

A SCHEMATIC DEFINITION OF QUANTUM POLYNOMIAL TIME COMPUTABILITY

TOMOYUKI YAMAKAMI

Abstract. In the past four decades, the notion of quantum polynomial-time computability has been mathematically modeled by quantum Turing machines as well as quantum circuits. This paper seeks the third model, which is a quantum analogue of the schematic (inductive or constructive) definition of (primitive) recursive functions. For quantum functions mapping finite-dimensional Hilbert spaces to themselves, we present such a schematic definition, composed of a small set of initial quantum functions and a few construction rules that dictate how to build a new quantum function from the existing ones. We prove that our schematic definition precisely characterizes all functions that can be computable with high success probabilities on well-formed quantum Turing machines in polynomial time, or equivalently uniform families of polynomial-size quantum circuits. Our new, schematic definition is quite simple and intuitive and, more importantly, it avoids the cumbersome introduction of the well-formedness condition imposed on a quantum Turing machine model as well as of the uniformity condition necessary for a quantum circuit model. Our new approach can further open a door to the descriptive complexity of quantum functions, to the theory of higher-type quantum functionals, to the development of new first-order theories for quantum computing, and to the designing of programming languages for real-life quantum computer

§1. Background, motivation, and the main results. In early 1980s emerged a groundbreaking idea of exploiting quantum physics to build mechanical computing devices, dubbed as *quantum computers*, which have completely altered the way we used to envision “computers.” Subsequent discoveries of more efficient quantum computations for factoring positive integers [29] and searching unstructured databases [13, 14] than classical computations prompted us to look for more mathematical and practical problems that can be solvable effectively on the quantum computers. Efficiency in quantum computing has since then rapidly become an important research subject of computer science as well as physics.

As a mathematical model to realize quantum computation, Deutsch [10] introduced a notion of *quantum Turing machine* (or QTM, for short), which was later discussed by Yao [39] and further refined by Bernstein and Vazirani [4]. This mechanical model largely expands the classical model of (probabilistic) Turing machine by allowing a physical phenomenon, called *quantum interference*, to take place on its computation. A different Hamiltonian formalism of Turing machine was also suggested by Benioff [3]. A QTM has an ability of computing a *quantum function* mapping a finite-dimensional Hilbert space to itself by evolving

Received December 1, 2019.

2020 *Mathematical Subject Classification.* 81P68, 68Q12, 68Q10, 03D75.

Key words and phrases. quantum computing, quantum function, quantum Turing machine, quantum circuit, schematic definition, descriptive complexity, polynomial-time computability, normal form theorem.

© 2020, Association for Symbolic Logic
0022-4812/20/8504-0008
DOI:10.1017/jsl.2020.45

unitarily a superposition of (classical) configurations of the machine, starting with a given input string and an initial inner state. A more restrictive use of the term of “quantum function” is found in, e.g., [37], in which quantum functions take classical input strings and produce either classical output strings of QTMs or acceptance probabilities of QTMs. Throughout this paper, nevertheless, quantum functions refer only to functions acting on Hilbert spaces of arbitrary dimensions.

To ensure the unitary nature of quantum computation, a QTM requires its mechanism to meet the so-called *well-formedness condition* on a single-tape model of QTM [4] and a multitape model [35, 37] (as well as [26]). Refer to Section 2.2 for their precise definitions.

Bernstein and Vazirani further formulated a new complexity class, denoted by BQP, as the collection of all languages recognized by well-formed QTMs running in polynomial time with error probability bounded from above by $1/3$. Furthermore, QTMs equipped with output tapes can compute string-valued functions in place of languages, and those functions form a function class, called FBQP.

From a different viewpoint, Yao [39] expanded Deutsch’s notion of *quantum network* [11] and formalized a notion of *quantum circuit*, which is a quantum analogue of classical Boolean circuit. Different from the classical Boolean circuit model, a quantum circuit is composed of *quantum gates*, each of which represents a unitary transformation acting on a Hilbert space of a small, fixed dimension. To act as a “programmable” unitary operator, a family of quantum circuits requires the so-called *uniformity condition*, which ensures that a blueprint of each quantum circuit is easily rendered. Yao further demonstrated that a uniform family of quantum circuits is powerful enough to simulate a well-formed quantum Turing machine. As Nishimura and Ozawa [25] pointed out, the uniformity condition of a quantum circuit family is necessary to precisely capture quantum polynomial-time computation. With this uniformity condition, BQP and FBQP are characterized exactly by uniform families of quantum circuits made up of polynomially many quantum gates.

This current paper boldly takes the third approach toward the characterization of quantum polynomial-time computability. Unlike the aforementioned mechanical device models, our approach is to extend the schematic (inductive or constructive) definition of (*primitive*) *recursive functions* on natural numbers. Such a schematic definition was thought in the 19th century by Peano [27], opposed to the definition given by Turing’s machine model [31]. This classical scheme comprises a small set of initial functions and a small set of rules, which dictate how to construct a new function from the existing ones. For instance, every primitive recursive function is built from the constant, successor, and projection functions by applying composition and primitive recursion finitely many times. In particular, the primitive recursion introduces a new function whose values are defined *by induction*. Recursive functions (in the form of μ -*recursive functions* [18, 19]) further require an additional scheme, known as the minimization (or the least number) operator. These functions coincide with the Herbrand-Gödel formalism of *general recursive functions* (see [9]). For a historical account of these notions, refer to, e.g., [28]. Similar schematic approaches to capture classical polynomial-time computability have already been sought in the literature [6–8, 23, 33, 34]. Those approaches have led to quite different research subjects from what the Turing machine model provides.

Our purpose in this paper is to give a schematic definition of quantum functions to capture the notion of quantum polynomial-time computability and, more importantly, to make such a definition simpler and more intuitive for a practical merit of our own. Our schematic definition (Definition 3.1) includes a set of initial quantum functions, I (identity), NOT (negation of a qubit), $PHASE_\theta$ (phase shift by $e^{i\theta}$), ROT_θ (rotation around xy -axis by angle θ), $SWAP$ (swap between two qubits), and $MEAS$ (partial projective measurement), as well as construction rules, composed of composition ($Compo[\cdot, \cdot]$), branching ($Branch[\cdot, \cdot]$), and multiqubit quantum recursion ($kQRec[\cdot, \cdot]$). Our choice of these initial quantum functions and construction rules stems mostly from a universal set of quantum gates in use in the past literature. Our quantum recursion, on the contrary, is quite different in nature from the primitive recursion used to build primitive recursive functions. Instead of using the successor function to count down the number of inductive iterations in the primitive recursion, the quantum recursion uses a divide-and-conquer strategy of reducing the number of accessible qubits needed for performing a specified quantum function. Within our new framework, we can implement typical unitary operators, such as the Walsh-Hadamard transform (WH), the controlled-NOT (CNOT), and the global phase shift (GPS).

An immediate merit of our schematic definition is that we can avoid the cumbersome introduction of the well-formedness condition imposed on the QTM model and the uniformity condition on the quantum circuit model. Another advantage of our schemata is that each scheme has its own inverse; namely, for any quantum function g defined by one of the schemata, its inverse g^{-1} is also defined by the same kind of scheme. For instance, the inverses of the quantum functions ROT_θ and $kQRec_t[g, h, p | \{f_s\}_{s \in \{0,1\}^k}]$ introduced in Definition 3.1 are exactly $ROT_{-\theta}$ and $kQRec_t[g^{-1}, p^{-1}, h^{-1} | \{f_s^{-1}\}_{s \in \{0,1\}^k}]$, respectively (Proposition 3.5).

For a further explanation of our main contributions, it is time to introduce a succinct notation of \square_1^{QP} (where \square is pronounced “square”) to denote the set of all quantum functions built from the initial quantum functions and by a finite series of sequential applications of the construction rules. Since the partial measurement ($MEAS$) is not a unitary operator, we denote the class obtained from \square_1^{QP} without use of $MEAS$ by $\widehat{\square}_1^{QP}$. Briefly, let us discuss clear differences between our schematic definition and the aforementioned two formalisms of polynomial-time quantumly computable functions in terms of QTMs and quantum circuits. Two major differences are listed below.

1) While a single quantum circuit takes a fixed number of input qubits, our quantum function takes an “arbitrary” number of qubits as an input. This situation is similar to QTMs because a QTM has an infinite tape and uses an arbitrary number of tape cells during its computation as extra storage space. On the contrary to the QTMs, a $\widehat{\square}_1^{QP}$ -function is constructed using the same number of qubits as its original input in such a way that a quantum circuit has the same number of input qubits and output qubits.

2) The two machine models exhort an algorithmic description to dictate the behavior of each machine; more specifically, a QTM uses a transition function, which algorithmically describes how each step of the machine acts on certain qubits, and a family of quantum circuits uses its uniformity condition to render the design of

quantum gates in each quantum circuit. Unlike these two models, no \square_1^{QP} -function has any mechanism to store information on the description of the function itself but the construction process itself specifies the behavior of the function.

As a consequence, the above mentioned differences help the \square_1^{QP} -functions take a distinctive position among all possible computational models that characterize quantum polynomial-time computability, and therefore we expect them to play an important role in analyzing the features of quantum polynomial-time computation from a quite different perspective.

In Section 3.1, we will formally present our schematic definition of \square_1^{QP} -functions (as well as $\widehat{\square_1^{\text{QP}}}$ -functions) and show in Section 4.1 that \square_1^{QP} (also $\widehat{\square_1^{\text{QP}}}$) can characterize all functions in FBQP. More precisely, we assert in the main theorem (Theorem 4.1) that any function from $\{0,1\}^*$ to $\{0,1\}^*$ in FBQP can be characterized by a certain polynomial p and a certain quantum function $g \in \square_1^{\text{QP}}$ in such a way that, by using an appropriate coding scheme, in the final quantum state of g on instances x and the runtime bound $p(|x|)$, we observe an output value $f(x)$ with high probability. This theorem will be split into two lemmas, Lemmas 4.2 and 4.3. The former lemma will be proven in Section 4.1; however, the proof of the latter lemma is so lengthy that it will be postponed until Section 5. In this proof, we will construct a \square_1^{QP} -function that can simulate the behavior of a given QTM.

Notice that, since BQP is a special case of FBQP, BQP is also characterized by our model. In our proof of the characterization theorem (Theorem 4.1), we will utilize a main result of Bernstein and Vazirani [4] and that of Yao [39] extensively. In Section 4.2, we will apply our characterization, in the help of a universal QTM [4, 25], to obtain a quantum version of Kleene's *normal form theorem* [18, 19], in which there is a universal pair of primitive recursive predicate and function that can describe the behavior of every recursive function.

Unlike classical computation on natural numbers (equivalently, *strings* over finite alphabets by appropriate coding schemata), quantum computation refers to a series of certain manipulations of a single vector in a finite-dimensional Hilbert space and we need only high precision to approximate each function in FBQP by such a vector. This fact allows us to choose a different set of schemata (initial quantum functions and construction rules) to capture the essence of quantum computation. In Section 6.1, we will discuss this issue using an example of a general form of the *quantum Fourier transform* (QFT). This transform may not be "exactly" computed in our current framework of \square_1^{QP} but we can easily expand \square_1^{QP} to compute the generalized QFT exactly if we include an additional initial quantum function, such as *CROT* (controlled rotation).

Concerning future research on the current subject, we will discuss in Section 6 four new directions of the subject. Our schematic definition provides not only a different way of describing languages and functions computable quantumly in polynomial time but also a simple way of measuring the "descriptive" complexity of a given language and a function restricted to instances of specified length. This new complexity measure will be useful to prove basic properties of $\widehat{\square_1^{\text{QP}}}$ -functions in Section 3. Its future application will be briefly discussed in Section 6.2.

Kleene [20, 21] defined recursive functionals of higher types by extending the aforementioned recursive functions on natural numbers. A more general study of *higher-type functionals* has been conducted in computational complexity theory for decades [7, 8, 16, 23, 30, 33, 34]. In a similar spirit, our schematic definition enables us to study *higher-type quantum functionals*. In Section 6.3, using *oracle functions* (*function oracles* or *oracles*), we will define type-2 quantum functionals, which may guide us to a rich field of research in the future.

A schematic definition of how to construct a target \square_1^{QP} -function can be viewed as a “program” that describes a series of instructions on which schemata to use. Hence, our schematic formulation opens a door to a new, practical application to the designing of succinct programming languages to control the operations of real-life quantum computers. In Section 6.4, we will briefly argue on an application of the schematic definition to the future development of “quantum programming languages.” As a further application of our schematic definition, we can look into a new aspect of first-order theories and their subtheories. Earlier, a quantum analogue of NP (nondeterministic polynomial time class) and beyond were sought in [36] with the use of bounded quantifiers over quantum states in finite-dimensional Hilbert spaces. In a similar vein, we expect that \square_1^{QP} will serve as a foundation to the introduction of first-order theories and their subtheories over quantum states in Hilbert spaces.

§2. Fundamental notions and notation. We begin with explaining basic notions and notation necessary to read through the subsequent sections. Let us assume the reader’s familiarity with classical Turing machines (see, e.g., [15]). For the foundation of quantum information and computation, in contrast, the reader refers to basic textbooks, e.g., [17, 24].

2.1. Numbers, languages, and qustrings. The notation \mathbb{Z} indicates the set of all *integers* and \mathbb{N} expresses the set of all *natural numbers* (that is, non-negative integers). For convenience, we set $\mathbb{N}^+ = \mathbb{N} - \{0\}$. Moreover, \mathbb{Q} denotes the set of all *rational numbers* and \mathbb{R} indicates the set of all *real numbers*. For two numbers $m, n \in \mathbb{Z}$ with $m \leq n$, the notation $[m, n]_{\mathbb{Z}}$ denotes an *integer interval* $\{m, m+1, m+2, \dots, n\}$, compared to a real interval $[\alpha, \beta]$ for $\alpha, \beta \in \mathbb{R}$ with $\alpha \leq \beta$. In particular, $[n]$ is shorthand for $[1, n]_{\mathbb{Z}}$ for any $n \in \mathbb{N}^+$. By \mathbb{C} , we express the set of all *complex numbers*. Given $\alpha \in \mathbb{C}$, α^* expresses the *complex conjugate* of α . *Polynomials* are assumed to have natural numbers as their coefficients and they thus produce nonnegative values from nonnegative inputs. A real number α is called *polynomial-time approximable*¹ if there exists a multitape polynomial-time deterministic Turing machine M (equipped with a write-only output tape) that, on each input of the form 1^n for a natural number n , produces a finite binary fraction, $M(1^n)$, on its designated output tape with $|M(1^n) - \alpha| \leq 2^{-n}$. Let $\tilde{\mathbb{C}}$ be the set of complex numbers whose real and imaginary parts are both polynomial-time approximable. For a bit $a \in \{0, 1\}$, \bar{a} indicates $1 - a$. Given a matrix A , A^T denotes its *transpose* and A^\dagger denotes the *transposed conjugate* of A .

¹Ko and Friedman [22] first introduced this notion under the name of “polynomial-time computable.” To avoid reader’s confusion in this paper, we prefer to use the term “polynomial-time approximation.”

An *alphabet* is a finite nonempty set of “symbols” or “letters.” Given such an alphabet Σ , a *string* over Σ is a finite series of symbols taken from Σ . The *concatenation* of two strings u and w is expressed as $u \cdot w$ or more simply uw . The *length* of a string x , denoted by $|x|$, is the number of all occurrences of symbols in x . In particular, the *empty string* has length 0 and is denoted λ . We write Σ^n (resp., $\Sigma^{\leq n}$) for the subset of Σ^* consisting only of all strings of length n (resp., length at most n) and we set $\Sigma^* = \bigcup_{n \in \mathbb{N}} \Sigma^n$ (the set of all strings over Σ). A *language* over Σ is a subset of Σ^* . Given a language S , its *characteristic function* is also expressed by S ; that is, $S(x) = 1$ for all $x \in S$ and $S(x) = 0$ for all $x \notin S$. A function on Σ^* (i.e., from Σ^* to Σ^*) is *polynomially bounded* if there exists a polynomial p satisfying $|f(x)| \leq p(|x|)$ for all strings $x \in \Sigma^*$.

For each natural number $k \geq 1$, \mathcal{H}_k expresses a *Hilbert space* of dimension k and each element of \mathcal{H}_k is expressed as $|\phi\rangle$ using Dirac’s “ket” notation. In this paper, we are interested only in the case where k is a power of 2 and we implicitly assume that k is of the form 2^n for a certain $n \in \mathbb{N}$. Any element of \mathcal{H}_2 that has the unit norm is called a *quantum bit* or a *qubit*. By choosing a standard computational basis $B_1 = \{|0\rangle, |1\rangle\}$, every qubit $|\phi\rangle$ can be expressed as $\alpha_0|0\rangle + \alpha_1|1\rangle$ for an appropriate choice of two values $\alpha_0, \alpha_1 \in \mathbb{C}$ (called *amplitudes*) satisfying $|\alpha_0|^2 + |\alpha_1|^2 = 1$. We also express $|\phi\rangle$ as a *column vector* of the form $\begin{pmatrix} \alpha_0 \\ \alpha_1 \end{pmatrix}$; in particular, $|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ and $|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$. In a more general case of $n \geq 1$, we use $B_n = \{|s\rangle \mid s \in \{0, 1\}^n\}$ as a computational basis of \mathcal{H}_{2^n} with $|B_n| = 2^n$. Given any number $n \in \mathbb{N}^+$, a *qustring of length n* is a vector $|\phi\rangle$ of \mathcal{H}_{2^n} with unit norm; namely, it is of the form $\sum_{s \in \{0, 1\}^n} \alpha_s |s\rangle$, where each amplitude α_s is in \mathbb{C} with $\sum_{s \in \{0, 1\}^n} |\alpha_s|^2 = 1$. Notice that a qubit is a qustring of length 1. The exception is the *null vector*, denoted simply by $\mathbf{0}$, which has norm 0. Although the null vector could be a qustring of “arbitrary” length n , we instead refer to it as the *qustring of length 0* for convenience. We use the notation Φ_n for each $n \in \mathbb{N}$ to denote the collection of all qustrings of length n . Finally, we set $\Phi_\infty = \bigcup_{n \in \mathbb{N}} \Phi_n$ (the set of all qustrings).

When $s = s_1 s_2 \dots s_n$ with $s_i \in \{0, 1\}$ for any index $i \in [n]$, the qustring $|s\rangle$ coincides with $|s_1\rangle \otimes |s_2\rangle \otimes \dots \otimes |s_n\rangle$, where \otimes denotes the *tensor product* and is expressed as, for example, $|00\rangle = (1 \ 0 \ 0 \ 0)^T$, $|01\rangle = (0 \ 1 \ 0 \ 0)^T$, and $|11\rangle = (0 \ 0 \ 0 \ 1)^T$. The *transposed conjugate* of $|s\rangle$ is denoted by $\langle s|$ (with the “bra” notation). For instance, if $|\phi\rangle = \alpha|01\rangle + \beta|10\rangle$, then $\langle\phi| = \alpha^*\langle 01| + \beta^*\langle 10|$. The *inner product* of $|\phi\rangle$ and $|\psi\rangle$ is expressed as $\langle\phi|\psi\rangle$ and the *norm* of $|\phi\rangle$ is thus $\sqrt{\langle\phi|\phi\rangle}$. When we *observe* or *measure* $|\phi\rangle$ in the computational basis B_n , we obtain each string $s \in \{0, 1\}^n$ with probability $|\langle s|\phi\rangle|^2$.

Let $\mathcal{H}_\infty = \bigcup_{n \in \mathbb{N}^+} \mathcal{H}_{2^n}$. We extend the “length” notion to arbitrary quantum states in \mathcal{H}_∞ . Given each non-null vector $|\phi\rangle$ in \mathcal{H}_∞ , the *length* of $|\phi\rangle$, denoted by $\ell(|\phi\rangle)$, is the minimal number $n \in \mathbb{N}$ satisfying $|\phi\rangle \in \mathcal{H}_{2^n}$; in other words, $\ell(|\phi\rangle)$ is the logarithm of the dimension of the vector $|\phi\rangle$. Conventionally, we further set $\ell(\mathbf{0}) = \ell(\alpha) = 0$ for the null vector $\mathbf{0}$ and any scalar $\alpha \in \mathbb{C}$. By this convention, if $\ell(|\phi\rangle) = 0$ for a quantum state $|\phi\rangle$, then $|\phi\rangle$ must be the qustring of length 0. A qustring $|\phi\rangle$ of length n is called *basic* if $|\phi\rangle = |s\rangle$ for a certain binary string s and we often identify such a basic qustring $|s\rangle$ with the corresponding classical binary string s for convenience. Since all basic qustrings in Φ_n form B_n , \mathcal{H}_{2^n} is spanned by all elements in Φ_n .

The *partial trace* over a system B of a composite system AB , denoted by tr_B , is a quantum operator for which $tr_B(|\phi\rangle\langle\phi|)$ is a vector obtained by tracing out B from the *outer product* $|\phi\rangle\langle\phi|$ of a quantum state $|\phi\rangle$. Regarding a quantum state $|\phi\rangle$ of n qubits, we use a handy notation $tr_k(|\phi\rangle\langle\phi|)$ to mean the quantum state obtained from $|\phi\rangle$ by tracing out all qubits except for the first k qubits. For example, it follows for $\sigma_1, \sigma_2, \tau_1, \tau_2 \in \{0, 1\}$ that $tr_1(|\sigma_1\rangle\langle\sigma_2| \otimes |\tau_1\rangle\langle\tau_2|) = |\sigma_1\rangle\langle\sigma_2| \cdot tr(|\tau_1\rangle\langle\tau_2|)$, where $tr(B)$ denotes the trace of a matrix B . The *trace norm* $\|A\|_{tr}$ of a square matrix A is defined by $\|A\|_{tr} = tr(\sqrt{AA^\dagger})$. The *total variation distance* between two ensembles $p = \{p_i\}_{i \in A}$ and $q = \{q_i\}_{i \in A}$ of real numbers over a finite index set A is $\frac{1}{2} \|p - q\|_1 = \frac{1}{2} \sum_{i \in A} |p_i - q_i|$.

Throughout this paper, we take special conventions concerning three notations, $|\cdot\rangle$, \otimes , and $\|\cdot\|$, which respectively express quantum states, the tensor product, and the ℓ_2 -norm. These conventions slightly deviate from the standard ones used in, e.g., [24], but they make our mathematical descriptions in later sections simpler and more succinct.

NOTATIONAL CONVENTIONS. We freely abbreviate $|\phi\rangle \otimes |\psi\rangle$ as $|\phi\rangle|\psi\rangle$ for any two vectors $|\phi\rangle$ and $|\psi\rangle$. Given two binary strings s and t , $|st\rangle$ means $|s\rangle \otimes |t\rangle$ or $|s\rangle|t\rangle$. Let k and n be two integers with $0 < k < n$. Any qustring $|\phi\rangle$ of length n is expressed in general as $|\phi\rangle = \sum_{s:|s|=k} |s\rangle|\phi_s\rangle$, where each $|\phi_s\rangle$ is a qustring of length $n - k$. This qustring $|\phi_s\rangle$ can be viewed as a consequence of applying a partial projective measurement to the first k qubits of $|\phi\rangle$, and therefore it is possible to express $|\phi_s\rangle$ succinctly as $\langle s|\phi\rangle$. With this new, convenient notation, $|\phi\rangle$ coincides with $\sum_{s:|s|=k} |s\rangle \otimes \langle s|\phi\rangle$, which is simplified as $\sum_{s:|s|=k} |s\rangle\langle s|\phi\rangle$. Notice that, when $k = n$, $\langle s|\phi\rangle$ is a scalar, say, $\alpha \in \mathbb{C}$. Hence, $|s\rangle \otimes \langle s|\phi\rangle$ is $|s\rangle \otimes \alpha$ and it is treated as a column vector $\alpha|s\rangle$; similarly, we identify $\alpha \otimes |s\rangle$ with $\alpha|s\rangle$. In these cases, \otimes is treated merely as the *scalar multiplication*. As a consequence, the equality $|\phi\rangle = \sum_{s:|s|=k} |s\rangle\langle s|\phi\rangle$ holds even when $k = n$. Concerning the null vector $\mathbf{0}$, we also take the following special treatment: for any vector $|\phi\rangle \in \mathcal{H}_\infty$, (i) $0 \otimes |\phi\rangle = |\phi\rangle \otimes 0 = \mathbf{0}$, (ii) $|\phi\rangle \otimes \mathbf{0} = \mathbf{0} \otimes |\phi\rangle = \mathbf{0}$, and (iii) when $|\psi\rangle$ is the null vector, $\langle\phi|\psi\rangle = \langle\psi|\phi\rangle = 0$. Associated with those conventions on the partial projective measurement $\langle\phi|\psi\rangle$, we also extend the use of the norm notation $\|\cdot\|$ to scalars. When $\ell(|\phi\rangle) = \ell(|\psi\rangle)$, $\|\langle\phi|\psi\rangle\|$ expresses the absolute value $|\langle\phi|\psi\rangle|$; more generally, for any number $\alpha \in \mathbb{C}$, $\|\alpha\|$ means $|\alpha|$. With these extra conventions, when $|\phi\rangle$ has the form $\sum_{s:|s|=k} |s\rangle\langle s|\phi\rangle$, the equation $\|\phi\|^2 = \sum_{s:|s|=k} \|\langle s|\phi\rangle\|^2$ always holds for any index $k \in [n]$.

