

DISS. ETH NO. 16737

Descriptional and Computational Complexity in Relation to the Practicability of Computation and Computational Modes

A dissertation submitted to the
SWISS FEDERAL INSTITUTE OF TECHNOLOGY ZURICH

for the degree of
Doctor of Sciences

presented by

JOACHIM KUPKE

Dipl.-Inform., RWTH Aachen University, Germany

born Sep 18, 1978

citizen of Germany

accepted on the recommendation of

Prof. Juraj Hromkovič, examiner
Prof. Juhani Karhumäki, co-examiner
Prof. Peter Widmayer, co-examiner

2006

Abstract

In this dissertation, we will investigate lower bounds on the complexity of various problems. Depending on what kind of computational resource the term “complexity” refers to, establishing a lower bound, typically expressed asymptotically in some parameter, means to prove that a given task can only be solved if we are given as many computational resources as the lower bound indicates. Usually, the most valuable computational resource is time, and hence, complexity theory revolves around the idea of lower-bounding the amount of time necessary to solve a computational problem, e. g., an optimization problem.

Depending on the chosen mode of computation, this quantity will vary. Nondeterministic computations (i. e., randomized computations, with an arbitrarily small chance of success), like approximation algorithms (i. e., we put up with a feasible solution to an optimization problem which is not optimal, but only close to optimal), are generally believed to be strictly more powerful than deterministic exact computations. While for practical purposes, it is safe to make this assumption, theoretical research focuses on what, precisely, might constitute this alleged power of nondeterminism.

Accordingly, this dissertation is subdivided into two parts. Part I deals with the theoretical question of separating classes of finite automata of different “amounts” of nondeterminism, where the amount of nondeterminism is measured by means of the *ambiguity* of an automaton. It is a longstanding open problem whether there are regular languages such that automata of constant ambiguity would need to contain substantially more states than automata of polynomial ambiguity to recognize them. Here, we will outline a new approach to proving the existence of such a gap, including a new technique to lower-bound the number of states of constantly ambiguous automata. This discussion will involve *weighted* finite automata, and we will also have a look at their application in signal processing and how binary operations on digital signals represented by finite automata can and cannot, respectively, be evaluated.

Our goal in Part II is to investigate the approximability of some generalizations of TSP. The most important of these generalizations is TSP with time windows, where for each vertex of the input graph, an opening and a closing point of time for a window is given, and the respective vertex must be visited while the window is open. This generalized problem naturally and frequently appears in a number of applications, the most prominent of which may be vehicle routing.

By combining methods from the field of approximation algorithms with ideas from the theory of parameterized complexity, we obtain the concept of parameterized approximation, using which we devise tractable algorithms for some of our TSP generalizations.

Moreover, we prove that neither approximation nor parameterization alone is sufficient to this end.

But there is also a limit to parameterized approximation: We will establish large lower bounds on the approximability of TSP with time windows, even in the parameterized setting. More precisely, we will prove that while TSP remains approximable if the input deviates a little from the requirement to be metric, even fixed-parameter TSP with time windows becomes inapproximable in the same situation.

We will then turn to a novel paradigm of optimization, where in addition to the problem instance, a solution to a related instance (i. e., one that was modified locally to yield the current input instance) is given as part of the input. We will show how we can outperform all existing TSP algorithms, not only for metric, but also for near-metric inputs. However, as for TSP with time windows, knowing an optimal solution to a related input instance does not help a single bit: We will establish precisely the same lower bounds on its approximability, using however different tools.

Zusammenfassung

In dieser Dissertation werden wir untere Schranken auf die Komplexität verschiedener Probleme untersuchen. In Abhängigkeit davon, auf welche Art von für eine Berechnung relevante Ressource sich der Begriff „Komplexität“ bezieht, bedeutet die Aufstellung einer unteren Schranke, die typischerweise in einem gewissen Parameter asymptotisch ausgedrückt wird, zu beweisen, dass eine gegebene Aufgabe nur gelöst werden kann, wenn die berechnungsrelevante Ressource in der durch die untere Schranke angegebenen Menge zur Verfügung steht. Für gewöhnlich ist die wertvollste Ressource bei Berechnungen die Zeit, und folglich dreht sich die Komplexitätstheorie um die Vorstellung, eine untere Schranke auf die zur Lösung eines algorithmischen Problems, etwa eines Optimierungsproblems, notwendige Zeit anzugeben.

