A Quantum Test Algorithm

Jacob Biamonte and Marek Perkowski

Abstract— **Current processes validation methods rely on diverse input states and exponential applications of state tomography. Through generalization of classical test theory exceptions to this rule are found. Instead of expanding a complete operator basis to validate a process, the objective is to utilize quantum effects making each gate realized in the process act on a complete set of characteristic states and next extract functional information. Random noise, systematic errors, initialization inaccuracies and measurement faults must also be detected. This concept is applied to the switching class comprising the search oracle. In a first approach, the test set cardinality is held constant to six; both testability and added depth complexity of an additional ”design-for-test” circuit are related to the function realized in the oracle. Oracles realizing affine functions are shown to generate no net entanglement and are thus the easiest to test, where oracles realizing bent functions are the most difficult to test. A second approach replaces extraction complexity with a linear growth in experiment count. An interesting corollary of this study is the success found when addressing the classical test problem quantum mechanically. The validation of all classical degrees of freedom in a quantum switching network were found to necessitate exponentially fewer averaged observables than the number of tests in the classical lower bound.**

Keywords: Reversible Computers, Quantum Computers, Quantum Process Validation, Test Pattern Generation.

EDICS Category: B.1.3 Control Structure Reliability, Testing, and Fault-Tolerance, B.1.3.a Diagnostics, B.1.3.b Error-checking

PACS numbers: 03.67.Lx, 03.67.−a, 06.20.Dk, 76.60.−k

I. Introduction

TEST THEORY is now over 70 years old. The materialization of which emerged to avoid expanding the full set of binary basis vectors used to characterize classical networks [1][2]. These methods are well established for classical circuits, may they be generalized to quantum circuits?

The classical theory of computation implies local realism in all states of a sequential program's execution and is therefore inconsistent with physical reality [3]. Furthermore, quantum circuits often arise as a measure of algorithmic complexity [4]. For example, Adiabatic, Cluster State and Type-II Quantum Algorithms rely on computational models with no direct classical equivalent [5][6][7]. How then could purely quantum mechanical circuits be tested with ideas from this celebrated classical theory? Many models of quantum computation use circuits as a way to describe the actions on, and the interactions between collections of bi-state systems (qubits) sought to compute [8]. These interactions are induced under the perturbation of a classical force, where the quantum state of one system may alter the timed change of a second. This forms a depiction of nodes, wires and gates in time-dependent diagrams named quantum circuits. The design [9], realization [10], and test [11][12][13][14] of the component circuits required to assemble quantum computational devices continues to be a subject of much study.

Quantum computers will first impact society by simulating physical systems intractable via classical means [15]. Successful simulations are conjectured to necessitate as few as fifty qubits [16]. Experimental physicists who build quantum circuits have not yet experienced much need to research optimized testing methods due to the current attainable qubit count. All approaches to the quantum test problem are consequently exhaustive. The main approach now is to use process tomography such that for a system of n qubits 2^n initial states necessitate 2^n measurements, for a complexity of $\Theta(2^{2n})$ and a growth rate proportional to the experimental accuracy desired [13][14]. In a second approach (known as ancillary assisted process tomography [17]), n qubits are *mirrored* replacing 2^n initial states with an n dimensional state space entangled with each of the 2^n basis states of the system under test. However, in this approach any reduction in initial states increases measurement complexity, therefore the only offered advantage is experimental simplification (such as in optics [18]). The time required to test quantum circuits using current validation methods is just as intractable as the very problems these circuits will be built to solve. The quantum test problem must therefore be addressed.

Quantum computers offer a speed up over many classical combinatorial algorithms (such as quantum search [20] and counting [21]) that rely on quantum oracles [22]. As shown in Sec. I-B, oracles are constructed as classical switching networks whose implementation is quantum mechanical. Useful quantum oracles require large numbers of qubits, making process validation time even less tractable. Because of such wide use in quantum algorithms [23], designing test strategies specifically for oracles is one of the areas that classical test theory is shown here to improve.

The difficulty of extending the classical test theory has been a subject of discussion in recent times with the attempts outlined in [24]. Despite this interest, no connection has been made between established classical methods and any of the subtleties of quantum computation, making this study an important element to foster some growth in the field of quantum test engineering. Classically, the testability of the circuit class comprising the oracle has already received much attention after the 1972 paper by Sudhakar M. Reddy [2]. This paper presents a quantum mechanical switching network generalization of classical methods.

1) Structure of the paper: We begin to address these questions by first, in Sec. I-A and I-B giving an introduction to quantum mechanics and oracle construction. Sec. II discusses the quantum fault models used in this study. The intended audience are engineers and test theorists wishing to extend classical ideas to respective quantum counterparts. The Quantum Test Algorithm is presented in Sec. III followed by the conclusion in Sec. IV wherein we close with a short discussion of some open problems.

J.B. and M.P. are with Portland State University, 1900 SW Fourth Avenue, P.O. Box 751, Portland, Oregon 97201, USA; J.B. is the author with whom electronic correspondence shall be addressed: biamonte@ieee.org

A. Background

In quantum computation, classical bit registers are replaced with collections of qubits described by a corresponding density operator, $\rho = \sum_i p_i \left|\psi_i\right\rangle \left\langle\psi_i\right|$, where $\left|\psi_i\right\rangle$ represents a state vector, and ρ has trace one. When $tr(\rho^2) = 1$ the pure states description is complete and when $tr(\rho^2) < 1$ the mixed state of the system lacks information for complete description. The n dimensional state space of quantum computation is a composite complex vector space formed from an algebraic tensor product $(\rho_0 \otimes \rho_1 \otimes \ldots \otimes \rho_n)$ of density matrices representing component physical systems, ρ acts on this state space.

A set of measurement operators (*observables*) $\{M_m\}$ acting on the state space of a quantum system must be defined, in which index m references the measurement outcomes [23] and $\sum_m M_m M_m^\dagger = I_m$. Consider for example a collection of measurement operators on a two qubit system:

$$\{M_m\} = \{\left|00\right\rangle\left\langle 00\right|, \left|01\right\rangle\left\langle 01\right|, \left|10\right\rangle\left\langle 10\right|, \left|11\right\rangle\left\langle 11\right|\}. \quad (1)$$

This collection is complete since their sum is the 4×4 identity matrix, $\left|00\right\rangle\left\langle 00\right| + \left|01\right\rangle\left\langle 01\right| + \left|10\right\rangle\left\langle 10\right| + \left|11\right\rangle\left\langle 11\right| = I_4$. If ρ is found in eigenstate m, the resulting joint quantum state of the system will be $\rho_m = (M_m \rho M_m^\dagger)/tr(M_m^\dagger M_m \rho)$. The probability of result m is $p(m) = tr(M_m^\dagger M_m \rho)$. In the case of Eqn. 3, the probability that the system will be found in state $M_0 = \left|00\right\rangle\left\langle 00\right|$ is calculated as $tr(\left|00\right\rangle\left\langle 00\right|\left|00\right\rangle\left\langle 00\right|\rho_e) = \frac{1}{2}$. It is helpful to consider that each real number indexed by m along the diagonal of density matrix ρ corresponds to the probability of measuring a quantum system in the basis with corresponding index and the sum of all m diagonal entries is 1. System measurement allows m bits of classical information to be extracted. If one or more of these m bits is different than expected, the quantum switching network contained an error.

A quantum program is represented as evolution of a (ideally closed) system and described by a unitary transformation U (a matrix). A program must be decomposed into a product of physically realizable operations (matrices), and each elementary operation can be represented as a gate in a quantum circuit. The qubits in the system are initialized to state ρ, and the system evolves according to $\rho' = U\rho U^\dagger$. During evolution it is possible for a register of qubits to reside in superpositions of classical states. Superposition states may be factored, but only to the level of description that is local with respect to single qubits, such as:

$$\begin{aligned}\rho_s &= \frac{1}{2}\left|00\right\rangle\left\langle 00\right| + \frac{1}{2}\left|01\right\rangle\left\langle 01\right| + \frac{1}{2}\left|10\right\rangle\left\langle 10\right| + \frac{1}{2}\left|11\right\rangle\left\langle 11\right| \\ &= \left(\frac{\left|0\right\rangle\left\langle 0\right| + \left|1\right\rangle\left\langle 1\right|}{\sqrt{2}}\right) \otimes \left(\frac{\left|0\right\rangle\left\langle 0\right| + \left|1\right\rangle\left\langle 1\right|}{\sqrt{2}}\right). \quad (2)\end{aligned}$$

Evolution may also lead to entangled states that may not be factored to local descriptions, like this one:

$$\rho_e = \frac{1}{2}\left(\left|00\right\rangle\left\langle 00\right| + \left|11\right\rangle\left\langle 00\right| + \left|00\right\rangle\left\langle 11\right| + \left|11\right\rangle\left\langle 11\right|\right). \quad (3)$$

Regardless of physical separation, action of a witness on an entangled component has a composite impact. Furthermore, for an entangled system, component observation leads to classically impossible information gain regarding the state of the composite system, as a consequence of altering all states.

We conclude this section with a comment on notational conventions. Normalization constants are often omitted, as is an introduction to state vectors.[1] Shorthand notation for some common states must be defined, $\left|+\right\rangle = \left|0\right\rangle + \left|1\right\rangle$, $\left|-\right\rangle = \left|0\right\rangle - \left|1\right\rangle$, and $\psi = \left|\psi\right\rangle\left\langle\psi\right|$. The number of qubits considered is $(k+1)$ often denoted as n and N represents the maximum number of items in an oracle (2^{n-1}). The general notational conventions and vocabulary terms outlined in the textbook by Nielsen and Chuang [23] are used. A construction method for quantum oracles is next given.

B. Constructing Quantum Oracle Search Spaces

A classical oracle may be viewed as a boolean function $f : \{0,1\}^k \longrightarrow \{0,1\}$ in a black box, whose standard action leaves the top k input variables unchanged. The oracle's binary response to a given query $f(x_1, x_2, ..., x_k)$ is read on the $(k+1)^{th}$ bit. A query leading to a response of *binary-one* is called a *solution*. Unlike a classical oracle, quantum oracles respond to simultaneous queries by appending solutions with relative phases and leaving the bottom $(k+1)^{th}$ qubit unnoticeably changed, but how would one construct such a device?

