Available online at www.sciencedirect.com
ScienceDirect
Electronic Notes in Theoretical Computer Science 253 (2010) 57–70
Electronic Notes in Theoretical Computer Science
www.elsevier.com/locate/entcs

Effect of BDD Optimization on Synthesis of Reversible and Quantum Logic

Robert Wille[1] Rolf Drechsler[2]

Institute of Computer Science
University of Bremen
Bremen, Germany

Abstract

Synthesis of reversible and quantum logic has become an intensely studied topic in the last years. However, most synthesis methods are limited, since they rely on a truth table representation of the function to be synthesized. BDD-based synthesis offers an alternative. Here, reversible or quantum circuits are derived from a function given as *Binary Decision Diagram* (BDD) by substituting all nodes of the BDD with a cascade of Toffoli or elementary quantum gates, respectively. As a result, the application of the approach is not limited by the truth table of the function but by the (quite more efficient) BDD representation. Furthermore, many optimization techniques for BDDs exist which can be exploited.
In this work, we evaluate the effect of three optimization methods for BDDs (namely shared nodes, complement edges, and advanced orderings) on the resulting reversible and quantum circuits. We describe in detail the adjustments, which have to be done to support these optimizations for synthesis, and discuss possible improvements and drawbacks. In a case study, the effects are experimentally evaluated. The results showed, that applying these optimization techniques leads to significant smaller circuits (with respect to number of gates and lines) in most of the cases.

Keywords: Synthesis, Reversible Circuits, Quantum Circuits, Binary Decision Diagrams

1 Introduction

Reversible and quantum logic [10,1,20] has applications in domains like low-power design [10], quantum computing [15], optical computing [4], DNA computing [1], and nanotechnologies [13]. Since synthesis of reversible and quantum circuits significantly differs from traditional design (e.g. fan-out and feedback are not allowed), it has become an intensely studied research area in the recent years.

However, many synthesis approaches are limited. Exact (see e.g. [7,23]) as well as heuristic (see e.g. [18,11,6,12]) methods have been proposed. But both are applicable only for relatively small functions. Exact approaches reach their limits with

[1] Email: rwille@informatik.uni-bremen.de
[2] Email: drechsler@uni-bremen.de

doi:10.1016/j.entcs.2010.02.006

functions containing more than 6 variables [23] while heuristic methods are able to synthesize functions with at most 30 variables [6]. Moreover, often a significant amount of run-time is needed to achieve these results.

These limitations are mainly caused by the underlying techniques. The existing synthesis approaches often rely on truth tables (or similar descriptions like permutations) of the function to be synthesized (e.g. in [18,14]). But even if more compact data structures like BDDs [9], positive-polarity Reed-Muller expansion [6], or Reed-Muller spectra [12] are used, the same limitations can be observed since all of them apply similar strategies (namely selecting reversible gates so that the chosen function representation becomes the identity).

As an alternative, in [21] a new synthesis approach has been introduced that can cope with significantly larger functions. Here, reversible or quantum circuits are derived from a function given as BDD [3] by substituting all nodes of the BDD with a cascade of Toffoli or elementary quantum gates, respectively. As a result, the synthesis approach is not limited by the truth table of the function but by the (quite more efficient) BDD representation. However, since for BDDs many optimization techniques have been developed (e.g. *shared nodes* [3], *complement edges* [2], and reordering strategies like *sifting* [17]) it seems obvious to exploit these optimizations for the synthesis of reversible and quantum logic as well. But this requires new methods to derive circuits from the BDD.

In this work, we describe an improved BDD-based synthesis approach that supports shared nodes, complement edges, and different orderings for BDD-based synthesis of reversible and quantum logic and discuss possible improvements and drawbacks. In a case study, we evaluate the effect of these optimization methods on the resulting circuit sizes. It turned out, that applying these optimization techniques leads to significant smaller circuits in most of the cases.

The remainder of the paper is structured as follows: Section 2 provides the basics of reversible and quantum logic as well as of BDDs. Afterwards, in Section 3 the synthesis approach as proposed in [21] is briefly reviewed. Section 4 describes the new BDD-based synthesis approach that supports shared nodes, complement edges, and reordering for BDD-based synthesis of reversible and quantum logic. Finally, in Section 5 the effect of these optimization techniques on the resulting circuits is experimentally evaluated while the paper is concluded in Section 6.

2 Preliminaries

To keep the paper self-contained this section briefly reviews the basic concepts of reversible and quantum logic. We also describe the basics of BDDs which are used as the underlying data structure by the synthesis approach.

2.1 Reversible Logic

A logic function is reversible if it maps each input assignment to a unique output assignment. Such a function must have the same number of input and output variables $X := \{x_1, \ldots, x_n\}$. Since fanout and feedback are not allowed in reversible