2.2. Quantum Turing machines. We assume the reader’s fundamental knowledge on the notion of quantum Turing machine (or QTM) defined in [4]. As was done in [35], we allow a QTM to equip multiple tapes and to move its multiple tape heads nonconcurrently either to the right or to the left, or to make the tape heads stay still. Such a QTM was also discussed elsewhere (e.g., [26]) and is known to polynomially equivalent to the model proposed in [4].

To compute functions from Σ^* to Σ^* over a given alphabet Σ , we generally introduce QTMs as machines equipped with output tapes on which output strings are written in a certain specified way by the time the machines halt. By identifying languages with their characteristic functions, such QTMs can be seen as language acceptors as well.

Formally, a *k-tape quantum Turing machine* (referred to as *k-tape QTM*), for $k \in \mathbb{N}^+$, is a sextuple $(Q, \Sigma, \Gamma_1 \times \dots \times \Gamma_k, \delta, q_0, Q_f)$, where Q is a finite set of inner states including the initial state q_0 and a set Q_f of final states with $Q_f \subseteq Q$, each Γ_i is an alphabet used for tape i with a distinguished blank symbol $\#$ satisfying $\Sigma \subseteq \Gamma_1$, and δ is a quantum transition function from $Q \times \tilde{\Gamma}^{(k)} \times Q \times \tilde{\Gamma}^{(k)} \times \{L, N, R\}^k$ to \mathbb{C} , where $\tilde{\Gamma}^{(k)}$ stands for $\Gamma_1 \times \dots \times \Gamma_k$. For convenience, we identify L, N , and R with $-1, 0$, and $+1$, respectively, and we set $D = \{0, \pm 1\}$. For more information, refer to [35].

All tape cells of each tape are indexed sequentially by integers. The cell indexed 0 on each tape is called the *start cell*. At the beginning of the computation, M is in the inner state q_0 , all the tapes except for the input tape are blank, and all tape heads are scanning the start cells. A given input string $x_1x_2 \dots x_n$ is initially written on the input tape in such a way that, for each index $i \in [n]$, x_i is in cell i (not cell $i - 1$). When M enters a final state, an *output* of M is the content of the string written on an output tape (if M has only a single tape, then an output tape is the same as the tape used to hold inputs) from the start cell, stretching to the right until the first blank symbol. A *configuration* of M is expressed as a triplet $(p, (h_i)_{i \in [k]}, (z_i)_{i \in [k]})$, which indicates that M is currently in inner state p having k tape heads at cells indexed by h_1, \dots, h_k with tape contents z_1, \dots, z_k , respectively. The notion of configuration will be slightly modified in Sections 4–5 to make the proof of our main theorem simpler. An *initial configuration* is of the form $(q_0, 0, x)$ and a *final configuration* is a configuration having a final state. The configuration space is spanned by the basis vectors in $\{|q, h, z\rangle \mid q \in Q, h \in \mathbb{Z}^k, z \in \Gamma_1^* \times \dots \times \Gamma_k^*\}$. For a nonempty string z_i and an index $h \in [|z_i|]$, $z_i[h]$ denotes the h th symbol in z_i . For example, if $z_i = 01101$, then $z_i[1] = 0$, $z_i[2] = 1$, and $z_i[5] = 1$. The *time-evolution operator* U_δ of M acting on the configuration space is induced from δ as

$$U_\delta |p, h, z\rangle = \sum_{q, w, d} \delta(p, z_h, q, z'_h, d) |q, h_d, z'\rangle,$$

where $p \in Q$, $h = (h_i)_{i \in [k]} \in \mathbb{Z}^k$, $z = (z_i)_{i \in [k]} \in \Gamma_1^* \times \dots \times \Gamma_k^*$, $z_h = (z_i[h_i])_{i \in [k]}$, $h_d = (h_i + d_i)_{i \in [k]}$, and $z' = (z'_i)_{i \in [k]}$, where each z'_i is the same as z except for the h_i -th symbol. Moreover, in the summation, variables $q, z'_h = (z'_i[h_i])_{i \in [k]}$, and $d = (d_i)_{i \in [k]}$ respectively range over $Q, \tilde{\Gamma}^{(k)}$, and D^k . Any entry of U_δ is called an *amplitude*. Quantum mechanics demand the time-evolution operator U_δ of the QTM to be unitary.

Each step of M consists of two phases: first apply δ and then perform a *partial projective measurement*, in which we check whether M is in a final state (i.e., an inner state in Q_f). Formally, a *computation* of M on input x is a series of superpositions of configurations produced by sequential applications of U_δ and measurements, starting from an initial configuration of M on x . If M enters a final state along a computation path, this computation path terminates; otherwise, its computation must continue.

A *k-tape QTM* $M = (Q, \Sigma, \tilde{\Gamma}^{(k)}, \delta, q_0, Q_f)$ is *well-formed* if δ satisfies three local conditions: unit length, separability, and orthogonality. To explain these conditions, as presented in [35, Lemma 1], we first introduce the following

notations. For our convenience, we set $E = \{0, \pm 1, \pm 2\}$ and $H = \{0, \pm 1, \mathfrak{h}\}$. Given elements $(p, \sigma, \tau) \in Q \times (\tilde{\Gamma}^{(k)})^2$, $\varepsilon = (\varepsilon_i)_{i \in [k]} \in E^k$, and $d = (d_i)_{i \in [k]} \in D^k$, we define $D_\varepsilon = \{d \in D^k \mid \forall i \in [k] (|2d_i - \varepsilon_i| \leq 1)\}$ and $E_d = \{\varepsilon \in E^k \mid d \in D_\varepsilon\}$. Moreover, let $h_{d,\varepsilon} = (h_{d_i,\varepsilon_i})_{i \in [k]}$, where $h_{d_i,\varepsilon_i} = 2d_i - \varepsilon_i$ if $\varepsilon_i \neq 0$ and $h_{d_i,\varepsilon_i} = \mathfrak{h}$ otherwise. Finally, we define $\delta(p, \sigma) = \sum_{q, \tau, d} \delta(p, \sigma, q, \tau, d) |q, \tau, d\rangle$ and $\delta[p, \sigma, \tau | \varepsilon] = \sum_{q \in Q} \sum_{d \in D_\varepsilon} \delta(p, \sigma, q, \tau, d) |E_d|^{-1/2} |q\rangle |h_{d,\varepsilon}\rangle$, where $\sigma, \tau \in \tilde{\Gamma}^{(k)}$ and $d \in D^k$.

1. (unit length) $\|\delta(p, \sigma)\| = 1$ for all $(p, \sigma) \in Q \times \tilde{\Gamma}^{(k)}$.
2. (orthogonality) $\delta(p_1, \sigma_1) \cdot \delta(p_2, \sigma_2) = 0$ for any distinct pair $(p_1, \sigma_1), (p_2, \sigma_2) \in Q \times \tilde{\Gamma}^{(k)}$.
3. (separability) $\delta[p_1, \sigma_1, \tau_1 | \varepsilon] \cdot \delta[p_2, \sigma_2, \tau_2 | \varepsilon'] = 0$ for any distinct pair $\varepsilon, \varepsilon' \in E^k$ and for any pair $(p_1, \sigma_1, \tau_1), (p_2, \sigma_2, \tau_2) \in Q \times (\tilde{\Gamma}^{(k)})^2$.

The well-formedness of a QTM captures the unitarity of its time-evaluation operator.

LEMMA 2.1 (Well-Formedness Lemma of [35]). *A k -tape QTM M with a transition function δ is well-formed iff the time-evolution operator of M preserves the ℓ_2 -norm.*

Given a nonempty subset K of \mathbb{C} , we say that a QTM is of K -amplitude if all values of its quantum transition function belong to K . It is of significant importance to limit the choice of amplitude within an appropriate set K of reasonable numbers. With the use of such a set K , we introduce two important complexity classes BQP_K and $FBQP_K$.

DEFINITION 2.2. Let K be any nonempty subset of \mathbb{C} and let Σ be any alphabet.

1. A subset S of Σ^* is in BQP_K if there exists a multitape, polynomial-time, well-formed QTM M with K -amplitudes such that, for every string x , M outputs $S(x)$ with probability at least $2/3$ [4].
2. A single-valued function f from Σ^* to Σ^* is called *bounded-error quantum polynomial-time computable* if there exists a multitape, polynomial-time, well-formed QTM M with K -amplitudes such that, on every input x , M outputs $f(x)$ with probability at least $2/3$. Let $FBQP_K$ denote the set of all such functions [37].

The use of *arbitrary* complex amplitudes turns out to make BQP_K quite powerful. As Adleman, DeMarrais, and Huang [1] demonstrated, $BQP_{\mathbb{C}}$ contains all possible languages, and thus $BQP_{\mathbb{C}}$ is no longer recursive. Therefore, we usually pay our attention only to polynomial-time approximable amplitudes and, for this reason, when $K = \tilde{\mathbb{C}}$, we always drop subscript K and briefly write BQP and $FBQP$ instead of BQP_K and $FBQP_K$, respectively. It is also possible to further limit the amplitude set K to $\{0, \pm 1, \pm \frac{3}{5}, \pm \frac{4}{5}\}$ because $BQP = BQP_{\{0, \pm 1, \pm \frac{3}{5}, \pm \frac{4}{5}\}}$ holds [1].

2.3. Quantum circuits. A k -qubit quantum gate, for $k \in \mathbb{N}^+$, is a unitary operator acting on a Hilbert space of dimension 2^k . Since any quantum state is a vector in a certain Hilbert space, each entry of such a quantum state is customarily called an *amplitude*. The Walsh-Hadamard transform (WH) and the controlled-NOT

transform (CNOT) defined as

$$WH = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \text{ and } CNOT = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix},$$

are typical quantum gates acting on 1 qubit and 2 qubits, respectively. If a quantum gate U acting on k qubits is applied to a k -qubit quantum state $|\phi\rangle$, then we obtain a new quantum state $U|\phi\rangle$. Notice that every quantum gate preserves the norm of any quantum state given as an input. A *quantum circuit* is a product of a finite number of layers, where each layer is a Kronecker product of allowed quantum gates. We often concentrate on a particular set of quantum gates to construct quantum circuits. Let us consider the specific quantum gates: the CNOT gate and three one-qubit gates of the form

$$Z_{1,\theta} = \begin{pmatrix} e^{i\theta} & 0 \\ 0 & 1 \end{pmatrix}, \quad Z_{2,\theta} = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{pmatrix}, \text{ and } R_\theta = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix},$$

where θ is a real number with $0 \leq \theta < 2\pi$. Notice that WH equals $R_{\frac{\pi}{4}}$. Those gates form a *universal set* of quantum gates [2] since WH and $Z_{2,\frac{\pi}{4}}$ (called the $\pi/8$ gate) can approximate any single-qubit unitary operator to arbitrary accuracy. For convenience, we call them *elementary gates*. The set of $CNOT$, WH , and $Z_{2,\frac{\pi}{4}}$ is also known to be universal [5].

Given an amplitude set K , a quantum circuit C is said to have K -*amplitudes* if all entries of each quantum gate used inside C are drawn from K . For any k -qubit quantum gate and any integer $n > k$, $G^{(n)}$ denotes $G \otimes I^{\otimes n-k}$, the n -qubit expansion of G . An n -qubit quantum circuit is formally defined as a finite sequence $(G_m, \pi_m), (G_{m-1}, \pi_{m-1}), \dots, (G_1, \pi_1)$ such that each G_i is an n_i -qubit quantum gate with $n_i \leq n$ and π_i is a permutation on $\{1, 2, \dots, n\}$. This quantum circuit represents the unitary operator $U = U_m U_{m-1} \cdots U_1$, where U_i is of the form $V_{\pi_i}^\dagger G_i^{(n)} V_{\pi_i}$ and $V_{\pi_i}(|x_1 \cdots x_n\rangle) = |x_{\pi_i(1)} \cdots x_{\pi_i(n)}\rangle$ for each $i \in [m]$. The *size* of a quantum circuit is the total number of quantum gates in it. Yao [39] and later Nishimura and Ozawa [25] showed that, for any k -tape QTM and a polynomial p , there exists a family of quantum circuits of size $O(p(n)^{k+1})$ that exactly simulates M .

A family $\{C_n\}_{n \in \mathbb{N}}$ of a quantum circuit is said to be P -*uniform* if there exists a deterministic (classical) Turing machine that, on input 1^n , produces a code of C_n in time polynomial in the size of C_n , provided that we use a fixed, efficient coding scheme to describe each quantum circuit.

PROPOSITION 2.3 ([39] (see also [25])). *For any language L over an alphabet $\{0, 1\}$, L is in BQP iff there exist a polynomial p and a P -uniform family $\{C_n\}_{n \in \mathbb{N}}$ of quantum circuits having \tilde{C} -amplitudes such that $\|\langle L(x) | C_{|x|} | x 10^{p(|x|)} \rangle\|^2 \geq \frac{2}{3}$ holds for all $x \in \{0, 1\}^*$, where L is seen as the characteristic function of L .*

Yao’s inspiring proof [39] of Proposition 2.3 gives a foundation to our proof of Lemma 4.3, which provides in Section 3.1 a simulation of a well-formed QTM by an appropriately chosen \square_1^{OP} -function.

§3. A new, simple schematic definition. As noted in Section 1, the “schematic” definition of recursive function means an inductive (or constructive) way of defining the set of computable functions and it involves a small set of so-called *initial functions* as well as a small set of *construction rules*, which are sequentially applied finitely many times to build more complex functions from certain functions that have been already constructed. A similar schematic characterization is known for polynomial-time computable functions (as well as languages) [6–8, 23, 34]. Along this line of work, we wish to present a new, simple schematic definition composed of a small set of initial quantum functions and a small set of construction rules, intending to make this schematic definition appropriately capture polynomial-time computable quantum functions, where a *quantum function* is a function mapping \mathcal{H}_∞ to \mathcal{H}_∞ . As remarked briefly in Section 1, it is important to note that our term of “quantum function” is quite different from the one used in, e.g., [37], in which “quantum function” refers to functions that take classical input strings and produce either classical output strings or acceptance probabilities of multitape polynomial-time well-formed QTMs and thus it maps Σ^* to either Σ^* or the real unit interval $[0, 1]$, where Σ is an appropriate alphabet.

3.1. Definition of \square_1^{QP} -functions. Our schematic definition induces a special function class, called \square_1^{QP} (where \square is pronounced “square”), capturing polynomial-time computable quantum functions mapping \mathcal{H}_∞ to \mathcal{H}_∞ , which is composed of a small set of initial quantum functions and four special construction rules: composition, swapping, branching, and multiqubit quantum recursion. Definition 3.1 formally presents our schematic definition.

Hereafter, we say that a quantum function f from \mathcal{H}_∞ to \mathcal{H}_∞ is *dimension-preserving* if, for every quantum state $|\phi\rangle \in \mathcal{H}_\infty$, $\ell(|\phi\rangle) = \ell(f(|\phi\rangle))$.

DEFINITION 3.1. Let \square_1^{QP} denote the collection of all quantum functions that are obtained from the initial quantum functions in Scheme I by a finite number (including zero) of applications of construction rules II–IV to quantum functions that have been already constructed, where Schemata I–IV² are given as follows. Let $|\phi\rangle$ be any quantum state in \mathcal{H}_∞ .

- I. The initial quantum functions. Let $\theta \in [0, 2\pi) \cap \tilde{\mathbb{C}}$ and $a \in \{0, 1\}$.
 - 1) $I(|\phi\rangle) = |\phi\rangle$. (identity)
 - 2) $PHASE_\theta(|\phi\rangle) = |0\rangle\langle 0|\phi\rangle + e^{i\theta}|1\rangle\langle 1|\phi\rangle$. (phase shift)
 - 3) $ROT_\theta(|\phi\rangle) = \cos\theta|\phi\rangle + \sin\theta(|1\rangle\langle 0|\phi\rangle - |0\rangle\langle 1|\phi\rangle)$. (rotation around xy -axis at angle θ)
 - 4) $NOT(|\phi\rangle) = |0\rangle\langle 1|\phi\rangle + |1\rangle\langle 0|\phi\rangle$. (negation)
 - 5) $SWAP(|\phi\rangle) = \begin{cases} |\phi\rangle & \text{if } \ell(|\phi\rangle) \leq 1, \\ \sum_{a,b \in \{0,1\}} |ab\rangle\langle ba|\phi\rangle & \text{otherwise.} \end{cases}$ (swapping of 2 qubits)
 - 6) $MEAS[a](|\phi\rangle) = |a\rangle\langle a|\phi\rangle$. (partial projective measurement)

²The current formalism of Schemata I–IV corrects discrepancies caused by the early formalism given in the extended abstract [38].

- II. The composition rule. From g and h , we define $Compo[g,h]$ as follows:
 $Compo[g,h](|\phi\rangle) = g \circ h(|\phi\rangle) (= g(h(|\phi\rangle)))$.
- III. The branching rule. From g and h , we define $Branch[g,h]$ as:
 - (i) $Branch[g,h](|\phi\rangle) = |\phi\rangle$ if $\ell(|\phi\rangle) \leq 1$,
 - (ii) $Branch[g,h](|\phi\rangle) = |0\rangle \otimes g(|0\rangle) + |1\rangle \otimes h(|1\rangle)$ otherwise.
- IV. The multiqubit quantum recursion rule. From g, h , dimension-preserving p , and $k, t \in \mathbb{N}^+$, we define $kQRec_t[g,h,p|\mathcal{F}_k]$ as:
 - (i) $kQRec_t[g,h,p|\mathcal{F}_k](|\phi\rangle) = g(|\phi\rangle)$ if $\ell(|\phi\rangle) \leq t$,
 - (ii) $kQRec_t[g,h,p|\mathcal{F}_k](|\phi\rangle) = h(\sum_{s:|s|=k} |s\rangle \otimes f_s(\langle s|\psi_{p,\phi}\rangle))$ otherwise,
 where $|\psi_{p,\phi}\rangle = p(|\phi\rangle)$ and $\mathcal{F}_k = \{f_s\}_{s \in \{0,1\}^k} \subseteq \{kQRec_t[g,h,p|\mathcal{F}_k], I\}$. To emphasize “ k ,” we call this rule by the k -qubit quantum recursion. In the case of $k = 1$, we write $QRec_t[g,h,p|f_0,f_1]$ in place of $1QRec_t[g,h,p|\{f_0,f_1\}]$ for brevity.

In Scheme I, $PHASE_\theta$ and ROT_θ correspond respectively to the matrices $Z_{2,\theta}$ and R_θ given in Section 2.3. Latter, we will argue that an angle θ in $PHASE_\theta$ and ROT_θ could be fixed to $\pi/4$. The quantum function $MEAS$ is associated with a *partial projective measurement*, in the computational basis $\{0,1\}$, applied to the first qubit of $|\phi\rangle$, when $\ell(|\phi\rangle) \geq 1$, and it obviously follows that $\ell(MEAS[i](|\phi\rangle)) \leq \ell(|\phi\rangle)$.

Before proceeding further, to help the reader understand the behaviors of the initial quantum functions listed in Scheme I, we briefly illustrate how these functions transform basic qustrings of length 3. For bits $a,b,c,d \in \{0,1\}$ with $d \neq a$, it follows that $I(|abc\rangle) = |abc\rangle$, $PHASE_\theta(|abc\rangle) = e^{i\theta a}|abc\rangle$, $ROT_\theta(|abc\rangle) = \cos\theta|abc\rangle + (-1)^a \sin\theta|\bar{a}bc\rangle$, $NOT(|abc\rangle) = |\bar{a}bc\rangle$, $SWAP(|abc\rangle) = |bac\rangle$, $MEAS[a](|abc\rangle) = |abc\rangle$, and $MEAS[d](|abc\rangle) = \mathbf{0}$, where $\bar{a} = 1 - a$.

Scheme IV is a core of the definition of \square_1^{QP} . The standard recursion rule used to define a primitive recursive function f from two functions g and h has the form: $f(0,x) = g(x)$ and $f(n+1,x) = h(n,x,f(n,x))$ for any $n \in \mathbb{N}$. This rule requires an internal counter (in the first argument place of f) that controls the number of iterated applications of h . In Scheme IV, however, we do not use such a counter. Instead, we use a divide-and-conquer strategy to slice a given quantum state qubit by qubit. At each inductive step, we deal with k -qubit shorter quantum states until the quantum state has become length t or less. We wish to provide two concrete examples of how Scheme IV works in the cases of $k = 1, 2$. Notice that, in the case of $k = 1$, Scheme IV becomes the 1-qubit (or single-qubit) quantum recursion rule described as:

$$\begin{aligned}
 & \text{(i')} \quad QRec_t[g,h,p|f_0,f_1](|\phi\rangle) = g(|\phi\rangle) && \text{if } \ell(|\phi\rangle) \leq t, \\
 & \text{(ii')} \quad QRec_t[g,h,p|f_0,f_1](|\phi\rangle) = h(|0\rangle \otimes f_0(\langle 0|\psi_{p,\phi}\rangle) + |1\rangle \otimes f_1(\langle 1|\psi_{p,\phi}\rangle)) && \text{otherwise,}
 \end{aligned}$$

where each of f_0 and f_1 must be either I or $QRec_t[g,h,p|f_0,f_1]$.

EXAMPLE 1. We first consider the case of $k = 1$. In this example, we set $t = 1$, $g = NOT$, $h = SWAP$, and $p = NOT$, and we briefly write F for $QRec_1[NOT,SWAP,NOT|f_0,f_1]$ with $f_0 = I$ and $f_1 = F$. It is worth mentioning that $\mathbf{0}$ is a special object and we obtain $g(\mathbf{0}) = \mathbf{0}$ for any quantum function g by Lemma 3.4(1). Let $|\phi\rangle$ denote any quantum state in \mathcal{H}_∞ given as an input to F .

(1) Assume that $|\phi\rangle$ has length 1 and is of the form $\alpha|0\rangle + \beta|1\rangle$ in general. By Lemma 3.4(2), it suffices for us to consider the computational basis $B_1 = \{|0\rangle, |1\rangle\}$. Since $\ell(|\phi\rangle) \leq t$, the outcomes of f for those basis qubits $|0\rangle$ and $|1\rangle$ are calculated as follows.

- (i) $F(|0\rangle) = g(|0\rangle) = NOT(|0\rangle) = |1\rangle$.
- (ii) $F(|1\rangle) = g(|1\rangle) = NOT(|1\rangle) = |0\rangle$.

Obviously, $F(\mathbf{0}) = \mathbf{0}$ holds. Since $|\phi\rangle$ is a superposition of the form $\alpha|0\rangle + \beta|1\rangle$, the above calculations instantly imply

$$F(|\phi\rangle) = F(\alpha|0\rangle + \beta|1\rangle) = NOT(\alpha|0\rangle + \beta|1\rangle) = \alpha NOT(|0\rangle) + \beta NOT(|1\rangle) = \alpha|1\rangle + \beta|0\rangle.$$

(2) Next, let us assume that $|\phi\rangle$ is of length 2. In the case where $|\phi\rangle$ is the basis quantum state $|00\rangle$ in $B_2 = \{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$, we obtain $|\psi_{p,\phi}\rangle = |\psi_{NOT,00}\rangle = NOT(|00\rangle) = |10\rangle$. Similarly, if $|\phi\rangle$ is $|01\rangle$, $|10\rangle$, and $|11\rangle$, then the quantum state $|\psi_{p,\phi}\rangle$ is $|11\rangle$, $|00\rangle$, and $|01\rangle$, respectively. Concerning the partial projective measurement, it follows that, for any bit $a \in \{0,1\}$, $\langle 1|1a\rangle = |a\rangle$, $\langle 0|0a\rangle = |a\rangle$, $\langle 1|0a\rangle = \mathbf{0}$, and $\langle 0|1a\rangle = \mathbf{0}$. Note also that $I(\mathbf{0}) = \mathbf{0}$. Using these equalities together with $F(|0\rangle) = |1\rangle$ and $F(|1\rangle) = |0\rangle$ obtained in (1), we can calculate the outcome $F(|\phi\rangle)$ as follows.

- (i) $F(|00\rangle) = h(|0\rangle \otimes f_0(\langle 0|\psi_{NOT,00}\rangle) + |1\rangle \otimes f_1(\langle 1|\psi_{NOT,00}\rangle)) = h(|0\rangle \otimes I(\langle 0|10\rangle) + |1\rangle \otimes F(\langle 1|10\rangle)) = h(|0\rangle \otimes I(\mathbf{0}) + |1\rangle \otimes F(|0\rangle)) = h(|1\rangle \otimes F(|0\rangle)) = SWAP(|1\rangle \otimes |1\rangle) = SWAP(|11\rangle) = |11\rangle$.
- (ii) $F(|10\rangle) = h(|0\rangle \otimes f_0(\langle 0|\psi_{NOT,10}\rangle) + |1\rangle \otimes f_1(\langle 1|\psi_{NOT,10}\rangle)) = h(|0\rangle \otimes I(\langle 0|00\rangle) + |1\rangle \otimes F(\langle 1|00\rangle)) = h(|0\rangle \otimes I(|0\rangle) + |1\rangle \otimes F(\mathbf{0})) = h(|0\rangle \otimes I(|0\rangle)) = SWAP(|0\rangle \otimes |0\rangle) = SWAP(|00\rangle) = |00\rangle$.
- (iii) $F(|01\rangle) = h(|0\rangle \otimes f_0(\langle 0|\psi_{NOT,01}\rangle) + |1\rangle \otimes f_1(\langle 1|\psi_{NOT,01}\rangle)) = h(|0\rangle \otimes I(\langle 0|11\rangle) + |1\rangle \otimes F(\langle 1|11\rangle)) = h(|0\rangle \otimes I(\mathbf{0}) + |1\rangle \otimes F(|1\rangle)) = h(|1\rangle \otimes F(|1\rangle)) = SWAP(|1\rangle \otimes |0\rangle) = SWAP(|10\rangle) = |01\rangle$.
- (iv) $F(|11\rangle) = h(|0\rangle \otimes f_0(\langle 0|\psi_{NOT,11}\rangle) + |1\rangle \otimes f_1(\langle 1|\psi_{NOT,11}\rangle)) = h(|0\rangle \otimes I(\langle 0|01\rangle) + |1\rangle \otimes F(\langle 1|01\rangle)) = h(|0\rangle \otimes I(|1\rangle) + |1\rangle \otimes F(\mathbf{0})) = h(|0\rangle \otimes I(|1\rangle)) = SWAP(|0\rangle \otimes |1\rangle) = SWAP(|01\rangle) = |10\rangle$.

