Information and Computation 251 (2016) 252–262

Contents lists available at ScienceDirect

Information and Computation

www.elsevier.com/locate/yinco

A generalization of Spira's theorem and circuits with small segregators or separators

Anna Gál [a,*,1], Jing-Tang Jang [b,2]

[a] Dept. of Computer Science, University of Texas at Austin, Austin, TX 78712-1188, USA
[b] Google, Mountain View, CA, USA

ARTICLE INFO

Article history:
Received 27 June 2013
Received in revised form 16 August 2016
Available online 30 September 2016

Keywords:
Boolean circuits
Circuit size
Circuit depth
Spira's theorem
Space complexity

ABSTRACT

Spira showed that any Boolean formula of size s can be simulated in depth $O(\log s)$. We generalize Spira's theorem and show that any Boolean *circuit* of size s with segregators (or separators) of size $f(s)$ can be simulated in depth $O(f(s)\log s)$. This improves and generalizes a simulation of polynomial-size Boolean circuits of constant treewidth k in depth $O(k^2 \log n)$ by Jansen and Sarma. Our results imply that the class of languages computed by non-uniform families of polynomial-size circuits with constant size segregators equals non-uniform NC^1.

As a corollary, we show that the Boolean Circuit Value problem for circuits with constant size segregators is in deterministic $SPACE(\log^2 n)$. Our results also imply that the Planar Circuit Value problem, which is known to be P-Complete, is in $SPACE(\sqrt{n}\log n)$; and that the Layered Circuit Value and Synchronous Circuit Value problems, which are both P-complete, are in $SPACE(\sqrt{n})$.

1. Introduction

Spira [37] proved the following theorem.

Theorem A. *[37] Let F be any Boolean formula of size s. Then F can be simulated by an equivalent formula of depth $O(\log s)$.*

There are several results improving or extending Spira's theorem. Bonet and Buss [4] improved the constants in the depth bounds and the size of the simulation for Boolean formulas. Spira originally considered formulas over the $\{\wedge, \vee, \neg\}$ basis. Savage [35] generalized the result to all complete bases. Wegener [39] proved the statement for monotone Boolean formulas. Brent [6], Bshouty et al. [7] extended it to arithmetic formulas. All these results study formulas, i.e. tree-like circuits with fan-out 1.

Valiant, Skyum, Berkowitz and Rackoff [38] showed that arithmetic circuits of size s and degree d can be simulated by arithmetic circuits of size $O((sd)^{O(1)})$ and $O(\log s \log d)$ depth. However, very little is known for size vs. depth for general Boolean circuits. The strongest results so far for general Boolean circuits by Paterson and Valiant [29], and Dymond and Tompa [15] give a simulation of arbitrary Boolean circuits of size s in depth $O(s/\log s)$.

* Corresponding author.
E-mail addresses: panni@cs.utexas.edu (A. Gál), keithjtjang@google.com (J.-T. Jang).
1 Supported in part by NSF Grant CCF-1018060.
2 Supported in part by MCD fellowship from Dept. of Computer Science, University of Texas at Austin, and NSF Grant CCF-1018060.

http://dx.doi.org/10.1016/j.ic.2016.09.008
0890-5401/

In this paper, we generalize Spira's technique to circuits with small segregators or small separators. Informally, the separator of a graph is a subset of the nodes whose removal yields two subgraphs of comparable sizes. (See the following section for a formal definition.) Graphs with small separators include trees, planar graphs [26], graphs with bounded genus [19], graphs with excluded minors [1], as well as graphs with bounded treewidth [33].

Segregators are a relaxed version of separators of directed acyclic graphs. Paul et al. [30], and Santhanam [34] used segregators to study the computation graph of Turing machines. Directed acyclic graphs with small separators also have small segregators, but the reverse may not necessarily hold. See Section 2 for more details.

Jansen and Sarma [22] studied the question of simulating Boolean circuits with bounded treewidth by small-depth circuits. They showed that polynomial-size circuits with constant treewidth k can be simulated in depth $O(k^2 \log n)$, and thus the class of languages with non-uniform polynomial-size bounded treewidth circuits equals non-uniform NC^1.

