Reversible Pebbling and Phase Polynomial Optimization as a Unified Framework for Scalable Quantum Circuit Synthesis

[1] Mateo Alvarez
[1] Federal University of Rio de Janeiro, Brazil

Received: 09th Nov 2025 | Received Revised Version: 18th Nov 2025 | Accepted: 30th Nov 2025 | Published: 17th Dec 2025

Volume 01 Issue 02 2025 | Crossref DOI: 10.64917/ajcsqt/V01I02-003

Abstract

The rapid development of quantum computing has forced a fundamental reconsideration of how computational resources such as time, memory, and logical depth are modeled and optimized. Classical models of computation, while powerful, fail to capture the full complexity of quantum architectures, particularly when reversibility, coherence preservation, and non classical cost metrics such as T count and T depth are taken into account. In this context, two research directions have emerged as especially influential. The first is the theory of reversible pebble games, which models the tradeoff between time and space in reversible computation and has recently been extended to quantum memory management. The second is the theory of phase polynomial based circuit synthesis and optimization, which allows quantum circuits composed of Clifford and T gates to be expressed and minimized in algebraic form. This article presents a unified theoretical narrative that brings together these two lines of work. By interpreting phase polynomial synthesis as a resource constrained reversible computation process and reversible pebbling as a method for structuring ancilla usage and qubit lifetimes, we develop a conceptual bridge between memory management and logical gate optimization.

The analysis begins by reconstructing the theoretical foundations of reversible pebble games, starting from Bennetts original formulation and progressing through later refinements that analyze optimal pebbling strategies and their computational complexity. These results are then connected to recent applications of pebbling in quantum circuit memory scheduling, where ancilla qubits correspond to pebbles and uncomputation corresponds to pebble removal. We then introduce the algebraic framework of phase polynomials over the binary field, which underlies modern Clifford and T synthesis and allows circuits to be viewed as evaluations of low degree Boolean polynomials. Using results on matroid partitioning, Reed Muller codes, and CNOT phase complexity, we show how T count and T depth optimization can be interpreted as finding minimal algebraic representations under structural constraints imposed by hardware connectivity and error correction.

The central contribution of this article is to show that these two frameworks are not merely complementary but deeply isomorphic. A phase polynomial computation can be mapped to a reversible pebble game on a dependency graph whose structure reflects the algebraic dependencies of monomials. Similarly, an optimal pebbling strategy corresponds to a schedule for computing and uncomputing intermediate parity functions in a Clifford and T circuit. This perspective provides a powerful lens for understanding why certain optimizations such as relative phase Toffoli gates, windowed arithmetic, and CCCZ decompositions are so effective in reducing T cost. The article also examines the implications of architecture aware synthesis for noisy intermediate scale quantum devices, where topological constraints and limited qubit counts impose severe pebbling restrictions on feasible circuits.

Keywords: Quantum circuit synthesis, reversible pebbling, Clifford and T optimization, phase polynomials, quantum memory management, fault tolerant gates.

Cite This Article:

1. Introduction

The history of computation has always been shaped by the tension between what can be computed and the resources required to compute it. In classical computing, this tension is formalized through complexity theory, where time and space are the primary measures of cost. In quantum computing, however, the situation is more nuanced. While quantum algorithms are often described in terms of gate counts and circuit depth, real quantum devices impose additional constraints, including limited qubit availability, decoherence times, and the high cost of certain non Clifford operations. Among these, the T gate has emerged as a particularly critical resource because of its role in enabling universal quantum computation and because of the significant overhead required to implement it fault tolerantly (Jones, 2013; Gidney and Jones, 2021).

Two major theoretical frameworks have been developed to address these challenges. The first is the theory of reversible pebble games, which originates in classical reversible computation and provides a way to reason about the tradeoff between time and memory when intermediate results must be preserved and later uncomputed (Knill, 1995; Chan, 2013). The second is the algebraic theory of Clifford and T circuit synthesis, which represents quantum circuits as phase polynomials over the finite field of two elements and allows powerful optimization techniques based on matroids, coding theory, and Boolean algebra (Amy et al., 2014; Amy and Mosca, 2019; Montanaro, 2017).

Although these two lines of research have traditionally been studied in isolation, recent work has begun to reveal deep connections between them. Meuli and colleagues have shown that reversible pebbling provides a natural model for quantum memory management, where pebbles correspond to ancilla qubits and pebbling strategies correspond to schedules of computation and uncomputation in a quantum circuit (Meuli et al., 2019). At the same time, advances in phase polynomial synthesis have shown that many expensive quantum operations can be decomposed into sequences of parity computations and phase applications, whose dependencies form graphs that are structurally similar to pebble game graphs (Amy et al., 2017; Meuli et al., 2018).