By Lemma 3.4(2), when $|\phi\rangle$ is of the form $\alpha|00\rangle + \beta|10\rangle$ for example, we obtain

$$F(|\phi\rangle) = F(\alpha|00\rangle + \beta|10\rangle) = \alpha F(|00\rangle) + \beta F(|10\rangle) = \alpha|11\rangle + \beta|00\rangle.$$

(3) Let us consider the case where the length of $|\phi\rangle$ is 3. For the computational basis $B_3 = \{|000\rangle, |001\rangle, |010\rangle, \dots, |111\rangle\}$, we calculate the outcomes $F(|\phi\rangle)$ only for inputs $|\phi\rangle$ in $\{|001\rangle, |101\rangle\}$. Notice that $|\psi_{NOT,001}\rangle = |101\rangle$ and $|\psi_{NOT,101}\rangle = |001\rangle$. Recall from (1)–(2) that $F(|01\rangle) = |01\rangle$ and $F(\mathbf{0}) = \mathbf{0}$.

- (i) $F(|001\rangle) = h(|0\rangle \otimes f_0(\langle 0|\psi_{NOT,001}\rangle) + |1\rangle \otimes f_1(\langle 1|\psi_{NOT,001}\rangle)) = h(|0\rangle \otimes I(\langle 0|101\rangle) + |1\rangle \otimes F(\langle 1|101\rangle)) = h(|0\rangle \otimes I(\mathbf{0}) + |1\rangle \otimes F(|01\rangle)) = h(|1\rangle \otimes F(|01\rangle)) = SWAP(|1\rangle \otimes |01\rangle) = SWAP(|101\rangle) = |011\rangle$.
- (ii) $F(|101\rangle) = h(|0\rangle \otimes f_0(\langle 0|\psi_{NOT,101}\rangle) + |1\rangle \otimes f_1(\langle 1|\psi_{NOT,101}\rangle)) = h(|0\rangle \otimes I(\langle 0|001\rangle) + |1\rangle \otimes F(\langle 1|001\rangle)) = h(|0\rangle \otimes I(|01\rangle) + |1\rangle \otimes F(\mathbf{0})) = h(|0\rangle \otimes I(|01\rangle)) = SWAP(|0\rangle \otimes |01\rangle) = SWAP(|001\rangle) = |001\rangle$.

If $|\phi\rangle$ is of the form $\alpha|001\rangle + \beta|101\rangle$ for example, Lemma 3.4(2) implies that $F(|\phi\rangle) = F(\alpha|001\rangle + \beta|101\rangle) = \alpha F(|001\rangle) + \beta F(|101\rangle) = \alpha|011\rangle + \beta|001\rangle$.

EXAMPLE 2. We see another example for the case of $k = 2$. By setting $t = 3$, $g = NOT$, $h = SWAP$, and $p = NOT$ for instance, we write F for $2QRec_3[NOT, SWAP, NOT|f_{00}, f_{01}, f_{10}, f_{11}]$ with $f_{00} = I$, $f_{01} = NOT$, $f_{10} = F$, and $f_{11} = F$. Let $|\phi\rangle$ denote any quantum state in \mathcal{H}_∞ .

- (1) When the length of $|\phi\rangle$ is at most 3, we instantly obtain $F(|\phi\rangle) = g(|\phi\rangle) = NOT(|\phi\rangle)$. For example, $F(|0\rangle) = |1\rangle$, $F(|10\rangle) = |00\rangle$, and $F(|101\rangle) = |001\rangle$.
- (2) Assuming that the length of $|\phi\rangle$ is 4, let us consider the basis quantum states in $B_4 = \{|0000\rangle, |0001\rangle, |0010\rangle, \dots, |1111\rangle\}$. Here, we are focused only on two cases: $|\phi\rangle \in \{|0101\rangle, |1011\rangle\}$. We remark that $|\psi_{NOT,0010}\rangle = NOT(|0010\rangle) = |1010\rangle$ and $|\psi_{NOT,1011}\rangle = NOT(|1011\rangle) = |0011\rangle$.
 - (i) $F(|0010\rangle) = h(|00\rangle \otimes f_{00}(\langle 00|\psi_{NOT,0010}\rangle) + |01\rangle \otimes f_{01}(\langle 01|\psi_{NOT,0010}\rangle) + |10\rangle \otimes f_{10}(\langle 10|\psi_{NOT,0010}\rangle) + |11\rangle \otimes f_{11}(\langle 11|\psi_{NOT,0010}\rangle)) = h(|00\rangle \otimes I(\langle 00|0101\rangle) + |01\rangle \otimes NOT(\langle 01|0101\rangle) + |10\rangle \otimes F(\langle 10|0101\rangle) + |11\rangle \otimes F(\langle 11|0101\rangle)) = h(|00\rangle \otimes I(\mathbf{0}) + |01\rangle \otimes NOT(\mathbf{0}) + |10\rangle \otimes F(|10\rangle) + |11\rangle \otimes F(\mathbf{0})) = h(|10\rangle \otimes F(|10\rangle)) = SWAP(|10\rangle \otimes |00\rangle) = SWAP(|1000\rangle) = |0100\rangle$.
 - (ii) $F(|1101\rangle) = h(|00\rangle \otimes f_{00}(\langle 00|\psi_{NOT,1101}\rangle) + |01\rangle \otimes f_{01}(\langle 01|\psi_{NOT,1101}\rangle) + |10\rangle \otimes f_{10}(\langle 10|\psi_{NOT,1101}\rangle) + |11\rangle \otimes f_{11}(\langle 11|\psi_{NOT,1101}\rangle)) = h(|00\rangle \otimes I(\langle 00|0101\rangle) + |01\rangle \otimes NOT(\langle 01|0101\rangle) + |10\rangle \otimes F(\langle 10|0101\rangle) + |11\rangle \otimes F(\langle 11|0101\rangle)) = h(|00\rangle \otimes I(\mathbf{0}) + |01\rangle \otimes NOT(|01\rangle) + |10\rangle \otimes F(\mathbf{0}) + |11\rangle \otimes F(\mathbf{0})) = h(|01\rangle \otimes NOT(|01\rangle)) = SWAP(|01\rangle \otimes |11\rangle) = SWAP(|0111\rangle) = |1011\rangle$.

3.2. A subclass of \square_1^{QP} and basic properties. Among the six initial quantum functions in Scheme I, *MEAS* makes a quite different behavior. It can change the norm as well as the dimensionality of quantum states, and this fact makes it irreversible in nature. For this reason, it is often beneficial in practice to limit our attention to a subclass of \square_1^{QP} , called $\widehat{\square_1^{QP}}$, which entirely prohibits the use of *MEAS*.

DEFINITION 3.2. The notation $\widehat{\square_1^{QP}}$ denotes the subclass of \square_1^{QP} defined by Schemata I-IV except for *MEAS* listed in Scheme I.

With no use of *MEAS* in Scheme I, every quantum function f in $\widehat{\square_1^{QP}}$ preserves the dimensionality of inputs; in other words, f satisfies $\ell(f(|\phi\rangle)) = \ell(|\phi\rangle)$ for any input $|\phi\rangle \in \mathcal{H}_\infty$.

Next, let us demonstrate how to construct typical unitary gates using our schematic definition.

LEMMA 3.3. The following functions are in $\widehat{\square_1^{QP}}$. Let $|\phi\rangle$ be any element in \mathcal{H}_∞ .

- 1. $CNOT(|\phi\rangle) = \begin{cases} |\phi\rangle & \text{if } \ell(|\phi\rangle) \leq 1, \\ |0\rangle\langle 0|\phi\rangle + |1\rangle \otimes NOT(\langle 1|\phi\rangle) & \text{otherwise.} \end{cases}$ (controlled-*NOT*)
- 2. $Z_{1,\theta}(|\theta\rangle) = e^{i\theta}|0\rangle\langle 0|\phi\rangle + |1\rangle\langle 1|\phi\rangle$.

3. $zROT_\theta(|\phi\rangle) = e^{i\theta}|0\rangle\langle 0|\phi\rangle + e^{-i\theta}|1\rangle\langle 1|\phi\rangle$. (*rotation around the z-axis*)
4. $GPS_\theta(|\phi\rangle) = e^{i\theta}|\phi\rangle$. (*global phase shift*)
5. $WH(|\phi\rangle) = \frac{1}{\sqrt{2}}|0\rangle \otimes (\langle 0|\phi\rangle + \langle 1|\phi\rangle) + \frac{1}{\sqrt{2}}|1\rangle \otimes (\langle 0|\phi\rangle - \langle 1|\phi\rangle)$. (*Walsh-Hadamard transform*)
6. $CPHASE_\theta(|\phi\rangle) = \begin{cases} |\phi\rangle & \text{if } \ell(|\phi\rangle) \leq 1, \\ \frac{1}{\sqrt{2}} \sum_{b \in \{0,1\}} (|0\rangle\langle b|\phi\rangle + e^{i\theta b}|1\rangle\langle b|\phi\rangle) & \text{otherwise.} \end{cases}$
(*controlled-PHASE*)

PROOF. It suffices to build each of the quantum functions given in the lemma from the initial quantum functions and by applying the construction rules. These target functions are constructed as follows.

- (1) $CNOT = Branch[I, NOT]$. Note that, when $\ell(|\phi\rangle) \leq 1$, Scheme III(i) implies $CNOT(|\phi\rangle) = |\phi\rangle$, matching Item 1 for $CNOT$.
- (2) $Z_{1,\theta} = NOT \circ PHASE_\theta \circ NOT$.
- (3) $zROT_\theta = Z_{1,\theta} \circ PHASE_{-\theta}$.
- (4) $GPS_\theta = Z_{1,\theta} \circ PHASE_\theta$.
- (5) $WH = ROT_{\frac{\pi}{4}} \circ NOT$.
- (6) $CPHASE_\theta = Branch[WH, f]$, where $f = Branch[I, GPS_\theta] \circ WH \circ NOT$. \dashv

Since any \square_1^{QP} -function f is constructed by applying Schemata I–IV, an application of one of the schemata is viewed as a basic step of the construction process of generating f . This fact helps us define the *descriptive complexity* of f to be the minimal number of times we use those schemata in order to construct f . For instance, all the initial functions have descriptive complexity 1 because they use Scheme I only once. As demonstrated in the proof of Lemma 3.3, the quantum functions $CNOT$, $Z_{1,\theta}$, and WH have descriptive complexity at most 3 and $zROT_\theta$ and GPS_θ have descriptive complexity at most 4, whereas $CPHASE_\theta$ is of descriptive complexity at most 15. This complexity measure is essential in proving, e.g., Lemma 3.4 since the lemma will be proven by induction on the descriptive complexity of a target quantum function. In Section 6.2, we will give a short discussion on this complexity measure for a future study.

Fundamental properties of $\widehat{\square_1^{QP}}$ -functions are given in the following lemma. A quantum function from \mathcal{H}_∞ to \mathcal{H}_∞ is called *norm-preserving* if $\|f(|\phi\rangle)\| = \||\phi\rangle\|$ holds for all quantum states $|\phi\rangle$ in \mathcal{H}_∞ .

LEMMA 3.4. *Let f be any quantum function in $\widehat{\square_1^{QP}}$ and let $|\phi\rangle, |\psi\rangle \in \mathcal{H}_\infty$ and $\alpha \in \mathbb{C}$.*

1. $f(\mathbf{0}) = \mathbf{0}$, where $\mathbf{0}$ is the null vector.
2. $f(|\phi\rangle + |\psi\rangle) = f(|\phi\rangle) + f(|\psi\rangle)$.
3. $f(\alpha|\phi\rangle) = \alpha \cdot f(|\phi\rangle)$.
4. f is dimension-preserving and norm-preserving.

PROOF. Let f be any $\widehat{\square_1^{QP}}$ -function, let $|\phi\rangle, |\psi\rangle \in \mathcal{H}_\infty$, and let $\alpha \in \mathbb{C}$ and $\theta \in [0, 2\pi)$ be constants. As mentioned earlier, we will prove the lemma by induction of the *descriptive complexity* of f . If f is one of the initial quantum functions in Scheme I, then it is easy to check that it satisfies Conditions 1–4 of the lemma. In particular, when $|\phi\rangle$ is the null vector $\mathbf{0}$, all of $e^{i\theta}|\phi\rangle$, $\cos\theta|\phi\rangle$, $\sin\theta|\phi\rangle$, $\langle 0|\phi\rangle$, $\langle 1|\phi\rangle$, and $\langle ba|\phi\rangle$

used in Scheme I are $\mathbf{0}$; thus, $|b\rangle \otimes \langle 1|\phi\rangle$ and $|b\rangle \otimes \langle 0|\phi\rangle$ are also $\mathbf{0}$ for each bit $b \in \{0,1\}$. Therefore, Condition 1 follows.

Among Schemata II–IV, let us consider Scheme IV since the other schemata are easily shown to meet Conditions 1–4. Let $g, h,$ and p be quantum functions in $\widehat{\square}_1^{\text{QP}}$. By induction hypothesis, we assume that $g, h,$ and p satisfy Conditions 1–4. For readability, we write f for $k\text{QRec}_t[g, h, p|\mathcal{F}_k]$. In what follows, we are focused only on Conditions 2 and 4 since the other conditions are easy to check. Our argument will employ induction on the length of input $|\phi\rangle$ given to f .

- (i) Our goal is to show that f satisfies Condition 2. First, consider the case of $\ell(|\phi\rangle) \leq t$. It then follows that $f(|\phi\rangle + |\xi\rangle) = g(|\phi\rangle + |\xi\rangle) = g(|\phi\rangle) + g(|\xi\rangle)$ since g is assumed to meet Condition 2. Next, we consider the case where $\ell(|\phi\rangle) > t$. From the definition of f , we obtain $f(|\phi\rangle + |\xi\rangle) = h(\sum_{s:|s|=k} |s\rangle \otimes f_s(\langle s|\psi_{p,\phi,\xi}\rangle))$, where $|\psi_{p,\phi,\xi}\rangle = p(|\phi\rangle + |\xi\rangle)$. Since $p(|\phi\rangle + |\xi\rangle) = p(|\phi\rangle) + p(|\xi\rangle)$ by induction hypothesis, we conclude that $\langle s|\psi_{p,\phi,\xi}\rangle = \langle s|\psi_{p,\phi}\rangle + \langle s|\psi_{p,\xi}\rangle$ for each string $s \in \{0,1\}^k$. It then follows by induction hypothesis that $f_s(\langle s|\psi_{p,\phi,\xi}\rangle) = f_s(\langle s|\psi_{p,\phi}\rangle) + f_s(\langle s|\psi_{p,\xi}\rangle)$, leading to the equation $|s\rangle \otimes f_s(\langle s|\psi_{p,\phi,\xi}\rangle) = |s\rangle \otimes f_s(\langle s|\psi_{p,\phi}\rangle) + |s\rangle \otimes f_s(\langle s|\psi_{p,\xi}\rangle)$. Using Condition 2 for h , we obtain $h(\sum_{s:|s|=k} |s\rangle \otimes f_s(\langle s|\psi_{p,\phi,\xi}\rangle)) = \sum_{s:|s|=k} h(|s\rangle \otimes f_s(\langle s|\psi_{p,\phi}\rangle)) + \sum_{s:|s|=k} h(|s\rangle \otimes f_s(\langle s|\psi_{p,\xi}\rangle))$. We then conclude that $f(|\phi\rangle + |\xi\rangle) = f(|\phi\rangle) + f(|\xi\rangle)$.
- (ii) We want to show that f satisfies Condition 4. By induction hypothesis, it follows that, for any $s \in \{0,1\}^k$, $\|f_s(\langle s|\psi_{p,\phi}\rangle)\| = \|\langle s|\psi_{p,\phi}\rangle\|$ and $\|h(|\phi\rangle)\| = \|\phi\|$. These equalities imply that $\|f(|\phi\rangle)\|^2 = \|h(\sum_{s:|s|=k} |s\rangle \otimes f_s(\langle s|\psi_{p,\phi}\rangle))\|^2 = \|\sum_{s:|s|=k} |s\rangle \otimes f_s(\langle s|\psi_{p,\phi}\rangle)\|^2 = \sum_{s:|s|=k} \|f_s(\langle s|\psi_{p,\phi}\rangle)\|^2$, which equals $\sum_{s:|s|=k} \|\langle s|\psi_{p,\phi}\rangle\|^2$. The last term coincides with $\|\psi_{p,\phi}\|^2$, which equals $\|p(|\phi\rangle)\|^2$. This implies Condition 4 because $\|p(|\phi\rangle)\| = \|\phi\|$.

Lemma 3.4(4) indicates that all quantum functions in $\widehat{\square}_1^{\text{QP}}$ also serve as functions mapping Φ_∞ to Φ_∞ in place of \mathcal{H}_∞ to \mathcal{H}_∞ .

Given a quantum function g that is dimension-preserving and norm-preserving, the inverse of g is a unique quantum function f satisfying the condition that, for every $|\phi\rangle \in \mathcal{H}_\infty$, $f \circ g(|\phi\rangle) = g \circ f(|\phi\rangle) = |\phi\rangle$. This special quantum function f is expressed as g^{-1} .

The next proposition guarantees that $\widehat{\square}_1^{\text{QP}}$ is closed under “inverse” since \square_1^{QP} lacks MEAS.

PROPOSITION 3.5. *For any quantum function $g \in \widehat{\square}_1^{\text{QP}}$, g^{-1} exists and belongs to $\widehat{\square}_1^{\text{QP}}$.*

PROOF. We prove this proposition by induction on the aforementioned *descriptive complexity* of g . If g is one of the initial quantum functions, then its inverse g^{-1} satisfies that $I^{-1} = I$, $\text{PHASE}_\theta^{-1} = \text{PHASE}_{-\theta}$, $\text{ROT}_\theta^{-1} = \text{ROT}_{-\theta}$, $\text{NOT}^{-1} = \text{NOT}$, and $\text{SWAP}^{-1} = \text{SWAP}$. If g is obtained from another quantum function or functions by one of the construction rules, then its inverse satisfies that $\text{Compo}[g, h]^{-1} = \text{Compo}[h^{-1}, g^{-1}]$, $\text{Branch}[g, h]^{-1} = \text{Branch}[g^{-1}, h^{-1}]$, and $k\text{QRec}_t[g, h, p|\{f_s\}_{s \in \{0,1\}^k}]^{-1} = k\text{QRec}_t[g^{-1}, p^{-1}, h^{-1}|\{f_s^{-1}\}_{s \in \{0,1\}^k}]$. −1

Our choice of Schemata I–IV is motivated by a particular universal set of quantum gates. Notice that a different choice of initial quantum functions and construction rules may lead to a different set of \square_1^{QP} -functions. Scheme I uses an “arbitrary” angle of θ in $[0, 2\pi) \cap \widehat{\mathbb{C}}$ to introduce $PHASE_\theta$ and ROT_θ ; however, we can restrict θ to a unique value of $\frac{\pi}{4}$ since $PHASE_\theta$ and ROT_θ for an arbitrary value $\theta \in [0, 2\pi)$ can be approximated with any desired accuracy using WH and $Z_{2, \frac{\pi}{4}}$. This last part comes from the fact that any single-qubit unitary operator can be approximated to any accuracy from the quantum gates WH and $Z_{2, \frac{\pi}{4}}$ [5] (see also [24]). For a further discussion on the choice of the schemata, see Section 6.1.

3.3. Construction of more complicated quantum functions. Before presenting the main theorem (Theorem 4.1), we wish to prepare useful quantum functions and new construction rules derived directly from Schemata I–IV. These quantum functions and construction rules will be used for the proof of our key lemma (Lemma 4.3), which supports the main theorem.

For each $k \in \mathbb{N}^+$, let us assume the standard lexicographic ordering $<$ on $\{0, 1\}^k$ and all elements in $\{0, 1\}^k$ are enumerated lexicographically as $s_1 < s_2 < \dots < s_{2^k}$. For example, when $k = 2$, we obtain $00 < 01 < 10 < 11$. Given each string $s \in \{0, 1\}^n$, s^R denotes the reversal of s ; that is, $s^R = s_n s_{n-1} \dots s_2 s_1$ if $s = s_1 s_2 \dots s_{n-1} s_n$. We expand this notion to quantum states in the following manner. Given a quantum state $|\phi\rangle \in \mathcal{H}_{2^n}$, the reversal of $|\phi\rangle$, denoted by $|\phi^R\rangle$, is of the form $\sum_{s:|s|=n} \langle s|\phi\rangle \otimes |s^R\rangle$, where $\langle s|\phi\rangle$ is merely a scalar. For instance, if $|\phi\rangle = \alpha|01\rangle + \beta|10\rangle$, then $|\phi^R\rangle = \alpha|10\rangle + \beta|01\rangle$.

LEMMA 3.6. *Let $k \in \mathbb{N}$ with $k \geq 2$, let $g, h \in \widehat{\square_1^{QP}}$, and let $\mathcal{G}_k = \{g_s \mid s \in \{0, 1\}^k\}$ be a set of $\widehat{\square_1^{QP}}$ -functions. The following quantum functions all belong to $\widehat{\square_1^{QP}}$. The lemma also holds even if $\widehat{\square_1^{QP}}$ is replaced by \square_1^{QP} . Let $|\phi\rangle$ be any quantum state in \mathcal{H}_∞ .*

1. $Compo[\mathcal{G}_k](|\phi\rangle) = g_{s_1} \circ g_{s_2} \circ \dots \circ g_{s_{2^k}}(|\phi\rangle)$. (multiple composition)
2. $Switch_k[g, h](|\phi\rangle) = g(|\phi\rangle)$ if $\ell(|\phi\rangle) < k$ and $Switch_k[g, h](|\phi\rangle) = h(|\phi\rangle)$ otherwise. (switching)
3. $LENGTH_k[g](|\phi\rangle) = |\phi\rangle$ if $\ell(|\phi\rangle) < k$ and $LENGTH_k[g](|\phi\rangle) = g(|\phi\rangle)$ otherwise.
4. $REMOVE_k(|\phi\rangle) = |\phi\rangle$ if $\ell(|\phi\rangle) < k$ and $REMOVE_k(|\phi\rangle) = \sum_{s:|s|=k} \langle s|\phi\rangle \otimes |s\rangle$ otherwise.
5. $REP_k(|\phi\rangle) = |\phi\rangle$ if $\ell(|\phi\rangle) < k$ and $REP_k(|\phi\rangle) = \sum_{s:|s|=n-k} \langle s|\phi\rangle \otimes |s\rangle$ otherwise.
6. $SWAP_k(|\phi\rangle) = |\phi\rangle$ if $\ell(|\phi\rangle) < 2k$ and $SWAP_k(|\phi\rangle) = \sum_{s:|s|=k} \sum_{t:|t|=k} |st\rangle \langle ts|\phi\rangle$ otherwise.
7. $REVERSE(|\phi\rangle) = |\phi^R\rangle$.
8. $Branch_k[\mathcal{G}_k](|\phi\rangle) = |\phi\rangle$ if $\ell(|\phi\rangle) < k$ and $Branch_k[\mathcal{G}_k](|\phi\rangle) = \sum_{s:|s|=k} |s\rangle \otimes g_s(\langle s|\phi\rangle)$ otherwise.
9. $RevBranch_k[\mathcal{G}_k](|\phi\rangle) = |\phi\rangle$ if $\ell(|\phi\rangle) < k$ and $RevBranch_k[\mathcal{G}_k](|\phi\rangle) = \sum_{s:|s|=k} g_s \left(\sum_{u:|u|=n-k} \langle us|\phi\rangle \otimes |u\rangle \right) \otimes |s\rangle$ otherwise, where $n = \ell(|\phi\rangle)$.

The difference between $REMOVE_k$ and REP_k is subtle but $REMOVE_k$ moves the first k qubits of an input to the end, whereas REP_k moves the last k qubits to the front. For basic qustrings of length 4, for instance, it holds that

$REP_1(|a_1a_2a_3a_4\rangle) = |a_4a_1a_2a_3\rangle$ and $REMOVE_1(|a_1a_2a_3a_4\rangle) = |a_2a_3a_4a_1\rangle$. Similarly, $Branch_k[\mathcal{G}_k]$ applies each g_s in \mathcal{G}_k to the quantum state obtained from $|\phi\rangle$ by eliminating the first k qubits whereas $RevBranch_k[\mathcal{G}_k]$ applies g_s to the quantum state obtained by eliminating the last k qubits, whenever $k \leq \ell(|\phi\rangle)$. For example, if $\mathcal{G}_k = \{h\}_{s \in \{0,1\}^k}$ for a single quantum function h , then, for every string $s \in \{0,1\}^k$, we obtain $Branch_k[\mathcal{G}_k](|s\rangle|\phi\rangle) = |s\rangle \otimes h(|\phi\rangle)$ and $RevBranch_k[\mathcal{G}_k](|\phi\rangle|s\rangle) = h(|\phi\rangle) \otimes |s\rangle$.

