

Prüfungsprotokoll Mathematik

Fach: ADM I
Studiengang: Mathematik

Vordiplom
 Zwischenprüfung
 Diplom

Bachelor
 Master

Prüfer/in: Prof. Skutella
Datum: 26.8.19
Prüfungsdauer: ~40 Min.

Beisitzer/in: Daniel Schmidt
Note: 2,0
Anzahl der Kandidaten: 1

Vorbereitungszeit: ca. 3 Wochen, mit Pausentagen

Literatur: Vorlesungsfolien, Korte/Vygen (Kombinatorische Optimierung),
Cook/Cunningham/Pulleyblank/Schrijver, Bertsimas/Tsitsiklis

Beurteilung der Prüfung und des/r Prüfers/in:

- nette Atmosphäre, er hilft einem sehr gut weiter, lässt aber auch erstmal Zeit zum selber überlegen
- Beisitzer hat nichts gesagt
- Skutella sagt immer, wenn es richtig war („ja“, „stimmt“) oder nicht

Fragen:

- Wahlthema: Goldberg-Tarjan-Algorithmus
 - optimality criterion (Is-Edipath in \mathbb{D}^x)
 - Definition von preflow, valid labelling usw. genannt
 - warum funktioniert Algo? (immer valid preflow & labelling)
 - Laufzeit (er fragte: „terminiert der Algo immer?“)
 - ↳ $O(n^2m)$ non-saturating pushes, Beweisidee, warum terminiert
 - (*) der Algo genau deshalb (white-loop)
- Linear Programming: „Was war das wichtigste Resultat?“ (wusste nicht, was er da wollte, habe daher gesagt, dass simplex ja exponentiell sein kann und wir deshalb ellipsoid methode gemacht haben), er hat dann erstmal zu simplex gefragt:
 - Was macht der simplex? (Anschauung, kein Beweis)
 - Def. basic direction, basic solution
 - „Wie sieht die basic solution aus?“ → wollte auf Satz 3.14 raus (Beweis für \Rightarrow)
 - jetzt Klee-Minty-cube (genaue Aussage von 11.1 nennen)
 - separation problem (was ist das?)
 - wieso sollte man nicht $Ax=b, x \geq 0$ immer in polynomieller Zeit überprüfen können? → exponentiell viele constraints möglich
 - Bsp. LP mit exponentieller Anzahl an constraints (max-flow auf graph $\cdot \xrightarrow{\cdot} \cdot \xrightarrow{\cdot} \cdot \dots \xrightarrow{\cdot} \cdot$) Folie 274
 - path-based LP-formulation dafür, dual aufstellen Folie 274
 - separation problem in linear time lösbar? → ja, wie? (shortest path berechnen)
- (*) → könnte man den auch benutzen für feasible b-transshipment? → ja (Folie 229, Observation 10.2 + Beweis)

Prüfungsprotokoll Mathematik

Fach:	ADM I	<input checked="" type="checkbox"/> Bachelor
Studiengang:	Mathematik	<input type="checkbox"/> Master

(Sonstiges bitte von Hand eintragen.)

Prüfer in:	Prof. Skutella	Beisitzer in:	Daniel Schmidt
Datum:	17.05.2019	Note:	2,0
Prüfungsdauer:	~40 Minuten	Anzahl der Kandidaten:	1

Vorbereitungszeit: 2 Wochen (plus 2 ½ Wochen für Erstversuch)

Literatur: Vorlesungsfolien

Beurteilung des Prüfers:

Sehr angenehme Atmosphäre. Sofern man die Antwort nicht direkt weiß, gibt er Tipps (entweder direkt („Denken Sie an...“) oder durch weitere Fragen („was wäre, wenn...?“). Wenn Antwort richtig war, gibt er entsprechendes Feedback. Prüfung wurde mit der Frage eröffnet, welches Wunschthema ich denn gerne hätte.

Fragen:

