Seminar-Themen

„Algorithmen und Komplexität“

und

„Komplexität“

Wintersemester 2016/17 — Prof. Dr. Georg Schnitger

Im Folgenden finden Sie Vortragsthemen für das Bachelor-Seminar „Algorithmen und Komplexität“ und das Master-Seminar „Komplexität“ im Wintersemester 2016/17. Die Vergabe der Themen erfolgt in der Vorbesprechung. Bachelor-Studierende dürfen jedes Thema wählen, das mit einem Ⓑ markiert ist. Entsprechend kommen für Master-Studierende nur Themen in Frage, die mit Ⓜ gekennzeichnet sind. Sind Themen sowohl mit Ⓑ als auch mit Ⓜ markiert, stehen sie beiden Studiengängen offen, wobei der geforderte Umfang für Bachelor-Studierende entsprechend reduziert wird.

Bitte machen Sie sich vorab mit den Themen vertraut, sodass Sie in der Vorbesprechung bereits eine ungefähre Vorstellung haben, welche Themen für Sie in Frage kommen.

Wenn Sie ein eigenes Thema vorschlagen wollen, setzen Sie sich bitte vor der Vorbesprechung mit uns in Verbindung.

Inhaltsverzeichnis

Berechnungswelten und (un)gewöhnliche Berechnungsmodelle

① **Fun with semirings: a functional pearl on the abuse of linear algebra** Ⓑ

Stephen Dolan
ICFP 2013: 101-110. http://dblp.org/rec/conf/icfp/Dolan13

Kurzbeschreibung: Semiringe sind algebraische Strukturen, die in der Informatik häufig auftreten, bspw. bildet die Menge $\{0, 1\}$ mit der Addition „$\vee$" und der Multiplikation „$\wedge$" den booleschen Semiring. In der Arbeit werden typische Probleme aus der Informatik mithilfe von Semiringen modelliert.

Bemerkungen: Die Beschreibung der Datenstrukturen und Algorithmen findet in dieser Arbeit in der Programmiersprache Haskell statt. Im Seminar soll der Fokus auf die Modellierung durch Semiringe gelegt werden. Die funktionale Implementierung ist zweitrangig.

② **Faktorgraphen und Belief Propagation auf Bäumen** Ⓑ

Christopher M. Bishop
Kapitel 8.4 in *Pattern Recognition and Machine Learning*. Springer, 2006

Kurzbeschreibung: Sei $g(x, y, z) = f_1(x, y) \cdot f_2(y) \cdot f_3(x, z)$ eine Faktorisierung der Funktion g. Ein Faktorgraph für g ist ein bipartiter Graph mit der Variablenknotenmenge $\{x, y, z\}$, der Faktorknotenmenge $\{f_1, f_2, f_3\}$ und der Kantenmenge $\big\{\{f_1, x\}, \{f_1, y\}, \{f_2, y\}, \{f_3, x\}, \{f_3, z\}\big\}$. Die Baumstruktur des Graphen kann ausgenutzt werden, um die Summe $\sum_{x,y,z} g(x, y, z)$ auf geschickte Weise zu berechnen („Belief Propagation" bzw. „Sum-product algorithm"). Semiringe und ihr Distributivgesetz spielen dabei eine besondere Rolle und ermöglichen interessante Anwendungen in der Welt der kombinatorischen Optimierung.

③ **A Fast Quantum Mechanical Algorithm for Database Search** Ⓑ Ⓜ

Lov K. Grover
STOC 1996: 212-219. http://dblp.org/rec/html/conf/stoc/Grover96

Kurzbeschreibung: Auf klassischen Computern ist eine lineare Suche in der Laufzeit $\Theta(n)$ möglich. Der Grover-Algorithmus für Quantencomputer beschleunigt die „lineare" Suche, sodass sie in der Laufzeit $\mathcal{O}(\sqrt{n})$ durchgeführt werden kann.

Bemerkungen: Eine gute Beschreibung des Algorithmus sowie seiner Analyse befindet sich z. B. in Kapitel 10.4 aus *Computational Complexity: A Modern Approach*, Cambridge University Press, 2009 von Sanjeev Arora und Baoz Barak. Der Vortrag soll auch eine kurze Einführung in die Funktionsweise eines Quantenrechners beinhalten.

④ **Closed Timelike Curves Make Quantum and Classical Computing Equivalent** Ⓑ Ⓜ

Scott Aaronson, John Watrous
Electronic Colloquium on Computational Complexity (ECCC) 15(092) (2008). http://dblp.org/rec/journals/eccc/AaronsonW08a

Kurzbeschreibung: Wie mächtig sind Schaltkreise, deren Eingabe- und Ausgabebits Zeitreisen unternehmen können? Closed Timelike Curves (CTCs) sind ein physikalisches Modell und beschreiben die theoretischen Voraussetzungen für solche Zeitreisen. Die Arbeit behandelt Schaltkreise mit CTC-Bits und zeigt, dass Polynomialzeitberechnungen mit CTC-Bits und PSPACE übereinstimmen, d. h. es gilt $\mathsf{P}_{\mathsf{CTC}} = \mathsf{PSPACE}$. Außerdem unterscheiden sich Quantencomputer mit CTC-Bits nicht von klassischen Computern mit CTC-Bits.