PROOF OF LEMMA 3.6. Let $k \in \mathbb{N}^+$, $g \in \widehat{\square}_1^{\text{QP}}$, and $\mathcal{G}_k = \{g_s \mid s \in \{0,1\}^k\} \subseteq \widehat{\square}_1^{\text{QP}}$. For each index $i \in [k]$, let s_i denote lexicographically the i th element of $\{0,1\}^k$.

- 1) We first set $f_{2^k} = g_{s_{2^k}}$ and inductively define $f_{i-1} = Compo[g_{s_{i-1}}, f_i]$ for every index $i \in [2, 2^k]_{\mathbb{Z}}$ to obtain $f_1 = Compo[\mathcal{G}_k]$. The resulted quantum function f_1 clearly belongs to $\widehat{\square}_1^{\text{QP}}$ because k is a fixed constant independent of inputs to f_1 .
- 2) This is a special case of the single-qubit quantum recursion rule (or the 1-qubit quantum recursion rule) in which $t = k - 1$, $p = I$, and $f_0 = f_1 = I$. Therefore, $Switch_k[g, h]$ belongs to $\widehat{\square}_1^{\text{QP}}$.
- 3) Since $LENGTH_k[g] = Switch_k[I, g]$, $LENGTH_k$ is in $\widehat{\square}_1^{\text{QP}}$.
- 4) We begin with the case of $k = 1$. The desired quantum function $REMOVE_1$ is defined as

$$REMOVE_1(|\phi\rangle) = \begin{cases} |\phi\rangle & \text{if } \ell(|\phi\rangle) \leq 1, \\ \sum_{a \in \{0,1\}} |a\rangle \otimes REMOVE_1(\langle a | \psi_{SWAP, \phi} \rangle) & \text{otherwise.} \end{cases}$$

More formally, we set $REMOVE_1 = QRec_1[I, I, SWAP | REMOVE_1, REMOVE_1]$. In the case of $k = 2$, we define $REMOVE_2 = REMOVE_1 \circ REMOVE_1$. For each index $k \geq 3$, $REMOVE_k$ is obtained as follows. Let h'_k be the k compositions of $REMOVE_1$ and define $REMOVE_k$ to be $LENGTH_k[h'_k]$. We remark that $LENGTH_k$ is necessary in our definition because h'_k is not guaranteed to equal $REMOVE_k$ when $\ell(|\phi\rangle) \leq k - 1$.

- 5) First, we define REP_1 as $REP_1 = QRec_1[I, SWAP, I | REP_1, REP_1]$, that is,

$$REP_1(|\phi\rangle) = \begin{cases} |\phi\rangle & \text{if } \ell(|\phi\rangle) \leq 1, \\ \sum_{a \in \{0,1\}} SWAP(|a\rangle \otimes REP_1(\langle a | \phi \rangle)) & \text{otherwise.} \end{cases}$$

Clearly, REP_1 belongs to $\widehat{\square}_1^{\text{QP}}$. For a general index $k > 1$, we define REP_k in the following way. We first set h'_k to be the k compositions of REP_1 . Finally, we set $REP_k = LENGTH_k[h'_k]$.

- 6) We first realize the quantum function $SWAP_{i,i+j}$, which swaps between the i th and the $(i + j)$ th qubits of any input. This goal is achieved by inductively constructing $SWAP_{i,i+j}$ in the following way. Initially, we set $SWAP_{i,i+1} = REP_{i-1} \circ SWAP \circ REMOVE_{i-1}$. For any index $j \in [k - i]$, we define $SWAP_{i,i+j} = SWAP_{i+j-1,i+j} \circ SWAP_{i,i+j-1} \circ SWAP_{i+j-1,i+j}$. We then set $g = SWAP_{k,2k} \circ SWAP_{k-1,2k-1} \circ \dots \circ SWAP_{2,k+2} \circ SWAP_{1,k+1}$. At last, it suffices to define $SWAP_k$ to be $LENGTH_{2k}[g]$.

7) The desired quantum function *REVERSE* can be defined as $REVERSE = QRec_1[I, REMOVE_1, I | REVERSE, REVERSE]$, namely,

$$REVERSE(|\phi\rangle) = \begin{cases} |\phi\rangle & \text{if } \ell(|\phi\rangle) \leq 1, \\ REMOVE_1(\sum_{a \in \{0,1\}} (|a\rangle \otimes REVERSE(|a|\phi\rangle))) & \text{otherwise.} \end{cases}$$

8) Note that, when $k = 2$, $Branch_k[\mathcal{G}_k]$ coincides with *Branch* and it thus belongs to $\widehat{\square}_1^{QP}$. Hereafter, we assume that $k \geq 3$. For each string $s \in \{0,1\}^k$, let $g_s^{(0)} = g_s$. For each index $i \in \mathbb{N}$ with $i < k$ and each string $s \in \{0,1\}^*$ with $|s| = k - i - 1$, we inductively define $g_s^{(i+1)}$ to be $Branch[g_{s0}^{(i)}, g_{s1}^{(i)}]$, that is,

$$g_s^{(i+1)}(|\phi\rangle) = \begin{cases} |\phi\rangle & \text{if } \ell(|\phi\rangle) \leq 1, \\ |0\rangle \otimes g_{s0}^{(i)}(|\phi\rangle) + |1\rangle \otimes g_{s1}^{(i)}(|\phi\rangle) & \text{otherwise.} \end{cases}$$

Finally, we set $Branch_k[\mathcal{G}_k] = g_\lambda^{(k)}$, where λ is the empty string.

9) Assuming $k \geq 2$, let $RevBranch_k[\{g_s\}_{s \in \{0,1\}^k}] = REMOVE_k \circ Branch_k[\{g_s\}_{s \in \{0,1\}^k}] \circ REP_k$. ←

The next lemma shows that we can extend any classical bijection on $\{0,1\}^k$ to its associated $\widehat{\square}_1^{QP}$ -function, which behaves in exactly the same way as the bijection does on the first k bits of its input.

LEMMA 3.7. *Let k be a constant in \mathbb{N}^+ . For any bijection f from $\{0,1\}^k$ to $\{0,1\}^k$, there exists a $\widehat{\square}_1^{QP}$ -function g_f such that, for any quantum state $|\phi\rangle \in \mathcal{H}_\infty$,*

$$g_f(|\phi\rangle) = \begin{cases} |\phi\rangle & \text{if } \ell(|\phi\rangle) \leq k - 1, \\ \sum_{s \in \{0,1\}^k} |f(s)\rangle \langle s | \phi \rangle & \text{otherwise.} \end{cases}$$

PROOF. Given a bijection f on $\{0,1\}^k$, it suffices to show the existence of $\widehat{\square}_1^{QP}$ -function h satisfying $h(|s\rangle|\phi\rangle) = |f(s)\rangle|\phi\rangle$ for any string $s \in \{0,1\}^k$ and any quantum state $|\phi\rangle \in \mathcal{H}_\infty$ since g_f is obtained from h simply by setting $g_f = LENGTH_k[h]$. Notice that, if $h \in \widehat{\square}_1^{QP}$, $h(\mathbf{0}) = \mathbf{0}$ makes $g_f(\mathbf{0})$ equal $\mathbf{0}$.

A bijection on $\{0,1\}^k$ is, in essence, a *permutation* on $\{s_1, s_2, \dots, s_{2^k}\}$, where each s_i is lexicographically the i th string in $\{0,1\}^k$, and thus it can be expressed as the multiplication of a finite number of *transpositions*, each of which swaps between two distinct numbers. This can be done by an application of the multiple composition of the corresponding $SWAP_{i,i+j}$, which is defined in the proof of Lemma 3.6(6). Therefore, h belongs to $\widehat{\square}_1^{QP}$. ←

Given a quantum state $|\phi\rangle$, it is possible to apply simultaneously a quantum function f to the first k qubits of $|\phi\rangle$ and another quantum function g to the rest.

LEMMA 3.8. *Given $f, g \in \widehat{\square}_1^{QP}$ and $k \in \mathbb{N}^+$, the quantum function $f^{\leq k} \otimes g$, which is defined by*

$$(f^{\leq k} \otimes g)(|\phi\rangle) = \begin{cases} f(|\phi\rangle) & \text{if } \ell(|\phi\rangle) \leq k, \\ \sum_{s \in \{0,1\}^k} f(|s\rangle) \otimes g(|s|\phi\rangle) & \text{otherwise,} \end{cases}$$

belongs to $\widehat{\square}_1^{QP}$.

PROOF. Given a quantum function f , we restrict the application of f to the last k qubits of any input $|\phi\rangle$ by setting $f' = QRec_k[f, I, I|f_0, f_1]$, where f_0 and f_1 are f' . It follows that $f'(|\phi\rangle|s\rangle) = |\phi\rangle \otimes f(|s\rangle)$ for any $|\phi\rangle \in \mathcal{H}_\infty$ and $s \in \{0, 1\}^k$.

Let $h = REP_k \circ f' \circ REMOVE_k$. This quantum function h satisfies that $h(|s\rangle|\phi\rangle) = f(|s\rangle) \otimes |\phi\rangle$ for any $s \in \{0, 1\}^k$. We then define $\mathcal{G}_k = \{g_s\}_{s \in \{0, 1\}^k}$ with $g_s = g$ for every string $s \in \{0, 1\}^k$ and set $g' = h \circ Branch_k[\mathcal{G}_k]$. It then follows that the desired quantum function $f^{\leq k} \otimes g$ equals $Switch_{k+1}[f, g']$. \dashv

Next, we present Lemma 3.9, which is useful for the proof of our key lemma (Lemma 4.3) in Section 5. The lemma allows us to skip, before applying a given quantum function, an arbitrary number of 0s until we read a fixed number of 1s.

LEMMA 3.9. *Let f be a quantum function in $\widehat{\square}_1^{QP}$ and let k be a constant in \mathbb{N}^+ . There exists a quantum function g in $\widehat{\square}_1^{QP}$ such that $g(|0^m 1^k\rangle \otimes |\phi\rangle) = |0^m 1^k\rangle \otimes f(|\phi\rangle)$ and $g(|0^{m+1}\rangle) = |0^{m+1}\rangle$ for any number $m \in \mathbb{N}$ and any quantum state $|\phi\rangle \in \mathcal{H}_\infty$. We write this g as $Skip[f]$. The lemma also holds when $\widehat{\square}_1^{QP}$ is replaced by \square_1^{QP} .*

PROOF. Let $k \geq 2$. Given a quantum function $f \in \widehat{\square}_1^{QP}$, we first expand f to f' so that $f'(|1^{k-1}\rangle|\phi\rangle) = |1^{k-1}\rangle \otimes f(|\phi\rangle)$ for any $|\phi\rangle \in \mathcal{H}_\infty$ and $f'(|0^{m+1}\rangle) = |0^{m+1}\rangle$ for any $m \in \mathbb{N}$. This quantum function f' can be obtained inductively as follows. We set $f_{k-1} = Branch[I, f]$, $f_i = Branch[I, f_{i+1}]$ for each $i \in [k-2]$, and finally define f' to be f_1 . When $k = 1$, we simply set $f' = f$. The desired quantum function g in the lemma must satisfy

$$g(|\phi\rangle) = \begin{cases} |\phi\rangle & \text{if } \ell(|\phi\rangle) \leq 1, \\ |0\rangle \otimes f'(g(|0|\phi\rangle)) + |1\rangle\langle 1|\phi\rangle & \text{otherwise.} \end{cases}$$

This g is formally defined as $g = QRec_1[I, Branch[f', I], I|g, I]$. This completes the proof. \dashv

Within our framework, it is possible to construct a ‘‘restricted’’ form of the quantum Fourier transform (QFT). Given a binary string $s = s_1 s_2 \dots s_k$ of length k with $s_i \in \{0, 1\}$, we denote by $num(s)$ the integer of the form $\sum_{i=1}^k s_i 2^{k-i}$. For instance, $num(011) = 1 \cdot 2^1 + 1 \cdot 2^0 = 5$ and $num(1010) = 1 \cdot 2^3 + 1 \cdot 2^1 = 10$. Moreover, let $\omega_k = e^{2\pi i/2^k}$, where $i = \sqrt{-1}$.

LEMMA 3.10. *Let k be any fixed constant in \mathbb{N}^+ . The following k -qubit quantum Fourier transform belongs to $\widehat{\square}_1^{QP}$. For any element $|\phi\rangle$ in \mathcal{H}_∞ , let*

$$F_k(|\phi\rangle) = \begin{cases} |\phi\rangle & \text{if } \ell(|\phi\rangle) < k, \\ \frac{1}{2^{k/2}} \sum_{t:|t|=k} \sum_{s:|s|=k} \omega_k^{num(s)num(t)} |s\rangle\langle t|\phi\rangle & \text{otherwise.} \end{cases}$$

PROOF. When $k = 1$, F_1 coincides with WH and therefore F_1 belongs to $\widehat{\square}_1^{QP}$ by Lemma 3.3. Next, assume that $k \geq 2$. It is known that, for any $x_1, x_2, \dots, x_k \in \{0, 1\}$,

$$F_k(|x_1 x_2 \dots x_k\rangle) = \frac{1}{2^{k/2}} (|0\rangle + \omega_1^{x_k} |1\rangle)(|0\rangle + \omega_1^{x_{k-1}} \omega_2^{x_k} |1\rangle) \dots (|0\rangle + \prod_{i=1}^k \omega_i^{x_i} |1\rangle). \quad (3.1)$$

For this fact and its proof, refer to, e.g., [24].

Let us recall the special quantum function $SWAP_{i,i+j}$ from the proof of Lemma 3.6(6), which swaps between the i th and the j th qubits. Using $CPHASE_\theta$, for any

index pair i, j with $i < j$, we define $CPHASE_{\theta}^{(i,j)}$ to be $SWAP_{1,i} \circ SWAP_{2,j} \circ CPHASE_{\theta} \circ SWAP_{2,j} \circ SWAP_{1,i}$, in which we apply $CPHASE_{\theta}$ to the i th and the j th qubits. We first want to construct $G_k^{(1)} = F_k \circ REVERSE$, which works similarly to F_k but takes $|\phi^R\rangle$ as an input instead. To achieve this goal, we define $\{G_j^{(i)}\}_{i,j \in \mathbb{N}^+}$ inductively as follows. Initially, we set $G_0^{(i)} = I$ for any $i \in \mathbb{N}^+$. Next, we define $G_1^{(i)}$ as $G_1^{(i)} = H$ if $i = 1$, and $G_1^{(i)} = REP_{i-1} \circ H \circ REMOVE_{i-1}$ otherwise. For any index $k \geq 2$, $G_k^{(i)}$ is defined to be $G_{k-1}^{(i)} \circ CPHASE_{\frac{\pi}{2^{k-1}}}^{(i,i+k-1)} \circ (G_{k-2}^{(i)})^{-1} \circ G_{k-1}^{(i+1)}$. It is not difficult to show that $G_k^{(1)}$ coincides with $F_k \circ REVERSE$ by Eqn. (1).

Since $G_k^{(1)} = F_k \circ REVERSE$, it suffices to define F_k as $G_k^{(1)} \circ REVERSE$. \dashv

A general form of QFT, in which k is not limited to a particular constant, will be discussed in Section 6.1 in connection to our choice of Schemata I–IV that form the function class \square_1^{QP} .

§4. Main contributions. In Section 3.1, we have introduced the \square^{QP_1} -functions and the $\widehat{\square_1^{QP}}$ -functions mapping \mathcal{H}_{∞} to \mathcal{H}_{∞} by applying Schemata I–IV for finitely many times. Our main theorem (Theorem 4.1) asserts that \square_1^{QP} can precisely characterize all functions in FBQP mapping $\{0,1\}^*$ to $\{0,1\}^*$, and therefore characterize all languages in BQP over $\{0,1\}$ by identifying languages with their corresponding characteristic functions. This theorem will be proven by using two key lemmas, Lemmas 4.2 and 4.3.

4.1. A new characterization of FBQP. Our goal is to demonstrate the power of \square_1^{QP} -functions (and thus $\widehat{\square_1^{QP}}$ -functions) by showing in Theorem 4.1 that \square_1^{QP} -functions (as well as $\widehat{\square_1^{QP}}$ -functions) precisely characterize FBQP-functions on $\{0,1\}^*$. For this purpose, unfortunately, there are two major difficulties to overcome.

The first difficulty arises in dealing with tape symbols of QTMs by qubits alone. Notice that QTMs working over nonbinary input alphabets are known to be simulated by QTMs taking the binary input alphabet $\{0,1\}$. Even if we successfully reduce the size of input alphabets down to 2, in fact, the simulation of such QTMs must require the proper handling of *nonbinary tape symbols*, in particular, the distinguished blank symbol. For our later convenience, we use “ b ” to denote the blank symbol, instead of #. In a similar vein, suppose that the outcome of a \square_1^{QP} -function f is composed of an important, meaningful portion and the other “garbage” portion, which is a remnant of the computation process. Since we use only qubits ($|0\rangle$ and $|1\rangle$) to express inputs and outputs, how can we distinguish between the meaningful portion and the garbage portion? In order to simulate QTMs by \square_1^{QP} -functions, we therefore need to “encode” all tape symbols into qubits.

This paper introduces the following simple coding scheme. We set $\hat{0} = 00$, $\hat{1} = 01$, and $\hat{b} = 10$. We also set $\hat{2} = 11$ and $\hat{3} = 10$ for later use. The input alphabet $\Sigma = \{0,1\}$ is thus translated into $\{\hat{0}, \hat{1}\}$ and the tape alphabet $\Gamma = \Sigma \cup \{b\}$ is encoded into $\{\hat{0}, \hat{1}, \hat{b}\}$. Given each binary string $s = s_1 s_2 \cdots s_n$ with $s_i \in \{0,1\}$ for every index $i \in [n]$, a *code* \tilde{s} of s indicates the string $\hat{s}_1 \hat{s}_2 \cdots \hat{s}_n \hat{2}$, where the last item $\hat{2}$ serves as an endmarker, which

marks the end of the code. It then follows that $|\tilde{s}| = 2|s| + 2$. As quick examples, we obtain $0\tilde{1}\tilde{1}0 = \hat{0}\hat{1}\hat{1}\hat{0}\hat{2} = 0001010011$ and $\tilde{0}\tilde{1} \cdot \tilde{1}\tilde{1} = \hat{0}\hat{1}\hat{2}\hat{1}\hat{1}\hat{2} = 000111010111$. Later, we will show in lemma 5.1 that the encoding of strings and the decoding of encoded strings can be carried out by suitable $\widehat{\square}_1^{\text{QP}}$ -functions.

Another difficulty comes from the inability of $\widehat{\square}_1^{\text{QP}}$ -functions to expand their qubits. Notice that a QTM is designed to freely use additional storage space by moving its tape head simply to new, blank tape cells beyond its *input area* in which an input is initially written. To simulate such a QTM, we need to simulate its entire activities made in the freely expanding tape space as well. On the contrary, every $\widehat{\square}_1^{\text{QP}}$ -function is dimension-preserving by Lemma 3.4(4), and thus the number of input qubits must match that of output qubits. If we want to simulate extra storage space of the QTM, then we need to feed the same amount of extra qubits to the target $\widehat{\square}_1^{\text{QP}}$ -function for use at the very beginning.

We resolve this second issue by extending each input of quantum functions by adding extra 0s whose length is associated with the running time of the QTM. For any polynomial p and any quantum function g on \mathcal{H}_∞ , we define $|\phi^p(x)\rangle = |0^{|x|}1\rangle|0^{p(|x|)}10^{11p(|x|)+6}1\rangle|x\rangle$ and $|\phi_g^p(x)\rangle = g(|\phi^p(x)\rangle)$ for every string $x \in \{0, 1\}^*$. Similarly, for any function f on $\{0, 1\}^*$, we set $|\phi^{p \cdot f}(x)\rangle = |0^{|f(x)|}1\rangle|0^{|f(x)|+1}1\rangle|\phi^p(x)\rangle$ and $|\phi_g^{p \cdot f}(x)\rangle = g(|\phi^{p \cdot f}(x)\rangle)$.

Any FBQP-function takes classical input strings and produces classical strings that are outcomes of a polynomial-time quantum Turing machine with high probability. In contrast, our $\widehat{\square}_1^{\text{QP}}$ -functions f are to transform each quantum state in \mathcal{H}_∞ to another one in \mathcal{H}_∞ . To obtain classical output strings, we need to observe the outcomes of f .

The main theorem (Theorem 4.1) roughly asserts the following: for any FBQP-function f , there always exists a $\widehat{\square}_1^{\text{QP}}$ -function g such that, when we observe the first $|f(x)|$ qubits of the outcome $g(|\phi^p(x)\rangle) (= |\phi_g^p(x)\rangle)$ of g on input $|\phi^p(x)\rangle$, we correctly obtain $f(x)$ with high probability. Notice that $g(|\phi^p(x)\rangle)$ also contains extra qubits, called “garbage” qubits, which are left unobserved in the process of calculating $f(x)$. Those garbage qubits are actually the remnant of the computation process of f . It is, however, possible to remove those qubits with high probability by partly reversing the whole computation, if we know the output size $|f(x)|$ ahead of the computation (by expanding $|\phi^p(x)\rangle$ to $|\phi^{p \cdot f}(x)\rangle$ and $|\phi_g^p(x)\rangle$ to $|\phi_g^{p \cdot f}(x)\rangle$).

THEOREM 4.1 (Main Theorem). *Let f be a polynomially-bounded function on $\{0, 1\}^*$. The following three statements are logically equivalent.*

1. *The function f is in FBQP.*
2. *For any constant $\varepsilon \in [0, 1/2)$, there exist a quantum function g in $\widehat{\square}_1^{\text{QP}}$ and a polynomial p such that $|f(x)| \leq p(|x|)$ and $\|\langle f(x) | \phi_g^p(x) \rangle\|^2 \geq 1 - \varepsilon$ for all $x \in \{0, 1\}^*$.*
3. *For any constant $\varepsilon \in [0, 1/2)$, there exist a quantum function g in $\widehat{\square}_1^{\text{QP}}$ and a polynomial p such that $|f(x)| \leq p(|x|)$ and $|\langle \Psi_{f(x)} | \phi_g^{p \cdot f}(x) \rangle|^2 \geq 1 - \varepsilon$ for all $x \in \{0, 1\}^*$, where $|\Psi_{f(x)}\rangle = |f(x)\rangle|0^{|f(x)|+1}1\rangle|\phi^p(x)\rangle$.*

In Statements 2–3 of Theorem 4.1, $\langle f(x)|\phi_g^p(x)\rangle$ is a non-null vector whereas $\langle \Psi_{f(x)}|\phi_g^{p,f}(x)\rangle$ is just a scalar because $\ell(|f(x)\rangle|(0^{|f(x)|+1}1)^2)|\phi^p(x)\rangle = \ell(|\phi_g^{p,f}(x)\rangle)$.

Hereafter, we wish to prove the main theorem, Theorem 4.1. For a strategic reason, we split the theorem into two technical lemmas, Lemmas 4.2 and 4.3. To present the lemmas, we need to introduce additional terminology for QTMs. It is easier in practice to design multitape well-formed QTMs rather than single-tape ones. However, since multitape QTMs can be simulated by single-tape QTMs by translating multiple tapes to multiple tracks of a single tape, toward the proof of Theorem 4.1, it suffices to focus our attention only on single-tape QTMs.

A single-tape QTM is said to be in *normal form* if $\delta(q_f, \sigma) = |q_0\rangle|\sigma\rangle|R$ holds for any tape symbol $\sigma \in \Gamma$. If a QTM halts in a superposition of final configurations in which a tape head returns to the start cell, then we call such a machine *stationary*. Refer to [4] for their basic properties. For convenience, we further call a QTM *conservative* if it is well-formed, stationary, and in normal form. Moreover, a well-formed QTM is said to be *plain* if its transition function satisfies the following specific requirement: for every pair $(p, \sigma) \in Q \times \Gamma$, $\delta(p, \sigma)$ has the form either $\delta(p, \sigma) = e^{i\theta}|q, \tau, d\rangle$ or $\delta(p, \sigma) = \cos\theta|q, \tau, d\rangle + \sin\theta|q', \tau', d'\rangle$ for certain $\theta \in [0, 2\pi)$ and two distinct tuples (q, τ, d) and (q', τ', d') . Bernstein and Vazirani [4] claimed that any single-tape, polynomial-time, conservative QTM M can be simulated by an appropriate single-tape, polynomial-time, conservative, plain QTM M' . Additionally, when M is of \mathbb{C} -amplitudes, so is M' .

