der, völlig analog zu LFP, die Modallogik um kleinste Fixpunktoperatoren erweitert. Sprachen wie der μ-Kalkül oder Fragmente davon, z.B. die Logiken LTL, CTL oder CTL*, spielen sowohl in der Praxis als auch in der Theorie des Model-Checkings ein große Rolle.

Überblick und Organisation. Thema meiner Dissertation ist das Studium von Fixpunktlogiken in verschiedenen Bereichen innerhalb der Informatik. Im einzelnen werden kleinste, inflationäre, nicht-deterministische und partielle Fixpunkte im Bereich der Prädikaten- und Modallogik behandelt. Die Arbeit gliedert sich in drei Teile, deren wichtigste Egebnisse hier in den folgenden Kapiteln über Fixpunkterweiterungen der Prädikatenlogik, Modallogik und ihre Fixpunkterweiterungen und Fixpunktlogiken als Abfragesprachen für Constraint Datenbanken beschrieben werden.
Im Kern der Betrachtungen standen vor allem inflationäre Fixpunkte und ihre Abgrenzung von kleinsten Fixpunkten. Zum einen wurden die Logiken IFP und LFP hinsichtlich ihrer Ausdrucksstärke verglichen und durch den Beweis der Äquivalenz der beiden Logiken dieses lange offene Problem abschließend gelöst. Dies ist das Hauptergebnis der Arbeit und wird hier im nächsten Kapitel genauer erläutert.
Zum anderen wurden inflationäre Fixpunkte im Bereich der Modallogik betrachtet und hinsichtlich ihrer Eignung als Basis einer sowohl theoretisch interessanten als auch praktisch einsetzbaren Verifikationssprache untersucht. Die Ergebnisse, die hier in Kapitel 3 genauer präsentiert werden, zeigen klar, daß die entstandene Logik algorithmisch viel zu komplex ist um als praktisch einsetzbare Sprache zu dienen, aber sehr wohl eine Logik mit sehr interessanten Eigenschaften ergibt.

2 Fixpunkterweiterungen der Prädikatenlogik

In der Einleitung wurden schon zwei wichtige Fixpunktlogiken eingeführt. Logiken wie LFP und IFP haben eine besondere Bedeutung im Bereich der endlichen Modelltheorie bzw. der *deskriptiven Komplexitätstheorie*, die versucht, die Komplexität von Problemen anhand der benötigten logischen Ressourcen zu klassifizieren. Im Gegensatz zur klassischen Komplexitätstheorie stehen hier also nicht die Ressourcen *Zeit* und *Platz* im Zentrum der Betrachtungen, sondern die Komplexität der zur Beschreibung der Probleme nötigen Logiken, wie z.B. die Anzahl von Quantoren oder die Art und Stelligkeit der verwendeten Fixpunktoperatoren. Ein bedeutendes Resultat der deskriptiven Komplexitätstheorie ist der folgende Satz von Immerman und Vardi.

2.1 Theorem (Immerman, Vardi). *Eine Klasse endlicher geordneter Strukturen ist genau dann in Polynomialzeit entscheidbar, wenn sie durch eine Formel aus* LFP *definierbar ist.*

