Seminar-Themen

„Algorithmen und Komplexität“

Sommersemester 2019 — Prof. Dr. Georg Schnitger

Im Folgenden finden Sie Vortragsthemen für das Bachelor-Seminar „Algorithmen und Komplexität“ im Sommersemester 2019. Die Vergabe der Themen erfolgt in der Vorbesprechung. Sie dürfen alle Themen wählen, die mit B („Bachelor“) markiert sind. Themen, die mit M („Master“) gekennzeichnet sind, werden in Absprache mit Ihrem Betreuer entsprechend eingeschränkt.

Bitte machen Sie sich vorab mit den Themen vertraut, sodass Sie in der Vorbesprechung bereits wissen, welche Themen für Sie in Frage kommen! Wenn Sie ein eigenes Thema vorschlagen wollen, setzen Sie sich bitte rechtzeitig vor der Vorbesprechung mit uns in Verbindung.

Inhaltsverzeichnis

Algorithmen **2**

1	B	Auction algorithms for network flow problems: A tutorial introduction	2
2	B, M	The Directed Steiner Network Problem is Tractable for a Constant Number of Terminals	2
3	B	SAT-Algorithmen: Schöning-Algorithmus	2
4	B, M	SAT-Algorithmen: Moser-Scheder-Algorithmus	2
5	B	The Small-World Phenomenon: An Algorithmic Perspective	2

Komplexitätstheorie **3**

6	B, M	Interactive Proofs and Graph Isomorphism	3
7	B, M	Lower Bounds for Monotone Circuits	3
8	B	Complexity of Counting and #P-Completeness	3
9	B	The Complexity of Escaping Labyrinths and Enchanted Forests	3
10	B	Tetris is hard, even to approximate	4
11	B	Smoothed Complexity and Pseudopolynomial-Time Algorithms	4
12	B	Not being (super)thin or solid is hard: A study of grid Hamiltonicity	4
13	B	Succinct Representations of Graphs	4
14	B	Analysis of the Blockchain Protocol in Asynchronous Networks	5

Algorithmen

①	**Auction algorithms for network flow problems: A tutorial introduction**	B

Dimitri P. Bertsekas
Comp. Opt. and Appl. 1(1): 7-66 (1992). https://dblp.org/rec/journals/coap/Bertsekas92

Kurzbeschreibung: Auktionsalgorithmen sind für viele Graphprobleme wie kürzeste Wege oder Max-Flow bekannt. Der „Klassiker“ ist der Auktionsalgorithmus für gewichtetes bipartites Matching. Dabei wird die eine Seite der Knoten zu *Bietern*, die andere Seite zu *Gegenständen* mit anfänglichem Preis 0. Jeder Bieter wählt nun den Gegenstand mit dem für ihn besten Profit (Kantengewicht minus Preis). In mehreren Runden passt der Algorithmus die Preise an, bis keine zwei Bieter mehr denselben Gegenstand wählen und somit ein Matching gefunden ist.

Bemerkungen: Nur die Abschnitte 1 bis 3 müssen bearbeitet werden.

Nützliche Vorkenntnisse: Lineare Programmierung, Dualität

②	**The Directed Steiner Network Problem is Tractable for a Constant Number of Terminals**	B, M

Jon Feldman, Matthias Ruhl
SIAM J. Comput. 36(2): 543-561 (2006). https://dblp.org/rec/journals/siamcomp/FeldmanR06

Kurzbeschreibung: Im Directed Steiner Network Problem sind ein gerichteter Graph mit Kantengewichten, sowie p Quellen $s_1, \ldots, s_p$ und p Senken $t_1, \ldots, t_p$ gegeben. Gefunden werden soll ein leichtester gerichteter Wald (d. h. eine leichteste Vereinigung von gerichteten Pfaden) sodass es von jeder Quelle s_i einen Weg zur Senke t_i gibt. Während das Problem für allgemeine p als NP-vollständig nachgewiesen wurde, lässt es sich für beschränkte p in Polynomialzeit lösen.