Let us recall that any output string of a QTM begins at the start cell and stretches to the right until the first blank symbol although there may be tape symbols left unerased in other parts of the tape. For our later convenience, a QTM M is said to have *clean outputs* if, when M halts, no nonblank symbol appears in the left-side region of the output string, i.e., the region consisting of all cells indexed by negative integers. Take a polynomial p that bounds the running time of M on every input. Since M halts in at most $p(|x|)$ steps on every instance $x \in \{0, 1\}^*$, it suffices for us to pay attention only to its *essential tape region* that covers all tape cells indexed by numbers between $-p(|x|)$ and $+p(|x|)$. In practice, we redefine a configuration γ of M on input x of length n as a triplet $(q, h, \sigma_1 \cdots \sigma_{2p(n)+1})$, where $p \in Q$, $h \in \mathbb{Z}$ with $-p(n) \leq h \leq p(n)$, and $\sigma_1, \dots, \sigma_{2p(n)+1} \in \{0, 1, b\}$ such that, for every index $i \in [2p(n) + 1]$, σ_i is a tape symbol written at the cell indexed by $i - p(n) - 1$. Note that the start cell comes in the middle of $\sigma_1 \cdots \sigma_{2p(n)+1}$. For notational convenience, we further modify the notion of configuration by splitting the essential tape region into two parts (z_1, z_2) , in which $z_1 = \sigma_1 \sigma_2 \cdots \sigma_{p(n)}$ refers to the left-side region of the start cell, not including the start cell, and $z_2 = \sigma_{p(n)+1} \sigma_{p(n)+2} \cdots \sigma_{2p(n)+1}$ refers to the rest of the essential tape region. This provides us with a modified configuration of the form $(q, h, z_1 z_2)$. For a practical reason, we further alter it into (z_2, z_1, h, q) , which we call by a *skew configuration*. Associated with this alteration of configurations, we also modify the original time-evolution operator, U_δ , of M so that it works on skew configurations. Be aware that this new operator cannot be realized by the standard QTM model. To distinguish it from the original time-evaluation operator U_δ of M , we call it the *skew time-evolution operator* and write it as \widehat{U}_δ .

LEMMA 4.2. *Let f be any quantum function in \square_1^{QP} . There exist a polynomial p and a single-tape, conservative, plain QTM M producing clean outputs with \tilde{C} -amplitudes such that, for any quantum state $|\phi\rangle$ in \mathcal{H}_{2^n} , M starts with a non-null quantum state $|\phi\rangle$ given on its input/work tape and, when it halts after $p(n)$ steps, the superposition $|\eta_{M,\phi}\rangle$ of skew final configurations of M on $|\phi\rangle$ is of the form $f(|\phi\rangle) \otimes |0, q_f\rangle$, where $f(|\phi\rangle)$ appears as the content of the tape from the start cell to the right until the first blank symbol and q_f is a unique final inner state of M .*

PROOF. We first show that all the initial functions in Scheme I can be exactly computed in polynomial time on appropriate single-tape, \tilde{C} -amplitude, conservative, plain QTMs having clean outputs over the input/output alphabet $\{0, 1\}$. For clarity, our goal here is to demonstrate that, for every quantum function f in Scheme I, there exists a QTM with the lemma’s properties such that a superposition $|\eta_{M,\phi}\rangle$ of M ’s skew final configurations is of the form $f(|\phi\rangle) \otimes |0, q_f\rangle$, where $f(|\phi\rangle)$ is the content of M ’s tape from the start cell to the right until the first blank symbol.

Since I (identity) is easy to simulate, let us consider $PHASE_\theta$. In this case, we take a QTM that applies a transition of the form $\delta(q_0, \sigma) = e^{i\theta\sigma} |q_f, \sigma, N\rangle$ for any bit $\sigma \in \{0, 1\}$. Clearly, if we start with a skew initial configuration $|\phi\rangle|0\rangle|q_0\rangle$, then we halt with $PHASE_\theta(|\phi\rangle) \otimes |0\rangle|q_f\rangle$; in short, this QTM “exactly computes” $PHASE_\theta$. For ROT_θ , we use the transition defined by $\delta(q_0, \sigma) = \cos\theta |q_f, \sigma, N\rangle + (-1)^\sigma \sin\theta |q_f, \bar{\sigma}, N\rangle$ to exactly compute ROT_θ . To simulate NOT , we then define a QTM to have a transition of $\delta(q_0, \sigma) = |q_f, \bar{\sigma}, N\rangle$. In the case of $SWAP$, it suffices to prepare inner states p_{σ_1} and r_{σ_2} as well as the following transitions: $\delta(q_0, \sigma_1) = |p_{\sigma_1}, \#, R\rangle$, $\delta(p_{\sigma_1}, \sigma_2) = |r_{\sigma_2}, \sigma_1, L\rangle$, and $\delta(r_{\sigma_2}, \#) = |q_f, \sigma_2, N\rangle$ for bits $\sigma_1, \sigma_2 \in \{0, 1\}$. Concerning $MEAS[a](|\phi\rangle)$, we start with checking the first qubit of $|\phi\rangle$. If it is not a , then we make a QTM *reject* the input; otherwise, we do nothing. More formally, we define $\delta(q_0, \bar{a}) = |q_{rej}, \bar{a}, N\rangle$ and $\delta(q_0, a) = |q_f, a, N\rangle$. It is not difficult to see that $|\eta_{M,\phi}\rangle$ has the form $f(|\phi\rangle) \otimes |0, q_0\rangle$.

Next, we intend to simulate each of the construction rules on an appropriate QTM. By induction hypothesis, there exist three polynomial-time, single-tape, \tilde{C} -amplitude, conservative, plain QTMs M_g , M_h , and M_p that satisfy the lemma for g , h , and p , respectively. By installing an *internal clock* in an appropriate way with a certain polynomial r , we can make M_g , M_h , and M_p halt in exactly $r(n)$ time on any input of length $n \in \mathbb{N}$. In what follows, let $|\phi\rangle$ denote any input in \mathcal{H}_∞ .

[Composition] Consider the case of $f = Compo[g, h]$. We compute f as follows. We first run M_h on $|\phi\rangle$ and obtain a superposition of skew final configurations, say, $h(|\phi\rangle) \otimes |0, q'_f\rangle$ for a unique final inner state q'_f of M_h . Since M_h is stationary and in normal form, we can further run M_g on the resulted quantum state $h(|\phi\rangle)$, treating q'_f as its new initial inner state and generating $|\eta_{M,\phi}\rangle = g(h(|\phi\rangle)) \otimes |0, q_f\rangle$ for a unique final inner state q_f of M_g .

[Branching] Assume that $f = Branch[g, h]$. We check the first qubit of $|\phi\rangle$. If it is 0, then we run M_g on $\langle 0|\phi\rangle$; otherwise, we run M_h on $\langle 1|\phi\rangle$. This produces $|0\rangle \otimes g(\langle 0|\phi\rangle) \otimes |0, q_{g,f}\rangle + |1\rangle \otimes h(\langle 1|\phi\rangle) \otimes |0, q_{h,f}\rangle$, where $q_{g,f}$ and $q_{h,f}$ are respectively unique final inner states of M_g and M_h . Notice that we do not need to check whether $\ell(|\phi\rangle) \geq 2$. Formally, we add the following transitions to those of M_g and M_h : $\delta(q_0, 0) = |q_0, \#_g, R\rangle$ and $\delta(q_0, 1) = |q_0, \#_h, R\rangle$, where $\#_g$ and $\#_h$ are designated symbols marking the left of the new “start cells” for M_g and M_h , respectively. At

terminating, the tape head moves back to the start cell. We then replace $\#_g$ and $\#_h$ respectively with 0 and 1 by entering a new final inner state q_f ; namely, $\delta(q_{g,f}, \#_g) = |q_f, 0, N\rangle$ and $\delta(q_{h,f}, \#_h) = |q_f, 1, N\rangle$. This transition is unitarily possible.

[Multiqubit quantum recursion] Finally, let us demonstrate a simulation of the multiqubit quantum recursion introduced by $f = kQRec_t[g, h, p | \mathcal{F}_k]$ with $\mathcal{F}_k = \{f_s\}_{s \in \{0,1\}^k}$. For readability, let us consider only the simplest case where $k = 1$, $f_0 = f$, and $f_1 = I$. The other cases can be similarly treated. Consider the “multitape” QTM M_f that roughly behaves as described below. Let T be a sufficiently large positive constant.

- (1) In this initial phase, starting with input $|\phi\rangle$, we prepare a counter on a new work tape and an internal clock on another work tape. We use the clock to adjust the terminating timing of all computation paths. Count the number $\ell(|\phi\rangle)$ of qubits simply by incrementing the counter as moving the input tape head from the start cell to the right. Initially, we set the current quantum state, say, $|\xi\rangle$ expressed on the tape to be $|\phi\rangle$ and set the current counter k to be $\ell(|\xi\rangle)$ ($= n$). Go to the splitting phase.
- (2) In this splitting phase, we inductively perform the following procedure using the clock. (*) Assume that the input tape currently contains a quantum state $|\xi\rangle$ and the counter has $k = \ell(|\xi\rangle)$. If $k \leq t$, then idle until the clock hits T and then go to the processing phase. Otherwise, run M_p on $|\xi\rangle$ to generate $|\psi_{p,\xi}\rangle$ and observe the first qubit of $|\psi_{p,\xi}\rangle$ in the computational basis, obtaining $\langle b | \psi_{p,\xi} \rangle$ for each $b \in \{0,1\}$. If b is 1, then run M_h on $|1\rangle\langle 1 | \psi_{p,\xi} \rangle$, obtain $h(|1\rangle\langle 1 | \psi_{p,\xi} \rangle)$, which is viewed as $f(|1\rangle\langle 1 | \xi \rangle)$, idle until the clock hits T , and then start the processing phase. On the contrary, when b is 0, move this bit 0 to a separate tape to remember and then update both $|\xi\rangle$ and k to be $\langle 0 | \psi_{p,\xi} \rangle$ and $k - 1$, respectively. Continue (*).
- (3) In this processing phase, we start with a quantum state $|\xi\rangle$, which is produced in the splitting phase. Let $k = \ell(|\xi\rangle)$. We inductively perform the following procedure. (**) If $k \leq t$, then we run M_g on the input $|\xi\rangle$ and produce $g(|\xi\rangle)$, which is viewed as $f(|\xi\rangle)$. Update $|\xi\rangle$ to be the resulted quantum state. Otherwise, we move back the last stored bit 0 from the separate tape, run M_h on $|0\rangle|\xi\rangle$, obtain $h(|0\rangle|\xi\rangle)$, and then update $|\xi\rangle$ to be the obtained quantum state and k to be $k + 1$ since $\ell(|0\rangle|\xi\rangle) = k + 1$. If all b 's are consumed (equivalently, $k = n$), then idle until the clock hits $2T$, output $|\xi\rangle$, and halt. Otherwise, continue (**).

The running time of the above QTM is bounded from above by a certain polynomial in the length $\ell(|\phi\rangle)$ because each of the procedures (*) and (**) is repeated for at most $\ell(|\phi\rangle)$ times. Although M_f stores bits on the separate tape, those bits are all moved back and used up by the end of the computation. This fact shows that a superposition of M_f 's skew final configurations is of the form $f(|\phi\rangle) \otimes |0, q_f\rangle$ for a unique final inner state q_f . Finally, we convert this multitape QTM into a computationally equivalent single-tape QTM of the desired properties in the lemma.

This completes the proof of Lemma 4.2. ⊣

For the notational sake, we write $M[r]$ for an output string written in a skew final configuration r , which covers only an essential tape region of M . We write $FSC_{M,n}$

to denote the set of all possible skew final configurations of M produced for any input of length n .

The key to the proof of Theorem 4.1 is the following lemma, which ensures the existence of a \square_1^{QP} -function g whose outcome $g(|\phi^p(|\psi\rangle)\rangle) (= |\phi_g^p(|\psi\rangle)\rangle)$ “almost” characterize the encoded skew final configuration $\sum_{x:|x|=n} \sum_{r \in FSC_{M,n}} \langle x|\psi\rangle |\widetilde{M}[r]\rangle |\xi_{x,r}\rangle$ of M , where $\widetilde{M}[r]$ is the encoding of $M[r]$. Here, the encoding $\widetilde{M}[r]$ is needed for the construction of g in the proof of the lemma. However, as shown also in the proof, executing an extra decoding procedure for $\widetilde{M}[r]$ allows us to replace $\widetilde{M}[r]$ by $M[r]$.

LEMMA 4.3 (Key Lemma). *Let M be a single-tape, polynomial-time, \tilde{C} -amplitude, conservative, plain QTM having clean outputs over the input/output alphabet $\Sigma = \{0,1\}$ and the tape alphabet $\Gamma = \{0,1,b\}$. Assume that, when M halts on input $|\psi\rangle$ of length n , a superposition of coded skew final configurations is of the form $\sum_{x:|x|=n} \sum_{r \in FSC_{M,n}} \langle x|\psi\rangle |\widetilde{M}[r]\rangle |\xi_{x,r}\rangle$, where $|\xi_{x,r}\rangle$ denotes an appropriate quantum state describing the rest of the coded skew final configurations other than $\widetilde{M}[r]$. There exist a quantum function g in \square_1^{QP} and a polynomial p such that, for any number $n \in \mathbb{N}^+$ and every quantum state $|\psi\rangle \in \mathcal{H}_{2^n}$, $|\phi_g^p(|\psi\rangle)\rangle$ has the form $\sum_{x:|x|=n} \sum_{r \in FSC_{M,n}} \langle x|\psi\rangle |\widetilde{M}[r]\rangle |\widehat{\xi}_{x,r}\rangle$ for certain quantum states $\{|\widehat{\xi}_{x,r}\rangle\}_{x,r}$ satisfying that, for any $x, x' \in \{0,1\}^n$ and $r, r' \in FSC_{M,n}$, (i) $\|\langle \widehat{\xi}_{x,r} | \widehat{\xi}_{x',r'} \rangle\| = \|\langle \widehat{\xi}_{x',r'} | \widehat{\xi}_{x,r} \rangle\|$ and (ii) $\langle \widehat{\xi}_{x,r} | \widehat{\xi}_{x',r'} \rangle = 0$ if $r \neq r'$. Furthermore, it is possible to modify g to g' , which satisfies $|\phi_{g'}^p(|\phi\rangle)\rangle = \sum_{x:|x|=n} \sum_{r \in FSC_{M,n}} \langle x|\psi\rangle |M[r]\rangle |\widehat{\xi}_{x,r}^t\rangle$ for appropriate quantum states $\{|\widehat{\xi}_{x',r'}^t\rangle\}_{x',r'}$ satisfying Conditions (i)–(ii).*

The proof of Lemma 4.3 is lengthy and it is postponed until Section 5. Meanwhile, we return to Theorem 4.1 and present its proof using Lemmas 4.2 and 4.3.

PROOF OF THEOREM 4.1. Let $\varepsilon \in [0,1/2)$ be any constant and let f be any polynomially-bounded function mapping $\{0,1\}^*$ to $\{0,1\}^*$.

(1 \Rightarrow 2) Assume that f is in FBQP. Take a multitape, polynomial-time, \tilde{C} -amplitude, well-formed QTM N that computes f with bounded-error probability. Let us choose a polynomial p that bounds the running time of N on every input. It is possible to “simulate” N with high success probability by a certain single-tape, \tilde{C} -amplitude, conservative, plain QTM, say, M having clean outputs in such a way that the machine takes input x and terminates with generating $f(x)bw$ in the right-side region of the start cell of the input/work tape with bounded-error probability, where b is a unique blank tape symbol. Since any bounded-error QTM freely amplifies its success probability, we assume without loss of generality that the error probability of M is at most ε . Notice that the coded skew final configuration begins with an output string. Thus, the superposition of coded skew final configurations of M on input x of length n must be of the form $\sum_{r \in FSC_{M,n}} |\widetilde{M}[r]\rangle |\xi_{x,r}\rangle$ for appropriately chosen quantum states $\{|\xi_{x,r}\rangle\}_{r \in FSC_{M,n}}$. Since M computes f with error probability at most ε , we conclude that $\sum_{r \in FSC_{M,n}} \|\langle f(x) | \widetilde{M}[r]\rangle |\xi_{x,r}\rangle\|^2 \geq 1 - \varepsilon$ for all x . Lemma 4.3 further provides us with a special quantum function $g \in \square_1^{\text{QP}}$ such that, for any number $n \in \mathbb{N}$ and any string $x \in \{0,1\}^n$, $|\phi_g^p(x)\rangle$ has the form

$\sum_{r \in FSC_{M,n}} |M[r]\rangle |\widehat{\xi}_{x,r}\rangle$ for certain quantum states $\{|\widehat{\xi}_{x,r}\rangle\}_{r \in FSC_{M,n}}$ satisfying that, for all $r, r' \in FSC_{M,n}$, $\|\xi_{x,r}\rangle\| = \|\widehat{\xi}_{x,r}\rangle\|$ and $\langle \widehat{\xi}_{x,r} | \widehat{\xi}_{x,r'} \rangle = 0$ if $r \neq r'$. We thus conclude that $\|\langle f(x) | \phi_g^p(x) \rangle\|^2 = \sum_{r \in FSC_{M,n}} \|\langle f(x) | M[r] \rangle |\widehat{\xi}_{x,r}\rangle\|^2 = \sum_{r \in FSC_{M,n}} |\langle f(x) | \widehat{M}[r] \rangle|^2 \|\widehat{\xi}_{x,r}\rangle\|^2$ since $\langle f(x) | M[r] \rangle = \langle f(x) | \widehat{M}[r] \rangle$ and $\langle \widehat{\xi}_{x,r} | \widehat{\xi}_{x,r'} \rangle = 0$ if $r \neq r'$. Moreover, from $\|\xi_{x,r}\rangle\| = \|\widehat{\xi}_{x,r}\rangle\|$, it follows that the term $|\langle f(x) | \widehat{M}[r] \rangle|^2 \|\widehat{\xi}_{x,r}\rangle\|^2$ equals $\|\langle f(x) | \widehat{M}[r] \rangle |\xi_{x,r}\rangle\|^2$. Therefore, $\|\langle f(x) | \phi_g^p(x) \rangle\|^2$ is at least $1 - \varepsilon$.

(2 \Rightarrow 3) Let ε be any constant in $[0, 1/2)$ and set $\varepsilon' = 1 - \sqrt{1 - \varepsilon}$. Let us choose a polynomial p and a function $g \in \square_1^{\text{QP}}$ for which $|f(x)| \leq p(|x|)$ and $\|\langle f(x) | \phi_g^p(x) \rangle\|^2 \geq 1 - \varepsilon'$ for all strings $x \in \{0, 1\}^*$. Starting with an input $|\phi^{p,f}(x)\rangle (= |0^{|f(x)}\rangle |0^{|f(x)+1}\rangle |1\rangle |\phi^p(x)\rangle)$, we first apply g to the last part $|\phi^p(x)\rangle$ of $|\phi^{p,f}(x)\rangle$ and obtain $|0^{|f(x)}\rangle |0^{|f(x)+1}\rangle |1\rangle |\phi_g^p(x)\rangle$. Next, we apply Lemma 5.1 to encode the first $|f(x)|$ qubits of $|\phi_g^p(x)\rangle$ with the help of $|0^{|f(x)+1}\rangle$ and then obtain $|\eta_{f,x}\rangle = \sum_{s:|s|=|f(x)|} |0^{|f(x)}\rangle |\tilde{s}\rangle \otimes \langle s | \phi_g^p(x) \rangle$. Using the quantum function $COPY_2$ given in Lemma 5.2, we then copy each qustring $|\tilde{s}\rangle$ of $|\eta_{f,x}\rangle$ into $|0^{|f(x)}\rangle$ and generate a quantum state of the form $\sum_{s:|s|=|f(x)|} (|\tilde{s}\rangle |\tilde{s}\rangle \otimes \langle s | \phi_g^p(x) \rangle)$. We then decode the first and the second occurrences of $|\tilde{s}\rangle$ into $|0^{|f(x)+1}\rangle |s\rangle$. For later convenience, we transform the first occurrence of $|0^{|f(x)+1}\rangle |s\rangle$ to $|s\rangle |0^{|f(x)+1}\rangle$. In what follows, we abbreviate $|s\rangle \langle s | \phi_g^p(x) \rangle$ as $|\zeta_g^p[s]\rangle$. Finally, we locally apply g^{-1} to this part $|\zeta_g^p[s]\rangle$, producing $|\xi_x\rangle = \sum_{s:|s|=|f(x)|} (|s\rangle |0^{|f(x)+1}\rangle \otimes g^{-1}(|\zeta_g^p[s]\rangle))$.

Let $|\Psi_{f(x)}\rangle = |f(x)\rangle |0^{|f(x)+1}\rangle |\phi^p(x)\rangle$. Since $|\phi_g^p(x)\rangle = \sum_{s:|s|=|f(x)|} |\zeta_g^p[s]\rangle$ and $g^{-1}(|\phi_g^p(x)\rangle) = |\phi^p(x)\rangle$, we derive $|\phi^p(x)\rangle = \sum_{s:|s|=|f(x)|} g^{-1}(|\zeta_g^p[s]\rangle)$. Therefore, $|\Psi_{f(x)}\rangle$ coincides with $\sum_{s:|s|=|f(x)|} (|f(x)\rangle |0^{|f(x)+1}\rangle \otimes g^{-1}(|\zeta_g^p[s]\rangle))$. Let us consider the inner product $\langle \Psi_{f(x)} | \xi_x \rangle$. By a simple calculation, $\langle \Psi_{f(x)} | \xi_x \rangle$ equals $\sum_{s:|s|=|f(x)|} \langle f(x) | s \rangle \otimes \tau_x(s)$, where $\tau_x(s)$ is the inner product between $|\phi^p(x)\rangle$ and $g^{-1}(|\zeta_g^p[s]\rangle)$. Since g^{-1} belongs to \square_1^{QP} and is dimension-preserving and norm-preserving by Proposition 3.5 and Lemma 3.4, $\tau_x(s)$ equals $\langle \zeta_g^p[s] | \zeta_g^p[s] \rangle$. Since s is forced to take the value $f(x)$ in $\langle \Psi_{f(x)} | \xi_x \rangle$, it follows that $\langle \Psi_{f(x)} | \xi_x \rangle = \langle \zeta_g^p[f(x)] | \zeta_g^p[f(x)] \rangle$, which equals $\|\langle f(x) | \phi_g^p(x) \rangle\|^2$. Therefore, we conclude that $|\langle \Psi_{f(x)} | \xi_x \rangle|^2 = \|\langle f(x) | \phi_g^p(x) \rangle\|^4 \geq (1 - \varepsilon')^2 = 1 - \varepsilon$, as requested.

(3 \Rightarrow 1) Since f is polynomially bounded, take a polynomial p such that $|f(x)| \leq p(|x|)$ holds for all strings x . Since we do not know the length of $f(x)$, we want to expand f by setting $f_1(x) = f(x)0^{1+p(|x|)+2-|f(x)|}$ so that $|f_1(x)| = p(|x|) + 2$ for all strings x . Fix $\varepsilon \in [0, 1/2)$. Let us assume that there exist a function $g_1 \in \square_1^{\text{QP}}$ and a polynomial p_1 that satisfy $|f_1(x)| \leq p_1(|x|)$ and $|\langle \Psi_{f_1(x)} | \phi_{g_1}^{p_1, f_1}(x) \rangle|^2 \geq 1 - \varepsilon$ for all instances $x \in \{0, 1\}^*$.

Using Lemma 4.2 for the quantum function g_1 , we can take a single-tape, polynomial-time, conservative, plain QTM M with \mathbb{C} -amplitudes for which M on input $|\phi\rangle$ produces a clean output of $g_1(|\phi\rangle)$ on its tape. We consider the following machine. On input $x \in \Sigma^*$, we first compute the value $p(|x|)$ deterministically and generate $|\phi^{p,f_1}\rangle = |0^{3p(|x|)+7}\rangle |0^{|x|}\rangle |0^{p(|x|)}\rangle |10^{11p(|x|)+6}\rangle \otimes |x\rangle$. We then run M

on $|\phi^{p,f_1}(x)\rangle$ to produce $|\phi_{g_1}^{p,f_1}(x)\rangle$. Since $|\langle\Psi_{f_1(x)}|\phi_{g_1}^{p,f_1}(x)\rangle|^2 \geq 1 - \varepsilon$, $|\phi_{g_1}^{p,f_1}(x)\rangle$ contains $f_1(x)$ with probability at least $1 - \varepsilon$. From $f_1(x)$, we extract $f(x)$ and output it. This concludes that f is in *FBQP*. \dashv

4.2. Quantum normal form theorem. Our key lemmas, Lemmas 4.2 and 4.3, further lead to a quantum version of Kleene’s *normal form theorem* [18, 19], which asserts the existence of a primitive recursive predicate $T(e,x,y)$ and a primitive recursive function $U(y)$ such that, for any recursive function $f(x)$, an appropriate index (called a Gödel number) $e \in \mathbb{N}$ satisfies $f(x) = U(\mu y.T(e,x,y))$ for all inputs $x \in \mathbb{N}$, where μ is the *minimization operator*. This statement is, in essence, equivalent to the existence of universal Turing machine [31]. Here, we wish to prove a slightly weaker form of the *quantum normal form theorem* using Lemmas 4.2 and 4.3.

THEOREM 4.4 (Quantum Normal Form Theorem). *There exists a quantum function f in \square_1^{QP} such that, for any quantum function g in \square_1^{QP} and any constant $\varepsilon \in (0, 1/2)$, there exist a binary string e and a polynomial p satisfying $\|\psi_{g,x}\langle\psi_{g,x}| - \text{tr}_n(|\eta_{f,x}\rangle\langle\eta_{f,x}|)\|_{\text{tr}} \leq \varepsilon$ for any input $|x\rangle$ with $x \in \{0,1\}^n$, where $|\psi_{g,x}\rangle = g(|x\rangle)$ and $|\eta_{f,x}\rangle = f(|\tilde{e}\rangle|0^{p(|x|)}1\rangle|x\rangle)$. Such a function f is called universal.*

The extra term $|0^{p(|x|)}1\rangle$ in $|\eta_{f,x}\rangle$ is needed for providing g with enough work space as in the case of Theorem 4.1. To prove the theorem, we utilize the fact that there is a *universal QTM*, which can simulate all single-tape well-formed QTMs with polynomial slowdown with any desired accuracy. Such a universal machine was constructed by Bernstein and Vazirani [4, Theorem 7.1] and by Nishimura and Ozawa [25, Theorem 4.1]. We say that M_1 on input x_1 *simulates* M_2 on input x_2 with accuracy at most ε if the total variation distance between two probability distributions $\{\|\langle y|U_{M_1}^{p_1(|x_1|)}|c_{0,1}^{(x_1)}\rangle\|^2\}_{y \in \{0,1\}^{\leq n}}$ and $\{\|\langle y|U_{M_2}^{p_2(|x_2|)}|c_{0,2}^{(x_2)}\rangle\|^2\}_{y \in \{0,1\}^{\leq n}}$ is at most ε , where $n = \max\{p_1(|x_1|), p_2(|x_2|)\}$ and, for each index $i \in \{1, 2\}$, U_{M_i} is the time-evolution operator of M_i , $p_i(\cdot)$ expresses the running time of M_i , $c_{0,i}^{(x_i)}$ is the skew initial configuration of M_i on input x_i , and y ranges over all possible output strings of M_i .