Der Satz von Immerman und Vardi stellt eine sehr enge Verbindung zwischen der Komplexitätsklasse PTIME und der kleinsten Fixpunktlogik her. Ähnliche Resultate konnten auch für andere Komplexitätsklassen und Fixpunktlogiken bewiesen werden. Diese sogenannten *capturing Resultate* lösten ein großes Interesse an Fixpunktlogiken aus, da es nun möglich war, Ergebnisse über Fixpunktlogiken auf die Komplexitätstheorie zu übertragen. Insbesondere kann das schwierige Problem des Trennens von Komplexitätsklassen wie PTIME, NP oder PSPACE auf das Trennen von Fixpunktlogiken reduziert werden. Zu den wichtigsten in diesem Bereich behandelten Logiken gehören LFP und IFP.
In ihrer hier vorgestellten Form wurden LFP und IFP Anfang der 80er Jahre eingeführt. Kleinste und inflationäre Fixpunktinduktionen in abstrakter Form wurden allerdings schon in den 70er Jahren unter anderem in der deskriptiven Mengenlehre studiert (siehe [Mo74]). Hier lag der Fokus auf Fixpunktinduktionen über FO-Formeln in einer festgehaltenen unendlichen Struktur, z.B. der Arithmetik. Da in dieser Zeit also keine expliziten Fixpunktoperatoren betrachtet wurden, konnten Fixpunkte auch nicht geschachtelt oder negiert werden.
Wie oben schon bemerkt, ist die Logik LFP in IFP enthalten. Eine naheliegende und wichtige Frage ist, ob IFP ausdrucksstärker als LFP ist, oder ob die Logiken äquivalent sind. Eine erste Teilantwort zu diesem aus den Untersuchungen der 70er Jahre offen gebliebenen Problem gibt der Satz von Gurevich und Shelah (siehe [GS86]).

2.2 Theorem (Gurevich, Shelah, 1986). *Auf endlichen Strukturen ist jede Formel in* IFP *äquivalent zu einer Formel in* LFP, *d.h.* LFP = IFP *auf endlichen Strukturen.*

Der Beweis des Satzes benutzt die sogenannten *Stage Comparison Relationen* für LFP und IFP, die von Moschovakis [Mo74] eingeführt wurden. Mit Hilfe dieser Relationen definierten Gurevich und Shelah die Stufen einer inflationären Fixpunktinduktion durch eine LFP-Formel. Hierbei wird wesentlich die Eigenschaft genutzt, daß auf endlichen Strukturen jede Fixpunktinduktion nur endlich viele Schritte durchlaufen kann, bevor der Fixpunkt erreicht wird. Insbesondere hat jede Fixpunktinduktion auf endlichen Strukturen eine (definierbare) letzte Induktionsstufe, bei der der Fixpunkt erreicht wird. Entsprechend verallgemeinert sich der Beweis auch nicht auf unendliche Strukturen, bei denen Fixpunktinduktionen Limesschritte beinhalten können. Die Frage, ob die Logiken auch auf unendlichen Strukturen äquivalent sind, blieb ungeklärt. Die im folgenden Theorem gegebene Lösung dieses Problems ist das wichtigste Resultat meiner Dissertation.

2.3 Theorem. *Jede Formel aus* IFP *kann so zu einer* LFP *Formel transformiert werden, daß beide Formeln auf allen Strukturen äquivalent sind, d.h.* LFP = IFP *auf allen Strukturen.*

Der technische Kern des Beweises benutzt ebenfalls Stage Comparison Relationen, um inflationäre durch kleinste Fixpunkte zu beschreiben. Konnte jedoch im Fall endlicher Strukturen ein **ifp**-Operator durch einen einzelnen **lfp**-Operator ersetzt werden, so wird in der Transformation auf beliebigen Strukturen jeder **ifp**-Operator durch eine Formel ersetzt, in der mehrere **lfp**-Operatoren ineinander geschachtelt und die inneren negiert verwendet werden. Die Gesamtformel wird also bei der Transformation nach LFP stark vergröért und ist in ihrer Struktur erheblich komplizierter. Dies kann im allgemeinen auch

nicht vermieden gibt. Wie ich zeigen konnte, gibt es keine Umwandlung von IFP nach LFP, bei der die Anzahl von Alternationen zwischen **lfp**-Operatoren und Negationen nicht ansteigt.
Der Beweis des Satzes ergibt ferner, daß eine enge Beziehung zwischen der Alternationshierachie für LFP und der Verschachtelungshierarchie für IFP besteht.