Any boolean equation may be uniquely expanded to the fixed polarity Reed-Muller form [2] as:

$$\begin{aligned}f(x_1, x_2, ..., x_k) = c_0 \oplus c_1 x_1^{\sigma_1} \oplus c_2 x_2^{\sigma_2} \oplus \cdots \oplus c_n x_n^{\sigma_n} \oplus \\ c_{n+1} x_1^{\sigma_1} x_n^{\sigma_n} \oplus \cdots \oplus c_{2k-1} x_1^{\sigma_1} x_2^{\sigma_2}, ..., x_k^{\sigma_k}, \quad (4)\end{aligned}$$

where selection variable $\sigma_i \in \{0,1\}$, literal $x_i^{\sigma_i}$ represents a variable or its negation and any c term labeled c_0 through c_j is a binary constant 0 or 1. In Eqn. 4 only fixed polarity variables appear such that each is in either un-complemented or complemented form. The case where all variables in the expansion of Eqn. 4 appear in an un-complemented form will be considered in this work, this is known as a PPRM.[2]

Example:

$$\begin{aligned}f(x_1, x_2, x_3, x_4) = 1 \oplus x_1 \oplus x_2 \oplus x_3 \oplus x_3 x_4 \oplus x_1 x_3 x_4 \oplus \\ x_1 x_2 x_3 \oplus x_2 x_3 x_4 \quad (5)\end{aligned}$$

Each term in the expansion of Eqn. 5 is called a product term [26], and each variable x_i a literal. Here a total of seven product terms and fourteen un-complemented literals are given. For example, $x_3 \cdot x_4$ is such a product term, with literals x_3 and x_4 (constant 1 however, is not considered to be a product term). Each product term for a given PPRM expansion may be realized by the arbitrary quantum controlled-NOT gate ($k-$CN) given in Fig. 1.

In the quantum circuit model of computation, horizontal wires represent the passage of time from left to right, while gates and controls[3] represent both interactions between and actions on qubits. A control is denoted with a black dot (•), and may be connected with other black dots using wires. For control gates, each connection is a conjunctive path; each literal in a given product term receives one black dot on the quantum circuit diagram. A vertical wire is next placed, interconnecting all of the black dots and the target of the gate written as a NOT symbol ($\oplus$). Repeating this procedure for each product term in Eqn. 5 leads to the network realization given in Fig. 2. Above each gate is the label p_i, p refers to a product term in the expansion of Eqn. 5, and i the index used to label all seven products. The network realization given in Fig. 2 may be implemented via a unitary approximation [23] or by using controlled 4^{th} root of NOT gates in the design considered by Barenco et al., (see [9], page 17, § 7). This

[1] State vectors are referenced using Dirac Notation, such as arbitrary example, $\left|\psi\right\rangle = \alpha\left|0\right\rangle + \beta\left|1\right\rangle$ and respective conjugate, $\left\langle\psi\right| = \alpha^*\left\langle 0\right| + \beta^*\left\langle 1\right|$.

[2] PPRM: Positive Polarity Reed-Muller Expansion such that each literal appears only un-complemented.

[3] Controls are often called nodes.

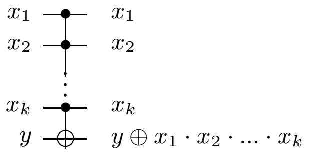

Fig. 1. $k-$CN Gate Realizing $y \oplus x_1 \cdot x_2 \cdot \ldots \cdot x_k$ on the $(k+1)^{th}$ qubit.

example will be used again so it is worth stating explicitly that p_0 corresponds to x_1, p_1 to x_2, p_3 to x_3, p_3 to x_3x_4, p_4 to $x_1x_3x_4$, p_5 to $x_1x_2x_3$ and finally p_6 to $x_2x_3x_4$.

The underpinning difference of operation between a classical and quantum *phase oracle* will now be made clear. Quantum gates exhibit a feature known as phase kick-back.[4] That is, if the input state of the target is an eigenvector of the control gate's operation, the eigenvalue of the target state traverses backwards to the activating state of the control qubit(s), leaving the target unchanged up to a global phase. The eigenvector states of $k-$CN gates are created using another gate known as the Hadamard operator: drawn schematically as $-\boxed{H}-$, defined algebraically in Eqn. 6 and it's action on some common states are: $|0\rangle \longrightarrow |+\rangle$, $|1\rangle \longrightarrow |-\rangle$, $|+\rangle \longrightarrow |0\rangle$ and $|-\rangle \longrightarrow |1\rangle$.

$$H = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)\langle 0| + \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)\langle 1| \qquad (6)$$

Typically Boolean function f is constructed by means of a $k-$CN network and placed in a black box with label $\mathcal{O}$ (for oracle). The bottom $(k+1)^{th}$ bit contains the realization of f to be read at the box's right. The top k inputs to the box begin in state $|0\rangle$ and the $(k+1)^{th}$ input (target) qubit starts in state $|1\rangle$. The Hadamard operation $H^{\otimes(k+1)}$ is next applied. Generally the black box takes as input:

$$H^{\otimes(k+1)} : |0\rangle^{\otimes k} \otimes |1\rangle \longrightarrow (|0\rangle + |1\rangle)^{\otimes k} \otimes (|0\rangle - |1\rangle) \qquad (7)$$

Inside the black box all of the targets act on state $|-\rangle$ (an eigenvector of the $k-$CN gate) and the top k qubits remain in a superposition of all possible classical states. The true minterms are inputs to a Boolean function that evaluate to 1 where the false minterms evaluate to 0. Each term in the superposition on the top k bits representing a true minterm in the switching function f realized in the oracle will be appended with a negative (relative) phase. The phase of states that do not represent true minterms are left invariant. This is seen by examining the truth table from Fig. 3. The action of an oracle $\mathcal{O}$, realizing a binary function $f(x_1, x_2, ..., x_k)$, is represented by the general transform $\mathcal{O} : |k\rangle \otimes |-\rangle \longrightarrow (-1)^{f(k)} |k\rangle \otimes |-\rangle$.

The oracle's introduction is complete. Before continuing on to Sec. II wherein the considered gate level quantum fault models are defined, it is now mentioned that quantum phase kickback is key to our study. Phase kickback faults impacting $k-$CN gates are addressed in Axioms 4 and 5. Sec. III presents the quantum test algorithm that extracts information from the phase of the quantum state to determine if a given oracle is functional.

II. Gate Level Quantum Fault Models

Classically, one defines a testability measure as the product of observability and controllability. A fault present in an entangled state generally results in probabilistic measurement outcomes thereby decreasing the observability of failures. Controllability allows one to propagate a specific input vector through a network, such that it will map a test vector to a place of fault. This represents an added challenge in the case of quantum circuits, since inputs will become entangled and in many cases specific (local) inputs to a certain fault location may not be possible. Functional quantum faults at the gate level were defined as Axioms that a complete test set must satisfy in Ref. [24]. These Axioms are used to logically test the gate level function of all network components and are presented here for completeness. As will be seen in Sec. III, the entanglement added to the state vector during a test must then be removed to properly observe failures.

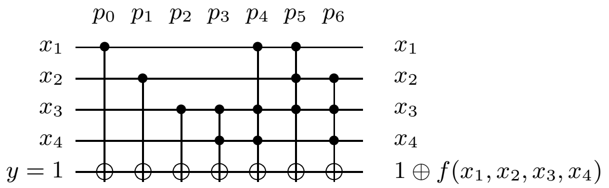

Fig. 2. Quantum Network Realization of Eqn. 5 built from arbitrary k-CN gates as shown in Fig. 1. The truth table of this oracle is given in Fig. 3.

phase	*state*	x_1	x_2	x_3	x_4	f
$+$	$\|0000\rangle$	0	0	0	0	0
$+$	$\|0001\rangle$	0	0	0	1	0
$-$	$\|0010\rangle$	0	0	1	0	1
$+$	$\|0011\rangle$	0	0	1	1	0
$-$	$\|0100\rangle$	0	1	0	0	1
$-$	$\|0101\rangle$	0	1	0	1	1
$+$	$\|0110\rangle$	0	1	1	0	0
$+$	$\|0111\rangle$	0	1	1	1	0
$-$	$\|1000\rangle$	1	0	0	0	1
$-$	$\|1001\rangle$	1	0	0	1	1
$+$	$\|1010\rangle$	1	0	1	0	0
$+$	$\|1011\rangle$	1	0	1	1	0
$+$	$\|1100\rangle$	1	1	0	0	0
$+$	$\|1101\rangle$	1	1	0	1	0
$+$	$\|1110\rangle$	1	1	1	0	0
$-$	$\|1111\rangle$	1	1	1	1	1

Fig. 3. Oracle Truth Table for Eqn. 5 implemented by the network in Fig. 2: Boolean function f is implemented quantum mechanically. Each of the 2^k terms in a superposition input that evaluate to *logic-one* will be marked with a negative phase (also shown in Eqn. 15, in Sec. III).

In quantum error correcting codes, error locations are between circuit stages, and have quantifiable error probabilities or strengths of occurrence [28]. For example, consider the single stage circuit shown in Fig. 4. The numbered locations of possible gate external faults are illustrated by placing an "$\times$" on the line representing a qubits time traversal and here, the gate, initial states $(|i_0\rangle, |i_1\rangle, |i_2\rangle)$ and measurements (m_0, m_1, m_2) may also contain errors. Error and Fault Locations are formally defined next in Def. 1.

***Definition** 1:* Error/Fault Location: The wire locations between stages as well as any node, gate initial state or measurement in a given network (see Fig. 4).

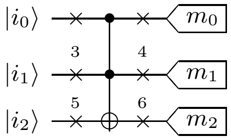

Fig. 4. $2-$CN gate with error locations.

A quantum test set is a set of initial state and measurement

[4] See [27], the 1999 PhD thesis of M. Mosca, *Quantum Computer Algorithms*, for background on using quantum phase for various quantum computational tasks.

pairs designed to drive a network to threshold limits. For example, one may develop a test set that first turns as many gates on as possible, next turns the highest possible number of gates off and then sends phase through as many gates during one test as the structure of the network would allow. It is the goal of this paper to develop complete test sets that sample failure rates. We therefore consider a set of error models adequately capturing the nature of fault types occurring in a given circuit, together with their locations. Ref. [24] introduced the concept of what is known as the *quantum single fault model*. This allows the separate consideration of all errors at each location for a given quantum circuit. We present first Def. 2 and next Conjecture 1, both related to this idea.

***Definition** 2:* Quantum Single Fault Model: For simplification the ”quantum single fault model” is assumed in this work. In the single fault model, test plans are optimized for all considered faults assuming that only a single failure perturbs the quantum circuit exclusively. Multiple faults will accumulate and be detected, but the single fault model makes it much easier to develop test plans.

***Conjecture** 1:* A test set designed to detect all considered single errors will detect and sample the accumulated impact of multiple errors at multiple locations.

The following definitions are used to define some of the fault types considered in this work. Complete fault coverage occurs after a test set has determined that the considered fault(s) are not physically present in a given circuit.