- Selbstgewähltes Thema: Preflow-Push-Algorithmus:
 - Operationen und Bedingungen (valid preflow etc.) definiert
 - Algorithmus beschrieben und formal aufgeschrieben
 - Laufzeit und Korrektheit bewiesen. Wieso funktioniert der Algorithmus?
 - Wieso sind non-saturation Push Ops. Laufzeitschranke für den Algorithmus?
 - Beweise zu:
 - Für alle aktiven Knoten ex. v-s-Dipath in D_x und kein s-t-Dipath
 - Saturating push operations in $O(n \cdot m)$
 - Non-saturating push operations in $O(n^2 \cdot m)$
 - Algo in $O(n^3)$, wenn v mit $d(v)=\max$ gewählt wird
- Äquivalenz von Vertex (woher kommt das c?), Basic feas. Solution, extreme point
- Beweis: Theorem 3.14 (x is BFS $\leftrightarrow Ax=b$ & es ex. Indizes, sodass $A(B(1)) \dots A(B(m))$ lin. unabhängig und $x_j=0$ für alle anderen.)
- Wie sieht Polytope aus, gegeben durch $A \cdot x=b$ (ohne $x \geq 0$)? Was Welche Lösungen gibt es für verschiedene Kostenvektoren? Wie sieht Polytope aus, wenn nur eine Zeile betrachtet wird (ohne nicht-Negativität). Was würde passieren, wenn so ein Polytope in einen LP-Solver geworfen wird?
- Komplexität von LPs:
 - Laufzeit exponentiell: Klee-Minty-Cubes erklärt (Pivoting Rules)
 - Ellipsoid-Method erklärt und separation problem
 - Was bringt das für das Finden einer Optimallösung, wenn E.-Methode einen Punkt im Polyeder finden kann?

Prüfungsprotokoll Mathematik

Fach: ADM I Bachelor
Studiengang: Naturwissenschaften in der Informationsgesells. Master

(Sonstiges bitte von Hand eintragen.)

Prüfer/in: Prof. Skutella Beisitzer/in: Christoph Hertrich
Datum: 10.02.2021 Note: 1,3
Prüfungsdauer: etwa 40min Anzahl der Kandidaten: 1

Vorbereitungszeit: ca. 4 Wochen (das Modul hatte ich jedoch 2 Jahre zuvor gehört)
Literatur: Folien der Vorlesung, Bertsimas & Tsitsiklis, Korte & Vygen, Cook & Cunningham

Beurteilung der Prüfung und des/r Prüfers/in:

Professor Skutella sorgte für eine freundliche und entspannte Atmosphäre. Er gab mir ausreichend Zeit zum Nachdenken und bestätigte richtige Aussagen direkt mit einem starken Nicken. Wenn ich etwas nicht wusste, versuchte er mir mit Hinweisen weiterzuhelfen oder gab am Ende auch selbst die richtige Antwort. In der Nachbesprechung meinte er, dass ich mein Wahlthema auch zügiger hätte vorstellen können. (Hier hätte ich mir mehr direkte Rückmeldung während der Prüfung gewünscht.)
Der Beisitzer sagte nichts während der Prüfung.

Fragen:

Zu Beginn durfte ich mir ein Thema aus der Vorlesung selbst aussuchen. Dazu hatte ich den Goldberg-Tarjan Algorithmus vorbereitet. Dabei erklärte ich: die Idee hinter dem Algorithmus + Optimalitätskriterien, Definitionen der Begriffe (preflow, active node, valid labelling und admissible arc), die obere Schranke durch den kürzesten v-t-Weg und die Beweisideen dazu, dass es keinen s-t-Weg im Residualgraphen gibt, aber für alle aktiven Knoten einen Weg zu s. Dann erklärte ich den Algorithmus selbst anhand eines kleinen selbst gewählten Beispiels. Zum Schluss fragte er mich noch zur Laufzeit und ich erläuterte die jeweiligen Ideen hinter den Beweisen. Weitere Frage zu Goldberg-Tarjan: Ist die Laufzeit polynomiell oder streng polynomiell? Warum? (Streng polynomiell) Wie würden Sie nun damit umgehen, wenn Sie (insbesondere von s ausgehende) Arcs mit unendlicher Kapazität hätten? Die Antwort: der Max Flow ist trotzdem durch den Min Cut bestimmt. Um im Algorithmus damit zu arbeiten, kann man sich eine Zahl M definieren, die größer ist als die Summe der endlichen Kapazitäten. Wenn ich diese als Startflow benutze, kann ich den Max Flow wie üblich im Algorithmus finden.

Nächstes Thema: Was war Ihrer Meinung nach das wichtigste Resultat in Bezug auf die lineare Programmierung? Ich antwortete damit, dass wir für ein pointed polyhedron mit bounded cost immer eine optimale Lösung haben, die eine Ecke ist und wir somit für polyhedra in standard form die optimalen Lösungen in den Ecken finden. Und sich darin dann eben auch das Simplex-Verfahren begründet.

Nachfrage: Was sind Ecken? Zu welchen Bezeichnungen ist dies noch äquivalent? (extreme point, basic solution)

Letztes Thema: Laufzeitanalysen zum Simplex-Verfahren: polynomiell je Iteration, aber eventuell exponentiell viele Iterationen. Hier fragte er mich wieder, ob das Simplex-Verfahren in streng polynomieller Laufzeit je Iteration läuft, was ich nicht direkt wusste, wir dann aber gemeinsam herleiteten.