⑤ Threshold Computation und Postselection Ⓑ Ⓜ

Yenjo Han, Lane A. Hemaspaandra, Thomas Thierauf. *Threshold Computation and Cryptographic Security*. SIAM J. Comput. 26(1): 59-78 (1997). http://dblp.org/rec/html/journals/siamcomp/HanHT97

Kurzbeschreibung: Während eine probabilistische Turingmaschine (TM) akzeptiert, wenn die akzeptierenden Berechnungen hinreichend *wahrscheinlich* sind, akzeptiert eine Threshold TM, falls der *Anteil akzeptierender Pfade* hinreichend groß ist. Dies entfesselt eine erstaunlich große Rechenkraft, die in klassischen Berechnungen äquivalent zur Fähigkeit der *Postselection* (s. u.) ist.

Scott Aaronson. *Quantum Computing, Postselection, and Probabilistic Polynomial-Time*. Electronic Colloquium on Computational Complexity (ECCC)(003) (2005). http://dblp.org/rec/html/journals/eccc/ECCC-TR05-003

Kurzbeschreibung: Eine TM mit *Postselection* darf nach einer probabilistischen Berechnung auf ein Ereignis (z. B. „das erste Bit ist 1") bedingen, d. h. alle Berechnungspfade, die das Ereignis nicht erfüllen, eliminieren. Polynomialzeit-Quantenrechner mit Postselection sind äquivalent zur (sehr mächtigen) Klasse PP. Dieses Resultat liefert auch Indizien dafür, warum die Quantenmechanik so ist, wie sie ist. Würde man bestimmte Stellschrauben leicht ändern, ließe sich Postselection effizient simulieren.

⑥ Rechnen mit Licht Ⓑ Ⓜ

Bemerkungen: Von den drei folgenden Arbeiten ist die erste sowie eine der anderen beiden auszuwählen.

Dominik Schultes: *Rainbow Sort: Sorting at the speed of light*, Natural Computing 5(1): 67-82 (2006). http://dblp.org/rec/journals/nc/Schultes06

Kurzbeschreibung: Diese Arbeit bildet einen Einstieg in das Thema „Rechnen mit Licht". Wie können wir eine Folge von Zahlen sortieren, wenn wir einen Laser und ein Prisma als Teil der „Berechnung" verwenden dürfen?

Mihai Oltean: *Solving the Hamiltonian path problem with a light-based computer*, Natural Computing 7(1): 57-70 (2008). http://dblp.org/rec/journals/nc/Oltean08

Kurzbeschreibung: Diese Arbeit beschreibt einen Computer mit einer graph-artigen Struktur. Entlang der Kanten werden Lichtstrahlen gesendet, welche in jedem passierten Knoten „markiert" werden. Auf diese Weise kann entschieden werden, ob ein Graph einen Hamilton-Weg enthält.

Mihai Oltean, Oana Muntean: *Solving the subset-sum problem with a light-based device*, Natural Computing 8(2): 321-331 (2009). http://dblp.org/rec/journals/nc/OlteanM09

Kurzbeschreibung: Es wird ein Computer mit Licht vorgestellt, der das Subset-Sum-Problem entscheidet: Gegeben eine Menge A von Zahlen und eine Zahl k, entscheide ob ein $B \subseteq A$ mit $\sum_{b \in B} b = k$ existiert.

⑦ Infinite-Time Turing Machines Ⓑ Ⓜ

Apostolos Syropoulos
Kapitel 4.2 in *Hypercomputation: Computing Beyond the Church-Turing Barrier*. Springer Science & Business Media, 2008. https://link.springer.com/book/10.1007/978-0-387-49970-3. Verfügbar aus dem Uni-Netzwerk.

Kurzbeschreibung: Hier geht es um Turing-Maschinen, die abzählbar unendlich lang in Schleifen stecken bleiben dürfen – und dann einfach weiterrechnen. Dies führt zu neuen Halteproblemen.

⑧ A Personal View of Average-Case Complexity Ⓑ Ⓜ

Russell Impagliazzo
Structure in Complexity Theory Conference 1995: 134-147. http://dblp.org/rec/conf/coco/Impagliazzo95

Kurzbeschreibung: Was wäre, wenn Annahmen aus der Komplexitätstheorie wie bpsw. „P = NP" wahr wären oder Einwegfunktionen existierten? Auf umgangssprachliche Weise werden fünf verschiedene Welten beschrieben: Algorithmica, Heuristica, Pessiland, Minicrypt und Cryptomania.