***Definition** 3:* Pauli Fault Model: The addition of an unwanted Pauli matrix in a quantum network, at any error location and with placement probability p. The Pauli matrices are given in Eqn. 8, 9 and 10.

$$\sigma_x = |1\rangle\langle 0| + |0\rangle\langle 1| \tag{8}$$

$$\sigma_y = i\,|0\rangle\langle 1| - i\,|1\rangle\langle 0| \tag{9}$$

$$\sigma_z = |0\rangle\langle 0| - |1\rangle\langle 1|, \tag{10}$$

***Definition** 4:* Initialization Error: A qubit that statistically favors correct preparation in one basis state over the other.

***Definition** 5:* Measurement Fault Model: A single functional measurement gate is replaced with a faulty measurement gate that statistically favors returning *logic-zero* or a *logic-one*.

The Initialization Errors (Axiom 3) and the Measurement Faults (Axiom 8) are considered to be largely part of the quantum computers' classical functionality and have clearly defined error locations. Test sets detecting quantum noise and systematic errors [30] satisfy Axioms 1, 2 and 3. To avoid the complications experienced with quantum test vector controllability the test sets in this work are shown to satisfy the following gate level functional Axioms: Lost Phase Faults (Axioms 4 and 5), Faded Control Faults (Axiom 6), and Forced Gate Faults (Axiom 7). In Sec. III a test algorithm in accordance with these Axioms that samples failure rates (Conjecture 1) will be given.

***Quantum Test Axiom** 1:* A bit flip (σ_x or σ_y) at any error location must be detectable. ■

***Quantum Test Axiom** 2:* A phase flip (σ_z or σ_y) at any error location must be detectable. ■

***Quantum Test Axiom** 3:* Each qubit must be initialized in both basis states $|0\rangle$ and $|1\rangle$. ■

***Quantum Test Axiom** 4:* With the target acting on state $|-\rangle$: Each gate must be shown to attach a relative phase to arbitrary activating state $|a\rangle$ with both positive and negative eigenvalues. Furthermore, each gate must be shown not to attach a relative phase to arbitrary non-activating state $|n\rangle$ with both positive and negative eigenvalues. The target state must remain globally invariant under both $|a\rangle$ and $|n\rangle$. ■

***Quantum Test Axiom** 5:* With the target acting on state $|+\rangle$: relative phase must be shown not to change under arbitrary activating state $|a\rangle$ with both positive and negative eigenvalues. Furthermore, relative phase must not change under arbitrary non-activating state $|n\rangle$ with both positive and negative eigenvalues. ■

***Quantum Test Axiom** 6:* For the target acting separately on basis state $|0\rangle$ and $|1\rangle$: All controls in a gate must be activated concurrently. Furthermore, each control must be addressed with a non-activating state. ■

***Quantum Test Axiom** 7:* Each target must separately act on basis state inputs $|0\rangle$ and $|1\rangle$. ■

***Quantum Test Axiom** 8:* Each qubit must be measured in both *logic-zero* and *logic-one* states. ■

Based on the Axioms and Definitions from Ref. [24], a discussion of a test set satisfying these Axioms for quantum oracles is discussed next is Sec. II-A.

A. Conclusions based on the Gate Level Fault Models

Traditionally test plans are optimized to detect all of the most common error types [32] and circuits are designed with ease of test in mind. A test plan is developed for the purpose of isolating a correct circuit from a circuit containing any of the considered errors. In practice, the choice of the fault model will be determined by a particular quantum circuit technology, as well as how the circuit will be used. In this work the functional use of $k-$CN networks are oracle search spaces. Building on an understanding of the different failures possible, here it is shown constructively that any $k-$CN gate exhibits twelve, functionally distinct actions. When used in a phase oracle, the gate level faults that need to be considered are the Phase Faults from Axioms 4 and 5. When classical inputs are considered, Faded Control Faults (Axiom 6) and Forced Gate Faults (Axiom 7) must be taken into account. Theorem 1 presents the four classical degrees of freedom possible in any $k-$CN gate.

***Theorem** 1:* A quantum $k-$CN gate is capable of four characteristic classical operations. (By characteristic it is meant that all other operations are variants of this basic set.)

Proof: The gate is able to act on a $|0\rangle$ and a $|1\rangle$ state when all controls are set to high. The two remaining functions are simply to act on $|0\rangle$ and $|1\rangle$ when one or more control(s) is addressed with a non-activating state (the action of course should be to do nothing). There are $2^k - 1$ input states that do not activate the gate, but these inputs all probe the *off* function. Similarly, each control has two logical functions. The first is to be addressed with a logical $|0\rangle$ and the second is to be addressed with a $|1\rangle$. (See test vectors v_0, v_1, v_2 and v_3 from Fig. 5. A similar situation arises with classical EXOR gates [1].) ■

Provided the state of the top k bits is some equal superposition and the target of the gate acts on a state with the following form: $|0\rangle + e^{\pm i\varphi}|1\rangle$. Under this condition, the inputs to a $k-$CN gate are expressed as:

$$|\psi_{in}\rangle \longrightarrow \left[\sum_{x\,=\,0}^{2^k-1} w_x\,|x\rangle\right] \otimes (|0\rangle + e^{\pm i\varphi}\,|1\rangle), \tag{11}$$

Minterm	*Target State*	*Minterm*	*Target State*
$e^{+i\phi}\vert true\rangle$	$(\vert 0\rangle + e^{+i\varphi}\vert 1\rangle)$	$e^{+i(\phi+\varphi)}\vert true\rangle$	$(\vert 0\rangle + e^{+i\varphi}\vert 1\rangle)$
$e^{-i\phi}\vert true\rangle$	$(\vert 0\rangle + e^{+i\varphi}\vert 1\rangle)$	$e^{-i(\phi-\varphi)}\vert true\rangle$	$(\vert 0\rangle + e^{+i\varphi}\vert 1\rangle)$
$e^{+i\phi}\vert false\rangle$	$(\vert 0\rangle + e^{+i\varphi}\vert 1\rangle)$	$e^{+i(\phi)}\vert false\rangle$	$(\vert 0\rangle + e^{+i\varphi}\vert 1\rangle)$
$e^{-i\phi}\vert false\rangle$	$(\vert 0\rangle + e^{+i\varphi}\vert 1\rangle)$	$e^{-i(\phi)}\vert false\rangle$	$(\vert 0\rangle + e^{+i\varphi}\vert 1\rangle)$
$e^{+i\phi}\vert true\rangle$	$(\vert 0\rangle + e^{-i\varphi}\vert 1\rangle)$	$e^{+i(\phi-\varphi)}\vert true\rangle$	$(\vert 0\rangle + e^{-i\varphi}\vert 1\rangle)$
$e^{-i\phi}\vert true\rangle$	$(\vert 0\rangle + e^{-i\varphi}\vert 1\rangle)$	$e^{-i(\phi+\varphi)}\vert true\rangle$	$(\vert 0\rangle + e^{-i\varphi}\vert 1\rangle)$
$e^{+i\phi}\vert false\rangle$	$(\vert 0\rangle + e^{-i\varphi}\vert 1\rangle)$	$e^{+i\phi}\vert false\rangle$	$(\vert 0\rangle + e^{-i\varphi}\vert 1\rangle)$
$e^{-i\phi}\vert false\rangle$	$(\vert 0\rangle + e^{-i\varphi}\vert 1\rangle)$	$e^{-i\phi}\vert false\rangle$	$(\vert 0\rangle + e^{-i\varphi}\vert 1\rangle)$

Fig. 6. A $k-$CN Gate Truth Table (Case: 2 top, Case: 1 bottom): Illustrating all of the different possible gate actions for orthogonal setting of variables ϕ and φ. A $|true\rangle$ minterm activates the gate, any $|false\rangle$ minterm does not.

$$\begin{array}{lcccccc} v_0 \rightarrow & 0 & 0 & 0 & 0 & \cdots & 1 \\ v_1 \rightarrow & 0 & 0 & 0 & 0 & \cdots & 0 \\ v_2 \rightarrow & 1 & 1 & 1 & 1 & \cdots & 1 \\ v_3 \rightarrow & 1 & 1 & 1 & 1 & \cdots & 0 \end{array}$$

Fig. 5. Classical test vectors (v_0, v_1, v_2, v_3) acting on binary basis vectors $\{0, 1\}$ with the gate first off (v_0, v_1) and then on (v_2, v_3). The rightmost bit in the figure is applied to the $(k + 1)^{th}$ bit.

where $w_x = e^{\pm i\phi}$. Similarly, as in the case of Theorem 1, certain operations define the gate's function. Furthermore, these actions are independent of the entanglements[5] experienced in the system prior to the application of the gate (the gate generates entanglement by acting on individual product terms in a superposition). The arbitrary quantum superposition state defined in Eqn. 11 allows one to consider each input as a separate state. In the column denoted minterm from Fig. 6, $|true\rangle$ minterms activate the gate while $|false\rangle$ terms do not. Under this consideration the following holds:

***Theorem** 2:* A $k-$CN gate is capable of eight characteristic quantum operations. (We consider quantum operations as those that manipulate quantum phase and non-classical superposition states; characteristic has the same meaning as in Theorem 1.)

Proof: The proof is constructive:

Case 1: When activated, quantum gates exhibit phase kickback when the state of the target is $|0\rangle + e^{-i\varphi}|1\rangle$. The activating state can have a phase of $+w_x$ or $-w_x$. Furthermore, a non-activating state can have a phase of $+w_x$ or $-w_x$ and of course, nothing should happen when acted on by the $k-$CN gate.

Case 2: (The opposite of Case 1.) The alternative case is that the target acts on state $|0\rangle + e^{+i\varphi}|1\rangle$. As before, the activating and non-activating states can have phases of $+w_x$ or $-w_x$. Nothing should happen under the case of both an activating and a non-activating state. This functionality is probed in four additional tests.

We draw the readers attention now to the table in Fig. 6 for the illustration of Case 1 and Case 2. Variables ϕ and φ are set to create states that are operated on by the $k-$CN gate, these are the combinations of actions considered. The Proof is concluded by mentioning that, all the quantum functions of the $k-$CN gate represent one variant of these eight cases when used in a phase oracle. ■

Thus according to Theorems 1 and 2 in total we need $4 + 8 = 12$ non-entangled tests to identify the function of any $k-$CN gate. Although Theorems 1 and 2 are simple in concept, only when these ideas are made clear is one able to fully characterize all of the gates in a given quantum search oracle. It is therefore safe to move on and, in the next section (III), present a test set in accordance with all the Axioms and principles of this section.

III. The Quantum Error Detection Algorithm

The quantum test algorithm introduced in this section utilizes entanglement as a controllability resource to combine test vectors and hence reduce test sets while the inherent reversibility of quantum circuits increases the observability of errors [31]. The Axioms mentioned in Sec. II are shown to be satisfied and the test algorithm is convergent. An explicit example is given illustrating how one would go about testing a quantum oracle.