PROPOSITION 4.5 [4, 16, 24, 25]. *There exists a single-tape, well-formed, stationary QTM U such that, for every constant $\varepsilon \in (0, 1)$, a number $t \in \mathbb{N}$, a single-tape well-formed QTM M with \tilde{C} -amplitudes, U on input $\langle M, x, t, \varepsilon \rangle$ simulates M on input x for t steps with accuracy at most ε with slowdown of a polynomial in t and $\log(1/\varepsilon)$, where $\langle M, x, t, \varepsilon \rangle$ refers to a fixed, efficient encoding of a quadruplet (M, x, t, ε) . Such a QTM is called universal.*

The improved factor $\log(1/\varepsilon)$ in Proposition 4.5 is attributed to Kitaev [16] and Solovay (cited in [24, Appendix 3]).

For the proof of Theorem 4.4, we need to simulate a universal QTM U provided by Proposition 4.5 on a certain conservative QTM even with lower accuracy. Concerning the form of inputs given to f , we need to split a quadruplet (M, x, t, ε) in the proposition into three parts (M, ε) , t , and x and then modify U so that U can take any input of the form $|\tilde{e}\rangle|0^t1\rangle|x\rangle$, where $e = \langle M, \varepsilon \rangle$, and mimic M on x within time t with accuracy at most ε . Furthermore, we need to force U to produce *clean outputs* by relocating all nonblank symbols appearing in the left-side region of any output string to elsewhere in time polynomial in t .

Theorem 4.4 now follows directly by combining Lemmas 4.2 and 4.3 and Proposition 4.5.

PROOF OF THEOREM 4.4. As explained above, let U denote a modified universal QTM that takes inputs of the form $|\tilde{e}\rangle|0^t1\rangle|x\rangle$ for any numbers $e, t \in \mathbb{N}^+$ and any string $x \in \{0, 1\}^*$ and produces clean outputs. Given any quantum function $g \in \square_1^{\text{QP}}$, Lemma 4.2 guarantees the existence of a polynomial p and a single-tape, $\tilde{\mathcal{C}}$ -amplitude, conservative, plain QTM M having clean outputs for which, on any input string $x \in \{0, 1\}^n$, M produces within $p(n)$ steps a superposition $g(|0^{p(n)}1\rangle|x\rangle) \otimes |0, q_f\rangle$ of skew final configurations composed of M 's essential tape region and M 's internal status. To be more precise, we denote by \widehat{U}_M the skew time-evolution operator of M and by $c_{0,M}^{(x)}$ the skew initial configuration of M on any input $|x\rangle$. We further set $|\zeta_{M,x}\rangle$ to be $\widehat{U}_M^{p(n)}|c_{0,M}^{(x)}\rangle$ and $g(|0^{p(n)}1\rangle|x\rangle)$ to be $|\psi_{g,x}\rangle$. Note that $|\zeta_{M,x}\rangle$ equals $|\psi_{g,x}\rangle \otimes |0, q_f\rangle$. Since $|\psi_{g,x}\rangle\langle\psi_{g,x}| = \text{tr}_n(|\psi_{g,x}\rangle\langle\psi_{g,x}| \otimes |0, q_f\rangle\langle 0, q_f|)$, $|\psi_{g,x}\rangle\langle\psi_{g,x}|$ can be expressed as $\text{tr}_n(|\zeta_{M,x}\rangle\langle\zeta_{M,x}|)$.

In contrast, we denote by \widehat{U}_δ the skew time-evolution operator of U and by $c_{0,U}^{(x_e)}$ the skew initial configuration of U on the input $|x_e\rangle$ for $e = \langle M, \varepsilon \rangle$ and $x_e = \tilde{e}0^{r(|x|)}1x$. Let $m = |x_e|$ and set $|\zeta_{U,x_e}\rangle$ to be $\widehat{U}_\delta^{s(p(m))}|c_{0,U}^{(x_e)}\rangle$, which is written as $\sum_{r \in \text{FSC}_{U,m}} |U[r]\rangle|\xi_{x_e,r}\rangle$ for an appropriate set $\{|\xi_{x_e,r}\rangle\}_{r \in \text{FSC}_{U,m}}$ of orthogonal quantum states. Proposition 4.5 then ensures that, by an appropriate choice of a polynomial s , the total variation distance between $\{\|\langle y|\widehat{U}_M^t|c_{0,M}^{(x)}\rangle\|^2\}_{y \in \{0,1\}^n}$ and $\{\|\langle y|\widehat{U}_\delta^{s(t)}|c_{0,U}^{(x_e)}\rangle\|^2\}_{y \in \{0,1\}^n}$ is at most $\varepsilon/2$ for any number $t \geq 0$.

We apply Lemma 4.3 and then obtain a \square_1^{QP} -quantum function f such that, for any quantum state $|\phi\rangle$, $f(|\phi\rangle)$ represents the result of $\widehat{U}_\delta^{s(p(n))}$ applied to $|\phi\rangle$. For convenience, we express $f(|x_e\rangle)$ as $|\eta_{f,x_e}\rangle$. Lemma 4.3 again implies that $|\eta_{f,x_e}\rangle = \sum_{r \in \text{FSC}_{U,m}} |U[r]\rangle|\widehat{\xi}_{x_e,r}\rangle$ with $\|\widehat{\xi}_{x_e,r}\rangle\| = \|\widehat{\xi}_{x_e,r}\rangle$ and $\langle\widehat{\xi}_{x_e,r}|\widehat{\xi}_{x_e,r'}\rangle = 0$ for all distinct pairs $r, r' \in \text{FSC}_{U,m}$. We thus obtain $\text{tr}_n(|\eta_{f,x_e}\rangle\langle\eta_{f,x_e}|) = \sum_{r,r' \in \text{FSC}_{U,m}} |U[r]\rangle\langle U[r']| \cdot \text{tr}(|\widehat{\xi}_{x_e,r}\rangle\langle\widehat{\xi}_{x_e,r'}|) = \sum_{r,r' \in \text{FSC}_{U,m}} \langle\widehat{\xi}_{x_e,r'}|\widehat{\xi}_{x_e,r}\rangle |U[r]\rangle\langle U[r']|$, which equals $\sum_{r \in \text{FSC}_{U,m}} \|\widehat{\xi}_{x_e,r}\rangle\|^2 |U[r]\rangle\langle U[r]|$. Similarly, we obtain $\text{tr}_n(|\zeta_{U,x_e}\rangle\langle\zeta_{U,x_e}|) = \sum_{r \in \text{FSC}_{U,m}} \|\xi_{x_e,r}\rangle\|^2 |U[r]\rangle\langle U[r]|$. From those calculations together with $\|\widehat{\xi}_{x_e,r}\rangle\| = \|\xi_{x_e,r}\rangle\|$, the equality $\text{tr}_n(|\eta_{f,x_e}\rangle\langle\eta_{f,x_e}|) = \text{tr}_n(|\zeta_{U,x_e}\rangle\langle\zeta_{U,x_e}|)$ follows immediately.

We thus conclude that $\|\|\psi_{g,x}\rangle\langle\psi_{g,x}| - \text{tr}_n(|\eta_{f,x}\rangle\langle\eta_{f,x}|)\|_{\text{tr}} = \|\text{tr}_n(|\zeta_{M,x}\rangle\langle\zeta_{M,x}|) - \text{tr}_n(|\zeta_{U,x_e}\rangle\langle\zeta_{U,x_e}|)\|_{\text{tr}}$. The last term is upper-bounded by $\sum_{y:|y|=n} \|\langle y|\widehat{U}_\delta^{s(p(n))}|c_{0,U}^{(x_e)}\rangle\|^2 - \|\langle y|\widehat{U}_M^{p(n)}|c_{0,M}^{(x)}\rangle\|^2$, which is clearly at most ε . Therefore, f is universal. \dashv

§5. Proof of the key lemma. To complete the proof of Theorem 4.1, we need to prove the key lemma, Lemma 4.3. This section intends to provide the lemma's desired proof. Our proof is inspired by a result of Yao [39], who demonstrated a quantum-circuit simulation of a QTM.

5.1. Functional simulation of QTMs. An essence of the proof of Lemma 4.3 is a direct step-by-step simulation of the behavior of a single-tape, $\tilde{\mathcal{C}}$ -amplitude, conservative, plain QTM $M = (Q, \Sigma, \Gamma, \delta, q_0, Q_f)$, which has clean outputs. For

simplicity, we assume that $\Sigma = \{0, 1\}$ and $\Gamma = \{0, 1, b\}$, where b here stands for a unique blank tape symbol, instead of $\#$ used in early sections. We further assume that $Q_f = \{q_f\}$ and $Q = \{0, 1\}^\ell$ for a certain fixed even number $\ell > 0$ with $q_0 = 0^\ell$ and $q_f = 1^\ell$. Let us assume that, starting with *binary* input string x written on the single input/work/output tape, M halts in at most $p(|x|)$ steps, where p is an appropriate polynomial associated only with M . We further assume that all computation paths of M on each input halt simultaneously. For convenience, we also demand that $p(n) > \ell$ for any $n \in \mathbb{N}$. It is important to remember that M eventually halts by entering the unique final inner state q_f and making the tape head stationed at the start cell and that no nonblank symbol appears in the left-side region of any output string.

Let $x = x_1x_2 \dots x_n$ be any input given to M , where x_i is a bit in $\{0, 1\}$ for each index $i \in [n]$. Associated with p , the essential tape region of M on x consists of all the tape cells indexed between $-p(n)$ and $+p(n)$. We express the tape content of the essential tape region as a string of the form $\sigma_1\sigma_2 \dots \sigma_{2p(n)+1}$ having length exactly $2p(|x|) + 1$ over the tape alphabet $\Gamma = \{0, 1, b\}$, where σ_i is a tape symbol written at the cell indexed $i - p(n) - 1$ for every index $i \in [2p(n) + 1]$. We trace the changes of these symbols as M makes its moves.

Since all \square_1^{QP} -functions are defined to handle quantum states in \mathcal{H}_∞ , we need to encode each tape symbol and thus a tape content. Let us define a new qustring that properly encodes a configuration $\gamma = (q, h, \sigma_1\sigma_2 \dots \sigma_{2p(n)+1})$ of M to be

$$|q\rangle \otimes |s_1, \hat{\sigma}_1\rangle \otimes |s_2, \hat{\sigma}_2\rangle \otimes |s_3, \hat{\sigma}_3\rangle \otimes \dots \otimes |s_{2p(n)+1}, \hat{\sigma}_{2p(n)+1}\rangle,$$

where each $\hat{\sigma}_i$ is in $\{\hat{0}, \hat{1}, \hat{b}\}$, each $s_i \in \{\hat{2}, \hat{3}\}$ indicates the presence of the tape head (where $\hat{2}$ means “the head rests here” and $\hat{3}$ means “no head is here”) at cell $i - p(n) - 1$, and q is an inner state in Q . In the subsequent subsections, we call such a qustring a *code* of the configuration γ of M and denote it by $|\hat{\gamma}\rangle$. This code $|\hat{\gamma}\rangle$ has length $\ell(|\hat{\gamma}\rangle) = 8p(n) + \ell + 4$, which is even and greater than n .

Given any binary input $x = x_1x_2 \dots x_n$ of length n , let us recall from Section 4.1 that $|\phi^p(x)\rangle = |0^{|x|}1\rangle|0^{p(|x|)}1\rangle|0^{11p(|x|)+6}1\rangle|x\rangle$ and $|\phi^{p.f}(x)\rangle = |\widetilde{0^{|f(x)|}}|0^{|f(x)|+1}1\rangle|\phi^p(x)\rangle$. Except for Step 1 in Section 5.2 as well as all steps in Section 5.4, we always ignore the prefix strings $\widetilde{0^{|f(x)|}}|0^{|f(x)|+1}1|0^{|x|}10^{p(|x|)}1$ in $|\phi^{p.f}(x)\rangle$ and $0^{|x|}10^{p(|x|)}1$ in $|\phi^p(x)\rangle$, and we pay our attention to the remaining qubits. The desired quantum function g will be constructed step by step through Sections 5.2–5.5.

5.2. Constructing a coded initial configuration. Given a binary input $x = x_1x_2 \dots x_n$, the initial configuration γ_0 of M on x is of the form $(q_0, 0, b \dots bxb \dots b)$, and thus the code $|\hat{\gamma}_0\rangle$ of γ_0 must have the form

$$|q_0\rangle \otimes |\hat{3}, \hat{b}\rangle \otimes \dots \otimes |\hat{3}, \hat{b}\rangle \otimes |\hat{2}, \hat{x}_1\rangle \otimes |\hat{3}, \hat{x}_2\rangle \otimes |\hat{3}, \hat{x}_3\rangle \otimes \dots \otimes |\hat{3}, \hat{x}_n\rangle \otimes |\hat{3}, \hat{b}\rangle \otimes \dots \otimes |\hat{3}, \hat{b}\rangle,$$

where \hat{x}_i is the code of x_i , $q_0 (= 0^\ell)$ is the initial inner state, and x_1 rests in cell 0. Notice that $\ell(|\hat{\gamma}_0\rangle) = 8p(n) + \ell + 4$. In what follows, we show how to generate this particular code $|\hat{\gamma}_0\rangle$ from the quantum state $|\phi^p(x)\rangle$. For simplicity, we will ignore the term $|\hat{q}_0\rangle$ in the following steps except for Step 8.

1) Starting with the input $|\phi^p(x)\rangle$, we first transform it to $|0^n1\rangle|0^{p(n)}1\rangle|10^{11p(n)+5}1\rangle|x\rangle$ by the quantum function $h_1 = \text{Skip}[NOT]$, which satisfies both $h_1(|0^m1\rangle|\psi\rangle) =$

$|0^m 1\rangle \otimes NOT(|\psi\rangle)$ and $h_1(|0^{m+1}\rangle) = |0^{m+1}\rangle$ for any number $m \in \mathbb{N}$ and any quantum state $|\psi\rangle \in \mathcal{H}_\infty$. Lemma 3.9 guarantees that h_1 actually exists in $\widehat{\square}_1^{\text{QP}}$.

From $|0^{p(n)} 1\rangle|10^{11p(n)+5} 1\rangle$, we wish to generate $|0^{p(n)} 1\rangle|0^{p(n)} 1\rangle|0^{10p(n)+5} 1\rangle$ by an appropriately constructed $\widehat{\square}_1^{\text{QP}}$ -function f_1 . To define the desired quantum function f_1 , we first construct another quantum function g_1 that maps $|0^{p(n)} 1\rangle|0^m 10^k 1\rangle$ to $|0^{p(n)} 1\rangle|0^{m+1} 10^{k-1} 1\rangle$ for any two integers $m \geq 0$ and $k \geq 1$ by setting

$$g_1(|\phi\rangle) = \begin{cases} |\phi\rangle & \text{if } \ell(|\phi\rangle) \leq 1, \\ |0\rangle \otimes g_1(\langle 0|\phi\rangle) + |1\rangle \otimes g_2(\langle 1|\phi\rangle) & \text{otherwise,} \end{cases}$$

where g_2 is introduced as

$$g_2(|\phi\rangle) = \begin{cases} |\phi\rangle & \text{if } \ell(|\phi\rangle) \leq 1, \\ SWAP(|0\rangle \otimes g_2(\langle 0|\phi\rangle) + |1\rangle\langle 1|\phi\rangle) & \text{otherwise.} \end{cases}$$

More formally, we set $g_2 = QRec_1[I, SWAP, I|g_2, I]$ and set $\hat{g}_2 = Branch[I, g_2]$. We then set $g_1 = QRec_1[I, I, \hat{g}_2|g_1, I]$. The quantum function f_1 is finally defined as $f_1 = QRec_1[I, g_1, I|f_1, I]$; namely,

$$f_1(|\phi\rangle) = \begin{cases} |\phi\rangle & \text{if } \ell(|\phi\rangle) \leq 1, \\ g_1(|0\rangle \otimes f_1(\langle 0|\phi\rangle) + |1\rangle\langle 1|\phi\rangle) & \text{otherwise.} \end{cases}$$

In a similar manner, we further transform $|0^{p(n)} 1\rangle|0^{10p(n)+5} 1\rangle$ to $|0^{p(n)} 1\rangle|0^{2p(n)+2} 1\rangle|0^{8p(n)+2} 1\rangle$. We then change $|0^n 1\rangle|0^{2p(n)+2} 1\rangle$ to $|0^n 1\rangle|0^n 1\rangle|0^{2p(n)-n+1} 1\rangle$ and $|0^n 1\rangle|0^{8p(n)+2} 1\rangle$ to $|0^n 1\rangle|0^n 1\rangle|0^{8p(n)-n+1} 1\rangle$. Overall, the input $|\phi^p(x)\rangle$ is turned into $|0^n 1\rangle|(\langle 0^p(n) 1\rangle^3)|(\langle 0^n 1\rangle^2)|0^{2p(n)-n+1} 1\rangle|0^{8p(n)-n+1} 1\rangle|x\rangle$.

(*) In what follows, by ignoring $|0^n 1\rangle|(\langle 0^p(n) 1\rangle^3)|(\langle 0^n 1\rangle^2)$, we assume that our input is temporarily $|0^{8p(n)-n+1} 1\rangle|x\rangle$.

2) For readability, we explain this step using an illustrative example of $|0^6 1\rangle|x_1 x_2 x_3\rangle$, which we intend to transform to $|00\rangle|\hat{3}\hat{x}_1 \hat{x}_2 \hat{x}_3\rangle$. For this purpose, we begin with changing $|0^6 1 x_1 x_2 x_3\rangle$ to $|x_3 x_2 x_1 10^6\rangle$ by applying *REVERSE*.

To obtain $|x_3 0 x_2 0 x_1 001\rangle|00\rangle$ from $|x_3 x_2 x_1\rangle|10^6\rangle$, we further apply $g_3 = SWAP \circ REP_1$, which changes $|x_3 x_2 x_1 10^6\rangle$ to $|x_3 0 x_2 x_1 10^5\rangle$. We repeatedly apply g_3 and transform $|x_3 x_2 x_1\rangle|10^6\rangle$ to $|x_3 0 x_2 0 x_1 0\rangle|10^3\rangle$. This process can be done by the quantum function $h_3 = 2QRec_2[I, h', g_3|\{h''_s\}_{s \in \{0,1\}^2}]$ defined by the 2-qubit quantum recursion, where $h' = Branch_2[\{h'_s\}_{s \in \{0,1\}^2}]$ with $h'_{a1} = SWAP$ and $h'_{a0} = I$ as well as $h''_{a1} = I$ and $h''_{a0} = h_3$ for each bit $a \in \{0, 1\}$; namely,

$$h_3(|\phi\rangle) = \begin{cases} |\phi\rangle & \text{if } \ell(|\phi\rangle) \leq 2, \\ \sum_{a \in \{0,1\}} (SWAP(|a1\rangle\langle a1|\psi_{g_3, \phi}) + |a0\rangle \otimes h_3(\langle a0|\psi_{g_3, \phi})) & \text{otherwise,} \end{cases}$$

where $|\psi_{g_3, \phi}\rangle$ stands for $g_3(|\phi\rangle)$. We further apply $g_4 = REVERSE \circ h_3$ to change $|x_3 x_2 x_1\rangle|10^6\rangle$ to $|0^3 1\rangle|\hat{x}_1 \hat{x}_2 \hat{x}_3\rangle$. In a general case, the above process transforms $|0^n 1\rangle|x_1 x_2 \dots x_m\rangle$ to $|1\rangle|\hat{x}_1 \hat{x}_2 \dots \hat{x}_n\rangle$.

Finally, we apply g_5 , which maps $|0^3 1\rangle|\hat{x}_1 \hat{x}_2 \hat{x}_3\rangle$ to $|00\rangle|\hat{3}\hat{x}_1 \hat{x}_2 \hat{x}_3\rangle$, defined by

$$g_5(|\phi\rangle) = \begin{cases} |\phi\rangle & \text{if } \ell(|\phi\rangle) \leq 1, \\ SWAP(|00\rangle \otimes g_5(\langle 00|\phi\rangle) + \sum_{y \in \{0,1\}^2 - \{00\}} (|y\rangle\langle y|\phi\rangle)) & \text{otherwise.} \end{cases}$$

In general, g_5 changes $|0^{n+1} 1\rangle|x_1 x_2 \dots x_n\rangle$ to $|\hat{3}\hat{x}_1 \hat{x}_2 \dots \hat{x}_n\rangle$.

(*) To explain our procedure further, for readability, we include more 0s to $|00\rangle|\hat{3}\hat{x}_1\hat{x}_2\hat{x}_3\rangle$, and hereafter we are focused on $|(00)^6\rangle|\hat{3}\hat{x}_1\hat{x}_2\hat{x}_3\rangle$.

3) We transform the first two bits 00 in $|(00)^6\rangle|\hat{3}\hat{x}_1\hat{x}_2\hat{x}_3\rangle$ to 11 ($=\hat{2}$) and then move them to the end of the qustring, resulting in $|(00)^5\rangle|\hat{3}\hat{x}_1\hat{x}_2\hat{x}_3\hat{2}\rangle$. This transformation is carried out as follows. Similarly to *CNOT*, we define $k_1 = NOT \circ SWAP \circ NOT$. Since $k_1(|00\rangle) = |11\rangle$, it suffices to define $f_3 = REMOVE_2 \circ k_1$.

4) In the beginning, our qustring is of the form $|\hat{3}\hat{x}_1\hat{x}_2\hat{x}_3\hat{2}\rangle$ by ignoring the leading bits $(00)^5$. We pay our attention to the qustring located between $\hat{3}$ and $\hat{2}$. We place the last two bits $\hat{2}$ into the location immediately right to $\hat{3}\hat{x}_1$ together with changing $\hat{2}$ to $\hat{3}$. We then obtain $|\hat{3}\hat{x}_1\hat{3}\hat{x}_2\hat{x}_3\rangle$. This process is realized by an appropriate \square_1^{QP} -function in the following fashion.

Let k_2 denote a bijection from $\{0,1\}^6$ to $\{0,1\}^6$ satisfying that $k_2(vw\hat{2}) = v\hat{2}w$ if $v, w \in \{\hat{0}, \hat{1}\}$, $k_2(\hat{3}w\hat{2}) = \hat{3}w\hat{3}$ if $w \in \{\hat{0}, \hat{1}\}$, $k_2(w\hat{3}\hat{2}) = w\hat{3}\hat{3}$ if $w \in \{\hat{0}, \hat{1}\}$, and $k_2(vwz) = vwz$ if $v, w, z \neq \hat{2}$. With this k_2 , we define g_5 as

$$g_5(|\phi\rangle) = \begin{cases} |\phi\rangle & \text{if } \ell(|\phi\rangle) \leq 6, \\ \sum_{y:|y|=6} |k_2(y)\rangle\langle y|\phi\rangle & \text{otherwise.} \end{cases}$$

By Lemma 3.7, g_5 belongs to $\widehat{\square_1^{QP}}$. We then define a quantum function h_5 by setting $h_5 = 2QRec_2[I, g_5, I | h_5, h_5, h_5, h_5]$, namely,

$$h_5(|\phi\rangle) = \begin{cases} |\phi\rangle & \text{if } \ell(|\phi\rangle) \leq 2, \\ \sum_{y \in \{\hat{0}, \hat{1}, \hat{2}, \hat{3}\}} g_5(|y\rangle \otimes h_5(\langle y|\phi\rangle)) & \text{otherwise.} \end{cases}$$

After placing $\hat{3}$ into the right of $\hat{3}\hat{x}_1$, we finally obtain the qustring $|(00)^5\rangle|\hat{3}\hat{x}_1\hat{3}\hat{x}_2\hat{x}_3\rangle$.