Partielle Fixpunktlogik. Die beiden bisher betrachteten Logiken sind Beispiele für induktive Fixpunktlogiken, d.h. Logiken, bei denen die Sequenz von Fixpunktstufen aufsteigend ist. Für manche Anwendungen ist dies aber zu restriktiv. Hier soll möglich sein, Elemente aus einer Fixpunktstufe zu entfernen, sie also nicht in die nächste zu übernehmen. Eine Fixpunktlogik, die dies erlaubt, ist die *partielle Fixpunktlogik* (PFP). Wiederum wird von einer Formel φ eine Sequenz von Mengen wie in (1) definiert, wobei nun aber keinerlei Einschränkungen mehr an die Formel gestellt werden. Daraus folgt, daß die Sequenz nicht mehr induktiv sein muß und im allgemeien auch keinen Fixpunkt erreicht. In diesem Fall definiert man den partiellen Fixpunkt als leer. Erreicht die Sequenz allerdings einen Fixpunkt, so wird dieser als der definierte Fixpunkt verwendet. Die Logik PFP hat ihren Ursprung in der endlichen Modelltheorie und wurde nur auf endlichen Strukturen betrachtet. Entsprechend hat die übliche Definition von PFP keine Regeln für Limesordinale und erlaubt somit nur endlich lange Fixpunktprozesse. Weiterhin läßt sie sich nur schwer auf den Bereich der Modallogik übertragen. Diese beiden Probleme werden in Kapitel 7 und 12 der Dissertation behandelt, in denen eine alternative Semantik für PFP vorgestellt wird, die die Standardsemantik erweitert und sowohl unendliche Fixpunktprozesse als auch eine einfache Übertragung auf den Bereich der Modallogik erlaubt.

3 Fixpunktlogiken in der Verifikation

Ziel der computerunterstützten Verifikation ist die Entwicklung möglichst automatischer Verfahren, die es erlauben, zu überprüfen, ob ein gegebener Prozeß, z.B. ein Hard- oder Softwaresystem, eine gewünschte Eigenschaft erfüllt. Ein für diese Zwecke erfolgreiches Verfahren ist das sogenannte Model-Checking. Hierbei wird der betrachtete Prozeß zunächst mittels eines Transitionssystems $\mathcal{T}$ modelliert, eines kanten- und knotenbeschrifteten Graphs, dessen Knoten die möglichen Zustände des Prozesses und dessen Kanten die durch Ereignisse oder Aktionen ausgelösten Zustandsübergänge repräsentieren. Zur Beschreibung eines Zustandes stehen sogenannte Propositionen, z.B. „Drucker druckt", „Drucker ist im Leerlauf"etc., zur Verfügung, mit denen die Knoten des Graphs beschriftet werden. Die möglichen Übergänge des Prozesses von einem Zustand zum nächsten, ausgelöst durch bestimme Ereignisse oder Aktionen, z.B. „Benutzer sendet Druckauftrag", werden durch mit den Aktionen beschriftete Kanten im Graph repräsentiert.

3.1 Definition. *Sei $\mathcal{P}$ eine Menge von Propositionssymbolen und $\mathcal{A}$ eine Menge von Aktionen. Ein Transitionssystem $\mathcal{T} := (V, (E_a)_{a\in\mathcal{A}}, (p)_{p\in\mathcal{P}})$ über $\mathcal{P}$ und $\mathcal{A}$ ist ein beschrifteter Graph, dessen Kanten durch Aktionen aus $\mathcal{A}$ und dessen Knoten durch Propositionen aus $\mathcal{P}$ beschriftet sind.*

Nach der Modellierung des Prozesses durch das Transitionssystem, werden die zu veri-

