

Quantum Algorithm for Exact Minimal Exclusive-OR Sum-of-Product Minimization and Reversible Synthesis

Background

A single-output **Boolean function** maps combinations of Boolean variables to a Boolean value. When every possible input combination has a defined output, the function is said to be **completely specified**; if not, the function is **incompletely specified**.

A minterm is a product term that uniquely represents an input combination by including each Boolean variable exactly once—either in its true or complemented form. A minterm that evaluates to 1 is called a **true minterm**, while one that evaluates to 0 is called a **false minterm**. If no value is assigned to a minterm, it is referred to as an unspecified or a **don't care** minterm.

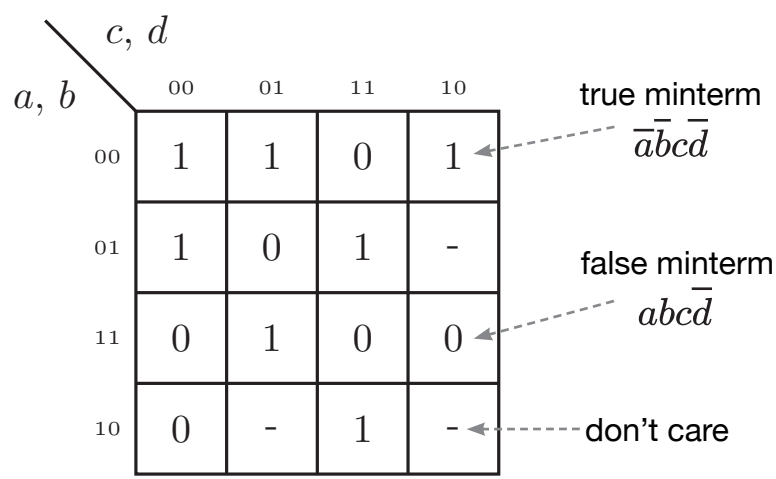


Figure 1: Karnaugh map representation of a Boolean function.

Any Boolean function can be expressed as an **Exclusive-OR Sum-of-Products (ESOP)** expression, which is an exclusive disjunction of product terms.

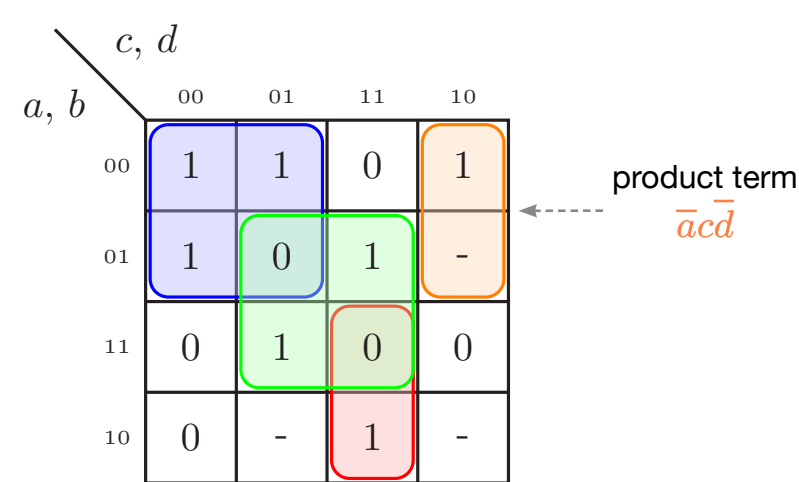


Figure 2: Karnaugh map representation of an ESOP covering of the Boolean function from Figure 1.

An ESOP expression is a two-level representation of a Boolean function, the first level **AND** and the second level **Exclusive-OR (XOR)**.