Tests T_1, T_2, T_5 and T_6 verify all classical degrees of freedom. Tests T_3 and T_4 verify the phase kickback features of the oracle. As a proof of concept the introduced method holds the test set cardinality to constant six, increasing the complexity of added stages for tests T_3 and T_4. This approach helps better tie classical ideas with quantum test set generation. This is due to the fact that classically, circuits realizing linear functions are easy to test due to their high level of controllability. Quantum mechanically, a search oracle realizing an affine function generates no net entanglement in the top k bits provided input state $|\pm\rangle^{\otimes k} \otimes |-\rangle$. Thus, it is easier to control these states, extract functional information or observe failure. Entanglement added by the network during tests T_3 and T_4 must be removed by additional circuit stages to return the system to a product state and allow a deterministic measurement outcome. An analysis of this presents powerful concepts to design test plans for quantum circuits that are both highly controllable and that allow high observability of errors. For example, Sec. III-F presents a second approach where T_3 and T_4 are replaced with other tests. These highly controllable tests have constant entanglement and reduce the quantum test problem to a cardinality of $(5 + 4\lceil k/2\rceil)$ by ”walking” $\lceil k/2\rceil$ EPR pairs [23] down the controls mirrored by $\lceil k/2\rceil$ Bell measurements. Classically, the additional circuitry used to generate test sequences is known as *BIST* (*Build In Self Test Circuit*). For completeness, Def. 6 is present.

***Definition** 6:* Quantum Build In Self Test Circuit (*QBIST*): A quantum circuit designed to test a second quantum circuit; the quantum circuit under test (*QCUT*). A *QBIST* circuit may be built at the input and/or output terminals of the *QCUT*, and the *QBIST* stage is always assumed to contain no errors.

Consider the example circuit presented in Fig. 2. The analysis given in the coming subsections begins by generating an input state that turns all the gates in the network *on* and *off*

[5] In terms of an ability to generate entanglement, the CN was shown to be the most robust gate in the presence of noise [33].

concurrently. This concurrent action tests all gates exhaustively on both computational basis states $|0\rangle$ and $|1\rangle$, (something classically impossible in just two tests). Denote these tests as T_1 and T_2, and their general form on a k variable function follows:

T_1: $(|0\rangle^{\otimes k} + |1\rangle^{\otimes k}) \otimes |0\rangle$
T_2: $(|0\rangle^{\otimes k} - |1\rangle^{\otimes k}) \otimes |1\rangle$

The classical equivalent of tests T_1 and T_2 was given in Fig. 5 (where T_1 corresponded to vectors v_0 and v_2, and T_2 corresponded to both v_1 and v_3). Together tests T_1 and T_2 will be shown to satisfy Axioms 1, 3, 6, 7 and 8 in Sec. III-A and III-B.

Sec. III-C considers tests T_5 and T_6. These tests are shown to satisfy Axiom 5 by using the following states as oracle inputs: $|+\rangle^{\otimes k} \otimes |+\rangle$ and $|-\rangle^{\otimes k} \otimes |+\rangle$. In both tests, the state at the controls will not impact the state at the target, leaving all qubits—ideally—unchanged (since no net entanglement is generated).

Sec. III-D and III-E investigate the ability of the network to both attach a relative phase to each activating term in the superposition and to leave non-activating states unaltered. This in general is a complex procedure, that in the first case can be done in two tests denoted as T_3 and T_4. Test T_3 utilizes state $|+\rangle^{\otimes k} \otimes |-\rangle$ and test T_4 utilizes state $|-\rangle^{\otimes k} \otimes |-\rangle$ as input to the oracle. However, additional "design-for-test" stages must be added to the end of the circuit. These stages remove the entanglements added by the oracle, returning the system to a local (factorable) description, thereby leading to a deterministic measurement. Tests T_3 and T_4 are shown to satisfy Axiom 4. Test T_1 is now considered.

A. *Test* T_1*:* $(|0\rangle^{\otimes k} + |1\rangle^{\otimes k}) \otimes |0\rangle$

In test T_1, all qubits are initialized as: $|0000\rangle \otimes |0\rangle$. The action of the first *QBIST*$_{11}$ stage (from Fig. 8) creates the following oracle input state:

$$\textit{QBIST}_{11} : |0000\rangle \otimes |0\rangle \longrightarrow \left(|0\rangle^{\otimes k} + |1\rangle^{\otimes k}\right) \otimes |0\rangle . \quad (12)$$

The left half of the entangled test sequence is $|0000\rangle \otimes |0\rangle$. It is clear that for a "*gold circuit*" not one gate turns on, and the target qubit will be left untouched. For the right half of the entangled test vector, each gate in the circuit turns on, and this cycles the $(k+1)^{th}$ qubit initially starting in $|0\rangle$ back and forth between basis states. The state of the last qubit after the oracle is $|0\rangle$.[6] The purpose of *QBIST*$_{12}$ is simply to remove the phase induced entanglement experienced on the top k qubits. The intermittent states at each stage of the circuit under test T_1 are shown in Fig. 7. The final step in the *QBIST*$_{12}$ circuit applies a Hadamard gate to the top qubit, resulting back in the starting state, $|0000\rangle \otimes |0\rangle$, thereby completing test T_1. The complexity of the added CN and H gates needed for test T_1 is $2(k-1)$CN+2H.

B. *Test* T_2*:* $(|0\rangle^{\otimes k} - |1\rangle^{\otimes k}) \otimes |1\rangle$

No physical change is made to the circuit from Fig. 8, however the qubits are now initialized to state $|1111\rangle \otimes |1\rangle$. The outcome is similar to test T_1, the bottom qubit is toggled a total of seven times resulting in the final state of $|1\rangle$. (*Each gate that acted on* $|0\rangle$ *in test* T_1 *now acts on* $|1\rangle$ *thereby exhaustively probing every classical input combination of each* $k-$*CN gate, seen in Fig. 8.*) The *QBIST*$_{12}$ again disentangles the test responses, resulting back in the initial state of $|1111\rangle \otimes |1\rangle$.

Stage	Action of Stage
$in \longrightarrow$	$\|0000\rangle \otimes \|0\rangle$
$QBIST_{11} \longrightarrow$	$(\|0000\rangle + \|1111\rangle) \otimes \|0\rangle$
$p_0 \longrightarrow$	$\|0000\rangle\,\|0\rangle + \|1111\rangle\,\|1\rangle$
$p_1 \longrightarrow$	$\|0000\rangle\,\|0\rangle + \|1111\rangle\,\|0\rangle$
$p_2 \longrightarrow$	$\|0000\rangle\,\|0\rangle + \|1111\rangle\,\|1\rangle$
$p_3 \longrightarrow$	$\|0000\rangle\,\|0\rangle + \|1111\rangle\,\|0\rangle$
$p_4 \longrightarrow$	$\|0000\rangle\,\|0\rangle + \|1111\rangle\,\|1\rangle$
$p_5 \longrightarrow$	$\|0000\rangle\,\|0\rangle + \|1111\rangle\,\|0\rangle$
$p_6 \longrightarrow$	$\|0000\rangle\,\|0\rangle + \|1111\rangle\,\|1\rangle$
$QBIST_{12} \longrightarrow$	$\|0000\rangle \otimes \|0\rangle$

Fig. 7. T_1 test pattern and impact at each gate in the circuit. Gates as labeled left to right p_1 to p_6.

In tests T_1 and T_2 each node is addressed with both activating and non-activating states. Furthermore, each qubit is initialized and measured in both basis states. Tests T_1 and T_2 have an added CN and H gate complexity of $4(k-1)$CN+4H. The following Theorems prove which faults have been detected with tests T_1 and tests T_2 and are general for n bit oracles:

Theorem *3:* Either test T_1 or test T_2 will detect σ_x and σ_y bit flips at any error location, thus satisfying Axiom 1.

Proof: Tests T_1 and T_2 both satisfy Axiom 1. The proof in this section is given for test T_1 and is nearly identical to the steps taken for test T_2. Consider now test T_1:

Case 1: The top (1^{st}) qubit is flipped: *QBIST*$_{12}$ receives state $(|1\rangle\,|0\rangle^{\otimes(k-1)} \pm |0\rangle\,|1\rangle^{\otimes(k-1)})$ as input. After successive applications of CN$_{i-1,i}$ from $i = k$ to $i = 2$ the state will be $(|11\rangle\,|0\rangle^{\otimes(k-2)} \pm |01\rangle\,|1\rangle^{\otimes(k-2)}) = (|0\rangle \pm |1\rangle) \otimes |1\rangle \otimes |0\rangle^{\otimes(k-2)}$. Thus, a bit flip impacting the 1^{st} bit is detectable on the 2^{nd} bit. Given a bit flip impacting any other qubit q, $(1 < q \leq k)$ *QBIST*$_{12}$ receives $(|0\rangle^{\otimes(q-1)}\,|a\rangle\,|0\rangle^{\otimes(k-q)} \pm |1\rangle^{\otimes(q-1)}\,|\bar{a}\rangle\,|1\rangle^{\otimes(k-q)})$ as input state. A similar relation holds such that a bit flip on the $(q-1)^{th}$ bit is detectable on the q^{th} and possibly the 1^{st} bit if the phase is also inverted. For errors impacting any qubit other than the 1^{st}, both the q^{th} bit as well as the $(q+1)^{th}$ (impacted bit) will show the error.

Case 2: Bottom $(k+1)^{th}$ qubit is flipped: Normally the top k bits and the bottom $(k+1)^{th}$ bits are factorable when entering the final *QBIST*$_{12}$ stage. Assume an even number of gates in the oracle and that instead of state: $(|0\rangle^{\otimes k} + |1\rangle^{\otimes k}) \otimes |0\rangle$ the final *QBIST*$_{12}$ receives the worst case state of $|0\rangle^{\otimes k}\,|0\rangle + |1\rangle^{\otimes k} \otimes |1\rangle$. The final *QBIST*$_{12}$ will not remove the entanglement associated with the $(k+1)^{th}$ bit. This is detectable based on p, the probability that a bit flip occurred in the computational basis in the first place, satisfying Axiom 1. This is the only fault that, when deterministically present interjects a probabilistic outcome in observability. ■

Theorem *4:* Together tests T_1 and T_2 initialize each qubit in both basis states so that Axiom 3 is satisfied.

