Performance Gains in Quantum SAT Solvers Using ESOP Encoding

Abhoy Kole, *Member, IEEE,* and Rolf Drechsler, *Fellow, IEEE,*

***Abstract*—The Boolean satisfiability problem (SAT) has been extensively studied in the context of quantum computing over the past few decades. A key challenge lies in the efficient representation of SAT instances—typically expressed in conjunctive normal form (CNF)—as quantum circuits. Among various encoding methods, the exclusive-sum-of-products (ESOP) approach has shown significant promise in minimizing clause complexity, thereby reducing the overall quantum resource overhead. The objective of this work is to analyze the effectiveness of the ESOP-based CNF (e-CNF) encoding in reducing the upper bounds on qubit requirements and Clifford+T gate counts, and to develop an approach for quantum circuit interpretation of e-CNF to validate these bounds.**

***Index Terms*—Boolean Satisfiability Problem (SAT), ESOP (Exclusive-Sum-of-Products) Encoding, Quantum Circuit Optimization**

I. Introduction

The Boolean satisfiability problem (SAT) is a fundamental problem in computer science and logic, with a wide range of applications. Various solution techniques have been developed over the years, including the Davis-Putnam-Logemann-Loveland (DPLL) algorithm [1], conflict-driven clause learning (CDCL) [2], and local search methods such as WalkSAT [3]. These algorithms form the core of many state-of-the-art SAT solvers, such as the Z3 Satisfiability Modulo Theories (SMT) solver [4]. SAT instances are typically represented as Boolean formulas, which must be transformed into conjunctive normal form (CNF) before processing.

Recent research has explored quantum approaches to solving SAT problems. These include formulating SAT as a quadratic unconstrained binary optimization (QUBO) problem for execution on quantum annealers [5], [6], and applying Grover's algorithm within gate-based quantum computing frameworks [7], [8]. A key challenge in using Grover's algorithm for SAT lies in efficiently encoding the problem as a quantum oracle. This requires translating the CNF representation into a quantum circuit, often introducing logical equivalence ($\Leftrightarrow$) relations in the process.

To address the complexity of this encoding, prior work [9] proposed the use of exclusive-sum-of-products (ESOP)-based CNF representations—referred to as e-CNF—for handling equivalence relations. This approach demonstrated up to a 60% reduction in clause count. Although initially applied to hardware equivalence checking, where all CNF clauses originate from $\Leftrightarrow$ relations, the method holds promise for broader applications in quantum SAT solving.

Direct ESOP-based transformation of $\Leftrightarrow$ relations offers a more resource-efficient quantum circuit representation. In contrast, applying Tseitin transformation [10] to $\Leftrightarrow$ before circuit synthesis can lead to significantly higher quantum cost. Motivated by this, the present work investigates the complexity of generating quantum circuits from both CNF and e-CNF expressions, evaluating them in terms of Clifford+T gate requirements. Additionally, we propose an automated framework for validating these complexity bounds and benchmarking e-CNF-based quantum circuit interpretations against their traditional CNF-based counterparts.

A. Kole is with German Research Center for Artificial Intelligence (DFKI), Bremen, Germany, E-mail: abhoy.kole@dfki.de.

R. Drechsler is with University of Bremen and German Research Center for Artificial Intelligence (DFKI), Bremen, Germany, E-mail: drechsler@uni-bremen.de.

II. SAT Encoding

A. Conventional CNF Representation

In SAT encoding, the basic logical operators—NOT ($\neg$), OR ($\vee$) and AND ($\wedge$)—are used to express more complex operations such as implication ($\Rightarrow$), equivalence ($\Leftrightarrow$), and exclusive-OR ($\oplus$) in conjunctive normal form (CNF). This form is particularly suitable because, in practice, many logical formulas emerge from the conjunction of multiple constraints that must hold simultaneously.