③	**SAT-Algorithmen: Schöning-Algorithmus**	B

Uwe Schöning, Jacobo Torán
Das Erfüllbarkeitsproblem SAT – Algorithmen und Analysen. Mathematik für Anwendungen 1, Lehmann 2012. http://dblp.org/rec/books/daglib/0028796

Kurzbeschreibung: Im SAT-Problem ist eine KNF-Formel gegeben und auf Erfüllbarkeit zu prüfen. Da das Problem NP-vollständig ist, sind hier keine Wunder zu erwarten, aber man möchte die exponentielle Laufzeit so niedrig wie möglich drücken. Der randomisierte Schöning-WalkSAT-Algorithmus durchsucht mittels einer Markov-Kette den Belegungsraum und erreicht die Laufzeit $\mathcal{O}\big(\mathrm{poly}(n) \cdot (\frac{4}{3})^n\big)$ für 3-SAT.

Nützliche Vorkenntnisse: Elementare Stochastik, Markov-Ketten

④	**SAT-Algorithmen: Moser-Scheder-Algorithmus**	B, M

Uwe Schöning, Jacobo Torán
Das Erfüllbarkeitsproblem SAT – Algorithmen und Analysen. Mathematik für Anwendungen 1, Lehmann 2012. http://dblp.org/rec/books/daglib/0028796

Kurzbeschreibung: Der Moser-Scheder-Algorithmus erreicht dieselbe Laufzeit wie der oben genannte Schöning-Algorithmus sogar deterministisch mithilfe von Überdeckungscodes und lokaler Suche.

Bemerkungen: Das Thema baut auf dem obigen („SAT-Algorithmen: Schöning-Algorithmus“) auf.

⑤	**The Small-World Phenomenon: An Algorithmic Perspective**	B

Jon Kleinberg
STOC, 2000. https://dl.acm.org/citation.cfm?id=335325

Kurzbeschreibung: In sozialen Netzwerken tritt das sogenannte Kleine-Welt-Phänomen auf: Je zwei Personen sind über eine sehr kurze Kette von Kontakten miteinander verbunden. Diese Arbeit beschäftigt sich mit den algorithmischen Grundlagen von Milgrams Experiment „übermittle einen Brief an eine Zielperson, von der du nur den Wohnort kennst, indem du ihn an einen deiner Bekannten weitergibst, von dem du vermutest, dass er der Zielperson näher steht als du“.

Nützliche Vorkenntnisse: parallele Algorithmen

Komplexitätstheorie

⑥ Interactive Proofs and Graph Isomorphism — B, M

Kapitel 8 bis inkl. Abschnitt 8.3 in *Computational Complexity: A Modern Approach*, Cambridge University Press 2009 von Sanjeev Arora und Baoz Barak. http://theory.cs.princeton.edu/complexity/

Kurzbeschreibung: Gegeben seien eine Sprache L und eine Eingabe x. In einem Interactive Proof (IP) müssen zwei Personen kooperieren, um die Frage $x \in L$*?* zu beantworten. Der *Prover* betrachtet die Eingabe, rechnet beliebig lange und schickt einen „Beweis“ an den *Verifier*. Dieser muss dann mithilfe des Beweises in Polynomialzeit entscheiden, ob $x \in L$ gilt. Ein prominentes Problem im Kontext der IPs ist Graph Isomorphism (GI). Es wird einerseits gezeigt, dass GI vermutlich nicht NP-vollständig ist – sonst würde die Polynomialzeithierarchie kollabieren –, andererseits ist auch kein Polynomialzeitalgorithmus für GI bekannt. Es wird somit vermutet, das GI „zwischen“ P und NP-vollständig liegt.