Proof: In test T_1 the initial state of the register is $|0\rangle^{\otimes k} \otimes |0\rangle$ and in test T_2 the initial state is $|1\rangle^{\otimes k} \otimes |1\rangle$, therefore Axiom 3 is satisfied. ■

Theorem *5:* Taken together tests T_1 and T_2 activate all controls concurrently and each control is addressed with a non-activating state while the target is separately in basis state $|0\rangle$

[6] If an an even number of gates were present a slight modification to the final half of the *QBIST*$_{12}$ circuit must be made. This modification is the removal of the first CN gate at the start of the *QBIST*$_{12}$ acting on the $(k+1)^{th}$ qubit and controlled by the k^{th} qubit. In general for an odd number of gates in a quantum network prior to the final *QBIST*$_{12}$ stage the circuit will be in state $|0\rangle^{\otimes k}\,|0\rangle \pm |1\rangle^{\otimes k}\,|1\rangle$. The addition of a CN$_{k,k+1}$ gate removes unwanted entanglement so that the final qubit will be left in a product state.

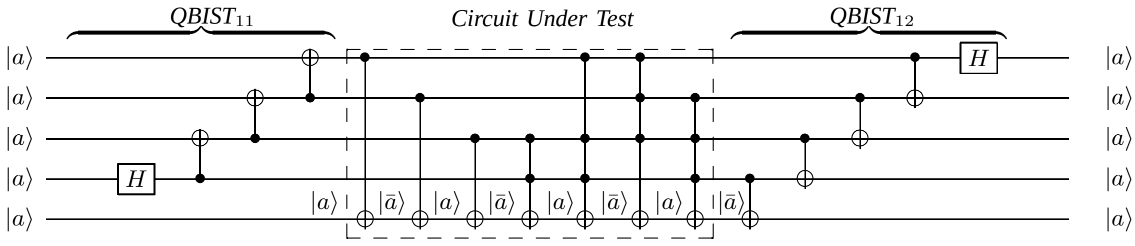

Fig. 8. Tests T_1 and T_2 (GHZ states): In Test T_1, $a = 0$ so the circuit starts off in state: $|0000\rangle$. *QBIST*$_{11}$ maps this state to the oracle's input as: $(|0000\rangle + |1111\rangle) \otimes |0\rangle$. In Test T_2, $a = 1$ and the input to the oracle is: $(|0000\rangle - |1111\rangle) \otimes |1\rangle$. *QBIST*$_{12}$ removes entanglement and returns the system to a product state.

and next $|1\rangle$ satisfying Axiom 6.

Proof: In tests T_1 and T_2 the test state prior to application of the oracle is $(|0\rangle^{\otimes k} \pm |1\rangle^{\otimes k}) \otimes |\bar{a}\rangle$. In both tests T_1 and T_2 the term $|0\rangle^{\otimes k}$ addresses each control with a non-activating state, the term $\pm |1\rangle^{\otimes k}$ activates all gates and in both tests the target is in a basis state. This satisfies Axiom 6. ■

***Theorem** 6:* Taken together tests T_1 and T_2 force each gate in the circuit to act on both basis states, thereby satisfying Axiom 7.

Proof: In both tests T_1 and T_2 the term $\pm |1\rangle^{\otimes k}$ activates all gates. Each gate in test T_1 that received target input state $|a\rangle$ received target input state $|\bar{a}\rangle$ in test T_2, thus satisfying Axiom 7. ■

***Theorem** 7:* After executing test T_1 and T_2 each qubit will be measured in both basis states, thus satisfying Axiom 8.

Proof: The result of test T_1 is $|0\rangle^{\otimes(k+1)}$ and the measured result pending the success of test T_2 is $|1\rangle^{\otimes(k+1)}$ thus satisfying Axiom 8. ■

C. Super Tests T_5 and T_6: $|+\rangle^{\otimes k} \otimes |+\rangle$ and $|-\rangle^{\otimes k} \otimes |+\rangle$

The two following tests are simple to conceptualize, as seen in Fig. 9 they have an added gate complexity of $4kH$. When $a = 0$ test T_5 generates input state $|++++\rangle \otimes |+\rangle$ and when $a = 1$ test T_6 generates input state $|----\rangle \otimes |+\rangle$. Since the eigenvalue of the target state is $+1$, no change in relative phase should result from propagation through the quantum circuit and the state of the register should not become entangled. Theorem 8 proves that test T_5 combined with test T_6 satisfy Axiom 5 with an added gate complexity of $4k$H.

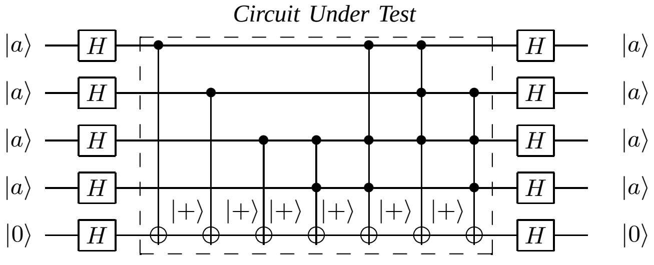

Fig. 9. Tests T_5 and T_6 (Super Tests): Test $|+\rangle^{\otimes k} \otimes |+\rangle$ is first generated ($a = 0$, T_5) and next test $|-\rangle^{\otimes k} \otimes |+\rangle$ is applied ($a = 1$, T_6). The target of each k-CN gate acts on state $|+\rangle$. No entanglement is added in either test, since no relative phase change of individual superposition term(s) will occur.

***Theorem** 8:* Together tests T_5 and T_6 satisfy Axiom 5.

Proof: In both tests T_5 and T_6 the state of the target qubit is $|+\rangle$. Any gate that was activated by a state with an eigenvalue $+1$ in test T_5 will be activated by a state with an eigenvalue -1 in test T_6. Relative phase will not change under arbitrary non-activating and activating states since the target state has an eigenvalue of $+1$, satisfying Axiom 5. ■

***Theorem** 9:* Either one of tests T_5 or T_6 detects σ_z or σ_y phase flips and therefore satisfies Axiom 2.

Proof: Here the Proof is done considering test T_5, however the steps are the same as those needed for test T_6. Consider state $|+\rangle^{\otimes k} \otimes |+\rangle$, this is a product state that may be expanded as: $|+\rangle \otimes \cdots \otimes |+\rangle \otimes |+\rangle \otimes |+\rangle \otimes \cdots \otimes |+\rangle$. The state of the target is $|+\rangle$ and therefore phase will not make the state non-local (with an exception of a phase flip on the $(k+1)^{th}$ bit, in that case the bottom bit will deterministically reveal the presence of an error). Given a σ_z fault impacting any qubit, the state becomes $|+\rangle \otimes \cdots \otimes |+\rangle \otimes |-\rangle \otimes |+\rangle \otimes \cdots \otimes |+\rangle$. In the final stage of *QBIST*$_{52}$ a Hadamard operation $H^{\otimes(k+1)}$ is applied to the register:

$$H^{\otimes(k+1)} \cdot |+\rangle \otimes \cdots \otimes |+\rangle \otimes |-\rangle \otimes |+\rangle \otimes \cdots \otimes |+\rangle \longrightarrow |0\rangle \otimes \cdots \otimes |0\rangle \otimes |1\rangle \otimes |0\rangle \otimes \cdots \otimes |0\rangle . \tag{13}$$

Since the σ_z bit flip impacts the global state of a qubit, it will be seen as a bit flip in the measured state of T_5 satisfying Axiom 2. The proof is concluded mentioning that this result coincides with observations drawn in [24], (Theorem 2, § 4). ■

The classical degrees of freedom for an oracle have been accounted for in tests T_1, T_2, T_5 and T_6 with an added gate complexity of only $4(k+1)$H+$4(k-1)$CN. The phase kickback features of the gates in the oracle are verified next in tests T_3 and T_4. Superposition input states that have retaliative phases to each other are difficult to control since they are often entangled and therefore not expressible in a product state description. Depending on the function realized in the oracle a different amount of entanglement will be added. Returning the system to a product state (removing this added entanglement) adds complication to tests that verify this property. The controllability of a circuit represents an ability to propagate a specific input vector through a network, such that it will map a state to a specific fault location. This represents an added challenge in the case of quantum circuits, since inputs will become entangled and in many cases specific inputs to a specific location may not be possible. However, after a discussion of the upper bounds of tests T_3 and T_4 in Sec. III-F more controllable test input vectors are proposed (Sec. III-G) replacing the added complexity of these tests with a linear increase in the number of experiments needed. Tests T_3 and T_4 introduce important concepts that will foster growth in this research area and allow one to gain a better understanding of the complexities of controllability and observability under the influence of quantum entanglement. A purpose of this paper is to introduce these concepts and connect ideas from quantum process validation and classical test theory.

Axioms (↓)	*Fault Types Tested* (↓) − *Tests* (→)	T_1	T_2	T_3	T_4	T_5	T_6	$T_1 \cup T_2$	$T_3 \cup T_4$	$T_5 \cup T_6$
Axiom 1	Any σ_x or σ_y bit flips occurring?	×	×					×		
Axiom 2	Any σ_z phase flips occurring?		○	○	○	×	×	○	○	×
Axiom 3	Is initialization into $\|0\rangle$ and $\|1\rangle$ O.K.?	○	○			○	○	×	×	
Axiom 4	With $\|-\rangle$ at target is phase kickback O.K.?			○	○				×	
Axiom 5	Any phase problems with $\|+\rangle$ at the target?					○	○			×
Axiom 6	Are the controls activated with $\|0\rangle$ and $\|1\rangle$?	○	○					×		
Axiom 7	Gate acts on basis $\|0\rangle$ and $\|1\rangle$ O.K.?	○	○					×		
Axiom 8	Is measurement in $\|0\rangle$ and $\|1\rangle$ O.K.?	○	○					×		

Fig. 10. Tests are depicted in columns $3-11$, fault types in column 2 and Axioms in column 1. A given test (column) with table entry × below it satisfies the Axiom listed in the row corresponding to that ×. Entries with ○ inside correspond to tests that cover some, but not all of the faults depicted in the corresponding row.