5) Let us define $h'_5 = h_5 \circ f_3$, which is the compositions of Steps 3–4. By applying h'_5 repeatedly, we can generally change $|(00)^{n-1}\rangle|\hat{3}\hat{x}_1\hat{x}_2 \dots \hat{x}_n\rangle$ to $|\hat{3}\hat{x}_1\hat{3}\hat{x}_2 \dots \hat{3}\hat{x}_n\rangle$. Furthermore, if we apply h'_5 twice to $|(00)^3\rangle|\hat{3}\hat{x}_1\hat{3}\hat{x}_2\hat{3}\hat{x}_3\rangle$ in our example, then it is possible to append $|\hat{3}\hat{b}\rangle$ to the end of the qustring by consuming $(00)^3$, and then we obtain $|\hat{3}\hat{x}_1\hat{3}\hat{x}_2\hat{3}\hat{x}_3\rangle|\hat{3}\hat{b}\rangle$. Using $|0^{2p(n)-n+1}\rangle$ in $|\phi^p\rangle$, we repeat Steps 3–4 $2p(n) - n + 1$ times to encode the content of the first $p(n) + 1$ tape cells indexed by nonnegative numbers. Returning to our example, if we take $|0^4\rangle$, then this process transforms $|0^4\rangle|(00)^6\rangle|\hat{3}\hat{x}_1\hat{x}_2\hat{x}_3\rangle$ into $|0^4\rangle|(00)^2\rangle|\hat{3}\hat{x}_1\hat{3}\hat{x}_2\hat{3}\hat{x}_3\rangle|\hat{3}\hat{b}\rangle$. To realize this transform by an appropriate \square_1^{QP} -function, since Steps 3–4 exclude $|0^4\rangle$, we first need to extend h'_5 to $H_1 = Skip[h'_5]$ by Lemma 3.9. The repetition of Steps 3–4 is done by the following quantum function f_5 :

$$f_5(|\phi\rangle) = \begin{cases} |\phi\rangle & \text{if } \ell(|\phi\rangle) \leq 1, \\ H_1(|0\rangle \otimes f_5(\langle 0|\phi\rangle) + |1\rangle\langle 1|\phi\rangle) & \text{otherwise.} \end{cases}$$

More generally, f_5 changes $|0^{2p(n)-n+1}\rangle|(00)^{2p(n)-n+1}\rangle|\hat{3}\hat{x}_1\hat{x}_2 \dots \hat{x}_n\rangle$ to $|0^{2p(n)-n+1}\rangle|\hat{3}\hat{x}_1\hat{3}\hat{x}_2 \dots \hat{3}\hat{x}_n\rangle|\hat{3}\hat{b}\hat{3}\hat{b} \dots \hat{3}\hat{b}\rangle$ with $p(n) - n + 1$ copies of $\hat{3}\hat{b}$.

6) Suppose that our qustring has the form $|(00)^2\rangle|\hat{3}\hat{x}_1\hat{3}\hat{x}_2\hat{3}\hat{x}_3\rangle|\hat{3}\hat{b}\rangle$ by ignoring $|0^4\rangle$. We want to change the leftmost $\hat{3}$ to $\hat{2}$, resulting in $|(00)^2\rangle|\hat{2}\hat{x}_1\hat{3}\hat{x}_2\hat{3}\hat{x}_3\rangle|\hat{3}\hat{b}\rangle$. For our purpose, we first choose a unique bijection p satisfying that $p(\hat{2}) = \hat{3}$, $p(\hat{3}) = \hat{2}$, and $p(y) = y$ for all other $y \in \{0,1\}^2$. Using Lemma 3.7, we expand this

bijection p to its associated quantum function g_p . The quantum function f_6 is then defined as

$$f_6(|\phi\rangle) = \begin{cases} |\phi\rangle & \text{if } \ell(|\phi\rangle) \leq 2, \\ g_p(|00\rangle) \otimes f_6(\langle 00|\phi\rangle) + \sum_{y \in \{0,1\}^{2-\ell(00)}} |y\rangle\langle y|\phi\rangle & \text{otherwise.} \end{cases}$$

7) We then change the series of 00's in $|(00)^2\rangle|\hat{2}\hat{x}_1\hat{3}\hat{x}_2\hat{3}\hat{x}_3\rangle|\hat{3}\hat{b}\rangle$ into 10's. To make this change, we prepare $h_6 = SWAP \circ NOT \circ CNOT \circ NOT$. Note that $h_6(|00\rangle) = |10\rangle$ and $h_6(|11\rangle) = |11\rangle$. We then set

$$f_7(|\phi\rangle) = \begin{cases} |\phi\rangle & \text{if } \ell(|\phi\rangle) \leq 2, \\ h_6(|00\rangle) \otimes f_7(\langle 00|\phi\rangle) + \sum_{y:|y|=2 \wedge y \neq 00} |y\rangle\langle y|\phi\rangle & \text{otherwise.} \end{cases}$$

This quantum function f_7 changes $|(00)^2\rangle$ to $|(10)^2\rangle$, which equals $|\hat{3}\hat{b}\rangle$, and thus we obtain $|\hat{3}\hat{b}\rangle|\hat{2}\hat{x}_1\hat{3}\hat{x}_2\hat{3}\hat{x}_3\rangle|\hat{3}\hat{b}\rangle$. To obtain $p(n)$ copies of $|\hat{3}\hat{b}\rangle$ in the left-side region of $|\hat{2}\rangle$ in general, the string $(00)^{2p(n)}$ is needed to consume.

8) In this final step, we include the term $|q_0\rangle (= |0^\ell\rangle)$ into our procedure. We combine Steps 1–7 to transform $|0^\ell\rangle|0^{2p(n)-n+1}1\rangle|0^{8p(n)-n+2}1\rangle|x\rangle$ to $|q_0\rangle|\hat{3}\hat{b} \dots \hat{3}\hat{b}\rangle|\hat{2}\hat{x}_1\hat{3}\hat{x}_2\hat{3}\hat{x}_3 \dots \hat{3}\hat{x}_n\rangle|\hat{3}\hat{b} \dots \hat{3}\hat{b}\rangle$ by the quantum function F_1 defined as $F_1 = RevBranch_\ell[\{g_s\}_{s \in \{0,1\}^\ell}]$, where $g_{0^\ell} = f_7$ and $g_s = I$ for any string s different from 0^ℓ .

5.3. Simulating a single step. To simulate an entire computation of M on any given input x , we need to simulate all steps of M one by one until M eventually enters the final inner state q_f . In what follows, we demonstrate how to simulate a single step of M by changing a head position, a tape symbol, and an inner state in a given configuration.

Note that M 's step involves only three consecutive cells, one of which is being scanned by the tape head. To describe such three consecutive cells together with M 's inner state, in general, we use an expression r of the form $ps_1\sigma_1s_2\sigma_2s_3\sigma_3$ using $p \in \{0,1\}^\ell$, $\sigma_i \in \{0,1,b\}$, and $s_i \in \{2,3\}$ for each index $i \in [3]$. Each expression with $s_i = 2$ indicates that M is in state q , scanning the i th cell of the three cells. The length of r is clearly $\ell + 6$. Let T be the set of all possible such r 's. Notice that T is a finite set. The code $|\hat{r}\rangle$ of $r = ps_1\sigma_1s_2\sigma_2s_3\sigma_3$ is $|q\rangle|\hat{s}_1, \hat{\sigma}_1\rangle|\hat{s}_2, \hat{\sigma}_2\rangle|\hat{s}_3, \hat{\sigma}_3\rangle$, which is of length $\ell + 12$.

For simplicity, we call r a target if $s_1 = 3, s_2 = 2, s_3 = 3$, and $\delta(q, \sigma_2)$ is defined. For later use, $s_1\sigma_1s_2\sigma_2s_3\sigma_3$ without q is called a pretarget if $qs_1\sigma_1s_2\sigma_2s_3\sigma_3$ is a target.

1) Let $r = qs_1\sigma_1s_2\sigma_2s_3\sigma_3$ be any fixed element in T . We prepare a flag qubit $|0\rangle$ in the end of $|\phi\rangle$ to mark that a simulation of M 's single step is in progress or has been already done. Let us define a quantum function f_8 , which transforms $|\hat{r}\rangle|0\rangle$ to either $|\hat{r}\rangle|0\rangle$ or $|\hat{s}\rangle|1\rangle$, where s is an expression obtained from r by applying δ once if r is a target. Let A denote the set of all targets in T and set $A^\circ = \{\hat{r} \mid r \in A\}$. Similarly, we define T° from T .

For this purpose, we first define a supporting quantum function $h_r(|b\rangle|\psi\rangle) = NOT(|b\rangle) \otimes |\hat{r}\rangle\langle \hat{r}|\psi\rangle + \sum_{s \in \{0,1\}^{\ell+12-A^\circ}} (|b\rangle \otimes |s\rangle\langle s|\psi\rangle)$ for each target $r \in T$ and any $b \in \{0,1\}$. Next, we define $\{g_r\}_{r \in T}$ as follows. If r is a non-target in T , then we set $g_r = I$. In what follows, we assume that r is a target. Let $g_r(|0\rangle|s\rangle|\phi\rangle) = |0\rangle|s\rangle|\phi\rangle$

for any string $s \in \{0, 1\}^{\ell+12}$. Since M is plain, M has only two kinds of transitions, shown in (i) and (ii) below.

(i) Consider the case where $\delta(q, \sigma_2) = e^{i\theta}|q', \tau, d\rangle$. When $d = L$, we set

$$g_r(|1\rangle|q\rangle|\hat{3}, \hat{\sigma}_1\rangle|\hat{2}, \hat{\sigma}_2\rangle|\hat{3}, \hat{\sigma}_3\rangle|\phi\rangle) = e^{i\theta}|1\rangle|q'\rangle|\hat{2}, \hat{\sigma}_1\rangle|\hat{3}, \hat{\tau}\rangle|\hat{3}, \hat{\sigma}_3\rangle|\phi\rangle.$$

When $d = R$, in contrast, we define

$$g_r(|1\rangle|q\rangle|\hat{3}, \hat{\sigma}_1\rangle|\hat{2}, \hat{\sigma}_2\rangle|\hat{3}, \hat{\sigma}_3\rangle|\phi\rangle) = e^{i\theta}|1\rangle|q'\rangle|\hat{3}, \hat{\sigma}_1\rangle|\hat{3}, \hat{\tau}\rangle|\hat{2}, \hat{\sigma}_3\rangle|\phi\rangle.$$

(ii) Consider the case where $\delta(q, \sigma_2) = \cos\theta|q_1, \tau_1, d_1\rangle + \sin\theta|q_2, \tau_2, d_2\rangle$. If $(d_1, d_2) = (R, L)$, then we define g_r as

$$g_r(|1\rangle|q\rangle|\hat{3}, \hat{\sigma}_1\rangle|\hat{2}, \hat{\sigma}_2\rangle|\hat{3}, \hat{\sigma}_3\rangle|\phi\rangle) = \cos\theta|1\rangle|q_1\rangle|\hat{3}, \hat{\sigma}_1\rangle|\hat{3}, \hat{\tau}_1\rangle|\hat{2}, \hat{\sigma}_3\rangle|\phi\rangle + \sin\theta|1\rangle|q_2\rangle|\hat{2}, \hat{\sigma}_1\rangle|\hat{3}, \hat{\tau}_2\rangle|\hat{3}, \hat{\sigma}_3\rangle|\phi\rangle.$$

The other values of (d_1, d_2) are similarly handled.

Notice that $\{g_r(|1\rangle|\tilde{r}\rangle)\}_{r \in A}$ forms an orthonormal set because M is well-formed, and thus δ satisfies all the conditions stated in Section 2.2. Once the flag qubit becomes $|1\rangle$, we do not need to apply $g_r \circ h_r$. Thus, we further set $g'_r = \text{Branch}[g_r \circ h_r, I]$. By combining all g'_r 's, we define g as $g = \text{Compo}\{\{g'_r\}_{r \in T}\}$, which implies

$$g(|0\rangle|\phi\rangle) = \sum_{r \in T} g_r(|1\rangle|\tilde{r}\rangle\langle\tilde{r}|\phi\rangle) + \sum_{s \in \{0,1\}^{\ell+12}-T^\diamond} (|0\rangle|s\rangle\langle s|\phi\rangle)$$

for any quantum state $|\phi\rangle \in \mathcal{H}_\infty$. We can claim that g is norm-preserving and it can be constructed from the initial quantum functions in Definition 3.1 by applying the construction rules. From this claim, g falls into $\overline{\square}_1^{\text{QP}}$. Finally, we define $f_8 = \text{REMOVE}_{1 \circ} (g^{\leq \ell+13} \otimes I)$, where $g^{\leq \ell+13} \otimes I$ is defined as in Lemma 3.8. Intuitively, f_8 changes the content of three consecutive tape cells whose middle cell is being scanned by the tape head.

2) We want to apply f_8 to all three consecutive tape cells. First, we find a code of a pretarget $\hat{s}_1\hat{\sigma}_1\hat{s}_2\hat{\sigma}_2\hat{s}_3\hat{\sigma}_3$ (using the quantum recursion), move an inner state q as well as a marker $|0\rangle$ forward (by $\text{REP}_{\ell+1}$) to generate a block of the form $|0\rangle|q\rangle|\hat{s}_1\hat{\sigma}_1\hat{s}_2\hat{\sigma}_2\hat{s}_3\hat{\sigma}_3\rangle$ and apply f_8 to this block. This changes $|0\rangle$ to $|1\rangle$ to record the execution of the current procedure. We then move the obtained inner state and the marker back to the end (by $\text{REMOVE}_{\ell+1}$). This entire procedure can be executed by an appropriate quantum function, say, F_2 .

To be more formal, we first set another quantum function p to be $\text{REMOVE}_{\ell+1} \circ \text{LENGTH}_{\ell+13}[f_8] \circ \text{REP}_{\ell+1}$. The quantum function F'_2 is then defined as

$$F'_2(|\phi\rangle|0\rangle|q\rangle) = \begin{cases} |\phi\rangle|0\rangle|q\rangle & \text{if } \ell(|\phi\rangle) < \ell + 13, \\ p(\sum_{s:|s|=4} (|s\rangle \otimes F'_2(\langle s|\phi\rangle|0\rangle|q\rangle))) & \text{otherwise.} \end{cases}$$

To complete the transformation, we further define $F_2 = \text{REP}_{\ell+1} \circ F'_2 \circ \text{REMOVE}_{\ell+1}$. After the application of F_2 , the first qubit of $|0\rangle|q\rangle|\phi\rangle$ turns to $|1\rangle$, marking an execution of M 's single step. If we want to repeat an application of F_2 , we need to reset $|1\rangle$ back to $|0\rangle$.

5.4. Completing the entire simulation. We have shown in Section 5.3 how to simulate a single step of M on x by F_2 . Here, we want to simulate all steps of M by applying F_2 inductively to any input of the form $|0^{p(|x|)}1\rangle \otimes |\psi\rangle$, where $|\psi\rangle$ represents a superposition of coded configurations of M on x . This process is implemented by a new quantum function F_3 , which repeatedly applies $NOT \circ F_2$ to $|0\rangle|\psi\rangle$, where NOT is used to reset $|1\rangle$ to $|0\rangle$.

We first take the quantum function $\hat{g} = Skip[NOT \circ F_2]$ given by Lemma 3.9. The desired quantum function F_3 must satisfy

$$F_3(|\phi\rangle) = \begin{cases} |\phi\rangle & \text{if } \ell(|\phi\rangle) \leq 1, \\ |0\rangle \otimes \hat{g}(F_3(\langle 0|\phi\rangle)) + |1\rangle \otimes I(\langle 1|\phi\rangle) & \text{otherwise.} \end{cases}$$

Note that the number of the applications of \hat{g} is exactly $p(|x|)$. This function F_3 can be realized with a use of the single-qubit quantum recursion of the form $F_3 = QRec_1[I, Branch[\hat{g}, I], I[F_3, I]]$.

5.5. Preparing an output. Assume that a superposition of coded skew final configurations of M on the given input x of length n has the form $\sum_{r \in FSC_M(x)} |\widehat{M}[r]\rangle | \zeta_{x,r} \rangle$ for a certain series $\{|\zeta_{x,r}\rangle\}_{r \in FSC_{M,n}}$ of quantum states, where $M[r]$ indicates M 's output string appearing in a skew final configuration r including only the essential tape region of M on x .

To show the first part of Lemma 4.3, in the end of our simulation, we need to generate $\widehat{M}[r]$ in the leftmost portion of the qustring obtained by the simulation. To achieve this goal, we first move the content of the left-side region of the start cell to the right end of the essential tape region.

As an illustrative example, suppose that we have already obtained the quantum state $|\hat{3}\hat{b}\hat{3}\hat{b}\hat{3}\hat{b}\rangle|\hat{2}\hat{0}\hat{3}\hat{0}\hat{3}\hat{1}\rangle|\hat{3}\hat{b}\rangle$ after executing steps in Section 5.4. In this case, the outcome of M in this skew final configuration r is 001, and thus $\widehat{M}[r] = \hat{0}\hat{0}\hat{1}\hat{2}$ holds. In what follows, we want to transform $|\hat{3}\hat{b}\hat{3}\hat{b}\hat{3}\hat{b}\rangle|\hat{2}\hat{0}\hat{3}\hat{0}\hat{3}\hat{1}\rangle|\hat{3}\hat{b}\rangle$ to $|\hat{0}\hat{0}\hat{1}\hat{2}\rangle|\hat{3}\hat{b}\hat{3}\hat{b}\hat{3}\hat{b}\rangle|\hat{1}\hat{3}\hat{3}\hat{3}\rangle$, which equals $|\widehat{M}[r]\rangle|\hat{3}\hat{b}\hat{3}\hat{b}\hat{3}\hat{b}\rangle|\hat{1}\hat{3}\hat{3}\hat{3}\rangle$, by an appropriate \square_1^{QP} -function, say, F_4 .

(i) Starting with $|\hat{3}\hat{b}\hat{3}\hat{b}\hat{3}\hat{b}\rangle|\hat{2}\hat{0}\hat{3}\hat{0}\hat{3}\hat{1}\rangle|\hat{3}\hat{b}\rangle$, to mark the end portion of the tape cell, we change the last marker $\hat{3}$ to $\hat{1}$ by applying NOT to $\hat{3}$ and then obtain $|\hat{3}\hat{b}\hat{3}\hat{b}\hat{3}\hat{b}\rangle|\hat{2}\hat{0}\hat{3}\hat{0}\hat{3}\hat{1}\rangle|\hat{1}\hat{b}\rangle$. This process is referred to as f_9 .

(ii) By moving repeatedly the leftmost $|\hat{3}\hat{b}\rangle$ in $|\hat{3}\hat{b}\hat{3}\hat{b}\hat{3}\hat{b}\rangle|\hat{2}\hat{0}\hat{3}\hat{0}\hat{3}\hat{1}\rangle|\hat{1}\hat{b}\rangle$ to the end of the qustring, we eventually produce $|\hat{2}\hat{0}\hat{3}\hat{0}\hat{3}\hat{1}\rangle|\hat{1}\hat{b}\hat{3}\hat{b}\hat{3}\hat{b}\hat{3}\hat{b}\rangle$. To realize this entire transform, we need to define

$$f_{10}(|\phi\rangle) = \begin{cases} |\phi\rangle & \text{if } \ell(|\phi\rangle) < 4, \\ \sum_{a \in \{0,1\}} |\hat{2}\hat{a}\rangle\langle\hat{2}\hat{a}|\phi\rangle + \sum_{z \in B_4} REMOVE_4(|z\rangle \otimes f_{10}(\langle z|\phi\rangle)) & \text{otherwise,} \end{cases}$$

where $B_4 = \{0, 1\}^4 - \{\hat{2}\hat{0}, \hat{2}\hat{1}\}$. More formally, we define f_{10} as $f_{10} = 2QRec_4[I, h'', I | \{f_z\}_{z \in \{0,1\}^4}]$, where $h'' = Branch[\{h''_z\}_{z \in \{0,1\}^4}]$ with $h''_{\hat{2}\hat{0}} = h''_{\hat{2}\hat{1}} = I$ and $h''_z = REMOVE_4$ as well as $f_{\hat{2}\hat{0}} = f_{\hat{2}\hat{1}} = I$ and $f_z = f_{10}$ for any $z \in B_4$.

(iii) The current qustring has the form $|\hat{2}\hat{0}\hat{3}\hat{0}\hat{3}\hat{1}\rangle|\hat{1}\hat{b}\rangle|\hat{3}\hat{b}\hat{3}\hat{b}\hat{3}\hat{b}\rangle$. We sequentially remove each of the markers $\{\hat{1}, \hat{2}, \hat{3}\}$ in $|\hat{2}\hat{0}\hat{3}\hat{0}\hat{3}\hat{1}\rangle|\hat{1}\hat{b}\rangle$ to the end of the entire qustring and produce $|\hat{0}\hat{0}\hat{1}\hat{b}\rangle|\hat{3}\hat{b}\hat{3}\hat{b}\hat{3}\hat{b}\rangle|\hat{1}\hat{3}\hat{3}\hat{2}\rangle$. To implement this process, we apply h_8 defined

by $h_8 = 2QREC_3[I, REMOVE_2, I | \{h_y\}_{y \in \{0,1\}^4}]$, where $h_{\hat{1}\hat{b}} = I$ and $h_y = h_8$ for all $y \in \{0,1\}^4 - \{\hat{1}\hat{b}\}$; in other words,

$$h_8(|\phi\rangle) = \begin{cases} |\phi\rangle & \text{if } \ell(|\phi\rangle) < 4, \\ REMOVE_2(|\hat{1}\hat{b}\rangle\langle\hat{1}\hat{b}| \phi) + \sum_{y \in \{0,1\}^4 - \{\hat{1}\hat{b}\}} |y\rangle \otimes h_8(\langle y|\phi\rangle) & \text{otherwise.} \end{cases}$$

(iv) At last, we change the rightmost \hat{b} in $|\hat{0}\hat{0}\hat{1}\hat{b}\rangle$ to $\hat{2}$ and then obtain $|\hat{0}\hat{0}\hat{1}\hat{2}\rangle (= |001\rangle)$. To produce such a qustring, we apply the quantum function h_6 defined by

$$h_9(|\phi\rangle) = \begin{cases} |\phi\rangle & \text{if } \ell(|\phi\rangle) < 2, \\ \sum_{y \in \{\hat{0}, \hat{1}\}} |y\rangle \otimes h_9(\langle y|\phi\rangle) + |\hat{b}\rangle\langle\hat{2}| \phi + |\hat{2}\rangle\langle\hat{b}| \phi & \text{otherwise.} \end{cases}$$

Formally, we set $h_9 = 2QRec_1[I, h'_9, I | \{h_y\}_{y \in \{0,1\}^2}]$, where $h'_9 = Branch[\{h''_y\}_{y \in \{0,1\}^2}]$ with $h''_2 = h''_{\hat{b}} = SWAP \circ NOT \circ SWAP$ and $h''_y = I$ together with $h_2 = h_{\hat{b}} = I$ and $h_y = h_9$ for any $y \in \{\hat{0}, \hat{1}\}$.

(vi) To perform Steps (i)–(v) at once, we combine all quantum functions used in Steps (i)–(v) and define a single quantum function $F_4 = h_9 \circ h_8 \circ f_{10} \circ f_9$. Overall, the resulted qustring is $|\hat{0}\hat{0}\hat{1}\hat{2}\rangle|\hat{3}\hat{b}\hat{3}\hat{b}\hat{3}\hat{b}\rangle|\hat{1}\hat{3}\hat{3}\hat{2}\rangle$.

The quantum state $|\phi_{F_4}^p(x)\rangle$ can be expressed as $\sum_{r \in FSC_{M,n}} |\widetilde{M}[r]\rangle |\widehat{\xi}_{x,r}\rangle$ for certain quantum states $\{|\widehat{\xi}_{x,r}\rangle\}_r$. Since we deal with an essential tape region of M , it instantly follows that, for every $x, x' \in \{0,1\}^n$ and $r, r' \in FSC_{M,n}$, $\|\langle \widehat{\xi}_{x,r} | \widehat{\xi}_{x',r'} \rangle\| = \|\langle \widehat{\xi}_{x,r} | \widehat{\xi}_{x',r'} \rangle\|$ and $\langle \widehat{\xi}_{x,r} | \widehat{\xi}_{x',r'} \rangle = 0$ if $r \neq r'$. Therefore, F_4 satisfies the condition of the first part of the lemma.

For the second part of the lemma, we further need to retrieve $|M[r]\rangle$ from the coded qustring $|\widetilde{M}[r]\rangle$. From the previous illustrative example $|\hat{0}\hat{0}\hat{1}\hat{2}\rangle|\hat{3}\hat{b}\hat{3}\hat{b}\hat{3}\hat{b}\rangle|\hat{1}\hat{3}\hat{3}\hat{2}\rangle$, we need to produce $|001\rangle|1\rangle|\hat{3}\hat{b}\hat{3}\hat{b}\hat{3}\hat{b}\rangle|\hat{1}\hat{3}\hat{3}\hat{2}\rangle|1000\rangle$. For this purpose, we use the quantum function g_7 defined by $g_7 = 2QRec_1[I, REMOVE_1, I | \{g'_z\}_{z \in \{0,1\}^2}]$, where $g'_{0y} = g_7$ and $g'_{1y} = I$ for any $y \in \{0,1\}$; namely,

$$g_7(|\phi\rangle) = \begin{cases} |\phi\rangle & \text{if } \ell(|\phi\rangle) < 2, \\ REMOVE_1(\sum_{y \in \{0,1\}} (|0y\rangle \otimes g_7(\langle 0y|\phi\rangle) + |1y\rangle\langle 1y|\phi)) & \text{otherwise.} \end{cases}$$

Using g_7 , we finally set $F_5 = g_7 \circ F_4$ to obtain the second part of the lemma.

This completes the proof of Lemma 4.3.

5.6. Simple applications of the simulation procedures. The proof of Lemma 4.3 provides useful procedures not only for the construction of the desired quantum function g but also for other special-purpose quantum functions. Hereafter, as simple applications of the simulation procedures given in Steps 1–6 of Section 5.2, we will explain how to encode/decode classical strings and how to duplicate classical information by $\widehat{\square}_1^{QP}$ -functions.

Steps 1–8 in Section 5.2 describe a transformation between classical strings and their encodings. A slight modification of Step 2 introduces an encoder *Encode*, which properly encodes binary strings s to \tilde{s} .