Nützliche Vorkenntnisse: Komplexitätsklassen wie P, NP und PSPACE, Elementare Stochastik

⑦ Lower Bounds for Monotone Circuits — B, M

Abschnitt 14.3 in *Computational Complexity: A Modern Approach*, Cambridge University Press 2009 von Sanjeev Arora und Baoz Barak. http://theory.cs.princeton.edu/complexity/

Kurzbeschreibung: Ein monotoner Schaltkreis besitzt OR- und AND-Gatter, aber keine NOT-Gatter. Eine Technik zur Herleitung unterer Schranken für die Größe monotoner Schaltkreise für Entscheidungsprobleme wird vorgestellt. Beispielsweise benötigt das Clique-Problem monotone Schaltkreise mindestens exponentieller Größe. Könnte man die Einschränkung der Monotonie weglassen, so wäre $\mathsf{P} \neq \mathsf{NP}$ gezeigt, doch leider benötigen monotone Schaltkreise auch exponentielle Größe für die Lösung des Matching-Problems.

Nützliche Vorkenntnisse: Schaltkreise, Elementare Stochastik

⑧ Complexity of Counting and #P-Completeness — B

Kapitel 17 bis inkl. Abschnitt 17.3.1 in *Computational Complexity: A Modern Approach*, Cambridge University Press 2009 von Sanjeev Arora und Baoz Barak. http://theory.cs.princeton.edu/complexity/

Kurzbeschreibung: Im Gegensatz zu Entscheidungsproblemen, wo nach der *Existenz* einer Lösung gefragt ist, wird in Zählproblemen nach der *Anzahl* der Lösungen gefragt. Viele solcher Zählprobleme, beispielsweise das Zählen perfekter Matchings oder einfacher Kreise in einem Graphen, haben sich als sehr schwierig herausgestellt. Diese Schwierigkeit wird durch das Konzept der #P-Härte analog zur NP-Härte formalisiert.

Nützliche Vorkenntnisse: Komplexitätsklassen, Polynomialzeitreduktionen

⑨ The Complexity of Escaping Labyrinths and Enchanted Forests — B

Florian D. Schwahn, Clemens Thielen
FUN 2018: 30:1-30:13. https://dblp.org/rec/conf/fun/SchwahnT18

Kurzbeschreibung: Die Komplexität der bekannten Brettspiele *Das verrückte Labyrinth* (engl. *The aMAZEing Labyrinth*) und *Sagaland* (engl. *Enchanted Forest*) wird analysiert. Das erste stellt sich als NP-vollständig heraus, sofern man alleine spielt, bzw. als PSPACE-vollständig, wenn man einen Gegenspieler hat. Das zweite hingegen ist in Polynomialzeit lösbar.

Nützliche Vorkenntnisse: Polynomialzeitreduktionen, PSPACE-Vollständigkeit

⑩ Tetris is hard, even to approximate — B

Ron Breukelaar, Erik D. Demaine, Susan Hohenberger, Hendrik Jan Hoogeboom, Walter A. Kosters, David Liben-Nowell
Int. J. Comput. Geometry Appl. 14(1-2): 41-68 (2004) https://dblp.org/rec/journals/ijcga/BreukelaarDHHKL04

Kurzbeschreibung: Der Videospiel-Klassiker Tetris wird hinsichtlich seiner Komplexität untersucht. Zentrale Fragestellungen wie „maximiere die Anzahl der vervollständigten Reihen“ stellen sich als NP-hart heraus.

Nützliche Vorkenntnisse: Polynomialzeitreduktionen

⑪ Smoothed Complexity and Pseudopolynomial-Time Algorithms — B

Tim Roughgarden
Skripte (Lecture 17 und 18) zur Vorlesung CS264 Fall'17 (Beyond Worst-Case Analysis). Stanford University. http://theory.stanford.edu/~tim/w17/w17.html

Kurzbeschreibung: Smoothed Complexity ist ein Hybrid zwischen Worst-Case Complexity und Average-Case Complexity und beantwortet die Frage „Wie hoch ist die erwartete Laufzeit eines Algorithmus, wenn Worst-Case-Eingaben zufällig verrauscht werden?“. Im Rahmen der Vorlesung wird ein zentrales Resultat von Beier und Vöcking präsentiert: Ein binäres Optimierungsproblem hat genau dann polynomielle Smoothed Complexity, wenn es durch einen Algorithmus mit pseudopolynomieller erwarteter Laufzeit gelöst werden kann.