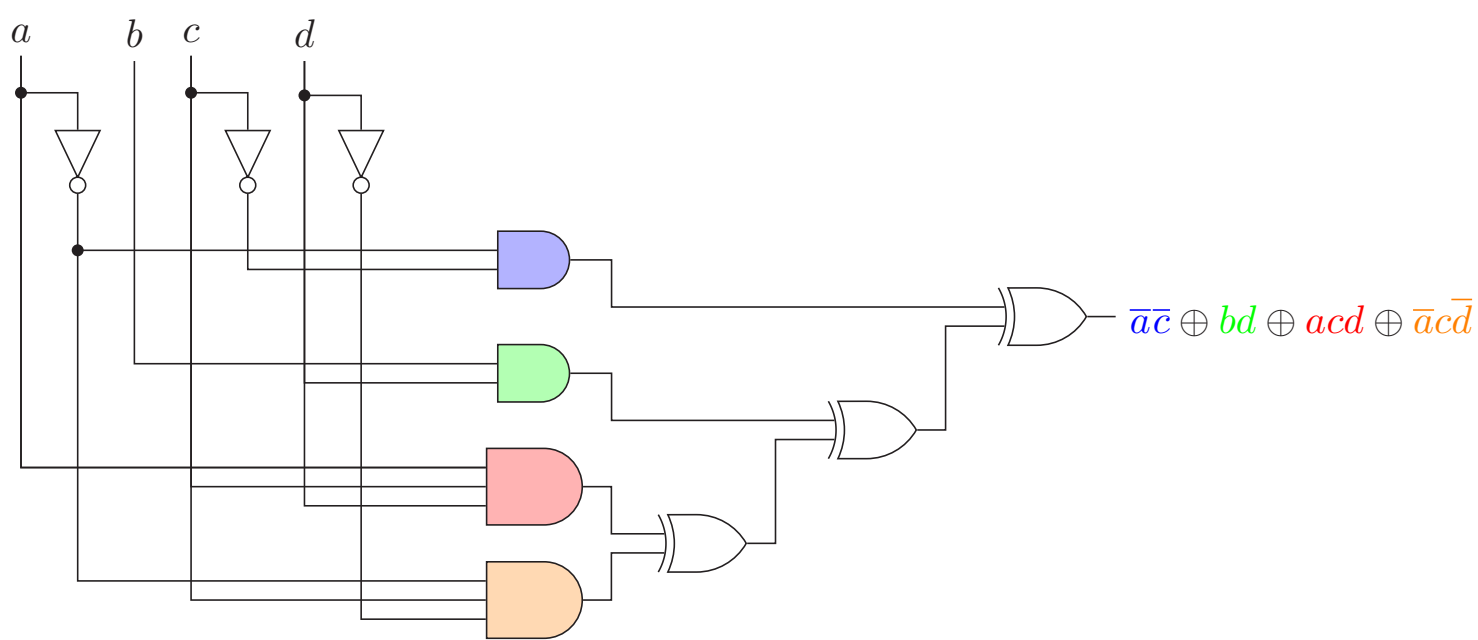


Figure 3: Classical digital circuit realization of the ESOP expression from Figure 2.

An ESOP can also be realized as a quantum circuit or classical reversible circuit.

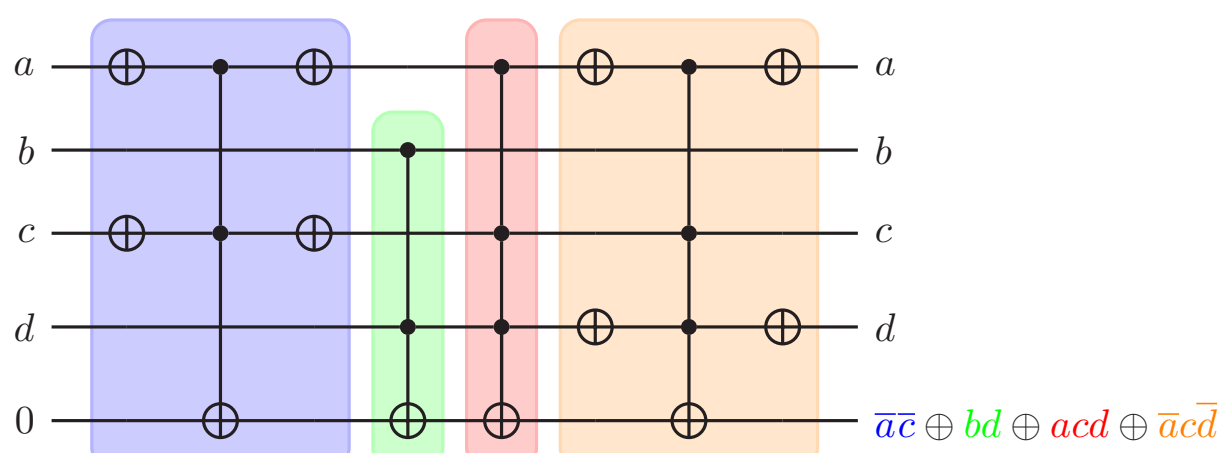


Figure 4: (Quantum) reversible circuit realization of the ESOP expression from Figure 2.