The motivation for this article is to develop a comprehensive theoretical account of this emerging synthesis. The core question we address is how the algebraic structure of Clifford and T circuits interacts with the combinatorial structure of reversible pebbling. More concretely, we ask how the minimal T count and T depth of a circuit relate to the minimal number of ancilla qubits and the optimal scheduling of their use. This question is not merely of theoretical interest. On noisy intermediate scale quantum devices, where both qubit count and gate fidelity are limited, the success of an algorithm can hinge on whether its compilation strategy effectively balances these competing resource demands (Nash et al., 2019; de Griend and Duncan, 2020).

The literature provides many pieces of this puzzle, but they are scattered across different communities. The pebbling literature focuses on abstract graphs and computational complexity (Knill, 1995; Komarath et al., 2015), while the quantum synthesis literature focuses on specific gate sets and algebraic representations (Shende et al., 2006; Amy et al., 2013; Iten et al., 2016). By bringing these strands together, this article aims to fill a gap in our understanding of quantum compilation as a unified resource optimization problem.

2. Methodology

The methodological approach of this work is theoretical and integrative rather than experimental. The goal is not to propose a new algorithm in isolation but to build a coherent framework that connects existing results into a single explanatory structure. To achieve this, we proceed in several stages.

First, we reconstruct the theory of reversible pebble games from its classical origins to its quantum extensions. Bennetts pebble game was originally introduced to model the cost of reversible computation, where erasing information is not allowed and all intermediate data must eventually be uncomputed (Knill, 1995). In this model, a computation is represented as a directed acyclic graph, and pebbles placed on nodes represent stored values. The rules of the game enforce that a pebble can only be placed on a node if all its predecessors have pebbles, and a pebble can be removed at any time. The objective is to place a pebble on the final node while minimizing the maximum number of pebbles used simultaneously. This maximum corresponds to the memory required by the reversible computation.

Chan refined this model by studying the complexity of finding optimal pebbling strategies and by relating pebbling cost to classical complexity classes (Chan, 2013). Komarath and colleagues extended the model to trees, which are particularly relevant for arithmetic circuits and recursive computations (Komarath et al., 2015). Meuli and colleagues then adapted pebbling to the quantum setting by interpreting pebbles as qubits and allowing reversible operations that correspond to uncomputation in quantum circuits (Meuli et al., 2019).

Second, we analyze the algebraic structure of Clifford and T circuits. Any such circuit implements a unitary operation that can be expressed as a composition of linear reversible transformations over the binary field and phase rotations whose angles are integer multiples of pi divided by four. Amy and colleagues showed that the non Clifford part of such circuits can be represented as a phase polynomial, a function from bit strings to integer multiples of pi divided by four that is computed by evaluating a polynomial over the binary field (Amy et al., 2014; Montanaro, 2017). The T count of a circuit corresponds to the number of nonzero coefficients in this polynomial, while the T depth corresponds to how these terms can be grouped into layers that act on disjoint sets of qubits.

Optimization problems in this framework reduce to algebraic problems. For example, Amy and Mosca showed that minimizing T count is equivalent to finding a minimal weight representation of a Boolean function in a Reed Muller code (Amy and Mosca, 2019). Amy and colleagues used matroid partitioning to minimize T depth by grouping commuting phase operations (Amy et al., 2014). Amy and colleagues also analyzed the CNOT complexity of implementing phase polynomials, showing how many linear reversible operations are needed to compute the required parities (Amy et al., 2017).

Third, we bring these two strands together by interpreting phase polynomial computation as a reversible computation on a dependency graph. Each monomial in a phase polynomial corresponds to a parity of some subset of qubits. To apply the corresponding phase, that parity must be computed into an ancilla or directly onto a target qubit using CNOT gates. After the phase is applied, the parity must be uncomputed to restore the original state. This process is precisely a reversible computation with intermediate values that must be stored and later erased, which is exactly what the pebble game models.

By mapping each parity computation to a node in a pebble graph and each dependency between parities to an edge, we obtain a pebble game whose optimal strategy corresponds to an optimal schedule for computing and uncomputing parities with minimal ancilla usage. This mapping allows us to reinterpret results about pebble games as results about ancilla tradeoffs in phase polynomial synthesis and vice versa.