The transformation into CNF involves applying De Morgan's laws to push negations $\neg$ inward toward atomic propositions, such as:

$$\neg(a \vee b) = \neg a \wedge \neg b \tag{1}$$

and distributing $\vee$ over $\wedge$:

$$(a \wedge b) \vee c = (a \vee c) \wedge (b \vee c). \tag{2}$$

However, such transformations can cause an exponential blow-up in the size of the formula. For instance:

$$\begin{aligned}(a_1 \wedge a_2 \wedge \cdots \wedge a_n) \vee (b_1 \wedge b_2 \wedge \cdots \wedge n_n) \\ = (a_1 \vee b_1) \wedge (a_2 \vee b_2) \wedge \cdots \wedge (a_n \vee b_n).\end{aligned} \tag{3}$$

To mitigate this, new auxiliary propositions are introduced to represent sub-formulas, coupled with constraints to preserve logical equivalence at the sub-formula level. For example, consider the formula:

$$\phi = (a_1 \wedge \neg a_2) \vee \neg(a_3 \wedge a_4) \vee \cdots \vee (a_{n-1} \wedge a_n) \tag{4}$$

We can introduce new propositions $p_1, p_2, \ldots, p_m$ for sub-formulas:

$$p_1 \Leftrightarrow a_1 \wedge \neg a_2,\ p_2 \Leftrightarrow a_3 \wedge a_4,\ \cdots,\ p_m \Leftrightarrow a_{n-1} \wedge a_n. \tag{5}$$

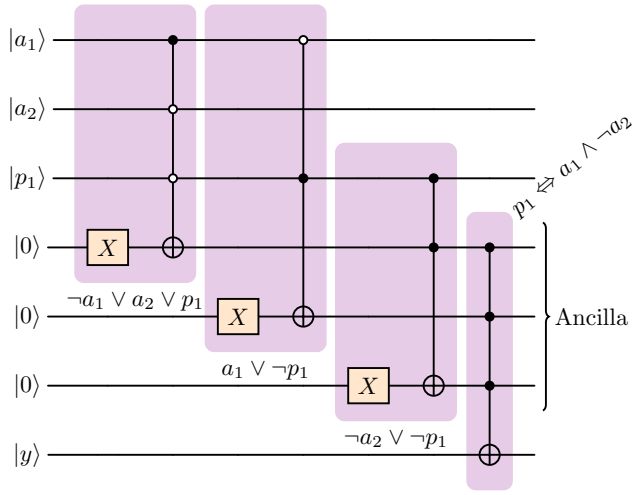

Fig. 1. Quantum circuit representation of proposition $p_1 \Leftrightarrow a_1 \wedge \neg a_2$ based on conventional CNF interpretation using three ancilla qubits, three Pauli X gates, and a pair of C^2X and C^3X gates with some controls of negative polarity.

Using these, the formula can be rewritten as:

$$\begin{aligned}\widehat{\phi} =&(p_1 \vee \neg p_2 \vee \cdots \vee p_m) \wedge (p_1 \Leftrightarrow a_1 \wedge \neg a_2) \\ &\wedge (p_2 \Leftrightarrow a_3 \wedge a_4) \wedge \cdots \wedge (p_m \Leftrightarrow a_{n-1} \wedge a_n).\end{aligned} \tag{6}$$

While ϕ and $\widehat{\phi}$ are not logically equivalent (i.e., $\phi \not\equiv \widehat{\phi}$), they are equisatisfiable, i.e., share the same satisfiability status:

$$\phi \equiv_{\text{SAT}} \widehat{\phi} \tag{7}$$

This technique is formalized through the Tseitin transformation [10], which introduces propositions of the form:

$$p_i \Leftrightarrow \mathcal{F} \tag{8}$$

where $\mathcal{F}$ is a sub-formula. This process leads to the generation of logically equivalent CNF expressions such as:

$$\begin{aligned}&p_1 \Leftrightarrow a_1 \wedge \neg a_2 \\ &\quad \equiv (\neg a_1 \vee a_2 \vee p_1) \wedge (a_1 \vee \neg p_1) \wedge (\neg a_2 \vee \neg p_1)\end{aligned} \tag{9}$$

The transformation ensures linear growth in formula size while preserving equisatisfiability, making it a practical tool for SAT solving. However, the quantum circuit realization of this transformation remains resource-intensive. Specifically, it requires three additional ancilla qubits per equivalence clause, along with a significant increase in gate complexity, as illustrated in Fig. 1.

