Randomized OBDDs and the Model Checking*

Marek Karpinski†

Abstract

We present some recent results on the computational power and the basic manipulation properties of the *randomized OBDDs* (or equivalently, *randomized read-once ordered branching programs*). We discuss here their utilizing properties for *randomized* formal verification and the model checking.

*Invited paper at the Workshop Probabilistic Methods in Verification, PROBMIV'98, Indianapolis, June 19 – 20, 1998.

†Dept. of Computer Science, University of Bonn, 53117 Bonn. Research partially supported by the International Computer Science Institute, Berkeley, California, by the DFG Grant KA 673/4-1, and by the ESPRIT BR Grants 7097, 21726, and EC-US 030, by DIMACS, and by the Max–Planck Research Prize.

Email: marek@cs.uni-bonn.de , URL: http://theory.cs.uni-bonn.de/~marek/ .

Branching programs have recently been found very useful in the field of formal verification and model checking for both hardware and software applications. The main problem of formal verification is to check whether a hardware circuit or a program has been correctly designed. A standard approach employed today is to transform independently the circuit or the program and their function specification to the common intermediate representation, and then check their equivalence. A model for intermediate representation must enjoy several formal manipulation properties, like the closure under boolean combinations, and the existence of an (efficient) algorithm for its satisfiability. A most commonly used model for the intermediate representation today is the model of an OBDD ("ordered binary decision diagram", or equivalently, *read-once ordered branching program*). The obvious boolean combination properties of OBDDs are necessary for the bottom-up algorithm that construct the OBDD's representation from the circuit or the program description. This strategy of using OBDDs for verification has an apparent shortcoming, in that we cannot hope to compute in general a *small* (*polynomial* in the *size* of the original circuit) representation. We recall that even unrestricted polynomial size branching programs compute the functions which are in non-uniform logspace. The above problem has largely been accepted as inherent, and not critical for this approach, since the functions to be transformed tend to be simple, and be computable in logspace. Independently, during the last decade there were several attemps to find manipulable generalizations of OBDDs for formal verification, strong enough to compute efficiently more complex funcitons.

The model of *randomized OBDDs* have been introduced recently by Ablayev and Karpinski [AK96], and proven to be exponentially more powerful than the classical model of the deterministic OBDDs. Surprisingly the exponential size advantage generalized even to nondeterministic read-k-times ordered branching programs [AK98a] (cf. also, [S97], [T98]).

One of the most important functions, and at the same time an elementary bottleneck, in formal deterministic verification, and hardware model checking, is the *integer multiplication* (cf., e.g. [P95]). It is well known that

computing the integer multiplication requires exponential size on deterministic read-k-times ordered branching programs even if $k = o(\log n)$, cf. [B91], [BSSW93], and [G94]. Ablayev and Karpinski [AK98b] succeeded in designing a *small* (polynomial size) *randomized* OBDDs for testing the function of integer multiplication, and proving at the same time an exponential lower bound on the size of any randomized OBDD computing exactly the integer multiplication. Interestingly, it is known that computing the test for integer multiplication with deterministic OBDDs is as hard as integer factorization [W94]. We discuss also some direct applications and extensions of the above results towards the formal verification of other functional problems.

It is not difficult to see that randomized OBDDs are closed under the boolean combinations. So the important algorithmic issues arise in formal verification, and the model checking on randomized OBBDs, namely, their *satisfiability* and *equivalence* problems. Agrawal and Thierauf [AT97] have proven recently that the general *satisfiability* (and *equivalence*) problem for randomized OBDDs is NP-complete, displaying at the same time, a *polynomial time* algorithm for satisfiability for randomized OBDDs with the very small (bounded by the inverse of their width) error probability. We discuss further some randomized algorithmic issues arising from removal of the variable ordering conditions in our model of randomized OBDDs, and also from the extensions to algebraic (branching condition) OBDDs similar to algebraic decision trees (cf. [GKY95], [GKMS96], [GKS97]). Very little is known about the restricted branching program with algebraic decision elements, despite their potential applications in algebraic and numeric computation, combinatorial optimization, and algorithmic geometry. Some open problems are presented.

References

[AK96] F. Ablayev and M. Karpinski, *On the Power of Randomized Branching Problems*, Proc. ICALP'96, LNCS 1099, Springer, 1996, pp. 348–356.

[AK98a] F. Ablayev and M. Karpinski, *On the Power of Randomized Ordered Branching Programs*, ECCC TR98-004 (1998), available at `http://www.eccc.uni-trier.de/eccc/` .

[AK98b] F. Ablayev and M. Karpinski, *A Lower Bound for Integer Multiplication on Randomized Read-Once Branching Programs*, ECCC TR98-011 (1998), available at `http://www.eccc.uni-trier.de/eccc/` .

[AT97] M. Agrawal and T. Thierauf, *The Satisfiability Problem for Probabilistic Ordered Branching Programs*, ECCC TR97-060 (1997), available at `http://www.eccc.uni-trier.de/eccc/` .

[BRS93] A. Borodin, A. Razborov and R. Smolensky, *On Lower Bounds for Read-k-Times Branching Programs*, Computational Complexity **3** (1993), pp. 1–18.

[BSSW93] B. Bollig, M. Sauerhoff, D. Sieling and I. Wegener, *Read k-Times Ordered Binary Decision Diagrams-Efficient Algorithms in the Presence of Null-Chains*, Technical Report Nr. 474, Univ. Dortmund, 1993.

[B91] B. Bryant, *On the Complexity of VLSI Implementations and Graph Representations of Boolean Functions with Applications to Integer Multiplication*, IEEE Trans. Comput. **40** (1991), pp. 205–213.

[G94] J. Gergov, *Time-Space Tradeoffs for Integer Multiplication on Various Types of Input Oblivious Sequential Machines*, Information Processing Letters **51** (1991), pp. 265–269.

[GKMS96] D. Grigoriev, M. Karpinski, F. Meyer auf der Heide and R. Smolensky, *A Lower Bound for Randomized Algebraic Decision Trees*, Proc. 28th ACM STOC (1996), pp. 612–619; also in Computational Complexity **6** (1997), pp. 357–375.

[GKS97] D. Grigoriev, M. Karpinski and R. Smolensky, *Randomization and the Computational Power of Analytic and Algebraic Decision Trees*, Computational Complexity **6** (1997), pp. 376–388.

[GKY95] D. Grigoriev, M. Karpinski and A.C. Yao, *An Exponential Lower Bound on the Size of Algebraic Decision Trees for MAX*, ECCC TR95-057 (1995), available at `http://www.eccc.uni-trier.de/eccc/` ; to appear in Computational Complexity.

[P95] S. Ponzio, *A Lower Bound for Integer Multiplication with Read-Once Branching Programs*, Proc. 27th ACM STOC (1995), pp. 130–139.

[S97] M. Sauerhoff, *A Lower Bound for Randomized Read-k-Times Branching Programs*, ECCC TR97-019 (1997), available at `http://www.eccc.uni-trier.de/eccc/` .

[T98] J.S. Thathacher, *On Seperating the Read-k-Times Branching Program Hierarchy*, ECCC TR98-002 (1998), available at `http://www.eccc.uni-trier.de/eccc/` ; to appear in Proc. 30th ACM STOC (1998).

[W94] I. Wegener, *Efficient Data Structures for Boolean Functions*, Discrete Mathematics **136** (1994), pp. 347–372.