fizierenden Eigenschaften durch eine Formel φ einer Spezifikationssprache formalisiert. Die Überprüfung, ob der gegebene Prozeß die gewünschte Eigenschaft besitzt, reduziert sich nun auf die Frage, ob das Transitionssystem die Formel φ erfüllt.
Für den Einsatz als Spezifikationssprachen eignen sich die Prädikatenlogik und ihre Fixpunkterweiterungen aus verschiedenen Gründen nicht. Stattdessen verwendet man eine viel einfachere Logik als Basis – die *Modallogik*.
Die reine Modallogik ist für Verifikationsaufgaben allerdings noch zu ausdrucksschwach, da sie nur konstant weit die Entwicklung eines Prozesses beschreiben kann, wohingegen viele häufig auftretenden Spezifikationen, z.B. die Frage, ob der Prozeß in einen deadlock oder einen anderen schlechten Zustand geraten kann, es erfordern, beliebig lange Abläufe des Prozesses zu beschreiben. Daher werden verschiedene Erweiterungen der Modallogik betrachtet, in denen solche Spezifikationen formuliert werden können. Zu den sowohl praktisch als auch theoretisch wichtigsten solcher Logiken gehören LTL, CTL, CTL* sowie der modale μ-Kalkül.
Beim Einsatz einer Logik als Spezifikationssprache sind vor allem drei Dinge wichtig: Neben der Frage, wie einfach sich die gewünschten Spezifikationen in der Logik formalisieren lassen, sind dies vor allem die Komplexität des *Erfüllbarkeitsproblems*, d.h. der Entscheidung, ob eine Formel der Logik erfüllbar ist, also ein Modell besitzt, und des *Auswertungs-* oder *„Model-Checking“-Problems*, d.h. der Überprüfung, ob ein gegebenes Transitionssystem eine gegebene Formel erfüllt. Das Auswertungsproblem ist der Kern des Model-Checkings. Jede hierfür geeignete Logik muß ein effizientes Auswertungverfahren erlauben.
Eine sehr gut untersuchte Verifikationslogik ist der 1982 von Dexter Kozen eingeführte μ-Kalkül. Er ist analog zu LFP als die Erweiterung der Modallogik um kleinste Fixpunktoperatoren definiert.

3.2 Definition. *Der* modale μ-Kalkül (L_μ) *ist definiert als die Erweiterung der Modallogik um Formeln ψ der Gestalt $\mu X.\varphi$, wobei φ eine L_μ-Formel, positiv in X, ist. Eine solche Formel ψ definiert den kleinsten Fixpunkt der durch φ induzierten Abbildung F_φ.*

Die Bedeutung des μ-Kalküls liegt in seiner guten Balance zwischen Ausdrucksstärke und algorithmischer Komplexität. So sind z.B. viele wichtige Verifikationssprachen wie LTL, CTL oder CTL* im μ-Kalkül enthalten. Eine sehr nützliche Charakterisierung der Ausdrucksstärke liefert der Satz von Janin und Walukiewicz, die gezeigt haben, daß genau die Eigenschaften von Transitionssystemen in L_μ ausgedrückt werden können, die invariant unter Bisimulation[2] und in der monadischen Logik zweiter Stufe (MSO) definierbar sind. Eine direkte Folgerung daraus ist, daß genau die regulären Sprachen im μ-Kalkül definiert werden können. Auch modelltheoretisch weist der μ-Kalkül gute Eigenschaften auf, z.B. die endliche Modell-Eigenschaft, die besagt, daß jede erfüllbare Formel ein endliches Modell hat.
Auf der anderen Seite bleibt der μ-Kalkül algorithmisch handhabbar. So ist Erfüllbarkeit von Formeln entscheidbar, nämlich EXPTIME-vollständig. Ferner liegt das Auswertungsproblem in NP $\cap$ CO-NP, und es wird vermutet, daß L_μ-Formeln sogar in Polynomialzeit ausgewertet werden können.

[2]Bisimulation ist eine Äquivalenzrelation auf Transitionssytemen, die „gleiches Verhalten“ der durch die Systeme repräsentierten Prozesse beschreibt.