According to [11], implementing a three-controlled X (C^3X) gate requires 33 Clifford+T gates, including 6 Hadamard (H), 15 T, and 12 CNOT (CX) gates. Similarly, a two-controlled X (C^2X) gate requires 15 Clifford+T gates (2 H, 7 T, and 6 CX). Therefore, realizing each proposition of the form in Eq. (6) entails: three ancilla qubits, and 99 (i.e., $2 \times (33 + 15) + 3$) Clifford+T gates per clause. Consequently, for m such propositions, the total resource cost becomes 3m ancilla qubits and 99m Clifford+T gates, posing a significant overhead for quantum implementation.

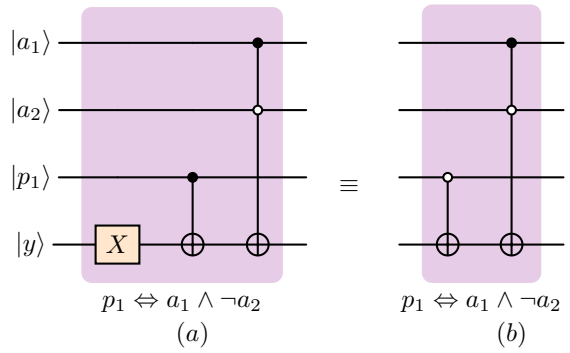

Fig. 2. Quantum circuit representations of proposition $p_1 \Leftrightarrow a_1 \wedge \neg a_2$ based on e-CNF interpretation using either (a) a Pauli X gate, a CX gate, and a C^2X gate or (b) a CX gate, and a C^2X gate with come controls of negative polarity.

B. e-CNF Representation

The exclusive-sum-of-products (ESOP) based CNF (e-CNF) representation extends traditional SAT encoding by incorporating the exclusive-OR ($\oplus$) operator. This added support for exclusive disjunction enables more compact and efficient encodings of certain logical propositions.

Specifically, propositions of the form Eq. (8) can be equivalently rewritten using XOR as:

$$p_i \Leftrightarrow \mathcal{F} \equiv 1 \oplus p_i \oplus \mathcal{F}. \tag{10}$$

For instance, the proposition from Eq. (9) can be expressed as:

$$\begin{aligned}p_1 \Leftrightarrow a_1 \wedge \neg a_2 &\equiv 1 \oplus p_1 \oplus a_1 \wedge \neg a_2 \\ &\equiv \neg p_1 \oplus a_1 \wedge \neg a_2\end{aligned} \tag{11}$$

This transformation corresponds to the quantum circuit shown in Fig. 2, which requires only 17 Clifford+T gates—comprising one C^2X gate, one CX gate, and one Pauli-X gate. In contrast, the quantum circuit derived from the conventional CNF-based transformation in Eq. (9) (see Fig. 1) is significantly more resource-intensive. The e-CNF-based approach eliminates the need for: three ancilla qubits, two C^3X gates, one C^2X gate and two additional Pauli-X gates. As a result, the total gate count reduction achieved using e-CNF over standard CNF is 82 (i.e., $2 \times 33 + 15 + 1$) Clifford+T gates.

Additionally, disjunctions ($\vee$) in Boolean formulas can also be represented using ESOP forms, such as:

$$a_1 \vee a_2 \vee \cdots \vee a_n = 1 \oplus \neg a_1 \wedge \neg a_2 \wedge \cdots \wedge \neg a_n. \tag{12}$$

Using this formulation, the e-CNF representation of Eq. (6) becomes:

$$\begin{aligned}\tilde{\phi} =&(1 \oplus \neg p_1 \wedge p_2 \wedge \cdots \wedge \neg p_m) \wedge (\neg p_1 \oplus a_1 \wedge \neg a_2) \\ &\wedge (\neg p_2 \oplus a_3 \wedge a_4) \wedge \cdots \wedge (\neg p_m \oplus a_{n-1} \wedge a_n)\end{aligned} \tag{13}$$

This representation preserves equisatisfiability, i.e.,

$$\phi \equiv_{\text{SAT}} \widehat{\phi} \equiv_{\text{SAT}} \tilde{\phi} \tag{14}$$

According to [11], the realization of an C^mX gate requires $18m - 21$ Clifford+T gates, including $4m - 6$ H gates, $8m - 9$ T gates, and $6m - 6$ CX gates. Realizing m propositions

of the form in Eq. (11) requires $17m$ Clifford+T gates. The expression $1 \oplus \neg p_1 \wedge p_2 \wedge \cdots \wedge \neg p_m$ additionally requires $18m - 20$ Clifford+T gates.

