Prüfungsprotokoll zur Diplomprüfung Informatik im

Vertiefungsfach Automaten, Logik und Verifikation

Prüfungsgebiete:

- Angewandte Automatentheorie (nach Skript, 4 SWS)
- Automaten und Reaktive Systeme (nach Skript, 4 SWS)
- Model-Checking (nach Vorlesung WS 2003/04, 2 SWS)
- Strukturtheorie regulärer und kontextfreier Sprachen (nach Vorlesung WS 2006/07, 2 SWS)

Prüfer:	Prof. Dr. Dr. h. c. Wolfgang Thomas
Beisitzerin:	Dipl.-Inform. Alex. Spelten
Datum:	03.04.2007 (11:30 Uhr)

Wenn ich mich richtig erinnere hat Prof. Thomas die Prüfungsgebiete zwar schon nacheinander behandelt, die Übergänge kamen mir aber eher fließend vor.

1. Angewandte Automatentheorie
   - Was ist das Ziel von Kongruenzen? (Zustandsminimierung)
   - Ist die Minimierung von NFAs effizient? (PSPACE-vollständig)
   - Wie zeigt man, dass dieses Problem PSPACE-hart ist? (Reduktion des Universalitätsproblems auf das Minimierungsproblem. Diesen Beweis musste ich in der Prüfung ausführlicher erklären.)
   - Was kann man denn dann statt Minimierung machen? (Bisimulationsäquivalenz)
   - Führt die zum minimalen Automaten? (im allgemeinen nicht)
   - Gibt es überhaupt *den* minimalen Automaten? (im allgemeinen auch nicht)
   - Und, was meinen sie, wie sieht es mit dem Universalitätsproblems für kontextfreie Sprachen (CFLs) aus? Ist das entscheidbar? (Ich habe vermutet, dass das nicht der Fall ist. Ist es auch nicht.)
   - Und ist das Inklusionsproblem für CFLs entscheidbar? (Nein)
   - Und das Äquivalenzproblem? (auch nicht)
   - Warum nicht? (Dann wäre auch das Universalitätsproblems für CFLs entscheidbar.)
   - Betrachten wir die Menge der Ableitungsbäume einer kontextfreien Grammatik (CFG). Ist diese Menge eine reguläre Baumsprache? (Ja, aber die Rückrichtung gilt nicht.)

- Wie zeigt man das, dass die Rückrichtung nicht gilt? Ich habe da eine bestimmte Baumsprache im Kopf. (Da bin ich nicht drauf gekommen. Das hat er dann schön intuitiv erklärt: Baumautomaten können die Existenz eines Pfades prüfen, CFGs können aber nicht prüfen, ob eine bestimmte Ableitungsregel tatsächlich verwendet wird.)
- Welche Entscheidungsprobleme für Petri-Netze (PN) kennen Sie? (Erreichbarkeit, Überdeckung, Beschränktheit, Verklemmungsfreiheit.)
- Ist Erreichbarkeit für PN entscheidbar? (Ja, haben wir aber in der Vorlesung nicht gemacht.)
- Und die anderen Probleme für PN? (Ja, z. B. mittels des Karp-Miller-Baums.)
- Womit sind die Knoten des Karp-Miller-Baum beschriftet? (Elemente aus $(\mathbb{N} \cup \{\infty\})^r$)
- Ist der Karp-Miller-Baum endlich? Warum? (Ja. Königs-Lemma, Lemma von Dixon.)
- Größe des Karp-Miller-Baums? (Lässt sich durch keine primitiv rekursive Funktion beschränken. Ich habe mich zunächst versprochen und 'nicht elementar' gesagt.)
- Ist denn jede elementare Funktion auch primitiv rekursiv? (Ja)
- Kennen Sie ein Beispiel für eine nicht primitiv rekursive Funktion? (Ja, die Ackermann-Funktion.)
- Ist das Beschränktheitsproblem des Karp-Miller-Baums in Polynomialzeit lösbar? (Ja, aber nicht durch Konstruktion des Karp-Miller-Baums.)
- Erreichbarkeitsproblem für Pushdownsystemen (PDS): Kann man entscheiden, ob die von einem Zustand erreichte Menge an Konfigurationen endlich ist? (Ja: Satz von Büchi, Saturierungsalgorithmus, Endlichkeitstest für NFAs)
- Wie stellt man fest, ob ein NFA eine unendliche Sprache erkennt? (Teste, ob eine erreichbare nicht-triviale starke Zusammenhangskomponente existiert, die einen Endzustand enthält.)
- Ist das Erreichbarkeitsproblem für kommunizierende endliche Automaten (CFSMs) entscheidbar? (Nein)