Betrachtet man seine insgesamt guten Eigenschaften und das große Forschungsinteresse am μ-Kalkül, so scheint es erstaunlich, daß keine allgemeineren Arten von Fixpunkten, wie z.B. inflationäre Fixpunkte, im Zusammenhang mit der Modallogik betrachtet wurden. Insbesondere bei Verifikationssprachen, mit denen Eigenschaften von Prozessen beschrieben werden, könnte sich der Wegfall der Beschränkung auf positive Formeln und die damit verbundene Vereinfachung in der Formalisierung von Eigenschaften positiv auswirken. Zusammen mit Anuj Dawar und Erich Grädel habe ich eine solche modale inflationäre Fixpunktlogik definiert und ihre Eigenschaften untersucht. Zu Details siehe Kapitel 11 meiner Dissertation bzw. [DGK02].

3.3 Definition. *Der* modale Iterationskalkül (MIC) *entsteht aus der Erweiterung der Modallogik um Formeln der Gestalt* $(\mathbf{ifp}\ X : \varphi(X))$*, die den inflationären Fixpunkt der durch* φ *induzierten Abbildung definieren. Hierbei ist* φ *eine beliebige* MIC*-Formel.*[3]

Mit dem gleichen Argument wie bei LFP und IFP kann man zeigen, daß MIC den μ-Kalkül erweitert. Das folgende Theorem listet einige der erzielten Resultate auf, deren Beweisideen im Anschluß kurz beschrieben werden.

3.4 Theorem. MIC *hat Unendlichkeitsaxiome, d.h. keine endliche Modelleigenschaft. Das Erfüllbarkeitsproblem ist unentscheidbar – es liegt sogar jenseits der arithmetischen Hierarchie – und das Auswertungsproblem ist* PSPACE*-vollständig. In* MIC *können Sprachen definiert werden, die zwar kontext-sensitiv, aber nicht kontext-frei sind. Folglich ist* MIC *nicht in* MSO *enthalten und echt ausdrucksstärker als* L_μ.

Als Unendlichkeitsaxiome dienen MIC-Formeln, deren Modelle nur Bäume unendlicher Höhe sind. Diese Formeln sind zwar erfüllbar, besitzen aber keine endlichen Modelle. Folglich hat MIC keine endliche Modelleigenschaft. Mit Hilfe solcher Bäume der Höhe ω kann nun die Unentscheidbarkeit des Erfüllbarkeitsproblems gezeigt werden, indem das Entscheidungsproblem der Theorie der Arithmetik, d.h. das Problem, die Gültigkeit einer FO-Formel über den natürlichen Zahlen mit Addition und Multiplikation zu entscheiden, auf das Erfüllbarkeitsproblem für MIC reduziert wird. Dazu werden die Operationen Addition und Multiplikation sowie All- und Existenzquantoren durch MIC-Formeln ausgedrückt, die die entsprechenden Operationen über der Höhe von Knoten in dem Baum definieren. Auf diese Weise lassen sich FO-Formeln über der Arithmetik in MIC-Formeln übersetzen, die genau dann erfüllbar sind, wenn die Originalformel in der Arithmetik gilt. Das Erfüllbarkeitsproblem für MIC liegt also nicht mehr in der arithmetischen Hierarchie und ist somit sehr hoch unentscheidbar. Der Beweis der PSPACE-Vollständigkeit des Auswertungsproblems geschieht durch Reduktion von QBF, der Frage, ob eine gegebene quantifizierte Boolesche Formel gilt.
Die erzielten Resultate zeigen deutlich, daß MIC sich in fast jeglicher Hinsicht erheblich vom μ-Kalkül unterscheidet, und im Bereich der Modallogik inflationäre Fixpunkte eine sehr viel ausdrucksstärkere Logik ergeben als kleinste Fixpunkte. Für die vergleichweise hohe Ausdrucksstärke zahlt man allerdings einen Preis: algorithmisch ist MIC sehr

[3] Die Originaldefinition von MIC erlaubt auch simultane Fixpunkte. Da aus Platzgründen dieses Konzept bisher nicht erwähnt wurde, ist die Definition hier angepaßt. Die hier definierte Logik wird in der Dissertation mit 1MIC bezeichnet. Es sei erwähnt, daß, im Gegensatz zu L_μ, simultane Fixpunkte die Ausdrucksstärke von MIC echt erhöhen, d.h. 1MIC $\subsetneq$ MIC.