Bemerkungen: Diese Arbeit ist als Ausgangspunkt für eine Formalisierung der beschriebenen Welten (Definitionen, Sätze, Beweisideen) anhand der dort zitierten Literatur aufzufassen.

⑨ Alternation Ⓑ Ⓜ

Ashok K. Chandra, Dexter Kozen, Larry J. Stockmeyer
J. ACM 28(1): 114-133 (1981). http://dblp.org/rec/html/journals/jacm/ChandraKS81

Kurzbeschreibung: Das Konzept der Alternation wird als Verallgemeinerung des Nichtdeterminismus eingeführt, in verschiedenen Maschinenmodellen formalisiert und jeweils deren Ausdruckskraft untersucht: Alternierende Turingmaschinen können in Polynomialzeit PSPACE-harte Probleme lösen. Alternierende endliche Automaten akzeptieren zwar auch „nur“ reguläre Sprachen, allerdings mit bis zu doppelt-exponentieller Ersparnis an Zuständen.

⑩ Zwei-Wege-Automaten – wie stark ist Nichtdeterminismus? Ⓑ

Kapitel 3.3.3 im Skript Theoretische Informatik 2 von Georg Schnitger

Kurzbeschreibung: Für gewöhnliche Automaten ist Kraft des Nichtdeterminimus geklärt: Ein NFA kann exponentiell weniger Zustände besitzen als ein dazu äquivalenter DFA. Die Situation für Zwei-Wege-Automaten – diese dürfen den Lesekopf nach rechts oder links bewegen – ist bis heute ungeklärt. Resultate gibt es aber für weiter eingeschränkte Automatenklassen.

Komplexität: Untere Schranken und Schwierigkeit von Problemen

11 The Byzantine Generals Problem — B

LESLIE LAMPORT, ROBERT E. SHOSTAK, MARSHALL C. PEASE
ACM Trans. Program. Lang. Syst. 4(3): 382-401 (1982) http://dblp.org/rec/html/journals/toplas/LamportSP82

Kurzbeschreibung: Beim verteilten Rechnen gilt es byzantinische Fehler zu vermeiden, die auf folgendes Problem zurückzuführen sind: Mehrere örtlich getrennte Generäle müssen sich auf eine einheitliche Strategie (*Angriff* oder *Rückzug*) verständigen, die Kommunikation ist aber nur zwischen je zwei Generälen mittels Boten möglich. Erschwerend kommt hinzu, dass sich unter den Generälen auch Verräter befinden können, die versuchen werden, die Einigung zu sabotieren.

12 Lower Bounds for Monotone Circuits — B

Kapitel 14.3 in *Computational Complexity: A Modern Approach*, Cambridge University Press 2009 von Sanjeev Arora und Baoz Barak

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 expontenielle Größe für die Lösung des Matching-Problems.

13 Hardness of Easy Problems: Basing Hardness on Popular Conjectures such as the Strong Exponential Time Hypothesis — B M

VIRGINIA V. WILLIAMS
IPEC 2015: 17-29. http://dblp.org/rec/conf/iwpec/Williams15

Kurzbeschreibung: Standardmäßig zeigen wir mithilfe von Polynomialzeit-Reduktionen, dass gewisse Probleme NP-schwer sind. Eine solche Unterscheidung in „effizient“ und „vermutlich nicht effizient“ ist leider nur sehr grob. Diese Arbeit beschäftigt sich mit feineren Reduktionen („fine-grained reductions“) für einfache Probleme (in P), um Fragen von der Form „Geht es besser als $\mathcal{O}(n^k)$?“ zu beantworten.

14 Gaming is a hard job, but someone has to do it — B M

GIOVANNI VIGLIETTA
Theory Comput. Syst. 54(4): 595-621 (2014). http://dblp.org/rec/journals/mst/Viglietta14

Kurzbeschreibung: Welche Spiele sind wirklich schwierig? Die Arbeit zeigt Metatheoreme über die komplexitätstheoretischen Konsequenzen von gewissen Spiel-Elementen wie Türen, Druckplatten oder Schlüsseln und beschreibt außerdem ein „PSPACE-Härte-Framework“. Anschließend werden die Metatheoreme und das Framework auf Spiele wie Pac-Man, Prince of Persia, Lemmings, Doom, Starcraft und weitere angewendet.

