

## **Gedächtnisprotokoll Diplomprüfung in Theoretischer Informatik**

Prüfer: Prof. Thomas  
Datum: 31. März 2003  
Note: 1,0

Vorlesungen: Applied Automata Theory  
Automata and reactive Systems  
Logikprogrammierung

### **Applied Automata Theory:**

*Wie sieht ein AFA aus? Was ist das besondere daran? Welche Vorteile bietet ein AFA gegenüber einem NFA?*

Ein Lauf eines AFAs ist ein Baum, bei dem jeder Ast in einem Endzustand enden muss. Fähigkeit, neue Prozesse zu erstellen.

*Kann man zu jedem AFA einen äquivalenten NFA konstruieren? Wie viele Zustände hat ein äquivalenter NFA oder DFA?*

Äquivalenter NFA hat  $2^n$  Zustände, DFA hat  $2^{2^n}$  Zustände.

*Wie sehen BLT-Formeln aus? Was kann man damit beschreiben?*

BLT-Formeln beschreiben genau die regulären Sprachen.

*Erstellen Sie eine BLT-Formel, die die Sprache beschreibt, deren Wörter zwischen zwei  $x_1$  immer ein  $x_2$  haben.*

$$\varphi = \forall t_1 \forall t_2 ((t_1 < t_2 \wedge x_1(t_1) \wedge x_1(t_2)) \rightarrow \exists s (t_1 < s \wedge s < t_2 \wedge x_2(s)))$$

*Was sind CFSM? Wie funktionieren Sie? Was ist ein globaler Zustand in einem CFSM?*

Globaler Zustand = Zustände der Automaten + Inhalte der Nachrichtenkanäle.

*Ist das Erreichbarkeitsproblem bei CFSM entscheidbar? Warum nicht? Wie kann man das Halteproblem für Turingmaschinen darauf reduzieren?*

Bei der Reduktion auf das Halteproblem hat Prof. Thomas recht genau nachgefragt, wie die Konstruktion funktioniert.

*Welche Probleme für Petri-Netze wurden in der Vorlesung besprochen? Was ist das Boundedness-Problem? Ist es entscheidbar? Wie?*

Reachability-Tree.

*Was bewirkt das Ersetzen einer Zahl durch  $\infty$  im Reachability-Tree?*

Reachability-Tree verliert an Information und wird endlich.

### **Automata and reactive Systems:**

*Was bedeutet co-Büchi-Akzeptanz?*

Ab einem bestimmten Zeitpunkt treten nur noch Endzustände auf.

*Ist die Sprache co-Büchi-erkennbar, deren Wörter ab einer bestimmten Stelle nur noch abwechselnd 0 und 1 enthalten?*

Ja.

*Wie kann man algorithmisch entscheiden, ob eine Sprache co-Büchi-erkennbar ist?*

Wenn es einen Muller-Automaten gibt, bei dem  $\mathcal{F}$  unter subtrees abgeschlossen ist. Hier hat Prof. Thomas allerdings eingesehen, dass diese Antwort so nicht direkt aus der Frage abzuleiten war und hat daher nachgeholfen.

*Wann ist eine  $\omega$ -Sprache regulär? Nennen Sie zwei Möglichkeiten.*

Wenn Sie Büchi-erkennbar und durch SIS beschreibbar ist.

*Geben Sie für oben genannte Sprache einen  $\omega$ -regulären Ausdruck an.*

$(0+1)^*(01)^\omega$

*Kann man zu einem nichtdeterministischen Büchi-Automaten effizient einen deterministischen Muller-Automaten erzeugen? Wie? Wie groß wird dieser Automat?*

Safra-Konstruktion erzeugt  $2^{O(n \log n)}$  Zustände. Weniger geht nicht!

*Wie sehen Knoten bei der Safra-Konstruktion aus? Warum gibt es nur endlich viele?*

Knoten sind Safra-Bäume. Nur endlich viele, weil Brüder nicht die selben Zustände enthalten können und die Zustandsmengen von Vater und Sohn nicht übereinstimmen dürfen.

*Worum geht es bei Spielen? Welche Arten von Gewinnstrategien gibt es?*

Gewinnstrategien und Gewinnbereiche sollen gefunden werden. Es gibt positionale und Automatenstrategien.

*Worum geht es bei Muller-Spielen? Was für eine Gewinnstrategie kann man hier finden?*

Nur Automatenstrategie.

*Worauf lässt sich ein Muller-Spiel reduzieren? Wie funktioniert das?*

Reduktion auf Paritätsspiel durch LAR-Automaten.

*Was für eine Gewinn-Strategie kann man zu einem Parity-Automaten finden? Welchen Aufwand hat das? Kann man diesen Aufwand verringern?*

Gewinn-Strategie kann gefunden werden mit einem Verfahren mit zeitlich exponentiellen Aufwand. Ob dieser Aufwand verringert werden kann, ist nicht bekannt.

*Was bedeutet determiniert? Gibt es Spiele, die nicht determiniert sind?*

Determiniert = Jeder Knoten gehört zu  $W_0$  oder  $W_1$ . Es gibt Spiele mit unendlichen Spielgraphen, die nicht determiniert sind.

### **Logikprogrammierung:**

*Wie funktioniert die aussagenlogische Resolution?*

Resolutionsregel erläutert.

*Was besagt der Resolutionssatz?*

Eine Formel ist unerfüllbar genau dann, wenn aus ihrer Klauselmengemenge die Leere Klausel folgt. Verfahren ist vollständig und korrekt.

*Wie funktioniert die FO-Resolution?*

FO-Resolutionsregel erläutert.

*Was hat es mit Unifikation auf sich? Wie funktioniert der Unifikationsalgorithmus und was berechnet er genau?*

Unifikationsalgorithmus beschrieben. Er berechnet den allgemeinsten Unifikator.

*Wie kann man einen Pfad als Logikprogramm darstellen und Pfade darin berechnen?*

```
Kante(a,b). Kante(a,c). ...  
Pfad(X, Y) :- Kante(X, Y).  
Pfad(X, Y) :- Kante(X, Z), Pfad(Z, Y).
```

*Wie wird ein solches Programm von Prolog abgearbeitet?*

Abarbeitung anhand des Programmes beschrieben.

*Welche Semantiken gibt es außer der prozeduralen Semantik? Wo liegt der Unterschied zwischen prozeduraler und deklarativer Semantik?*

Es gibt deklarative, prozedurale und Fixpunktsemantik. Prozedurale Semantik ist Operationalisierung der deklarativen Semantik.