A medley for computational complexity: With applications of information theory, learning theory, and Ketan Mulmuley's parametric complexity technique
Loff, B.F.L.

Citation for published version (APA):
Abstract

This thesis contains four parts. Each part studies a topic within computational complexity by applying techniques from other fields in theoretical computer science.

In Chapter 1 we will use Kolmogorov complexity to study probabilistic polynomial-time algorithms. Let $R$ denote the set of Kolmogorov-random strings, which are those strings $x$ whose Kolmogorov complexity $K(x)$ is at least as large as their length $|x|$. There are two main results. First, we show that any bounded-error probabilistic polynomial-time algorithm can be simulated by a deterministic polynomial-time algorithm that is allowed to make non-adaptive queries to $R$. Second, we show that for a time-bounded analogue of $R$ (defined using time-bounded Kolmogorov complexity), it holds that any polynomial-time algorithm that makes non-adaptive queries to $R$ can be simulated both in polynomial space and by circuits of polynomial size.

This indicates that we are near to an alternative characterization of probabilistic polynomial-time as being exactly deterministic polynomial-time with non-adaptive queries to $R$. Such characterizations ultimately aim at using techniques from Kolmogorov complexity and computability to study the relationship between different complexity classes. As can be expected, the proofs in this chapter make essential use of such techniques.

In Chapter 2 we make an effort at extending Mahaney’s theorem [74] to more general reductions, or — seen another way — strengthening the Karp-Lipton theorem [64] to a stronger collapse of the polynomial-time hierarchy. Mahaney’s theorem states that if SAT is $m$-reducible to a sparse set, then $P = NP$, and the Karp-Lipton theorem (more precisely, the strengthened version of Cai [40]) says that if SAT is Turing-reducible to a sparse set, then $PH \subseteq ZPP^{NP}$.

We prove that if a class of functions $C$ has a polynomial-time learning algorithm in Angluin’s bounded error learning model, then if SAT is $m$-reducible to $C$, it follows that $PH \subseteq P^{NP}$.

Then from the existence of such an algorithm for linear-threshold functions, we conclude that if SAT is $m$-reducible to a linear-threshold function, then $PH \subseteq P^{NP}$. It will be seen that both disjunctive and majority truth-table (non-adaptive) reductions to sparse sets are a special case of $m$-reductions to linear-threshold functions, and hence our results hold also for these kinds of
reductions.

We also prove a somewhat stronger result of independent interest. For such a class of functions $C$, it holds that if $\text{Sat}$ $m$-reduces to $C$, then we can answer any number of $\text{Sat}$-queries of length $n$ by asking only $n$ (larger) queries to $\text{Sat}$.

There are two main results in Chapter 3. First, we prove a more refined $\text{NP}$-hardness result for knapsack and related problems. We will construct a reduction from the satisfiability of fan-in-2 circuits of size $S$ with $k$ input bits to an instance of the subset-sum problem having bit-length $\mathcal{O}(S + k)$. A corollary of this is a simple proof that there is no approximation algorithm for the knapsack problem which gives a better-than-inverse-polynomial approximation ratio, unless the exponential-time hypothesis of Impagliazzo and Paturi [55] fails to hold.

Secondly, we will use the technique we just developed, together with Ketan Mulmuley’s parametric complexity technique, in order to prove an unconditional lower bound in Mulmuley’s parallel semi-algebraic PRAM model [77]. We will show that, in that model, there is no algorithm for solving the knapsack problem in time $o(x^{1/4})$ using $2^{o(n^{1/4})}$ processors, even when the bit-length of the weights is restricted to $n$. The significance of this result follows from the fact that pretty much every known parallel algorithm can be implemented in this model.

In Chapter 4, we turn to communication complexity and information complexity [28]. We prove several theorems: (1) we show a “Reverse Newman’s Theorem”, stating that a private-coin $q$-round protocol that reveals $I$ bits of information can be simulated by a public-coin $q$-round protocol that reveals $I + \tilde{O}(q)$ bits of information; (2) we show that public-coin protocols that reveal $I$ bits of information can be simulated by protocols that communicate only $\tilde{O}(I)$ bits (but possibly use many more rounds); (3) we prove a constant-round two-way variant of the Slepian–Wolf theorem, and use it to show that $q$-round public-coin protocols that reveal $I$ bits of information can be simulated by protocols that communicate only $O(I) + \tilde{O}(q)$ bits on average, and make use of $O(q)$ rounds, also on average; and (4) as a consequence of (1) and (3) we show a direct-sum theorem for bounded-round protocols, which states that, for any function $f$ which needs $C$ bits of (average) communication to be computed by randomized protocols in $O(q)$-average-rounds, computing $k$ copies of the function using a $q$-round protocol requires $\Omega(kC) - \tilde{O}(q)$ bits of communication.