In Abhängigkeit des gewählten Modus der Berechnung wird diese Grösse unterschiedlich ausfallen. Von nichtdeterministischen Berechnungen (d. h. randomisierten Berechnungen, bei denen die Erfolgswahrscheinlichkeit beliebig klein sein darf) glaubt man, genau wie von Approximationsalgorithmen (d. h., man gibt sich mit einer lediglich nahezu optimalen, jedoch nicht optimalen zulässigen Lösung zu einem Optimierungsproblem zufrieden), dass sie strikt stärker als deterministische exakte Berechnungen sind. Während man aus praktischen Erwägungen diese Annahme als gesichert betrachten kann, legt die theoretische Forschung den Schwerpunkt darauf, was genau diese unterstellte Stärke des Nichtdeterminismus ausmacht.

Dementsprechend ist diese Dissertation in zwei Teile gegliedert. Teil I behandelt die theoretische Fragestellung, Klassen endlicher Automaten verschiedener „Mengen“ von Nichtdeterminismus zu separieren, wobei die Menge von Nichtdeterminismus anhand der *Mehrdeutigkeit* eines Automaten gemessen wird. Es ist ein seit langer Zeit offenes Problem, ob es derartige reguläre Sprachen gibt, dass Automaten konstanter Mehrdeutigkeit eine beträchtlich grössere Anzahl von Zuständen enthalten müssten als Automaten polynomialer Mehrdeutigkeit, um sie zu erkennen. Hier werden wir einen neuen Ansatz umreissen, um die Existenz einer derartigen Lücke zu beweisen, einschliesslich einer neuen Technik, um die Mindestzustandszahl konstant mehrdeutiger Automaten abzuschätzen. Diese Diskussion wird *gewichtete* endliche Automaten involvieren, deren Anwendung in der Signalverarbeitung, sowie die Frage, welche binäre Operationen auf durch endliche Automaten dargestellten digitalen Signalen wie ausgewertet werden können und welche nicht, wir uns ebenfalls ansehen werden.

Unsere Absicht in Teil II ist es, die Approximierbarkeit einiger Verallgemeinerungen des TSP zu untersuchen. Um die wichtigste dieser Verallgemeinerungen handelt es sich bei TSP mit Zeitfenstern, bei dem zu jedem Knoten des eingegebenen Graphen ein Öffnungs-

und ein Schliessungszeitpunkt eines Fensters gegeben ist, so dass der betreffende Knoten besucht werden muss, während das Fenster offen ist. Dieses verallgemeinerte Problem tritt auf natürliche Weise und häufig in einer Reihe von Anwendungen auf, von denen die prominenteste die Streckenführung von Fahrzeugen sein könnte.

Indem wir Methoden aus dem Feld der Approximationsalgorithmen mit Ideen aus der Theorie der parametrisierten Komplexität verbinden, erhalten wir das Konzept der parametrisierten Approximation, mit Hilfe dessen wir für einige unserer TSP-Verallgemeinerungen praktikable Algorithmen entwerfen. Darüber hinaus beweisen wir, dass, um dieses Ziel zu erreichen, weder Approximation noch Parametrisierung alleine ausreichen.

Jedoch ist auch der parametrisierten Approximation eine Grenze gesetzt: Wir werden hohe untere Schranken auf die Approximierbarkeit von TSP mit Zeitfenstern aufstellen, sogar in der parametrisierten Situation. Genauer gesagt werden wir beweisen, dass, während TSP approximierbar bleibt, wenn die Eingabe etwas von der Anforderung abweicht, metrisch zu sein, TSP mit Zeitfenstern, sogar wenn der Parameter fixiert wird, unter denselben Umständen inapproximierbar wird.

Wir werden uns danach einem neuartigen Optimierungsparadigma zuwenden, bei dem zusätzlich zu der Problem Instanz eine Lösung zu einer verwandten Instanz (d. h. zu einer, aus der die jetzige Eingabeinstanz durch eine lokale Modifikation hervorgeht) als Teil der Eingabe gegeben ist. Wir werden zeigen, wie wir die Approximationsgarantien aller bekannten TSP-Algorithmen übertreffen können, und zwar nicht nur für metrische, sondern auch für beinahe-metrische Eingaben. Was jedoch TSP mit Zeitfenstern anbelangt, so hilft einem die Kenntnis einer optimalen Lösung zu einer verwandten Eingabeinstanz nicht im geringsten weiter: Wir werden genau die gleichen unteren Schranken auf seine Approximierbarkeit aufstellen, allerdings unter Zuhilfenahme anderer Werkzeuge.