LEMMA 5.1. *There exists a quantum function Encode in $\widehat{\square}_1^{QP}$ that satisfies $Encode(|0^{k+1}1\rangle \otimes |\phi\rangle) = \sum_{s \in \{0,1\}^k} (|\tilde{s}\rangle \otimes |s\rangle\langle s|\phi)$ for any number $k \in \mathbb{N}$ and any*

quantum state $|\phi\rangle \in \mathcal{H}_\infty$. Moreover, the quantum function $Decode = Encode^{-1}$ is also in $\widehat{\square}_1^{QP}$.

Given extra bits $0^{k+1}1$, the first k qubits of $|\phi\rangle$ is properly encoded by $Encode$ by consuming $0^{k+1}1$. For example, $Encode$ changes $|0^3 1\rangle|a_1 a_2\rangle$ to $|\hat{a}_1 \hat{a}_2 \hat{2}\rangle$ and $Decode$ returns $|\hat{a}_1 \hat{a}_2 \hat{2}\rangle$ back to $|0^3 1\rangle|a_1 a_2\rangle$. Notice that Proposition 3.5 ensures $Decode \in \widehat{\square}_1^{QP}$ from $Encode \in \widehat{\square}_1^{QP}$.

Quantum mechanics in general prohibits us from duplicating unknown quantum states; however, it is possible to copy each classical string quantumly. We thus have the following quantum function $COPY_2$, which copies the content of the first k qubits of any input.

LEMMA 5.2. *There exists a quantum function $COPY_2$ in $\widehat{\square}_1^{QP}$ that satisfies the following condition: for any $k \in \mathbb{N}^+$ and any $|\phi\rangle \in \mathcal{H}_\infty$,*

$$COPY_2(|\tilde{0}^k\rangle \otimes |\phi\rangle) = \sum_{s \in \{0,1\}^k} (|\tilde{s}\rangle \otimes |\tilde{s}\rangle \langle \tilde{s} | \phi \rangle).$$

The encoding $\tilde{0}^k$ of 0^k is needed to distinguish 0^k from any part of $|\phi\rangle$ because k is not a fixed constant. When $|\phi\rangle$ has the form $|\tilde{x}\rangle|\psi\rangle$, the quantum function $COPY_2$ works as $COPY_2(|\tilde{0}^k\rangle \otimes |\tilde{x}\rangle|\psi\rangle) = |\tilde{x}\rangle \otimes |\tilde{x}\rangle|\psi\rangle$.

PROOF OF LEMMA 5.2. We wish to construct the desired quantum function $COPY_2$ as follows. To make our construction process readable, we use an illustrative example of $|\tilde{0}^k\rangle|\phi\rangle = |\tilde{0}^2\rangle|\tilde{a}_1 \tilde{a}_2\rangle (= |\hat{0}\hat{0}\hat{2}\rangle|\hat{a}_1 \hat{a}_2 \hat{2}\rangle)$ to show how each constructed quantum function works.

- 1) Steps 1–6 of Section 5.2 transform $|\hat{0}\hat{0}\hat{1}\rangle|\hat{a}_1 \hat{a}_2\rangle$ to $|\hat{2}\hat{a}_1 \hat{3}\hat{a}_2\rangle|\hat{3}\rangle$. By a slight modification of these steps, it is possible to transform $|\hat{0}\hat{0}\hat{2}\rangle|\hat{a}_1 \hat{a}_2 \hat{2}\rangle$ to $|\hat{3}\hat{a}_1 \hat{3}\hat{a}_2 \hat{2}\rangle|\hat{2}\rangle$. We denote by f_1 a quantum function that realizes this transformation.
- 2) By copying \hat{a}_i in $\hat{3}\hat{a}_i$ onto $\hat{3}$ for each $i \in \{1,2\}$, we change $\hat{3}\hat{a}_i$ to $\hat{a}_i \hat{a}_i$ and then obtain $|\hat{a}_1 \hat{a}_1 \hat{a}_2 \hat{a}_2 \hat{2}\rangle|\hat{2}\rangle$. This step can be formally made in the following way. First, we define a quantum function h_2 that satisfies $h_2(|b_1 b_2\rangle|b_3 b_4\rangle \otimes |\phi\rangle) = |b_4 b_2 b_1 b_3\rangle \otimes |\phi\rangle$ for any $\phi \in \mathcal{H}_\infty$. Such a quantum function actually exists by Lemma 3.7. Second, we set DUP to be $h_2^{-1} \circ (SWAP \circ NOT \circ SWAP \circ CNOT) \circ h_2 \circ NOT$. It then follows that $DUP(|\hat{3}\rangle|\hat{a}\rangle \otimes |\phi\rangle) = |\hat{a}\rangle|\hat{a}\rangle \otimes |\phi\rangle$ for any bit $a \in \{0,1\}$ and any quantum state $|\phi\rangle \in \mathcal{H}_\infty$. With this DUP , we further define f_2 by the 4-qubit quantum recursion as

$$f_2(|\phi\rangle) = \begin{cases} |\phi\rangle & \text{if } \ell(|\phi\rangle) < 4, \\ DUP(\sum_{a \in \{0,1\}} |\hat{3}\hat{a}\rangle \otimes f_2(\langle \hat{3}\hat{a} | \phi \rangle) + \sum_{y \in B'_4} |y\rangle \langle y | \phi \rangle) & \text{otherwise,} \end{cases}$$

where $B'_4 = \{0,1\}^4 - \{\hat{3}\hat{0}, \hat{3}\hat{1}\}$.

- 3) Next, we transform $|\hat{a}_1 \hat{a}_1 \hat{a}_2 \hat{a}_2 \hat{2}\rangle|\hat{2}\rangle$ to $|\hat{a}_1 \hat{a}_2 \hat{2}\rangle|\hat{2}\hat{a}_2 \hat{a}_1\rangle$. This transformation can be done by f_3 defined as

$$f_3(|\phi\rangle) = \begin{cases} |\phi\rangle & \text{if } \ell(|\phi\rangle) < 4, \\ REMOVE_{E_2}(\sum_{y \in \{0,1\}^4 - \{\hat{2}\hat{2}\}} |y\rangle \otimes f_3(\langle y | \phi \rangle) + |\hat{2}\hat{2}\rangle \langle \hat{2}\hat{2} | \phi \rangle) & \text{otherwise.} \end{cases}$$

- 4) We then change $|\hat{a}_1\hat{a}_2\hat{2}\rangle|\hat{2}\hat{a}_2\hat{a}_1\rangle$ to $|\hat{2}\hat{a}_2\hat{a}_1\rangle|\hat{2}\hat{a}_2\hat{a}_1\rangle$ by removing each two qubits in the first part to the end. This process is precisely realized by f_4 defined as

$$f_4(|\phi\rangle) = \begin{cases} |\phi\rangle & \text{if } \ell(|\phi\rangle) < 4, \\ REMOVE_2(\sum_{y \in \{0,1\}^{2-\{\hat{2}\}}}(|y\rangle \otimes f_5(\langle y|\phi\rangle)) + |\hat{2}\rangle\langle\hat{2}|\phi\rangle) & \text{otherwise.} \end{cases}$$

- 5) Finally, we reverse the whole qustring to obtain $|\hat{a}_1\hat{a}_2\hat{2}\rangle|\hat{a}_1\hat{a}_2\hat{2}\rangle$ by applying *REVERSE*. This completes the proof of the lemma. ⊣

§6. Future challenges. In Definition 3.1, we have defined \square_1^{QP} -functions on \mathcal{H}_∞ and we have given in Theorem 4.1 a new characterization of FBQP-functions in terms of these \square_1^{QP} -functions. To point out the directions of future research, we wish to raise a challenging open question in Section 6.1 and to present in Sections 6.2 and 6.3 three possible implications of our schematic definition to the subjects of descriptonal complexity, first-order theories, and higher-type functionals. In Section 6.4, we will remark a practical application to the designing of quantum programming languages.

6.1. Seeking a more reasonable schematic definition. Our schematic definition (Definition 3.1) is composed of the initial quantum functions, which are derived from natural, simple quantum gates, and the construction rules, including the multiqubit quantum recursion, which significantly enriches the scope of constructed quantum functions. The choice of initial functions and construction rules that we have used in this paper directly affects the richness of \square_1^{QP} -functions. Although our \square_1^{QP} is sufficient to characterize FBQP, if we further seek for enriching the \square_1^{QP} -functions, one way is to supplement additional initial quantum functions. As a concrete example, let us consider the quantum Fourier transform (QFT), which plays an important role in, e.g., Shor’s factoring quantum algorithm [29]. We have demonstrated in Lemma 3.10 how to implement a restricted form of QFT working on a fixed number of qubits, and thus it belongs to $\widehat{\square_1^{QP}}$. Nonetheless, a more general form of QFT, acting on an “arbitrary” number of qubits, may not be realized *precisely* by $\widehat{\square_1^{QP}}$ -functions although it can be approximated to any desired accuracy by the $\widehat{\square_1^{QP}}$ -functions. To remedy the exclusion of QFT from our function class \square_1^{QP} , for instance, we can expand the current $\widehat{\square_1^{QP}}$ by including as an initial quantum function the quantum function defined as

$$CROT(|\phi\rangle|0^j\rangle) = |0\rangle\langle 0|\phi\rangle|0^j\rangle + \omega_j|1\rangle\langle 1|\phi\rangle|0^j\rangle \text{ (controlled rotation),}$$

where $\omega_j = e^{2\pi i/2^j}$, with an extra term 0^j . It is not difficult to construct QFT from quantum functions in this expanded $\widehat{\square_1^{QP}}$ obtained by adding *CROT*. As this example shows, it remains important to seek for a simpler, more reasonable schematic definition of quantum functions, which are capable of precisely characterizing both BQP and FBQP and also simplifying the proof of Theorem 4.1.

From the minimalist’s viewpoint, on the contrary, we may be able to eliminate certain schemata or replace them by simpler ones but still ensure the characterization result of BQP and FBQP in terms of \square_1^{QP} -functions. As a concrete example, we may

ask whether our multiqubit quantum recursion can be replaced by 1-qubit quantum recursion at the cost of adding extra initial quantum functions.

6.2. Introduction of descriptonal complexity and first-order theories. As noted in Section 3.1, our schematic definition of \square_1^{QP} -functions provides us with a natural means of assigning the *descriptonal complexity*—a new complexity measure—to each of those quantum functions in \square_1^{QP} . This complexity measure has been used to prove, for instance, Lemma 3.4. As a consequence of our main theorem, this complexity measure concept also transfers to languages in BQP and functions in FBQP, and thus it naturally helps us introduce the notion of the descriptonal complexities of such languages and functions.

It is further possible for us to extend this complexity measure to “arbitrary” languages and functions on $\{0,1\}^*$, which are not necessarily limited to FBQP and BQP, and to discuss their “relative” complexity to \square_1^{QP} . More formally, given a function f on $\{0,1\}^*$, the \square_1^{QP} -*descriptonal complexity of f at length n* is the minimal number of times we use initial quantum functions and construction rules to build a \square_1^{QP} -function g for which $|\langle \Psi_{f(x)} | \phi_g^{p,f}(x) \rangle|^2 \geq 2/3$ hold for a certain polynomial p with $|f(x)| \leq p(|x|)$ for all strings x of length exactly n . We write $dc(f)[n]$ to denote the \square_1^{QP} -descriptonal complexity of f at length n . Obviously, every \square_1^{QP} -function has *constant* \square_1^{QP} -descriptonal complexity at every length. In a similar spirit but based on quantum finite automata, Villagra and Yamakami [32] discussed the quantum state complexity restricted to inputs of length exactly n (as well as length at most n). It has turned out that such complexity measure is quite useful. Refer to [32] for the detailed definitions. Our new complexity measure $dc(f)[n]$ is also expected to be a useful tool in classifying languages and functions in descriptonal power in a way that is quite different from what QTMs and quantum circuits do.

In a much wider perspective, our schematic definition of polynomial-time quantum computability may lead to the future development of an appropriate form of first-order theories over quantum states in Hilbert spaces or *first-order quantum theories*, for short. In the literature, first-order theories and their natural subtheories have become a fruitful research subject in mathematical logic and recursion theory and they have also found numerous applications in other fields as well. A weak form of their subtheories has been studied in quantum complexity theory. For instance, using bounded quantifiers over quantum states in Hilbert spaces, quantum analogues of NP and the Meyer-Stockmeyer polynomial(-time) hierarchy have been discussed in [36]. Unfortunately, we are still far away from obtaining well-accepted first-order theories and useful subtheories for quantum computing.

6.3. Extension to type-2 quantum functionals. Conventionally, functions mapping Σ^* to Σ^* are categorized as *type-1 functionals*, whereas *type-2 functionals* are functions taking inputs from Σ^* together with type-1 functionals. In computational complexity theory, such *type-2 functionals* have been extensively discussed in, e.g., [7, 8, 16, 23, 30, 33, 34].

In analogy to the classical case, we call \square_1^{QP} -functions on \mathcal{H}_∞ *type-1 quantum functionals*. To introduce *type-2 quantum functionals*, we start with an arbitrary quantum function O mapping \mathcal{H}_∞ to \mathcal{H}_∞ , which is treated as a *function oracle* (an

oracle function or simply an oracle) in the following formulation. From such an oracle O , we define a new linear operator \tilde{O} as $\tilde{O}(|\tilde{x}\rangle) = \sum_{s:|s|=|\tilde{x}|} \alpha_s |\tilde{s}\rangle$ if $O(|x\rangle)$ is of the form $\sum_{s:|s|=|x|} \alpha_s |s\rangle$ for any string $x \in \{0,1\}^*$, where \tilde{s} is the code of s , defined in Section 4.1. Note that $\ell(\tilde{O}(|\tilde{x}\rangle)) = 2\ell(O(|x\rangle)) + 2$.

First, we expand our initial functions and construction rules given in Definition 3.1 by replacing any quantum function, say, $f(|\phi\rangle)$ in each scheme of the definition with $f(|\phi\rangle, O)$. Second, we introduce another initial quantum function, called the query function *QUERY*. For any two qustrings $|\phi\rangle$ and $|\psi\rangle$ of length n , let $|\phi\rangle \oplus |\psi\rangle = \sum_{s:|s|=n} \sum_{t:|t|=n} \langle s|\phi\rangle \langle t|\psi\rangle |s \oplus t\rangle$, where $s \oplus t$ means the bitwise XOR of s and t . As a special case, it follows that $\tilde{O}(|\tilde{x}\rangle) \oplus \tilde{O}(|\tilde{x}\rangle) = |0^{2|\tilde{x}|+2}\rangle$ for any $x \in \{0,1\}^*$. The query function is then defined as

$$\begin{aligned}
 & \text{QUERY}(|\phi\rangle, O) \\
 &= \sum_{n \in [t]} \sum_{x \in \Sigma^n} \sum_{s \in \Sigma^n} \left(|\tilde{x}\rangle \otimes (\tilde{O}(|\tilde{x}\rangle) \oplus |\tilde{s}\rangle) \otimes \langle \tilde{x}\tilde{s}|\phi\rangle + \sum_{y \in \Sigma^{4n+4} \wedge y \neq \tilde{x}\tilde{s}} |y\rangle \langle y|\phi\rangle \right)
 \end{aligned}$$

for all $|\phi\rangle \in \mathcal{H}_\infty$, where $t = \lfloor (\ell(|\phi\rangle) - 4)/4 \rfloor$. In particular, $\text{QUERY}(|\tilde{x}\rangle|\tilde{s}\rangle|\phi\rangle)$ equals $|\tilde{x}\rangle \otimes (\tilde{O}(|\tilde{x}\rangle) \oplus |\tilde{s}\rangle) \otimes |\phi\rangle$. It also follows that $\text{QUERY} \circ \text{QUERY}(|\phi\rangle, O) = I(|\phi\rangle)$ since $\tilde{O}(|\tilde{x}\rangle) \oplus \tilde{O}(|\tilde{x}\rangle) = |0^{2|\tilde{x}|+2}\rangle$.

Notice that if O is in \square_1^{QP} then the function $Q_O(|\phi\rangle) =_{\text{def}} \text{QUERY}(|\phi\rangle, O)$ also belongs to \square_1^{QP} . It is also possible to show similar results discussed in the previous sections. These basic results can open a door to a rich field of higher-type quantum computability and we expect fruitful results to be discovered in this new field.

6.4. Application to quantum programming languages. A practical application of our schematic definition can be found in the area of *quantum programming languages*. Since the early days of quantum computing research, a significant effort has been made by physicists, computer scientists, and computer engineers to draw a pragmatic road map to a real-life quantum computer.

Toward the realization of such quantum computers, most research has focused on their hardware construction. For the building of “multipurpose” quantum computers, however, it is more desirable to make them “programmable” in such a way that any run of an appropriate “quantum program” freely alters computing processes for different target problems without remodeling their hardware each time. A quantum program here refers to a finite series of instructions on how to operate the quantum computer step by step. To write such a quantum program, nevertheless, we need to develop well-structured programming languages for the quantum computer (dubbed as *quantum programming languages*). Various quantum programming languages have been discussed over two decades in due course of developing real-life quantum computers. Refer to surveys, e.g., [12] for a necessary background.

Our schematic definition provides a description of how to define a given \square_1^{QP} -function. This description can be viewed as a set of instructions, each of which instructs how to apply each scheme to build the desired quantum function and it thus resembles a program that dictates how to construct the quantum function. Therefore, our schematic description of a construction process of quantum functions may help us design appropriate quantum programming languages in the future.

Acknowledgments. This work was done while the author was at the University of Ottawa between 1999 and 2003, and it was financially supported by the Natural Sciences and Engineering Research Council of Canada.

An extended abstract appeared under the title “A recursive definition of quantum polynomial time computability (extended abstract)” in the Proceedings of the 9th Workshop on Non-Classical Models of Automata and Applications (NCMA 2017), Prague, Czech Republic, August 17–18, 2017, Österreichische Computer Gesellschaft 2017, the Austrian Computer Society, pp. 243–258, 2017. The current paper intends to correct erroneous descriptions in the extended abstract and provide more detailed explanations to all omitted proofs due to the page limit.

REFERENCES

- [1] L. M. ADLEMAN, J. DEMARRAIS, and M. A. HUANG, *Quantum computability*. *SIAM Journal on Computing*, vol. 26 (1997), pp. 1524–1540.
- [2] A. BARENCO, et al., *Elementary gates for quantum computation*. *Physical Review A*, vol. 52 (1995), pp. 3457–3467.
- [3] P. BENIOFF, *The computer as a physical system: A microscopic quantum mechanical Hamiltonian model of computers represented by turing machines*. *Journal of Statistical Physics*, vol. 22 (1980), pp. 563–591.
- [4] E. BERNSTEIN and U. VAZIRANI, *Quantum complexity theory*. *SIAM Journal on Computing*, vol. 26 (1997) pp. 1411–1473.
- [5] P. O. BOYKIN, et al., *On universal and fault-tolerant quantum computing*, preprint, 1999. arXiv:quant-ph/9906054.
- [6] A. COBHAM, The intrinsic computational difficulty of functions. *Proceedings of the 1964 Congress for Logic, Mathematics, and Philosophy of Science*, North-Holland, 1964, pp. 24–30.
- [7] R. L. CONSTABLE, Type two computational complexity. *Proceedings of the 5th ACM Symposium on Theory of Computing (STOC'73)*, Association for Computing Machinery, New York, 1973, pp. 108–121.
- [8] S. COOK and B. M. KAPRON, Characterizations of the basic feasible functionals of finite type. *Proceedings of the 30th IEEE Conference on Foundations of Computer Science (STOC'89)*, IEEE, 1989, pp. 154–159.
- [9] M. DAVIS, *The Undecidable. Basic Papers on Undecidable Propositions, Unsolvability Problems, and Computable Functions*. Raven Press, Hewlett-Packard, New York, 1965.
- [10] D. DEUTSCH, *Quantum theory, the Church-Turing principle, and the universal quantum computer*. *Proceedings Royal Society London, Series A*, vol. 400 (1985), pp. 97–117.
- [11] ———, *Quantum computational networks*. *Proceedings of the Royal Society of London Series*, vol. 425 (1989), pp. 73–90.
- [12] S. J. GAY, *Quantum programming languages: Survey and bibliography*. *Mathematical Structures in Computer Science*, vol. 16 (2006) pp. 581–600.
- [13] L. K. GROVER, A fast quantum mechanical algorithm for database search. *Proceedings of the 28th Annual ACM Symposium on the Theory of Computing (STOC'96)*, Association for Computing Machinery, New York, 1996, pp. 212–219.
- [14] L. GROVER, *Quantum mechanics in searching for a needle in a haystack*. *Physical Review Letters*, vol. 79 (1997), p. 325.
- [15] J. HOPCROFT and J. ULLMAN, *An Introduction to Automata Theory Languages and Computation*. Addison-Wesley, Reading, MA, 1979.
- [16] A. KITAEV, *Quantum computations: Algorithms and error correction*. *Russian Mathematical Surveys*, vol. 52 (1997), pp. 1191–1249.
- [17] A. Y. KITAEV, A. H. SHEN, and M. N. VYALYI, *Classical and Quantum Computation*, American Mathematical Society, Providence, RI, 2002.
- [18] S. C. KLEENE, *General recursive functions of natural numbers*. *Mathematische Annalen*, vol. 112 (1936), pp. 727–742.

- [19] ———, *Recursive predicates and quantifiers*. *Transactions of the American Mathematical Society*, vol. 53 (1943), pp. 41–73.
- [20] ———, *Recursive functionals and quantifiers of finite types I*. *Transactions of the American Mathematical Society*, vol. 91 (1959), pp. 1–52.
- [21] ———, *Recursive functionals and quantifiers of finite types II*. *Transactions of the American Mathematical Society*, vol. 108 (1963), pp. 106–142.
- [22] K. KO and H. FRIEDMAN, *Computational complexity of real functions*. *Theoretical Computer Science*, vol. 20 (1982), pp. 323–352.
- [23] K. MEHLHORN, *Polynomial and abstract subrecursive classes*. *Journal of Computer and System Sciences*, vol. 12 (1976), pp. 147–178.
- [24] M. A. NIELSEN and I. L. CHUANG, *Quantum Computation and Quantum Information*. Cambridge University Press, Cambridge, 2000.
- [25] H. NISHIMURA and M. OZAWA, *Computational complexity of uniform quantum circuit families and quantum turing machines*. *Theoretical Computer Science*, vol. 276 (2002), pp. 147–181.
- [26] M. OZAWA and H. NISHIMURA, *Local transition functions of quantum turing machines*. *RAIRO—Theoretical Informatics and Applications*, vol. 276 (2000), pp. 379–402.
- [27] G. PEANO, *Sul concetto di numero*. *Rivista di Matematica*, vol. 1 (1891), pp. 87–102, 256–267.
- [28] R. I. SOARE, *Computability and recursion*. *The Bulletin of Symbolic Logic*, vol. 2 (1996), no. 3, pp. 284–321.
- [29] P. W. SHOR, *Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer*. *SIAM Journal on Computing*, vol. 26 (1997), pp. 1484–1509.
- [30] M. TOWNSEND, *Complexity for type-2 relations*. *Notre Dame Journal of Formal Logic*, vol. 31 (1990), 241–262.
- [31] A. M. TURING, *On computable numbers with an application to the Entscheidungsproblem*. *Proceedings of the London Mathematical Society*, vol. 42 (1936), no. 2, pp. 230–265; Erratum ibid 43 (1937) 544–546.
- [32] M. VILLAGRA and T. YAMAKAMI, *Quantum state complexity of formal languages*. *Proceedings of the 17th International Workshop on Descriptive Complexity of Formal Systems (DFCS 2015)*, Lecture Notes in Computer Science, vol. 9118, Springer, New York, 2015, pp. 280–291.
- [33] T. YAMAKAMI, *Structural properties for feasibly computable classes of type two*. *Mathematical Systems Theory*, vol. 25 (1992), pp. 177–201.
- [34] ———, *Feasible computability and resource bounded topology*. *Information and Computation*, vol. 116 (1995), pp. 214–230.
- [35] ———, *A foundation of programming a multi-tape quantum Turing machine*. *Proceedings of the 24th Mathematical Foundations of Computer Science (MFCS'99)*, Lecture Notes in Computer Science, vol. 1672, 1999, pp. 430–441. arXiv:quant-ph/9906084.
- [36] ———, *Quantum NP and a quantum hierarchy*. *Proceedings of the 2nd IFIP International Conference on Theoretical Computer Science (TCS 2002)*, Kluwer Academic Press (under the name “Foundations of Information Technology in the Era of Network and Mobile Computing”), the series IFIP—The International Federation for Information Processing, vol. 96, 2002, (Track 1), pp. 323–336.
- [37] ———, *Analysis of quantum functions*. *International Journal of Foundations of Computer Science*, vol. 14 (2003), pp. 815–852. A preliminary version appeared in the *Proceedings of the 19th International Conference on Foundations of Software Technology and Theoretical Computer Science*, Lecture Notes in Computer Science, vol. 1738, 1999, pp. 407–419.
- [38] ———, *A recursive definition of quantum polynomial time computability (extended abstract)*. *Proceedings of the 9th Workshop on Non-Classical Models of Automata and Applications (NCMA 2017)*, Österreichische Computer Gesellschaft 2017, The Austrian Computer Society, 2017, pp. 243–258.
- [39] A. C. YAO, *Quantum circuit complexity*. *Proceedings of the 34th Annual IEEE Symposium on Foundations of Computer Science (FOCS'93)*, 1993, pp. 80–91.

FACULTY OF ENGINEERING
 UNIVERSITY OF FUKUI
 3-9-1 BUNKYO, FUKUI 910-8507, JAPAN
 E-mail: tomoyukiyamakami@gmail.com