ESOP Synthesis

The ESOP Minimization Problem: Find an ESOP expression that expresses a given Boolean function with the minimal number of product terms

The classical formulation of the ESOP minimization problem aims to minimize the number of the XOR gates required in an AND-XOR circuit. It is a well-known and actively researched problem in two-level logic optimization.

Decades of research have produced effective heuristic minimizers for completely specified functions, but few exact methods have been presented for non-trivial, incompletely specified functions due to the difficulty of exact minimal minimization.

Grover's Algorithm

Grover's algorithm, or the **quantum search algorithm**, is a method that performs an unsorted exhaustive search of a state-space given constraints specified by an oracle. An **oracle** is a function that outputs whether the constraints are satisfied for a given input. Grover's algorithm offers a quadratic speedup over equivalent classical unstructured search.

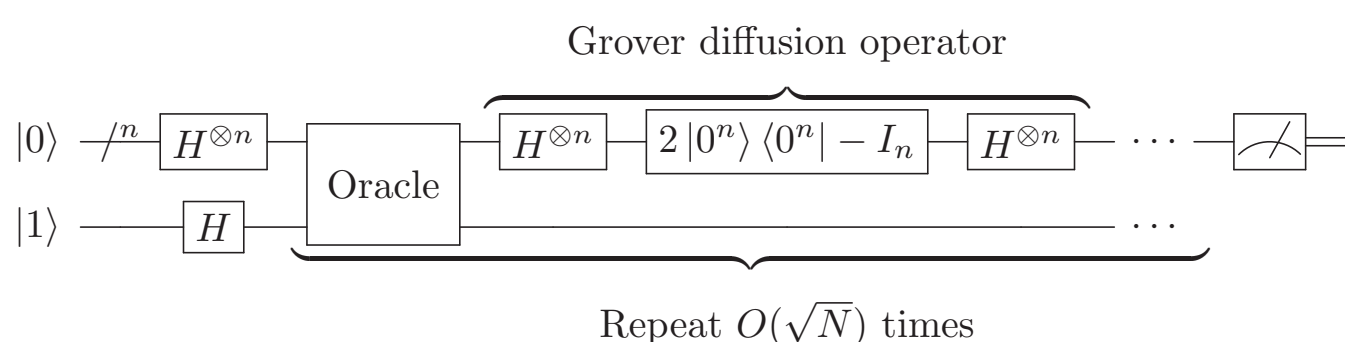


Figure 5: Grover's algorithm.

Note. Adapted from Grover's algorithm, by Bender2k14, Wikimedia Commons (https://commons.wikimedia.org/wiki/File:Grover_algorithm.svg). CC BY-SA 3.0. Modified by finalist.

Constraint Formulation

Product Encoding

An ESOP of n Boolean variables x_1, \dots, x_n can be expressed as

$$\bigoplus_{j=1}^k \left(\bigwedge_{i=1}^n x_i^{o_{i,j}} x_i^{p_{i,j}} \right)$$

where k is the number of product terms, $o_{i,j}, p_{i,j} \in \mathbb{B}$, and each expression

$$x_i^{o_{i,j}} = \begin{cases} 1, & \text{if } o_{i,j} = 0 \\ \bar{x}_i, & \text{if } o_{i,j} = 1 \end{cases}$$

$$x_i^{p_{i,j}} = \begin{cases} 1, & \text{if } p_{i,j} = 0 \\ x_i, & \text{if } p_{i,j} = 1 \end{cases}$$

for $1 \leq i \leq n$ and $1 \leq j \leq k$. The value of $o_{i,j}$ denotes a bit of the negative polarity string of a product within an ESOP expression and the value of $p_{i,j}$ denotes a bit of the positive polarity string.