D. Test T_3: $|+\rangle^{\otimes k} \otimes |-\rangle$

The goal of test T_3 is to verify that phase traverses backwards correctly amongst all gates. For test T_3 the Hadamard gates at the left of Fig. 11 are used to prepare the following superposition state as the oracle's input on the top k bits:

$$\begin{aligned} &\Longrightarrow |0000\rangle + |0001\rangle + |0010\rangle + |0011\rangle + |0100\rangle + |0101\rangle \\ &+ |0110\rangle + |0111\rangle + |1000\rangle + |1001\rangle + |1010\rangle + |1011\rangle \\ &+ |1100\rangle + |1101\rangle + |1110\rangle + |1111\rangle \end{aligned} \tag{14}$$

Observe that Eqn. 15 is like a truth table where all the true minterms of the function have phase factors of -1, (see Fig. 3). This often results in phase induced entanglement as shown in Eqn. 15.

$$\begin{aligned} &\Longrightarrow |0000\rangle + |0001\rangle - |0010\rangle + |0011\rangle - |0100\rangle - |0101\rangle \\ &+ |0110\rangle + |0111\rangle - |1000\rangle - |1001\rangle + |1010\rangle + |1011\rangle \\ &+ |1100\rangle + |1101\rangle + |1110\rangle - |1111\rangle \end{aligned} \tag{15}$$

In general, a product (*local*) superposition state may be written as:

$$\pm \bigotimes_{i=0}^{k-1} (|0\rangle + a_i |1\rangle) \tag{16}$$

where any a_i term is either $+1$ or -1. For the state in Eqn. 15 to be expressible as a product state, Eqn. 17 must be satisfied:

$$(|0\rangle + a_0 |1\rangle)(|0\rangle + a_1 |1\rangle)(|0\rangle + a_2 |1\rangle)(|0\rangle + a_3 |1\rangle). \tag{17}$$

Given Eqn. 17, any one of 2^i $(0 \leq i < k)$ possible choices for a_i results in a local description of the quantum state (the implications of which will be discussed in Sec. III-F). The general expansion of Eqn. 17 leads directly to the generic state:

$$\begin{aligned} &\Longrightarrow |0000\rangle + a_3 |0001\rangle + a_2 |0010\rangle + a_1 |0100\rangle + a_0 |1000\rangle \\ &+ a_0 \cdot a_1 |1100\rangle + a_0 \cdot a_2 |1010\rangle + a_0 \cdot a_3 |1001\rangle \\ &+ a_1 \cdot a_2 |0110\rangle + a_1 \cdot a_3 |0101\rangle + a_2 \cdot a_3 |0011\rangle \\ &+ a_0 \cdot a_1 \cdot a_2 |1110\rangle + a_0 \cdot a_2 \cdot a_3 |1011\rangle \\ &+ a_0 \cdot a_1 \cdot a_3 |1101\rangle + a_1 \cdot a_2 \cdot a_3 |0111\rangle \\ &+ a_0 \cdot a_1 \cdot a_2 \cdot a_3 |1111\rangle \end{aligned} \tag{18}$$

Comparing Eqns. 15 and 18 for the considered circuit, the system of arithmetic equations given in Eqn. 19 is obtained. This system is clearly not specifying a product state since Eqns. 15 and 18 matched with Eqn. 19 are inconsistent. The interfering terms $a_0 \cdot a_1 \cdot a_2$ and $a_2 \cdot a_3$ could be changed for the system to return to a local, product state description. This may be done by inserting the *QBIST*$_{32}$ circuit given in Fig. 11. *QBIST*$_{32}$ inverts the phase on terms $|1110\rangle$ and $|0011\rangle$ to $+1$, making the state factorable as $(|0\rangle + |1\rangle)(|0\rangle + |1\rangle)(|0\rangle + |1\rangle)(|0\rangle - |1\rangle) \otimes |-\rangle$.

$$\left(\begin{array}{rr} a_0 = -1 & a_1 \cdot a_3 = +1 \\ a_1 = -1 & a_2 \cdot a_3 = +1 \\ a_2 = -1 & a_0 \cdot a_1 \cdot a_2 = +1 \\ a_3 = +1 & a_0 \cdot a_1 \cdot a_3 = -1 \\ a_0 \cdot a_1 = +1 & a_0 \cdot a_2 \cdot a_3 = +1 \\ a_0 \cdot a_2 = +1 & a_1 \cdot a_2 \cdot a_3 = +1 \\ a_0 \cdot a_3 = -1 & a_0 \cdot a_1 \cdot a_2 \cdot a_3 = -1 \\ a_1 \cdot a_2 = +1 & \forall i, a_i \in \{-1, +1\} \end{array} \right) \tag{19}$$

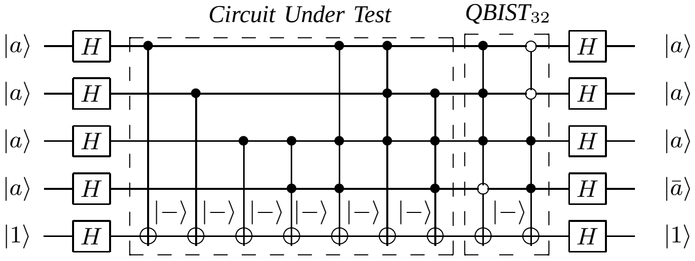

Fig. 11. Circuit Under Test T_3 and T_4: Test $T_3 \longrightarrow |+\rangle^{\otimes k} \otimes |-\rangle$ is first generated ($a = 0$, T_3) and next test $T_4 \longrightarrow |-\rangle^{\otimes k} \otimes |-\rangle$ is applied ($a = 1$, T_4). Nodes activated with $|0\rangle$ are denoted as (○). *QBIST*$_{32}$ removes entanglement returning the system to a product state and has the same form in both tests.

E. Test T_4: $|-\rangle^{\otimes k} \otimes |-\rangle$

Test T_4 is an exact dual to test T_3 and therefore, the needed *QBIST*$_{42}$ stage will have the exact same structure as the *QBIST*$_{32}$ already used. Now the register is initialized into state $|1111\rangle \otimes |1\rangle$ (by setting $a = 1$ in Fig. 11). The Hadamard operators map this initial state as follows:

$$\left(H^{\otimes(k+1)}\right) \cdot (|1111\rangle \otimes |1\rangle) \longrightarrow |----\rangle \otimes |-\rangle \tag{20}$$

and this acts as input to the oracle. The phase of each term is now opposite when compared with T_3. *QBIST*$_{42}$ inverts the phase on term $|1110\rangle$ and $|0011\rangle$ to -1, making the state factorable and resulting in this local state description $(|0\rangle - |1\rangle)(|0\rangle - |1\rangle)(|0\rangle - |1\rangle)(|0\rangle + |1\rangle) \otimes |-\rangle$.

Theorem 10 proves that test T_3 combined with test T_4 satisfy Axiom 4. Tests T_3 and T_4 have a worst case added gate complexity of at most $\Theta(N-k) + 4kH$, where Θ is a function of the number of controls needed in the disentanglement stage and the linearity of the oracle.

***Theorem** 10:* Together tests T_3 and T_4 satisfy Axiom 4.

Proof: In tests T_3 and T_4 the state of the target is $|-\rangle$. Any gate that was activated by a state with eigenvalues ± 1 during test T_3 is activated by a state with eigenvalues ∓ 1

in test T_4. Furthermore, both tests T_3 and T_4 contain non-activating terms, each with opposite eigenvalues. Tests T_3 and T_4 therefore satisfy Axiom 4. ■

To conclude the test algorithms presentation, the Table in Fig. 10 is now mentioned. This table provides a concise illustration of the sets of faults entirely covered by given test(s) (denoted by $\times$) as well as the sets of faults partially covered by a given test (denoted by $\circ$).

We have developed a quantum test algorithm for the quantum phase oracle. It has been shown that this test pattern satisfies all of the Axioms in Sec. II and the results coincide with Theorems 1 and 2. This test pattern therefore probes the logical function of each $k-$CN gate in the oracle. Now upper bounds on the extraction technique (*QBIST*$_{32}$ circuit stage) will be derived in Sec. III-F.

F. Upper Bounds for QBIST$_{32}$:

The concepts of the presented test algorithm are general and therefore work for any circuit. They do however require the successful design of the *QBIST*$_{32}$. This design varies between oracles and has an upper bound of added depth complexity that depends on the function realized in the oracle. The purpose of this section is to present an analysis of this test method designed for functional information extraction. It begins with the following definition.

***Definition** 7:* An affine Boolean function $A_f(x_1,...,x_k)$, on variables $x_1,...,x_k$ is any function the takes the form

$$A_f(x_1,x_2,...,x_k) = c_0 \oplus c_1 \cdot x_1 \oplus c_2 \cdot x_2 \oplus \cdots \oplus c_k \cdot x_k, \quad (21)$$

where $\cdot$ is Boolean AND, $\oplus$ is EXOR (modulo 2 addition), $c_i \in \{0,1\}$ and $i = 0,1,...,n$ are indices of coefficients. It is easy to see that there exist 2^{k+1} affine functions all of which have checkered cube patterns. A linear function is any one of the 2^k affine functions generated when coefficient $c_0 = 0$.

We present the following theorem (11) relating state separability to the function being realized by a given oracle. An observation made during this study is that oracles realizing affine functions produce no net entanglement on the top k qubits. However, an oracle search space realizing bent function produces maximal inseparability in state of the top k qubits when used as a search oracle.[7] Thus, an oracle realizing an affine function will correspond to, in the ideal case, a deterministic measurement result when interfered through H gates.

***Theorem** 11:* Consider oracle $\mathcal{O}$ for which test T_3 obtains only separable (local) measurements (requires no disentanglement). $\mathcal{O}$ necessarily realizes only affine functions over k variables.

Proof: The formal proof involves complex notation but is based on the straightforward generalization of the following example:

Assume input variables (x_1,x_2,x_3). The expression

$$\begin{aligned}&|000\rangle\,(+1) + |001\rangle\, a_2 + |010\rangle\, a_1 + |011\rangle\, a_1 \cdot a_2 + \\ &|100\rangle\, a_0 + |101\rangle\, a_0 \cdot a_2 + |110\rangle\, a_0 \cdot a_1 + \\ &|111\rangle\, a_0 \cdot a_1 \cdot a_2 \qquad (22)\end{aligned}$$

corresponds to a classical truth table with $\prod a_i$ expressions corresponding to sum-of-product canonical coefficients. Assuming the encoding

$$en(+1) = 0, \quad en(-1) = 1, \quad (23)$$

arithmetic expressions like $a_1 \cdot a_2$ are changed to Boolean values like $en(a_1) \oplus en(a_2)$. Normally one would consider the case that $b_0 = 0$ for linear functions. Because of global phase b_0 may take either binary value corresponding to all affine functions on k variables. It is well known from the canonical SOP to PPRM conversion method that PPRM = $b_0 \cdot 1 \oplus (b_0 \oplus b_1) \cdot x_3 \oplus (b_0 \oplus b_2) \cdot x_2 \oplus (b_0 \oplus b_1 \oplus b_2 \oplus b_3) \cdot x_2 \cdot x_3 \oplus (b_0 \oplus b_2 \oplus b_4 \oplus b_6) \cdot x_1 \cdot x_2 \oplus (b_0 \oplus b_4) \cdot x_1 \oplus (b_0 \oplus b_1 \oplus b_4 \oplus b_5) \cdot x_1 \cdot x_3 \oplus (b_0 \oplus b_1 \oplus b_2 \oplus b_3 \oplus b_4 \oplus b_5 \oplus b_6 \oplus b_7) \cdot x_1 \cdot x_2 \cdot x_3$, where b_i are coefficients of minterms, i.e. b_0 is a coefficient of $|000\rangle$, b_1 is a coefficient of $|001\rangle$, etc. The minterms of canonical SOP obtain thus the following encoding (symbol $\cdot$ is arithmetic multiplication)[8] $b_1 = en(a_2)$, $b_2 = en(a_1)$, $b_3 = en(a_1 \cdot a_2) = en(a_1) \oplus en(a_2) = b_2 \oplus b_1$, $b_4 = en(a_0)$, $b_5 = en(a_0 \cdot a_2) = en(a_0) \oplus en(a_2) = b_4 \oplus b_1$, $b_7 = en(a_0 \cdot a_1 \cdot a_2) = en(a_0) \oplus en(a_1) \oplus en(a_2) = b_4 \oplus b_2 \oplus b_1$.