To compute the final output, the conjunction of all $m + 1$ ESOP expressions (from Eq. (13)) necessitates one more C^mX gate and quantum uncomputation, which also requires $18m - 21$ and $35m - 20$ Clifford+T gates, respectively. Thus, the total gate cost for the e-CNF-based realization is $88m - 61$ (i.e., $2 \times (35m - 20) + 18m - 21$) Clifford+T gates. In contrast, the conventional CNF-based realization from Eq. (6) requires $252m - 61$ (i.e., $2 \times (117m - 20) + 18m - 21$) Clifford+T gates. This results in a net reduction of 164m gates, highlighting the substantial efficiency gain of the e-CNF-based quantum circuit construction over traditional CNF-based approaches.

III. Conclusion

This work analyzes the resource requirements for encoding SAT instances as quantum circuits, with a focus on evaluating the advantages of using e-CNF-based clause generation over the conventional CNF-based approach. In particular, for interpreting equivalence ($\Leftrightarrow$) propositions, the e-CNF method demonstrates a significant reduction in both qubit count and quantum gate complexity. For example, a proposition of the form $p_i \Leftrightarrow a_j \wedge b_k$ requires 3 additional qubits and 99 Clifford+T gates using the CNF-based approach, whereas the equivalent e-CNF-based circuit needs only 17 Clifford+T gates and no additional qubits. Future work will focus on developing automated techniques to evaluate the resource bounds of e-CNF-based quantum circuit interpretations and benchmarking their performance against CNF-based implementations.

References

[1] M. Davis, G. Logemann, and D. Loveland, "A machine program for theorem-proving," *Commun. ACM*, vol. 5, no. 7, p. 394–397, Jul. 1962. [Online]. Available: https://doi.org/10.1145/368273.368557

[2] J. Marques-Silva and K. Sakallah, "GRASP: a search algorithm for propositional satisfiability," *IEEE Transactions on Computers*, vol. 48, no. 5, pp. 506–521, 1999.

[3] H. Kautz and B. Selman, "Pushing the envelope: Planning, propositional logic, and stochastic search," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 1996, pp. 1194–1201.

[4] L. de Moura and N. Bjørner, "Z3: An efficient smt solver," in *Tools and Algorithms for the Construction and Analysis of Systems*, C. R. Ramakrishnan and J. Rehof, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 337–340.

[5] Z. Bian, F. Chudak, W. Macready, A. Roy *et al.*, "Solving sat and maxsat with a quantum annealer: Foundations and a preliminary report," in *Frontiers of Combining Systems*. Cham: Springer International Publishing, 2017, pp. 153–171.

[6] F. Glover, G. Kochenberger, and Y. Du, "A tutorial on formulating and using qubo models," *arXiv preprint arXiv:1811.11538 [cs.DS]*, 2018.

[7] E. Dantsin, V. Kreinovich, and A. Wolpert, "On quantum versions of record-breaking algorithms for sat," *SIGACT News*, vol. 36, no. 4, p. 103–108, Dec. 2005. [Online]. Available: https://doi.org/10.1145/1107523.1107524

[8] L. K. Grover, "A fast quantum mechanical algorithm for database search," in *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, ser. STOC '96. New York, NY, USA: Association for Computing Machinery, 1996, p. 212–219. [Online]. Available: https://doi.org/10.1145/237814.237866

[9] A. Kole, M. E. Djeridane, L. Weingarten, K. Datta, and R. Drechsler, "qSAT: Design of an efficient quantum satisfiability solver for hardware equivalence checking," *J. Emerg. Technol. Comput. Syst.*, Apr. 2025. [Online]. Available: https://doi.org/10.1145/3729229

[10] G. S. Tseitin, "On the complexity of derivation in propositional calculus," in *Automation of Reasoning: 2: Classical Papers on Computational Logic 1967–1970*, J. H. Siekmann and G. Wrightson, Eds. Springer Berlin Heidelberg, 1983, pp. 466–483. [Online]. Available: https://doi.org/10.1007/978-3-642-81955-1_28

[11] D. Maslov, "Advantages of using relative-phase toffoli gates with an application to multiple control toffoli optimization," *Phys. Rev. A*, vol. 93, p. 022311, Feb 2016. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevA.93.022311