viel komplexer als der μ-Kalkül. Dessen Ausgewogenheit zwischen Ausdrucksstärke und Komplexität verschiebt sich bei MIC also eindeutig zugunsten der Ausdrucksstärke. Aufgrund der hohen Komplexität der relevanten algorithmischen Probleme eignet sich MIC also sicherlich nicht als Basis einer Spezifikationssprache für die Verifikation. Andererseits erschließen sich durch die große Ausdrucksstärke wiederum neue potentielle Einsatzgebiete, für die L_μ zu ausdrucksschwach ist. Siehe z.B. [GK03] und Referenzen darin.
Wie gesehen, ist MIC echt ausdrucksstärker als L_μ. Im allgemeinen ist das Trennen von Logiken in Bezug auf ihre Ausdrucksstärke ein schwieriges Problem. Um zu zeigen, daß eine Eigenschaft in einer Logik definierbar ist, reicht die Angabe einer entsprechenden Formel. Will man jedoch beweisen, daß eine Eigenschaft in einer Logik *nicht* definierbar ist, muß man zeigen, daß sie durch *keine* Formel der Logik definiert werden kann. In der Literatur wurden verschiedene Methoden entwickelt, mit denen Beweise dieser Art geführt werden können. Ausgehend von dem Problem, MIC von anderen Logiken zu trennen, wird in der Arbeit eine weitere solche Methode entwickelt, die auf einer Charakterisierung von (modalen) Fixpunktlogiken durch Automaten basiert. Anders als bei den meisten Automatencharakterisierungen von Logiken wird hierbei nicht jeder Formel der betrachteten Logik ein Automat eines geeigneten Automatenmodells zugeordnet, der die gleiche Klasse von Transitionssystemen, Wörtern, Bäumen, etc. erkennt wie die Formel. Stattdessen wird für alle Logiken das gleiche Automatenmodell verwendet. Die durch eine Formel φ definierte Klasse $\mathcal{K}$ von Strukturen wird durch eine Familie von Automaten erkannt, so daß es für jedes n einen Automat gibt, der genau die Strukturen der Größe n erkennt, die die Formel erfüllen. Der sogenannte *Labelling Index* von $\mathcal{K}$ bzw. φ ist definiert als die Funktion, die jedem n die Größe des minimalen Automaten zuordnet, der genau die Strukturen der Größe höchstens n erkennt, die die Formel erfüllen. Die Unterscheidung zwischen Logiken wie MIC, L_μ etc. wird nun anhand der Labelling Indices der in den Logiken definierbaren Strukturklassen getroffen. Beispielsweise haben L_μ definierbare Klassen einen polynomiellen Labelling Index, d.h. die Automaten wachsen nur polynomiell in der Größe der Strukturen. Für MIC ergibt sich ein exponentielles und für das monadische Fragment von IFP nicht-elementares Wachstum. Ebenso wie den Modellklassen von Formeln kann man den Labelling Index von beliebigen Klassen von Transitionssystemen bestimmen. So erhält man z.B. für bestimmte Formen des *trace equivalence*-Problems einen doppelt exponentiellen Labelling Index, woraus sofort folgt, daß dieses Problem nicht in MIC definierbar ist. Auf diese Weise konnten wir für verschiedene Logiken sehr elegante Nicht-Definierbarkeitsbeweise führen und die entsprechenden Logiken voneinander trennen.

