Small Pseudo-Random Sets Yield Hard Functions: New Tight Explicit Lower Bounds for Branching Programs
(Extended Abstract)

Alexander E. Andreev\(^1\), Juri L. Baskakov\(^2\),
Andrea E.F. Clementi\(^3\), and José D.P. Rolim\(^4\)
\(^1\) LSI Logic, California, andreev@lsi.com
\(^2\) Dept. of Mathematics, University of Moscow, baskakov@matis.math.msu.su
\(^3\) Dept. of Mathematics, University “Tor Vergata” of Rome clementi@mat.uniroma2.it
\(^4\) Centre Universitaire d’Informatique, University of Geneva, CH, jose.rolim@cui.unige.ch

Abstract. In several previous works the construction of a computationally hard function with respect to a certain class of algorithms or Boolean circuits has been used to derive small pseudo-random spaces. In this paper, we revert this connection by presenting two new direct relations between the efficient construction of pseudo-random (both two-sided and one-sided) sets for Boolean affine spaces and the explicit construction of Boolean functions having hard branching program complexity.

In the case of 1-read branching programs (1-Br:Pr.), we show that the construction of non trivial (i.e. of cardinality \(2^{o(n)}\)) discrepancy sets (i.e. two-sided pseudo-random sets) for Boolean affine spaces of dimension greater than \(n/2\) yield a set of explicit Boolean functions having very hard 1-Br:Pr. size. By combining the best known construction of \(\epsilon\)-biased sample spaces for linear tests and a simple “Reduction” Lemma, we derive the required discrepancy set and obtain a Boolean function in \(P\) having 1-Br:Pr. size not smaller than \(2^{n-O(\log^2 n)}\) and a Boolean function in DTIME\((2^{O(\log^2 n)})\) having 1-Br:Pr. size not smaller than \(2^{n-O(\log n)}\). The latter bound is optimal and both of them are exponential improvements over the best previously known lower bound that was \(2^{n-3n^{1/2}}\) [21].

As for non-deterministic syntactic \(k\)-read branching programs (\(k\)-Br:Pr.), we introduce a new method to derive explicit, exponential lower bounds that involves the construction of hitting sets (one-sided pseudo-random sets) for affine spaces of dimension \(o(n/2)\). Using an appropriate “orthogonal” representation of small Boolean affine spaces, we efficiently construct these hitting sets thus obtaining an explicit Boolean function in\(P\) that has \(k\)-Br:Pr. size not smaller than \(2^{n^{1-o(1)}}\) for any \(k = o\left(\frac{\log n}{\log \log n}\right)\).

This improves over the previous best known lower bounds given in [8,17,11] for some range of \(k\).
1 Introduction

A Branching Program (Br. Pr.) is a directed acyclic graph where one of the nodes, called source, has fan-in 0 and some other nodes, called terminals, have fan-out 0. Each non terminal node is labelled by the index of an input Boolean variable and has fan-out 2. The two edges leaving a non terminal node are respectively labelled 0 and 1. A Br. Pr. computes a Boolean function $f : \{0, 1\}^n \rightarrow \{0, 1\}$ on a fixed input as follows. Starting the computation from the source node, if a generic node is reached, the corresponding input variable is tested and the computation chooses the edge corresponding to the actual value of this input variable. The process terminates when a sink node is reached and its label represents the output of $f$. Clearly, the Br. Pr. model can also be made non deterministic by allowing the existence of more edges labeled with the same value that leave the same node. The size of Br. Pr. is the number of its nodes and the length is the maximum length of a computation path from the source to a sink.

Theoretical research in Br. Pr.’s is extremely active since they represent an important abstraction of several computing models [23, 18]. The size and the length of a given Br. Pr. provide a measure of, respectively, the space and the time of the corresponding computation. In particular, a super-polynomial lower bound for the Br. Pr. size of a function $f$ would imply that $f$ is not computable in non-uniform log-space. However, the best known lower bound for explicit functions in NP is $\Omega(n^2 / \log^2 n)$ [15]. Another important aspect of this model lies in the study of time-space trade-offs where finding an explicit function $f$ that requires non-uniform super polynomial size Br. Pr.’s of linear length constitutes one of the most important open problems (see [7] for a survey of results and open problems in this field). This result in fact would separate the non-uniform log-space class from the linear-time one (this result is presently available only for uniform computations [9]).

Informally speaking, a linear upper bound on the length of a Br. Pr. implies that the average number of times that a fixed variable is tested during the program is only constant. This simple observation has led researchers [8, 11, 19, 20] to investigate the restricted variants of Br. Pr.’s that consider this property with the aim to achieve better lower bounds.

A read-$k$-times Br. Pr. is allowed to read each variable at most $k$ times along any valid computation and, in a syntactic read-$k$-times Br. Pr., this reading restriction holds for any path from the source to any sink. Notice that while a read-1-time Br. Pr. is always a syntactic read-1-time Br. Pr., for $k \geq 2$ this does not hold.

To our knowledge, the best known explicit lower bound for read-1-time-branching programs (1-Br. Pr.’s) due to Savicky and Zak [20, 21]; they derived a Boolean function in P having, for any sufficiently large $n$, 1-Br. Pr. size not smaller than $2^{n-s}$, where $s = O(n^{1/2})$. We also emphasize that for this model there are no known harder explicit functions even in classes larger than P, while a non explicit lower (and optimal) bound of size $\Theta(2^{n-\log n})$ is known.

Concerning non-deterministic syntactic read-$k$-times Br. Pr.’s ($k$-Br. Pr.’s) in the case $k \geq 2$, there are explicit lower bounds of exponential size when