Constraints

For every true minterm $t_1 t_2 \dots t_{n-1} t_n$,

$$\bigoplus_{j=1}^k \left(\left(\bigwedge_{i=1}^n \overline{t_i o_{i,j}} \right) \left(\bigwedge_{i=1}^n \overline{t_i p_{i,j}} \right) \right) = 1.$$

Similarly, for every false minterm $f_1 f_2 \dots f_{n-1} f_n$,

$$\bigoplus_{j=1}^k \left(\left(\bigwedge_{i=1}^n \overline{f_i o_{i,j}} \right) \left(\bigwedge_{i=1}^n \overline{f_i p_{i,j}} \right) \right) = 0.$$

Oracle Design

The mathematical constraints that each minterm must satisfy can be converted into reversible circuits, which determine whether a given constraint is satisfied. Each such circuit consists of a series of generalized Toffoli gates.

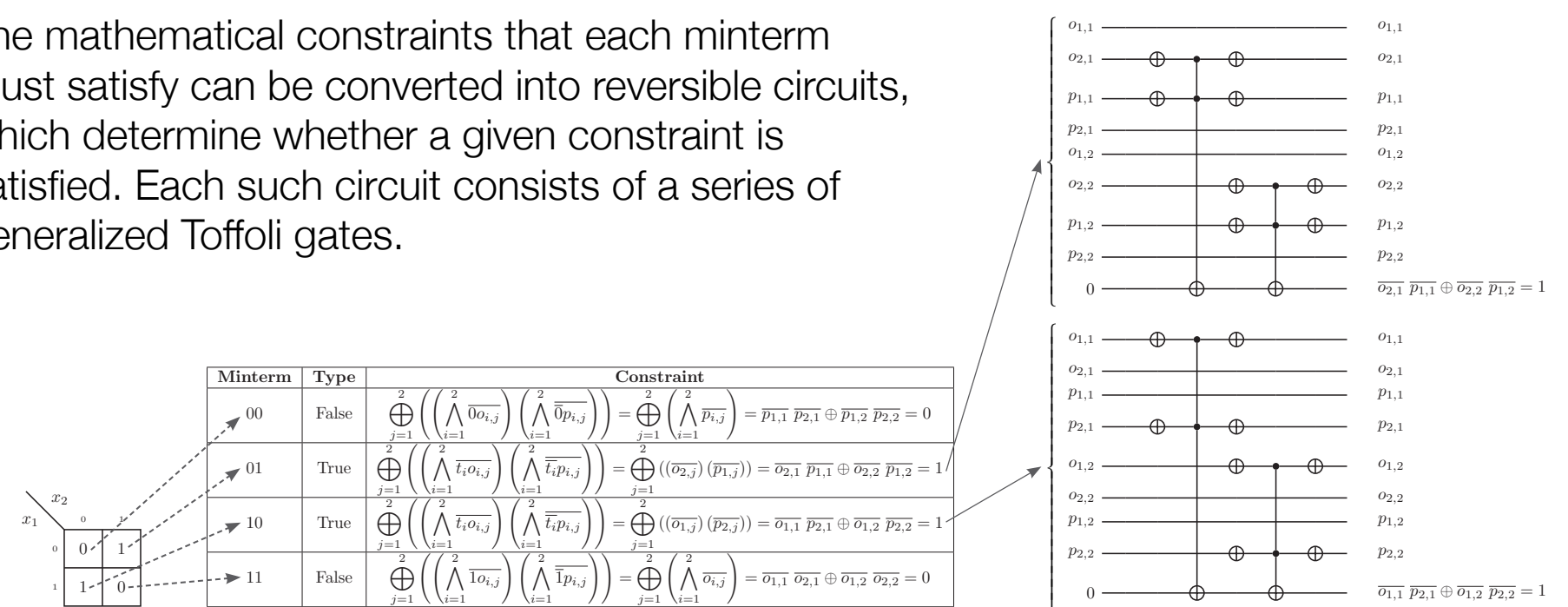


Figure 6: Translation of constraints into components of a Grover oracle.

This process is repeated for all minterms, using counters to increment every instance in which a minterm satisfies a constraint. Mirrors reset all values except the counter register. When the counting registers are properly initialized, all register bits will become 1 if and only if every counter is activated. A generalized Toffoli gate then verifies this condition, ensuring the oracle correctly determines whether a given ESOP of size k is correct.