4 Constraint Datenbanken

Der dritte Teil der Dissertation, auf den hier nur kurz eingegangen werden kann, beschäftigt sich mit Constraint Datenbanken. Hierbei handelt es sich um ein zu Beginn der 90er Jahre entwickeltes Datenbankmodell, bei dem die Datenbankrelationen durch Formeln repräsentiert werden (siehe [KLP00]). Es eignet sich daher besonders für geometrische Daten, die direkt durch Angabe der sie beschreibenden Gleichungen und Ungleichungen gespeichert werden. In solchen Datenbanken können auf einfache und natürliche Weise Abfragen durch den Benutzer gestellt werden. Allerdings wirkt sich in diesem Bereich die Ausdrucksschwäche der Prädikatenlogik noch stärker aus als bei klassischen Datenbanken, so daß nach geeigneteren Abfragesprachen gesucht wurde. Im dritten Teil der Dissertation wird der Einsatz von Fixpunktlogiken als Anfragesprachen für Constraint Datenbanken untersucht. Wie ich zeigen konnte, erhält man durch die Erweiterung der Prädikatenlogik um Operatoren zur Bildung transitiver Hüllen eine Turing-vollständige Sprache. Auf der anderen Seite lassen sich durch geeignete Einschränkungen von kleinsten Fixpunktoperatoren Abfragesprachen definieren, die genau die in Polynomialzeit berechenbaren Anfragen ausdrücken können.

Literatur

[DG02] Dawar, A. und Gurevich, Y.: Fixed-point logics. *Bulletin of Symbolic Logic*. 8(1):65–88. 2002.

[DGK02] Dawar, A., Grädel, E., und Kreutzer, S.: Inflationary fixed points in modal logic. *ACM Transactions on Computational Logic (TOCL)*. 2002. Accepted for publication. An extended abstract appeared in the Proc. of the 10th Conf. on Computer Science Logic (CSL), volume 2142 of LNCS, pages 277-291, Springer Verlag 2001.

[EF99] Ebbinghaus, H.-D. und Flum, J.: *Finite Model Theory*. Springer. 2nd. 1999.

[GK03] Grädel, E. und Kreutzer, S.: Will deflation lead to depletion? On non-monotone fixed-point inductions. In: *IEEE Symp. of Logic in Computer Science (LICS)*. 2003.

[GS86] Gurevich, Y. und Shelah, S.: Fixed-point extensions of first-order logic. *Annals of Pure and Applied Logic*. 32:265–280. 1986.

[KLP00] Kuper, G., Libkin, L., und Paredaens, J. (Hrsg.): *Constraint Databases*. Springer. 2000.

[Kr02] Kreutzer, S.: *Pure and Applied Fixed-Point Logics*. PhD thesis. RWTH Aachen. 2002.

[Mo74] Moschovakis, Y.: *Elementary Induction on Abstract Structures*. North Holland. 1974. ISBN 0 7204 2280 9.

Wissenschaftlicher Werdegang

Geboren wurde ich am 27.5.1974 in Aachen-Walheim. Nach Erwerb des Abiturs am Inda-

Gymnasium in Aachen-Kornelimünster fing ich zum Wintersemester 1993/94 an der RWTH Aachen ein Informatikstudium an. Im Januar 1999 schloß ich das Studium mit einem Diplom mit Auszeichnung ab, für das ich die Springorum-Medaille der RWTH Aachen erhielt. Meine Diplomarbeit zum Thema *Descriptive Complexity Theory for Constraint Databases* schrieb ich am Lehr- und Forschungsgebiet "Mathematische Grundlagen der Informatik" von Prof. Dr. Grädel an der RWTH Aachen. Unmittelbar nach Abschluß des Studiums habe ich eine Stelle als wissenschaftlicher Mitarbeiter am Lehr- und Forschungsgebiet "Mathematische Grundlagen der Informatik" bei Prof. Grädel angenommen und meine Promotion begonnen. Mit Vorlage der hier besprochenen Dissertation zum Thema *Pure and Applied Fixed-Point Logics* habe ich meine Promotion im Jahre 2002 mit Auszeichnung abgeschlossen und wurde dafür von der RWTH Aachen für den Dissertationspreis der Gesellschaft für Informatik vorgeschlagen. Nach meiner Promotion habe ich im Rahmen des EU Forschungsund Trainingsnetzwerks “GAMES” ein Stipendium für einen Aufenthalt an der Universität von Edinburgh in Schottland erhalten.