Applying now the encoding from Eqn. 23 and substituting into the above PPRM one obtains PPRM = $b_0 \cdot 1 \oplus (b_0 \oplus b_1) \cdot x_3 \oplus (b_0 \oplus b_2) \cdot x_2 \oplus [(b_0 \oplus b_1 \oplus b_2) \oplus (b_2 \oplus b_1)] x_2 \cdot x_3 \oplus [(b_0 \oplus b_2 \oplus b_4) \oplus (b_4 \oplus b_2)] x_1 \cdot x_2 \oplus (b_0 \oplus b_4) x_1 \oplus [(b_0 \oplus b_1 \oplus b_4) \oplus (b_4 \oplus b_1)] x_1 \cdot x_2 \oplus [(b_0 \oplus b_1 \oplus b_2) \oplus (b_2 \oplus b_1) \oplus (b_4) \oplus (b_4 \oplus b_1) \oplus (b_4 \oplus b_2) \oplus (b_4 \oplus b_2 \oplus b_1)] \cdot x_1 \cdot x_2 \cdot x_3 = b_0 \cdot 1 \oplus (b_0 \oplus b_1) x_3 \oplus (b_0 \oplus b_2) x_2 \oplus (b_0 \oplus b_4) \cdot x_1$. Thus, PPRM = $b_0 \oplus (b_0 \oplus b_1) x_3 \oplus (b_0 \oplus b_2) x_2 \oplus (b_0 \oplus b_4) x_1$ which corresponds to all affine functions on variables x_1, x_2, x_3. ■

If oracle $\mathcal{O}$ contains function $f(x_1,...,x_k)$ that is not affine, a modification to any one of the affine functions $A_i(x_1,...,x_k)$ must be made. This can be done by adding a circuit (such as *QBIST*$_{32}(x_1,...,x_k)$) and can be thought of as EXORing it with some function, like this:

$$f(x_1,...,x_k) \oplus BIST_i(x_1,...,x_k) = A_i(x_1,...,x_k). \quad (24)$$

Thus, $f(x_1,...,x_k) = BIST_i(x_1,...,x_k) \oplus A_i(x_1,...,x_k)$. The general disentanglement procedure is as follows:

1) Each function $A_i(x_1,...,x_k) \oplus BIST_i(x_1,...,x_k)$ is realized as an ESOP.
2) $BIST_i(x_1,...,x_k)$ with the minimum cost is selected.
3) Function $BIST_i(x_1,...,x_k)$ is added (XORed) after f as *QBIST*$_{32}$.

***Theorem** 12:* The minimum number of product terms in the ESOP realization of the BIST circuit ESOP[*BIST*$(x_1,...,x_k) \oplus A_i(x_1,...,x_k)$] where A_i is an arbitrary affine function on variables $x_1,...,x_k$ is equal to $p-k$ where p is the minimal number of product terms in ESOP(*BIST*$(x_1,...,x_k)$).

Proof: Given is the minimal ESOP, denoted by ESOP(BIST), of function $BIST(x_1,...,x_k)$. Let A be an arbitrary affine function on variables $x_1,x_2,...,x_k$ and $c_0 \oplus c_1 \cdot x_1 \oplus ...c_k \cdot x_k$, where $c_i \in \{0,1\}$. There are two of these functions that have the maximum number of variables equaling k; $x_1 \oplus x_2 \oplus ...x_k$ and $1 \oplus x_1 \oplus x_2 \oplus \cdots \oplus x_k = \bar{x_1} \oplus x_2 \oplus ...x_k$. Assuming that ESOP(BIST) has the minimal number of product terms, the following cube pair types must not be included in it: $x_i \cdot x_j \oplus x_i$, $x_i \cdot x_j \oplus x_i \cdot \bar{x_j}$, $x_i \cdot \bar{x_j} \oplus \bar{x_i} \cdot x_j$, $x_i \cdot x_j \oplus \bar{x_i} \cdot \bar{x_j}$. The only product terms possible in ESOP(BIST) are necessarily x_i, $\bar{x_i}$, $x_i \cdot x_j$, $x_i \cdot \bar{x_j}$, $x_i \oplus x_i \cdot x_j \cdot ... \cdot x_k$. If one writes ESOP(BIST$\oplus A_i$) as ESOP$(BIST) \oplus x_1 \oplus x_2 \oplus \ldots x_k$ provided all the best merging cases, then all variables (literals) from A are merged, each of them with some literal from ESOP(BIST), like this: $x_i \oplus x_i = 0$, $\bar{x_i} \oplus \bar{x_i} = 0$ and $x_i \oplus \bar{x_i}$. Each of these cases will decrease the ESOP cost by one.

[7] It is interesting to note that a quantum computer can distinguish all affine oracles with a single query; an exponential speed up over the classical case, with no known use yet, other than testing linear systems.

[8] This is also called the polarity table in which one considers a Boolean function over variables $\{-1,1\}$ instead of $\{0,1\}$. In this case, XOR ($\oplus$) over $\{0,1\}$ is equivalent to real multiplication over $\{-1,1\}$.

Merging x_i with $x_i \cdot x_j = x_i$; $x_i \oplus x_i \cdot x_j = x_i \cdot \bar{x_j}$ will not change the ESOP cost. All other mergings will increase the cost of the ESOP(*BIST*$\oplus A_i$) with respect to ESOP(BIST). Thus, the number of terms in the ESOP can be decreased by no more than k. Observe also that the highest decrease of cost is when BIST is already an affine function. ■

As proven by Gaidukov [34], the worst case complexity of an ESOP expression for k variables is $29 \cdot 2^{k-7}$ product terms for $k > 6$. It was however shown by exhaustive search [36] that for functions of four variables only 24 reach the maximum bound of 6 (counting constant 1 as a term) and only 3888 functions have 5 terms. There exist several ESOP minimizers that can efficiently realize large functions and give exponentially good results for arithmetic functions. In terms of exact solutions, the best ESOP minimizer can be found in [35].

Many useful functions (such as adders) are nearly linear. The method of extraction introduced for tests T_3 and T_4 may be considered as a discussion of controllability of quantum systems when the concept of test is an issue. A maximally non-linear Boolean function is known as a bent function, where the measurement of nonlinearity depends on Hamming distance. Bent functions have several applications in cryptography. Their use in ciphers and a discussion of the difficulties of finding a bent function can be found in [37] and the references therein.

G. Possible Extensions and Applications

An observation found in this study is that in some switching circuits, both phase terms and product terms may need to be changed in the final *QBIST*$_{\times 2}$ stage to make the states description local and extract information. Nonetheless, the principle of our algorithm is general and with an adjustment has application to arbitrary structures of $n \times n$ quantum mechanical switching networks (as opposed to single output quantum-realized functions). Respective test patterns for $n \times n$ networks should now also be developed. For example, the methods of making two non-adaptive oracle calls presented in [38] are easily adapted to reduce the number of classical tests atleast twice.

An alternative approach based on the theory outlined in tests T_3 and T_4 utilizes highly controllable test vectors. The growth in additional circuitry is thus replaced with linear growth in the number of experiments needed. The total cardinality in the number of experiments in this second method is $(5+4\lceil k/2\rceil)$. There is little added growth in circuit complexity. Tests T_3 and T_4 are replaced with first repeating the circuitry needed in test T_1. (All replaced tests of course have state $|-\rangle$ at the target.) Next, starting with the top 2 qubits (Fig. 12), an EPR pair is generated to test the oracle and mirrored with a measurement in the Bell basis. This is then moved down all the top k qubits (Fig. 13) a total of $2\lceil k/2\rceil$ times. The EPR generating circuitry is used to create inputs that are products of state $|01\rangle \pm |10\rangle$ and $|1\rangle$. These must be repeated with both positive and negative versions to satisfy Axiom 4. This results in something in classical test known as *walking-a-zero* [26] (except quantum mechanics allows two zeros to be walked at the same time). This alternative approach however, does not probe the oracle under the types of inputs experienced when used in a Grover search algorithm. It does however illustrate that the algorithm can be modified to reduce the complexity of the stages needed to extract information. Alternative applications of the methods presented in this paper also exist.

Disputing the implications of this testing method, we envision that after quantum systems become more controllable, methods such as ours will find more frequent use. The first idea will be to use the proposed test set with one key exception. Instead of computational basis measurements, perform tomographic state reconstruction and apply the distance measures given in Ref. [11]. In the future these distance measures (on a tailored test set) will be replaced with computational basis measurements—as proposed in this study. It is interesting that our method provides a non-classical speed increase when an ensemble quantum computer is used (NMR for instance or a machine capable of generating observable averages as output). Classically, the lower bound of this circuit class was found to be $(k + 4 + 2n_e)$ by Reddy [2] (where the $2n_e$ term depends on the function being realized). However, the classically impossible speed increase of this method was only an interesting counterpart of our main goal. Our goal was predominantly to generalize the classical test theory and combine this theory and methodologies with quantum process validation.

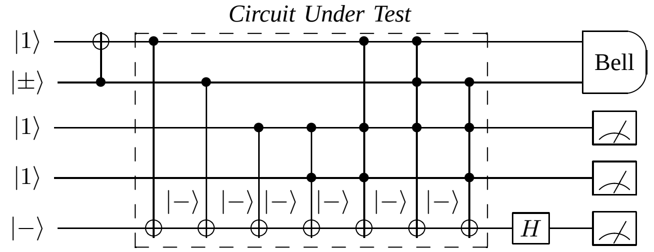

Fig. 12. Alternative setup for tests T_3 and T_4: Test $|0111\rangle \pm |1011\rangle$. The target of each k-CN gate acts on state $|-\rangle$. No entanglement is added in either test, since all relative phases will result in a product measurement in the Bell basis.

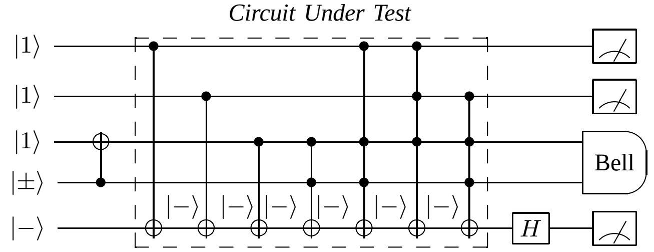

Fig. 13. Alternative setup for tests T_3 and T_4: Test $|1101\rangle \pm |1110\rangle$.