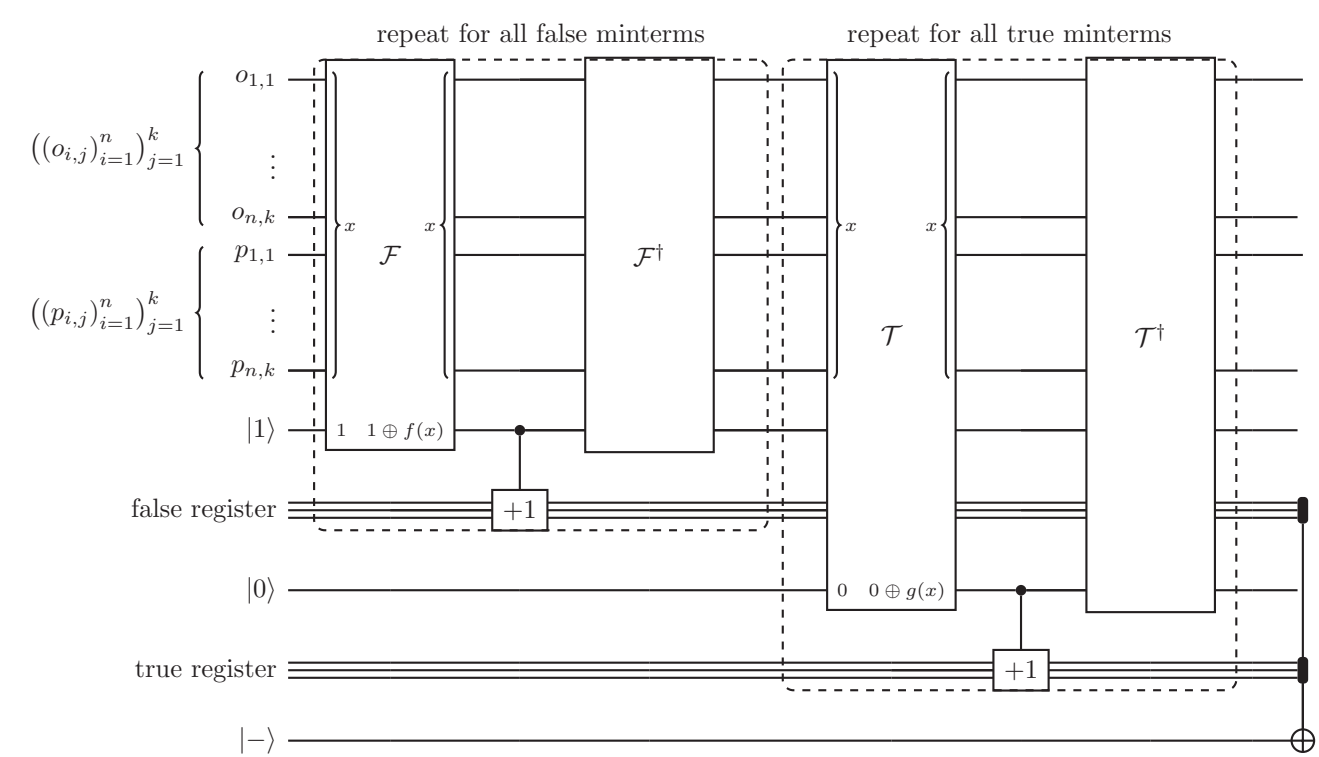


Figure 7: General unmirrored Grover oracle design for exact ESOP minimization.

The complete exact ESOP minimization procedure leverages Grover search to find ESOP expressions of size k , iteratively adjusting k until the minimal representation is identified.

Time Complexity Analysis

- The best comparable classical algorithm for finding an ESOP representation with k products has a worst-case time complexity of $O(4^{nk} 2^{km})$, where m denotes the number of minterms.
- The algorithm presented in this project achieves the same task with an oracle query complexity of $O(2^{nk})$ and an overall worst-case time complexity of $O(2^{nk} km)$, given that the complexity of the oracle itself is $O(km)$.
- Other quantum algorithms have been proposed, but do not scale well in terms of number of qubits, query complexity, or gate cost.

Key Findings & Conclusion

- Improved ESOP minimization enables more efficient (quantum) reversible circuit design.
- Developed the fastest known algorithm for the exact minimal ESOP synthesis of incompletely specified Boolean functions
- Verified oracle correctness through extensive quantum simulations in Qiskit
- Extended the method to synthesize multi-output Toffoli cascades and incorporate constraints that minimize quantum gate cost

All figures created by the finalist, unless otherwise noted, using LaTeX with the quantikz and tikz-karnaugh packages.