We extend this result to arbitrary circuits with small segregators and show that any Boolean circuit of size s with segregators (or separators) of size $f(s)$ can be simulated in depth $O(f(s)\log s)$. For circuits with segregators of size k, thus also for graphs with treewidth k, this gives a simulation in depth $k \log s$, improving the bound in [22]. If the segregator size is at least s^{ε} for some constant $\varepsilon > 0$, then we can obtain a simulation of depth $O(f(s))$. Our results imply that the class of languages computed by non-uniform families of polynomial-size circuits that have constant-size segregators equals non-uniform NC^1.

Barrington [2] showed that the class of languages decided by branching programs of polynomial size and constant width is the same as NC^1 (in both uniform and non-uniform settings). In the non-uniform setting, our results together with Barrington's result imply that the class of languages decided by constant-width branching programs of polynomial size is the same as the class of languages decided by polynomial-size circuits with constant-size segregators.

In [17] we observed that the two-person pebble game of Dymond and Tompa can be used to simulate circuits with small separator size in small depth, giving essentially the same dependence of the depth on the separator size as in the current paper. The approach in [17] based on the two person pebble game can also be extended to graphs with small segregators. However, our simulation based on the two person pebble game is non-uniform, and it seems that the resulting circuits cannot be produced efficiently using this approach. Jansen and Sarma's [22] simulation of bounded treewidth circuits is also non-uniform.

For circuits with constant-size segregators or separators, the simulating circuits we obtain in this paper can be generated in space $O(\log^2 s)$. We also note that our simulation works for any circuit, and if the circuit has a segregator of size $f(s)$, we obtain a simulating circuit of depth at most $O(f(s)\log s)$. The value $f(s)$ does not have to be provided in advance. In contrast, the simulation in [22] assumes that the treewidth k is known in advance, and a tree decomposition is available along with the description of the circuit to be simulated. It would be desirable to generate the simulating circuits even more efficiently with respect to space or circuit depth, especially in the case of polynomial-size circuits with constant-size segregators or separators, since in that case, as in the case of formulas in Spira's theorem, the resulting circuits are NC^1 circuits. Note however, that even in the case of formulas (tree-like circuits) Spira's theorem is non-uniform. In the uniform setting, it was shown that for Boolean formulas presented as parenthesized expressions the Boolean Formula Value Problem is in $SPACE(\log n)$ [27], and in $DLOGTIME$-uniform NC^1 [8,9]. For Boolean formulas presented as tree-like circuits, [10] showed that the Boolean Formula Value Problem can be solved in $SPACE(\log n)$ and in $ALOGTIME$.

We also consider the space complexity of the Circuit Value Problem (CVP). Ladner [24] showed that the Circuit Value Problem is P-complete. The space complexity of the CVP is not known to be $o(n/\log n)$ for general Boolean circuits. There are only a few restricted versions of the Circuit Value Problem that have been previously shown to have small-space complexity. It is a straightforward consequence of Borodin's theorem [5] (see Theorem C) that the CVP for logspace uniform depth d circuits is in $SPACE(d)$ for $d \geq \log n$. It is also easy to see that the CVP for small-width circuits can be solved in small space. The Monotone Planar Circuit Value Problem (MPCVP) is another restricted version of CVP with small-space complexity, where the circuits only have $\wedge$ and $\vee$ gates, positive input literals, and can be embedded on the plane without crossings. There are numerous results on this problem. The strongest result so far is by Limaye, Mahajan, and Sarma [25], who showed that MPCVP is in SAC^2. Also see [12,41,31] for results on MPCVP in the PRAM model. Stronger bounds are known for some restricted versions of MPCVP [11,20,14,3,23,32,41].

Our generalization of Spira's theorem allows us to bound the space complexity of the Circuit Value Problem for circuits with small separators and segregators. We show that the Boolean Circuit Value Problem for circuits with constant-size segregators (or separators) is in deterministic $SPACE(\log^2 n)$. Our results also imply that the Planar Circuit Value problem, which is known to be P-Complete [20], is in $SPACE(\sqrt{n}\log n)$.

In addition we show that the Layered Circuit Value and the Synchronous Circuit Value problems, which are both P-complete [21], are in $SPACE(\sqrt{n})$. However, since layered circuits and synchronous circuits do not necessarily have small separators or segregators, instead of using our generalization of Spira's theorem we use a different approach.

2. Preliminaries

2.1. Space bounded turing machines

For the space complexity of Turing machines, we follow the convention of considering Turing machines with a separate *read-only* input tape, and additional work tapes. If the machine has to produce an output string (instead of just accepting or