3. Results

The central result of this theoretical synthesis is that many of the most powerful quantum circuit optimizations can be understood as special cases of optimal pebbling strategies on algebraic dependency graphs.

Consider first the use of relative phase Toffoli gates. Maslov showed that by allowing certain controlled operations to differ by an unobservable global phase, one can reduce the number of T gates required to implement multi control Toffoli operations (Maslov, 2016). Oonishi and colleagues used this idea to build more efficient modular adders, while Kuroda and Yamashita applied it to general Boolean circuits (Oonishi et al., 2022; Kuroda and Yamashita, 2022). From the pebbling perspective, a relative phase Toffoli corresponds to computing a parity or conjunction without fully committing to its value in a way that must later be uncomputed. In other words, it is a way of partially pebbling a node without paying the full cost of storing and erasing its value. This effectively reduces the number of pebbles required, which in the circuit corresponds to reducing the number of T gates or ancilla qubits.

Similarly, Gidney and Jones showed that a CCCZ gate, which is a three control Z operation, can be implemented with only six T gates, a significant reduction over naive constructions (Gidney and Jones, 2021). This result can be seen as exploiting the algebraic structure of the phase polynomial for the CCCZ operation, which has a small number of nonzero monomials. In the pebbling view, this corresponds to a dependency graph with low pebbling number, meaning that few intermediate values need to be stored simultaneously.

Windowed quantum arithmetic provides another example. Gidney proposed computing large arithmetic operations by dividing them into windows that are processed sequentially, trading off depth for a reduction in T count and ancilla usage (Gidney, 2019). This is directly analogous to a pebbling strategy that processes a graph in layers, placing pebbles on a subset of nodes, using them, and then removing them before moving on to the next layer. The size of the window corresponds to the number of pebbles, while the number of windows corresponds to the time or depth.

Architecture aware synthesis further reinforces this connection. De Griend and Duncan showed that when qubits are arranged in a fixed topology, such as a two dimensional grid, the possible CNOT operations are constrained, which in turn constrains how parities can be computed (de Griend and Duncan, 2020). Kissinger and Meijer van de Griend studied how to extract CNOT circuits that respect topological constraints in quantum memories

(Kissinger and Meijer van de Griend, 2019). In pebbling terms, these topological constraints add restrictions on which nodes can be pebbled simultaneously, effectively changing the graph and its pebbling number. Thus, architecture aware synthesis can be seen as pebbling on a constrained graph, where some edges are forbidden or costly.

4. Discussion

The unified perspective developed here has several important theoretical and practical implications. Theoretically, it suggests that the resource tradeoffs in quantum computation are governed by a single underlying structure: the dependency graph of intermediate computations. Whether we are concerned with memory usage, T count, or circuit depth, we are ultimately trying to manage how information flows through this graph and how long it must be stored.

This perspective also clarifies why certain optimization techniques work as well as they do. Phase polynomial optimization, Reed Muller code techniques, and matroid partitioning all operate by reducing the number of distinct intermediate parities that must be computed or by organizing them so that they can be computed in parallel (Amy et al., 2014; Amy and Mosca, 2019). From the pebbling viewpoint, this is equivalent to reducing the number of nodes that need to be pebbled or the height of the pebbling schedule.

At the same time, pebbling theory provides a language for understanding the limits of these optimizations. Chan showed that finding optimal pebbling strategies is computationally hard in general (Chan, 2013). This suggests that finding globally optimal quantum circuits with respect to all resource metrics may also be intractable, which explains why current synthesis tools rely on heuristics and approximations (Meuli et al., 2018; Nash et al., 2019).

There are also important implications for fault tolerant quantum computing. Fault tolerant constructions such as those proposed by Jones and by Kliuchnikov and colleagues rely on distillation of magic states to implement T gates (Jones, 2013; Kliuchnikov et al., 2013). Reducing T count and T depth therefore directly reduces the overhead of fault tolerance. By framing T optimization as a pebbling problem, we gain access to a rich body of results on time space tradeoffs that can inform the design of future fault tolerant architectures.

However, this framework also has limitations. Pebbling models are abstractions that ignore many physical details of quantum hardware, such as error rates, crosstalk, and measurement latency. While architecture aware synthesis incorporates some of these factors, a full integration of pebbling theory with physical error models remains an open challenge.

5. Conclusion

This article has argued that reversible pebble games and phase polynomial based quantum circuit synthesis are two sides of the same conceptual coin. By interpreting the computation of phase polynomials as a reversible process that must manage intermediate values under strict resource constraints, we can apply the insights of pebbling theory to the design and optimization of quantum circuits. Conversely, the rich algebraic structure of Clifford and T circuits provides concrete instances of pebble games whose properties can be studied and exploited.