15 Classic Nintendo Games are (Computationally) Hard — B M

GREG ALOUPIS, ERIK D. DEMAINE, ALAN GUO, GIOVANNI VIGLIETTA
Theor. Comput. Sci. 586: 135-160 (2015). http://dblp.org/rec/journals/tcs/AloupisDGV15

Kurzbeschreibung: Als Nachfolge-Arbeit zu „Gaming is a hard job, but someone has to do it“ wird nun zusätzlich ein „NP-Härte-Framework“ eingeführt. Die Metatheoreme und Frameworks werden anschließend auf Super Mario, Donkey Kong, Legend of Zelda, Metroid und Pokémon angewendet.

Klassische und brandaktuelle Algorithmen

⑯ The Hungarian Method for the Assignment Problem Ⓑ

Harold W. Kuhn
50 Years of Integer Programming 2010: 29-47. http://dblp.org/rec/books/daglib/p/Kuhn10

Kurzbeschreibung: Das Zuordnungsproblem ist die Frage nach einem größtmöglichen Matching in einem bipartiten Graphen. Die sogenannte „Ungarische Methode“ (1955) bildet die Grundlage für primal-duale Algorithmen zur Lösung von ganzzahligen linearen Programmen (ILPs).

⑰ Künstliche Neuronale Netze Ⓑ

Martin Anthony
Kapitel 1 (Artificial Neural Networks) und 9 (Neural Network Learning) in *Discrete Mathematics of Neural Networks: Selected Topics*. Monographs on Discrete Mathematics and Applications. Society for Industrial and Applied Mathematics, 2001. Verfügbar im Präsenzbestand der Professur THI.

Kurzbeschreibung: Ein künstliches neuronales Netz (KNN) ist ein gerichteter, gewichteter Graph, dessen Knoten (Neuronen) reelle Zahlen als Nachrichten austauschen. Eine Anwendung eines KNN ist es, aus einer Menge von Trainingsinstanzen zu lernen, um im Anschluss weitere Instanzen klassifizieren zu können. Im Google Research Blog wurde ein Artikel über Bilderkennung durch KNNs veröffentlicht, siehe https://research.googleblog.com/2015/06/inceptionism-going-deeper-into-neural.html.

⑱ The Complexity of Weighted Greedy Matching Ⓑ Ⓜ

Argyrios Deligkas, George B. Mertzios, Paul G. Spirakis
CoRR abs/1602.05909 (2016). http://dblp.org/rec/journals/corr/DeligkasMS16

Kurzbeschreibung: Die exakte Berechnung eines schwersten Matchings in einem gewichteten Graphen gelingt in Polynomialzeit. Mit Greedy-Algorithmen erreicht man sogar häufig in Linearzeit eine gute Approximation. Doch wie sieht die Situation aus, wenn ein *optimales Greedy*-Matching gesucht ist, d.h. ein unter allen von einem Greedy-Algorithmus berechneten Matchings ein schwerstes ist? In dieser Arbeit wird das Problem GreedyMatching zunächst formalisiert und anschließend seine NP-Vollständigkeit gezeigt. Ein randomisierter Approximationsalgorithmus wird vorgestellt und für einige Klassen untersucht.

⑲ An Optimal Online Algorithm for Weighted Bipartite Matching and Extensions to Combinatorial Auctions Ⓑ Ⓜ

Thomas Kesselheim, Klaus Radke, Andreas Tönnis, Berthold Vöcking
ESA 2013: 589-600. http://dblp.org/rec/html/conf/esa/KesselheimRTV13

Kurzbeschreibung: In einem Online-Szenario soll ein wertvollstes Matching auf einem bipartiten Graphen gefunden werden. Dabei sind die Knoten auf der rechten Seite vorab bekannt, während die Knoten auf der linken Seite und die inzidenten Kanten nacheinander eintreffen. Eine frisch eingetroffene Kante muss entweder sofort oder gar nicht ins finale Matching aufgenommen werden. Durch eine Verallgemeinerung des *Secretary Problem* (bzw. dessen Lösung) erhält man hierfür einen guten Algorithmus.

⑳ Greedy Algorithms for Steiner Forest Ⓜ

Anupam Gupta, Amit Kumar:
STOC 2015: 871-878. http://dblp.org/rec/conf/stoc/Gupta015

Kurzbeschreibung: Das NP-harte Steinerbaumproblem lässt sich leicht durch Greedy-Algorithmen approximieren: Ein durch Prim oder Kruskal berechneter minimaler Spannbaum liefert bereits eine 2-Approximation. Die beiden Algorithmen lassen sich auch auf das Steinerwaldproblem übertragen: Hier wird verlangt, dass bestimmte Knotenpaare (in einem Baum des Waldes) verbunden sind. In dieser Arbeit wird gezeigt, dass der kruskal-artige Algorithmus immerhin einen konstanten Approximationsfaktor erreicht – dies gelang bis dahin nur mit linearer Programmierung.

Bemerkungen: Vorkenntnisse im Gebiet der Approximationsalgorithmen werden empfohlen.