IV. Conclusion

This work reduced the classical test problem by utilizing entanglement as a controllability resource in Sec. III. Quantum effects were used to test multiple classical degrees of freedom concurrently, hence the latter is used to verify the former and quantum process validation was reduced to a linear growth of $(5 + 4\lceil k/2\rceil)$ in experiment count. When testing an oracle, states become non-local due to the phase change undergone by all true minterms as seen in tests T_3 and T_4 in Sec III-D and III-E. It was shown in Sec. III-F that all affine oracles generate no net entanglement when used as a search oracle, while an oracle realizing a bent function requires the greatest effort to disentangle the state and return the system to a local product state. Since there are 2^{k+1} affine functions, Sec. III-F addressed the question of how close an arbitrary state is to a factorable state with phase terms that represent the spectrum of an affine function. The distance in many cases is close, but the upper bound is $\sim \Theta(N - k)$. Linear and Affine functions are very easy to test when realized quantum mechanically. Based on the potential limitations highly controllable test vectors were developed in Sec. III-G that do not undergo phase induced entanglement when propagation though a phase oracle occurs. This test set is an application of our main approach.

In a correspondence from Agrawal in 1981 [39], fault detection probability was shown to be the highest when the information output of a circuit is maximized. However, the information content into a classical circuit is fixed. An interesting result found in our study is that when a quantum information source is used to increase information input, the probability of detecting a fault is also increased. An information theoretic approach to quantum fault testing might lead to further useful insight into the quantum test problem.

The classical test problem is typically defined to be in the class **NP**. Other circuit structures will be shown to be exponentially easier to test using our methods. The high testability of a quantum information processing device, may well prove in fact to be yet another supporting argument to study quantum information theory. Of course, the overwhelming failure rate experienced with constructing quantum circuits at the time of this writing causes us to call the results of this paper somewhat ironic. ■

ACKNOWLEDGMENTS

The authors would like to thank P.J. Love, J.A. Jones, M. Jeong and J.S. Allen. The Quantum Circuit diagrams were drawn in LaTeX using Q-circuit [40]. Portland State University provided student funding, resources, space and support for this project. Support was also given by the Ronald E. McNair Post baccalaureate Achievement Program, and the Korean Advanced Institute of Science and Technology.

REFERENCES

[1] W. Kautz, *Testing faults in combinational cellular logic arrays*, Proceedings of 8th annu. Symp. Switching and Automata Theory, Oct. 1971, pp. 161-174.

[2] S. Reddy, ”*Easily Testable Realizations for Logic Functions,*” IEEE Transactions on Computers, Vol. C-21, No. 11, pages 1183 - 1188, November (1972).

[3] D. Deutsch, ”*Quantum theory, the Church-Turing Principle and the universal quantum computer,*” Proc. R. Soc. Lond. A, 400:96, (1985).

[4] A.C. Yao, ”*Quantum circuit complexity,*” Proc. of the 34^{th} Ann. IEEE Symp. on Fondations of Computer Science, pages 352-361, (1993).

[5] D. Aharonov, Wim van Dam, J. Kempe, Z. Landau, S. Lloyd and O. Regev, ”*Adiabatic Quantum Computation is Equivalent to Standard Quantum*, 30 pages, (2005), quant-ph/0405098.[9]

[6] M.A. Nielsen, ”*Cluster-state quantum computation,*” 15 pages, in Press. Rev.Math.Phys, (2005), quant-ph/0504097.

[7] P.J. Love and B. Boghosian, ”*Type-II Quantum Algorithms,*” To appear in Physica A, (2005), quant-ph/0506244.

[8] D. Deutsch, ”*Quantum computational networks,*” Proc. R. Soc. London A, 425:73, (1989).

[9] A. Barenco, C. Bennett, R. Cleve, D. DiVincenzo, N. Margolus, P. Shor. T. Sleator, J. Smolin, and H. Weinfurter, ”*Elementary gates of quantum computation,*” Phys.Rev.A, 52(5):3457-3467, (1995), quant-ph/9503016.

[10] J.A. Jones and M. Mosca, ”*Implementation of a Quantum Algorithm to Solve Deutsch's Problem on a Nuclear Magnetic Resonance Quantum Computer,*” Journal of Chemical Physics, Vol. 109, August 1^{st} 1998, pp. 1648-1653, quant-ph/9801027.

[11] A. Gilchrist, N. Langford and M. Nielsen, ”*Distance measures to compare real and ideal quantum processes,*” Phys. Rev. A 71, 062310 (2005), quant-ph/0408063.

[12] M. Bowdrey and J.A. Jones, ”*A Simple and Convenient Measure of NMR Rotor Fidelity,*” JAJ-QP-01-01, (2001), quant-ph/0103060.

[13] A. Childs, I. Chuang and D. Leung, ”*Realization of quantum process tomography in NMR,*” Phys. Rev. A 64, 012314 (2001), quant-ph/0012032.

[14] I. Chuang and M. Nielsen, ”*Prescription for experimental determination of the dynamics of a quantum black box,*” J. Mod. Opt. 44, 2455 (1997),quant-ph/9610001.

[15] R. Feynman, ”*Simulating physics with computers,*” Internat. J. Theoret. Phys., 21, pp. 467488 (1982).

[16] D. Abrams and S. Lloyd ”*A quantum algorithm providing exponential speed increase for finding eigenvalues and eigenvectors,*” Phys.Rev.Lett. 83, pages 5162-5165, (1999), quant-ph/9807070.

[17] G.M. D'Ariano and P. Lo Presti, ”*Measuring Experimentally the Matrix Elements of an Arbitrary Quantum Operation,*” Phys. Rev. Lett. 86, 41954198 (2001), DOI: 10.1103/PhysRevLett.86.4195.

[18] J.B. Altepeter, D. Branning, E. Jeffrey, T.C. Wei, P.G. Kwiat, R.T. Thew, J.L. O'Brien, M.A. Nielsen and A.G. White, ”*Ancilla-assisted quantum process tomography,*” Phys. Rev. Lett. 90, 193601 (2003), quant-ph/0303038.

[19] P.W. Shor, ”*Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer,*” SIAM J. Sci. Statist. Comput. 26, 1484, 28 pages (1997), quant-ph/9508027.

[20] L. Grover, ”*Quantum mechanics helps in searching for a needle in a haystack,*” Physical Review Letters, 79:325, (1997), quant-ph/9706033.

[21] G. Brassard, P. Hoyer and A. Tapp, ”*Quantum Counting,*” Proc. of ICALP (1998); see also quant-ph/9805082.

[22] D. Simon, ”*On the power of quantum computation,*” Proc. 35^{th} Annual Symposium on Foundations of Computer Science, IEEE Computer Society Press, Los Alamitos, CA, pp. 116 123, (1994); SIAM Journal on Computing Volume 26 , Issue 5, Pages: 1474 - 1483 (1997), DOI. 10.1137/S0097539796298637.

[23] M.A. Nielsen and I.L. Chuang, ”*Quantum Computation and Quantum Information,*” Cambridge Univ. Press, (2000).

[24] J.D. Biamonte, J.S. Allen and M.A. Perkowski, ”*Fault Models for Quantum Mechanical Switching Networks,*” 22 pages, (2005), quant-ph/0508147.

[25] J. Biamonte and M. Perkowski, *Testing a Quantum Computer*, Proceedings of, KIAS-KAIST Workshop on Quantum Information Science, Seoul Korea, August 29^{th}-31^{st}, 16 pages, (2004).

[26] U. Kalay, M. Perkowski and D. Hall ”*A Minimal Universal Test Set for Self-Test of EXOR-Sum-of-Product Circuits,*” IEEE Transactions on Computers, vol. 49, no. 3, pp. 267-276, (2000).

[27] M. Mosca, ”*Quantum Computer Algorithms,*” PhD thesis, University of Oxford, (1999), http://www.cacr.math.uwaterloo.ca/.

[28] E. Knill, R. Laflamme and W.H. Zurek, ”*Resilient Quantum Computation: Error Models and Thresholds,*”, Proc. Mathematical, Physical Engineering Sciences Vol. 454, 1997, pp. 365-384 quant-ph/9702058.

[29] M. Nielsen, ”*Quantum information theory,*” PhD thesis, University of New Mexico, 259 pages, Report UNM-98-08 (1998), quant-ph/0011036.

[30] H.K. Cummins, H. Llewellyn and J.A. Jones, ”*Tackling Systematic Errors in Quantum Logic Gates with Composite Rotations,*” Phys. Rev. A 67, 042308 (2003), quant-ph/0208092.

[31] W. Zurek, ”*Reversibility and Stability of Information Processing Systems,*” Phys. Rev. Lett. 53, pages 391-394, (1984), DOI: 10.1103/PhysRevLett.53.391.

[32] E. McCluskey and C.W. Tseng, ”*Stuck-fault tests vs. actual defects,*” in Proc. of Int. Test Conf., pp. 336 343, (2000).

[33] A.W. Harrow and M.A. Nielsen, ”*How robust is a quantum gate in the presence of noise?,*” Phys. Rev. A Vol. 68, 012308, 2003, 14 pages, quant-ph/0301108.

[34] A. Gaidukov, ”*An Algorithm to derive the minimum ESOP for 6-variable functions,*” Proc. 5^{th} International Workshop on Boolean Problems, Freiberg, Germany, (2002).

[35] S. Stergiou and G.K. Papakonstantinou, ”*Exact Minimum of ESOP Expressions with less than Eight Product Terms,*” J.Cir.Sys.Comp., Vol 13, No.1, pages 1-15, (2004).

[36] T. Sasao, ”*Easily Testable Realizations for Generalized Reed-Muller Expressions,*” IEEE Trans. Computers, 46(6), pp. 709-716, (1997).

[37] A. Dimovski and D. Gligoroski, ”*Generating highly nonlinear Boolean functions using a genetic algorithm,*” Proc. Bal. Con. on Informatics, Greece, (2003).

[38] Wim van Dam, ”*Two Classical Queries versus One Quantum Query,*” Report-no: CQC-2CQ:1QQ, 6 pages, (1998), quant-ph/9806090.

[39] V. Agrawal, ”*An Information Theoretic Approach to Digital Fault Testing,*” IEEE Transactions on Computers, vol. 30, pages 582 - 587, August (1981).

[40] B. Eastin and S.T. Flammia, ”*Q-circuit Tutorial,*” 7 pages, (2004), free online, quant-ph/0406003.

[9] Citations with a 'quant-ph/xxxxxxx' designation are on the internet at http://arXir.org/.