OKFDDs versus OBDDs and OFDDs *

Bernd Becker¹, Rolf Drechsler¹, Michael Theobald²

¹ Dept. of Computer Science
Johann Wolfgang Goethe-University
60054 Frankfurt am Main, Germany
² Dept. of Computer Science
Columbia University
New York, NY 10027, USA

Abstract. Ordered Decision Diagrams (ODDs) as a means for the representation of Boolean functions are used in many applications in CAD. Depending on the decomposition type, various classes of ODDs have been defined, the most important being the Ordered Binary Decision Diagrams (OBDDs), the Ordered Functional Decision Diagrams (OFDDs) and the Ordered Kronecker Functional Decision Diagrams (OKFDDs). In this paper we clarify the computational power of OKFDDs versus OBDDs and OFDDs from a (more) theoretical point of view. We prove several exponential gaps between specific types of ODDs. Combining these results it follows that a restriction of the OKFDD concept to subclasses, such as OBDDs and OFDDs as well, results in families of functions which lose their efficient representation.

1 Introduction

Ordered Decision Diagrams (ODDs) as a data structure for the representation and manipulation of Boolean functions are applied in many fields of electronic design automation (see [7] for an overview). The most popular type of ODD is the Ordered Binary Decision Diagram (OBDD) [5]. The more recent techniques, like dynamic variable ordering by sifting [17], have made it possible to handle (some) large functions without any basic variation of the OBDD concept itself. However, sifting tends to be very time consuming and the representation of large functions remains problematic; for certain classes of functions, notably multipliers, it has been known that OBDDs will be of exponential size irrespective of the order of the variables [6]. Thus research has been geared towards variations of OBDDs. Among these variations there are those that utilize less restricted Decision Diagrams and there are other ones which augment BDDs with additional constructs. The constructs such as General BDDs [8], or pBDDs [11], IBDDs [12], XBDDs [13], and free BDDs [18] (also known as “1-time branching programs”) remove the ordering constraint on BDDs at the expense of loosing the canonicity of the structure.

* The first and second author were supported by DFG grant Be 1176/4-2.
On the other hand there have been recent attempts at varying the nodes in OBDDs. While OBDDs are based on a recursive application of the Shannon decomposition, other decomposition types can be used to define new types of ODDs. These include Ordered Functional Decision Diagrams (OFDDs) [10], positive OFDDs (pOFDDs) [15], negative OFDDs (nOFDDs) and Ordered Kronecker Functional Decision Diagrams (OKFDDs) [9]. OFDDs, pOFDDs, nOFDDs and OKFDDs, all define canonical representations of Boolean functions. The nodes in OFDDs (pOFDDs, nOFDDs) are decomposed by (positive, negative) Davio decompositions, in OKFDDs positive, negative Davio as well as Shannon decompositions are allowed. OKFDDs thus are potentially more compact than OBDDs and OFDDs, both special kinds of OKFDDs. Recently, it has been shown [2], that OKFDDs are the most general type of ODD, that can be obtained by a variation of the decomposition type. In other words, all other decomposition types (like e.g. the equivalence decomposition [14]) are special cases of positive, negative Davio and Shannon decomposition. In this sense it is interesting and important to determine the computational power of OKFDDs versus OBDDs and OFDDs.

First promising results in this direction have been obtained in [9], where it is demonstrated that OKFDDs allow much more succinct representations of real Benchmark functions than OBDDs and pOFDDs. The relation between OBDDs and pOFDDs is studied in [3] from a more theoretical point of view. Exponential trade-offs between OBDDs and pOFDDs for a class of Boolean functions are proven. It follows that depending on the function class OBDDs and pOFDDs as well might be advantageous.

We consider trade-offs in the more general context of OKFDDs in this paper. The relation between OBDDs and OKFDDs (with a list \(d\) of decomposition types) is described by a bijective Boolean transformation \(\tau_d\), called the generalized \(\tau\)-operator. We use this property to derive several results on the computational power of OKFDDs and subclasses thereof. Classes of (generalized clique) functions are given that have exponentially more concise OFDDs than OBDDs, and vice versa, independent of the ordering of the variables. In particular, the exponential gaps between OBDDs and pOBDDs as proved in [3] follow as a special case. Combining our results it can be concluded that a restriction of the OKFDD concept to subclasses, such as OBDDs, pOFDDs, nOFDDs and OFDDs as well, results in families of functions which lose their efficient representation. Thus, OKFDDs in full generality offer significant advantages in terms of representation size.

The paper is structured as follows: In Sect. 2 (ordered) Kronecker Functional Decision Diagrams and subclasses are introduced. Section 3 studies the relation between the different data structures defined in Sec. 2. Based on the generalized \(\tau\)-operator ODD-sizes of specific function classes are compared in Sect. 4. We finish with a resume of the results in Sect. 5.