Information and Computation 261 (2018) 281–295

Contents lists available at ScienceDirect

Information and Computation

www.elsevier.com/locate/yinco

Local reduction [☆]

Hamidreza Jahanjou [a,*], Eric Miles [b,1], Emanuele Viola [a]

[a] Northeastern University, Boston, MA, United States
[b] Google Inc., Los Angeles, CA, United States

ARTICLE INFO

Article history:
Received 21 June 2015
Available online 8 February 2018

Keywords:
Computational complexity
Cook–Levin
Reductions
ACC0

ABSTRACT

We reduce non-deterministic time $T \geq 2^n$ to a 3SAT instance ϕ of quasilinear size $|\phi| = T \cdot \log^{O(1)} T$ such that there is an explicit NC^0 circuit C that encodes ϕ in the following way: on input a $(\log|\phi|)$-bit index i, C outputs the ith clause of ϕ. The previous best result was C in NC^1. Even in the simpler setting of polynomial size ($|\phi| = \mathrm{poly}(T)$), the previous best result was C in AC^0.

More generally, for any time $T \geq n$ and parameter $r \leq n$ we obtain $|\phi| = \max(T, 2^{n/r}) \cdot (n \log T)^{O(1)}$, and each output bit of C is a decision tree of depth $O(\log r)$.

As an application, we tighten Williams' connection between satisfiability algorithms and circuit lower bounds (STOC 2010; SIAM J. Comput. 2013).

© 2018 Elsevier Inc. All rights reserved.

1. Introduction and our results

The efficient reduction of arbitrary non-deterministic computation to 3SAT is a fundamental result with widespread applications. For many of these, two aspects of the efficiency of the reduction are at a premium. The first is the length of the 3SAT instance. A sequence of works shows how to reduce non-deterministic time-T computation to a 3SAT instance ϕ of quasilinear size $|\phi| = \tilde{O}(T) := T \log^{O(1)} T$ [1–6]. More recent works obtain reductions with the same parameters which, in addition, introduce a "gap" (and which give rise to probabilistically checkable proofs) [7–10].

The second aspect is the computational complexity of producing the 3SAT instance ϕ given a machine M, an input $x \in \{0,1\}^n$, and a time bound $T = T(n) \geq n$. It is well-known and easy to verify that a ϕ of size $\mathrm{poly}(T)$ is computable even by circuits from the restricted class NC^0. More generally, Agrawal, Allender, Impagliazzo, Pitassi, and Rudich show [11] that such NC^0 reductions exist whenever AC^0 reductions do.

A stronger requirement on the complexity of producing ϕ is critical for many applications. The requirement may be called *clause-explicitness*. It demands that the ith clause of ϕ be computable, given $i \leq |\phi|$ and $x \in \{0,1\}^n$, with resources $\mathrm{poly}(|i|) = \mathrm{poly}\log|\phi| = \mathrm{poly}\log T$. In the case $|\phi| = \mathrm{poly}(T)$, this is known to be possible by an unrestricted circuit D of size $\mathrm{poly}(|i|)$. (The circuit has either random access to x, or, if $T \geq 2^n$, it may have x hardwired.) As a corollary, so-called succinct versions of NP-complete problems are complete for NEXP. Arora, Steurer, and Wigderson [12] note that the circuit D may be taken from the restricted class AC^0. They use this to argue that, unless EXP = NEXP, standard NP-complete graph problems cannot be solved in time $\mathrm{poly}(2^n)$ on graphs of size 2^n that are described by AC^0 circuits of size $\mathrm{poly}(n)$.

[☆] All authors were supported by NSF grants CCF-0845003, CCF-1319206.

[*] Corresponding author.

E-mail addresses: hamid@ccs.neu.edu (H. Jahanjou), enmiles@gmail.com (E. Miles), viola@ccs.neu.edu (E. Viola).

[1] Research performed while a student at Northeastern University.

https://doi.org/10.1016/j.ic.2018.02.009
0890-5401/© 2018 Elsevier Inc. All rights reserved.

Interestingly, applications to unconditional complexity lower bounds rely on reductions that are clause-explicit and simultaneously optimize the length of the 3SAT instance ϕ and the complexity of the circuit D computing clauses. For example, the time–space tradeoffs for SAT need to reduce non-deterministic time T to a 3SAT instance ϕ of quasilinear size $\tilde{O}(T)$ such that the ith clause is computable in time $\text{poly}(|i|) = \text{poly}\log|\phi|$ and space $O(\log|\phi|)$, see e.g. [13] or Van Melkebeek's survey [14]. More recently, the importance of optimizing both aspects of the reduction is brought to the forefront by Williams' approach to obtain lower bounds by satisfiability algorithms that improve over brute-force search by a super-polynomial factor [15–19]. To obtain lower bounds against a circuit class C using this technique, one needs a reduction of non-deterministic time $T = 2^n$ to a 3SAT instance of size $\tilde{O}(T)$ whose clauses are computable by a circuit D of size $\text{poly}(n)$ that belongs to the class C. For example, for the ACC^0 lower bounds [16,19] one needs to compute them in ACC^0. However it has seemed "hard (perhaps impossible)" [16] to compute the clauses with such restricted resources.