2. Strukturtheorie regulärer und kontextfreier Sprachen
   - Sternhöhenproblem: Da haben wir zwei Probleme kennengelernt, welche? (das Sternhöhenproblem (sh-Problem) für reguläre Ausdrücke und das generalisierte Sternhöhenproblem (gsh-Problem) für verallgemeinerte reguläre Ausdrücke.)
   - Kann man eine Sprache L auf Sternfreiheit ($\mathrm{gsh}(L) = 0$) testen? (Ja, Transitionsmonoid bestimmen und diesen auf Gruppenfreiheit testen. Funktioniert wegen Satz von Schützenberger.)

- Was ist die einfache Richtung des Satzes von Schützenberger? (von sternfreien regulären Ausdrücken nach Gruppenfreiheit. Beweis durch Induktion über reguläre Ausdrücke.)
- Führt man bei der Rückrichtung des Beweises auch eine Induktion durch? (Ja)
- Worüber? (wusste ich leider nicht mehr: über *Ideale*)
- Gibt es eine Logik, in der man gerade die sternfreien regulären Sprachen ausdrücken kann? (Ja, FO.)
- Welches offene Problem haben wir bezüglich der Sternhöhe kennengelernt? (ob eine Sprache L mit $\mathrm{gsh}(L) > 1$ existiert)
- Welche Komplexitätsklassen unterhalb von P kennen Sie? Ich denke da an die Schaltkreise... (*Nick's class* NC)
- Wie sind die Komplexitätsklassen NC^k und AC^k definiert?
- In welcher dieser Klassen sind reguläre Sprachen? (NC^1)
- Sind die sternfreien Sprachen im Allgemeinen auch in einer Unterklasse davon? (Ja, in AC^0.)
- Wir haben eine Sprache kennengelernt, die in gewisser Weise für die kontextfreien Sprachen universell ist, welche? (die Dyck-Sprache D_n bzw. sogar D_2)
- Wie lautet der Zusammenhang? (Satz von Chomsky-Schützenberger. Darstellung: $L = h(D_n \cap R)$ bzw. $L = h(f^{-1}(D_2 \cap R))$)
- Beweisidee? (Konstruktion einer entsprechenden Grammatik für eine reguläre Sprache R. Er hat aber zum Glück nicht danach gefragt, wie diese Konstruktion funktioniert.)

3. Model-Checking (MC)
   - Gegeben seien eine LTL-Formel φ sowie eine Struktur $(\mathcal{M}, s_0)$. Wie kann man MC machen? ($(\mathcal{M}, s_0) \models \varphi \iff (\mathcal{M}, s_0) \times \mathcal{A}_{\neg\varphi} \neq \emptyset$. Ich musste dann die einzelnen Schritte erklären.)
   - Komplexität der Konstruktion beim LTL-MC? (Exponentiell in der Formelgröße. LTL-MC ist ferner PSPACE-vollständig.)
   - Wie funktioniert das CTL-MC? (sukzessives Markieren der Knoten mit Teilformeln. Ich sollte den Fall $E\varphi U\psi$ erläutern.)
   - Komplexität des CTL-MC? ($O(|\mathrm{cl}(\varphi)| \cdot (|S| + |R|))$ mit Knotenmenge S und Kantenrelation R der Kripke-Struktur)
   - Warum ist diese Laufzweit trozdem problematisch? (zum einen können CTL-Formeln exponentiell größer sein als LTL-Formeln, andererseits ist in der Regel die Größe der Kripke-Strukturen exponentiell in der Größe des zu modellierenden Systems.)
   - Ausweg? (symbolisches CTL-MC)

- Was benutzt man dafür? (OBBDs)
- Was beschreiben OBBDs? (Boolesche Funktionen)
- Wie beschreibt man eine Kripke-Struktur durch Boolesche Funktionen? (siehe Vorlesung Nr. 6, Seite 2)
- Welche schönen Eigenschaften haben OBBDs? (Habe ein wenig drumrumgeredet und effiziente Verknüpfung genannt.)
- Kann man zwei OBBDs effizient auf Äquivalenz testen? (Ja, der Äquivalenztest bezüglich einer Variablenordnung ist effizient.)
- Wofür braucht man den Äquivalenztest beim CTL-MC? (wusste ich nicht)
- Frage nach dem Algorithmus für $E\varphi U\psi$. (Er hat aber nur ein Bildchen gemalt, welches ich vervollständigen sollte, und zwar das von Vorlesung Nr. 6, Seite 5.)
- Wann kann man diesen Algorithmus abbrechen? (Da braucht man den Äquivalenztest. Für Prof. Thomas war es wohl eine Genugtuung, mich doch noch zu dieser Antwort bringen zu können.)