Bemerkungen: Aus Lecture 17 sind nur die Abschnitte 1 und 2 zu bearbeiten.

Nützliche Vorkenntnisse: Laufzeitanalyse, elementare Stochastik, lineare Programmierung

⑫ Not being (super)thin or solid is hard: A study of grid Hamiltonicity — B

Esther M. Arkin, Sándor P. Fekete, Kamrul Islam, Henk Meijer, Joseph S.B. Mitchell, Valentin Polishchuk, David Rappaport, Henry Xiao
Computational Geometry: Theory and Applications, 42(6-7), 582-605. https://www.sciencedirect.com/science/article/pii/S092577210800117X

Kurzbeschreibung: Das Hamiltonkreis-Problem ist eines der klassischen NP-vollständigen Probleme. Sofern $\mathsf{P} \neq \mathsf{NP}$ gilt, ist es im Allgemeinen nicht effizient lösbar. Doch wie sehen möglichst „einfache“ Graphklassen aus, für die das Hamiltonkreis-Problem trotzdem noch NP-vollständig ist? Und welche Graphklassen sind einfach und erlauben eine effiziente Lösung des Hamiltonkreis-Problems? In dieser Arbeit betrachten wir Gitter mit drei-, vier- und sechseckigen Maschen.

Nützliche Vorkenntnisse: Polynomialzeitreduktionen

⑬ Succinct Representations of Graphs — B

Bemerkung: Beide Quellen sind zu bearbeiten.

Hana Galperin, Avi Wigderson: *Succinct representations of graphs*, Information and Control, 56(3):183–198, 1983. https://dblp.org/rec/journals/iandc/GalperinW83

Kurzbeschreibung: Üblicherweise wird die Laufzeitkomplexität von Graphproblemen in Abhängigkeit von $|V|$ angegeben. Doch was passiert, wenn die Eingabeinstanz besonders platzsparend (mit polylog($|V|$) vielen Bits) beschrieben werden kann? Es stellt sich heraus, dass viele einfache Graphprobleme plötzlich NP-vollständig sind, wenn die Eingaben platzsparend beschrieben werden.

Christos H. Papadimitriou, Mihalis Yannakakis: *A Note on Succinct Representations of Graphs*, Information and Control 71(3): 181-185 (1986). https://dblp.org/rec/journals/iandc/PapadimitriouY86

Kurzbeschreibung: In dieser Arbeit wird gezeigt, dass klassische NP-vollständige Graphprobleme NEXP-vollständig sind, wenn ihre Eingaben platzsparend beschrieben werden.

Nützliche Vorkenntnisse: Komplexitätsklassen, Reduktionen

⑭	**Analysis of the Blockchain Protocol in Asynchronous Networks**	B

Rafael Pass, Lior Seeman, Abhi Shelat
EUROCRYPT (2) 2017: 643-673. https://dblp.org/rec/html/journals/iacr/MarcedonePS16

Kurzbeschreibung: Das bekannte *Bitcoin-Blockhain-Protokoll* muss die Konsistenz einer großen verteilten Datenbank in einem dezentralen Netzwerk sicherstellen. Dabei gibt es im Netzwerk keine zentrale Instanz, welche die Teilnahme reguliert. Anders als bei Filesharing-Protokollen muss die Datenbank insbesondere gegen bösartige Angreifer gefeit sein. Dies wird hier über das *Proof-of-Work*-Konzept sichergestellt: Wer Daten hinzufügen will, muss zuvor eine moderat schwierige Einweg-Funktion berechnen.

Nützliche Vorkenntnisse: Kryptographie, elementare Stochastik