Two workarounds have been devised [16,18]. Both exploit the fact that, under an assumption such as $\text{P} \subseteq \text{ACC}^0$, non-constructively there does exist such an efficient circuit computing clauses; the only problem is constructing it. They accomplish the latter using either nondeterminism [16] or brute-force [18] (cf. [20]). The overhead in these arguments limits the consequences of satisfiability algorithms: before this work, for a number of well-studied circuit classes C (discussed later) a lower bound against C did not follow from a satisfiability algorithm for circuits in C.

1.1. Our results

We show that, in fact, it is possible to reduce non-deterministic computation of time $T \geq 2^n$ to a 3SAT formula ϕ of quasilinear size $|\phi| = \tilde{O}(T)$ such that given an index of $\ell = \log|\phi|$ bits to a clause, one can compute (each bit of) the clause by looking at a constant number of bits of the index. Functions for which each output bit is computed from a constant number of input bits are also known as local, NC^0, or junta. More generally our results give a trade-off between $|\phi|$ and the decision-tree depth needed to compute C's output bits. The results apply to any time bound T, paying an inevitable loss of $O(|x|) = O(n)$ for T close to n.

Theorem 1 *(Local reductions). Let M be an algorithm running in time $T = T(n) \geq n$ on inputs of the form (x, y) where $|x| = n$. Let $r \leq n$ be any value. Then, for every $x \in \{0,1\}^n$, there is a 3CNF ϕ of size $|\phi| = \max(T, 2^{n/r}) \cdot (n \log T)^{O(1)}$ such that the following holds.*

1. *ϕ is satisfiable iff there is $y \in \{0,1\}^T$ such that $M(x, y) = 1$.*
2. *There is a circuit D with input length $\log|\phi|$ such that $D(i)$ outputs the ith clause of ϕ and each output bit of D is a decision tree of depth $O(\log r)$. Further, D can be constructed from x in time $\text{poly}(n, \log T)$.*

Note that for $T = 2^{\Omega(n)}$, by setting $r := n/\log T$ we get that D is in NC^0 and ϕ has size $T \cdot \log^{O(1)} T$. We also point out that the only place where decision-tree depth $O(\log r)$ (as opposed to $O(1)$) is needed in D is to select bits of the string x.

The previous best result was D in NC^1 [7]. Even in the simpler setting of $|\phi| = \text{poly}(T)$ the previous best result was D in AC^0 [12].

Tighter connections between satisfiability and lower bounds. The quest for non-trivial satisfiability algorithms has seen significant progress recently, see e.g. [16,21–25]. Our results lower the bar for obtaining new circuit lower bounds from such algorithms. Previously, a lower bound for circuits of depth d and size s was implied by a satisfiability algorithm for depth $c \cdot d$ and size s^c for a constant $c > 1$ (for typical settings of s and d). With our proof it suffices to have a satisfiability algorithm for depth $d + c$ and size $c \cdot s$ for a constant c. These results can be extended and optimized for several well-studied circuit classes. In particular we obtain the following new connections.

Corollary 2. *For each of the following classes C, if the satisfiability of circuits in C can be solved in time $2^n/n^{\omega(1)}$ then there is a problem $f \in \text{E}^{\text{NP}}$ that is not solvable by circuits in C:*

(1) linear-size circuits,
(2) linear-size series-parallel circuits,
(3) linear-size log-depth circuits,
(4) quasi-polynomial-size SYM-AND circuits.

Recall that available size lower bounds for unrestricted circuits are between $3n - o(n)$ and $5n - o(n)$, depending on the basis [26–28]. Although Corollary 2 and Corollary 3 below are stated in terms of linear-size circuits, the proofs provide a close correspondence between the running time for satisfiability and the parameters of the circuit class. In particular, the constant hidden by the circuit size in class (1) can be optimized, as discussed in the paragraph "Subsequent work" below. At the moment this approach does not match known lower bounds, due to the (in)efficiency of known satisfiability algorithms.

In 1977 Valiant [29] focused attention on classes (2) and (3). (Some missing details about series-parallel graphs are provided in [30].) The class (4) contains ACC [31,32], and can be simulated by number-on-forehead protocols with a polylogarithmic number of players and communication [33]. Williams [16] gives a quasilinear-time algorithm to evaluate a SYM-AND circuit on all inputs.

Download English Version:

https://daneshyari.com/en/article/6873781

Download Persian Version:

https://daneshyari.com/article/6873781

Daneshyari.com