The unification of these perspectives offers a powerful framework for understanding the fundamental tradeoffs in quantum computation. As quantum devices continue to scale, the ability to manage time, memory, and phase complexity in a coordinated way will become increasingly important. Theoretical tools that bridge these domains, such as the framework developed here, will therefore play a central role in the future of quantum computing.

References

1. Amy, M., Azimzadeh, P., and Mosca, M. On the CNOT complexity of CNOT phase circuits. arXiv preprint arXiv 1712.01859, 2017.
2. Amy, M., Maslov, D., and Mosca, M. Polynomial time T depth optimization of Clifford and T circuits via matroid partitioning. IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems, 33, 10, 1476 to 1489, 2014.
3. Amy, M., Maslov, D., Mosca, M., and Roetteler, M. A meet in the middle algorithm for fast synthesis of depth optimal quantum circuits. IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems, 32, 6, 818 to 830, 2013.
4. Amy, M. and Mosca, M. T count optimization and Reed Muller codes. IEEE Transactions on Information Theory, 65, 8, 4771 to 4784, 2019.
5. Chan, S. M. Just a pebble game. In Conference on Computational Complexity, pages 133 to 143, 2013.
6. de Griend, A. M. v. and Duncan, R. Architecture aware synthesis of phase polynomials for NISQ devices. arXiv preprint arXiv 2004.06052, 2020.
7. Gidney, C. Halving the cost of quantum addition.

Quantum, 2, 74, 2018.
8. Gidney, C. Windowed quantum arithmetic. arXiv preprint arXiv 1905.07682, 2019.
9. Gidney, C. and Jones, N. C. A CCCZ gate performed with 6 T gates. arXiv preprint arXiv 2106.11513, 2021.
10. Iten, R., Colbeck, R., Kukuljan, I., Home, J., and Christandl, M. Quantum circuits for isometries. Physical Review A, 93, 032318, 2016.
11. Jones, C. Low overhead constructions for the fault tolerant Toffoli gate. Physical Review A, 87, 022328, 2013.
12. Kissinger, A. and Meijer van de Griend, A. CNOT circuit extraction for topologically constrained quantum memories. arXiv preprint arXiv 1904.00633, 2019.
13. Kliuchnikov, V., Maslov, D., and Mosca, M. Asymptotically optimal approximation of single qubit unitaries by Clifford and T circuits using a constant number of ancillary qubits. Physical Review Letters, 110, 190502, 2013.
14. Komarath, B., Sarma, J., and Sawlani, S. Reversible pebble game on trees. In International Conference on Computing and Combinatorics, pages 83 to 94, 2015.
15. Kuroda, S. and Yamashita, S. Optimization of quantum Boolean circuits by relative phase Toffoli gates. In Proceedings of the International Conference on Reversible Computation, pages 20 to 27, 2022.
16. Maslov, D. Advantages of using relative phase Toffoli gates with an application to multiple control Toffoli optimization. Physical Review A, 93, 022311, 2016.
17. Meuli, G., Soeken, M., Roetteler, M., Bjorner, N., and De Micheli, G. Reversible pebbling game for quantum memory management. In Design Automation and Test in Europe, pages 288 to 291, 2019.
18. Meuli, G., Soeken, M., and De Micheli, G. SAT based CNOT and T quantum circuit synthesis. In International Conference on Reversible Computation, pages 175 to 188, 2018.
19. Montanaro, A. Quantum circuits and low degree polynomials over the finite field of two elements. Journal of Physics A Mathematical and Theoretical, 50, 084002, 2017.
20. Nash, B., Gheorghiu, V., and Mosca, M. Quantum circuit optimizations for NISQ architectures. arXiv preprint arXiv 1904.01972, 2019.
21. Oonishi, K., Tanaka, T., Uno, S., Satoh, T., Van Meter, R., and Kunihiro, N. Efficient construction of a control modular adder on a carry lookahead adder using relative phase Toffoli gates. IEEE Transactions on Quantum Engineering, 3, 1 to 18, 2022.
22. Selinger, P. Efficient Clifford and T approximation of single qubit operators. Quantum Information and Computation, 15, 159 to 180, 2015.
23. Shende, V. V., Bullock, S. S., and Markov, I. L. Synthesis of quantum logic circuits. IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems, 25, 6, 1000 to 1010, 2006.