4. Automaten und Reaktive Systeme
   - Fragen nach der Definition von Büchi- und Mullerautomaten.
   - Fragen nach den Abschlusseigenschaften der durch Büchi- und Mullerautomaten erkennbaren Sprachen.
   - Welchen Zusammenhang zwischen den von diesen Automaten definierten Sprachklassen kennen sie? (L ist Muller-erkennbar genau dann, wenn L ist Boolesche Kombination von Büchi-erkennbaren Sprachen.)
   - Fragen, die auf Schleifen und den Satz von Landweber abzielten.
   - Fragen nach Eigenschaften von Büchi- und Paritäts-Spiele? (determiniert, positionale Strategien)
   - Gibt es Spiele, die nicht determiniert sind? (Ja, die sind aber recht exotisch und gehörten nicht zum Vorlesungsstoff, wie z. B. Gale-Stewart-Spiele.)
   - Was für eine Strategie braucht man für Muller-Spiele? (Automatenstrategie)
   - Fragen nach der Konstruktion dieser Strategie und der Spielreduktion. (Reduktion auf Paritätsspiele durch LAR-Konstruktion)
   - Größe des Strategieautomaten? ($O(n!)$ für ein Muller-Spiel mit n Knoten)
   - Fragen nach Steiger-Wagner-Spielen und der Spielreduktion. (Reduktion auf schwache Paritätsspiele. Idee: Merken der bereits besuchten Zustände.)
   - Größe des Strategieautomaten? ($O(2^n)$ für ein Steiger-Wagner-Spiele mit n Knoten)

Fazit: Die Prüfung fand ich im Ganzen sehr spaßig. Leider hatte ich nur vier Wochen Vorbereitungszeit, was für eine Diplomprüfung meiner Meinung nach eigentlich zu wenig Zeit ist (ich empfehle eher sechs bis acht Wochen). Zum Prüfungsstil kann ich sagen, dass Prof. Thomas den Prüfungsstoff sehr informell und mehr in die Breite prüft. Mir kam es dabei so vor, dass er hauptsächlich nur versucht herauszufinden, ob man den Stoff verstanden hat. Dazu gehört aber auch, dass er einen mit kleineren Zwischenfragen gängelt, um zu testen, dass man einen Satz oder einen Beweis nicht einfach bloß auswendig gelernt hat, sondern auch tatsächlich das Verständnis vorhanden ist. Bei den Beweisen reichte es immer, "nur" die Beweisidee und eine grobe Beweisskizze zu kennen und diese auch verstanden zu haben. (Beispielsweise sollte man wissen, worüber eine Induktion durchgeführt wird, oder welche Fälle unterschieden werden müssen.) Wirklich formal ausführen musste ich nichts. Eine eloquente Präsentation des Stoffs scheint mir dabei für eine gute Note weniger nötig zu sein (ist wahrscheinlich sogar vollkommen unerheblich). Find ich gut. :)

Ich habe dieses Protokoll leider erst 10 Tage nach meiner Prüfung geschrieben und direkt nach der Prüfung hatte ich mir nur sehr wenig Stichpunkte gemacht. Sicherlich fehlen daher eine Menge der Fragen, die mir gestellt wurden. Außerdem waren die Fragen von Prof. Thomas sehr viel kleinschrittiger.

Zum Lernen verwendete Literatur:

- Offizielle i7-Skripte:
  `http://www-i7.informatik.rwth-aachen.de/teaching/skript.html`
- Eigene handschriftliche Vorlesungsmitschriften.
- Slides und Videos zu den Vorlesungen:
  `http://www-i7.informatik.rwth-aachen.de/teaching/ws0607/ars/media`
  `http://www-i7.informatik.rwth-aachen.de/teaching/ws0304/modelchk/`
  `http://www-i7.informatik.rwth-aachen.de/teaching/ss03/angauto/`
- Übungsblätter zu den Vorlesungen.
- Prüfungsprotokolle.
- Ein netter Vergleich zwischen Safra- und Muller-Shupp-Konstruktion:
  C. Schulte Althoff, W. Thomas, and N. Wallmeier. Observations on determinization of büchi automata. Theoretical Computer Science, 363(2):224-233, October 2006.
  `http://www-i7.informatik.rwth-aachen.de/download/papers/wallmeier/stw06.pdf`

Wünsche Euch viel Spaß und Erleuchtung bei Euren Prüfungen!