On Minimal Universal Quantum Gates

Christopher Gilbert

August 2024

Abstract

This paper investigates the construction of universal classical and quantum gates, and proposes two novel quantum sets which use the fewest types of gates for each of their constructions of universality. For a circuit with ancilla qubits, a zero-controlled dual-axis irrational rotation gate is universal, and for a maximally entangled circuit with no ancillas—a construction more similar to classical universality—the set containing the controlled 90-degree Z-rotation gate and the irrational Y-rotation gate is universal.

1 Introduction

1.1 Universal Gates

A logic gate is an operation which acts on bits or qubits. For example, the classical AND gate, denoted $\wedge$, takes in two bits and outputs a 1 if and only if both input bits were 1. Let a gate G that takes in n inputs and returns m outputs be denoted $G : n \to m$. A gate or set of gates is universal if some arrangement of the gates can replicate the behavior of every possible gate.

An example of such a $2 \to 1$ gate for classical universality is the NAND gate, $\overline{\wedge}$, which outputs a 0 if and only if both of its input bits are 1. The universality of $\overline{\wedge}$ can be proved either by exhaustively showing that every possible $2 \to 1$ gate can be constructed by $\overline{\wedge}$, or with a more formal proof which will be presented later on.

A gate set is said to be minimal if it is the smallest gate set possible to make. For example, $\{\overline{\wedge}\}$ is a minimal universal set because any computation is impossible without at least one operation, and so a set with only a single element is the smallest possible set to be universal.

1.2 Quantum Gates

1.2.1 Quantum Computing

A quantum computer uses the properties of quantum mechanics in order to perform calculations. By creating a superposition of multiple possible inputs, all operations in a circuit are simultaneously done on each input. By canceling out the probabilities of inputs with negative results using quantum phase, an operation can find only the correct inputs with a quadratic improvement over classical computers. For other less generalizable quantum algorithms, notably Shor's algorithm, the speedup over classical computers can be up to exponential.

1.2.2 Quantum State Vectors

The state of a classical system may be represented as a sequence of bits, where each bit represents a wire in a circuit. By interpreting this sequence of bits as a binary integer, every classical state can be represented with a single number.

Quantum states, on the other hand, may be in a superposition of multiple states and so may be represented as a list of probability amplitudes corresponding to each classical state. This list is called the state vector and can accurately represent any possible quantum superposition. Thus a classical state with three wires can be in one of eight states, and a quantum state with three qubits can be fully described by a normal vector in $\mathbb{C}^8$. The state vector must always have a length of 1, because the sum of the squared amplitudes has to be 1, just as the sum of a set of probabilities must always add up to 1.

A classical state with state number n may also be represented as the nth unit vector, because it has a 100% chance to be in the state n.

1.2.3 Quantum Operations

The state vector representation of a classical or quantum state is extremely helpful in formulating universality mathematically using linear algebra. For an n-wire circuit with $N = 2^n$ states, a matrix

$$(\vec{c_1} \; \ldots \; \vec{c_N})$$

will take a classical state vector in the state i to the state vector $\vec{c_i}$. Thus, any operation can be directly described as a matrix of the output state vectors for each input.

In most cases, an operation is done only on a subset of the wires in a circuit. In order to apply an operation matrix, it first needs to be combined with identity matrices until it achieves the correct dimensions. For a circuit with n wires, in order to operate O on wires a through b, the following matrix M must be applied to the state vector of the entire circuit:

$$M_{a,b} = I_{2^a} \otimes O \otimes I_{2^{n-b}}$$

where $\otimes$ is the Kronecker product and I_x is the $x \times x$ identity matrix.

Each matrix M represents one way that O can be applied to a circuit, so the problem of universality can be described in terms of a matrix decomposition of any arbitrary operation into matrices of the form of M.

The proofs of universality presented in this paper will rely on decomposing each operation in an already proven quantum gate set into operations from a smaller gate set.

2 Classical Universality

2.1 Classical Circuits

Classical circuits are generally presented as electrical circuit diagrams. For example, an XOR gate is made from NAND gates as follows:

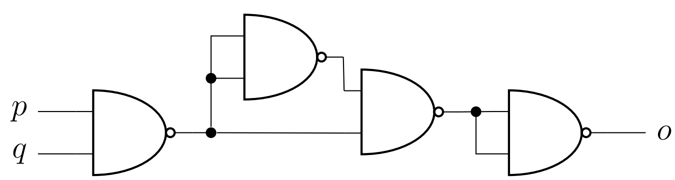

Figure 1: XOR Circuit made from NAND Gates

This classical diagram has a number of properties which differ from most quantum circtuits:

- Wires are allowed to cross and move
- Wires may be copied or duplicated
- Gates may change the number of wires
- No ancilla wires (in the 0 state) are provided

In order to rectify the first disparity, all quantum circuits used to prove universal quantum gate sets will also be allowed to use the SWAP gate, represented with $\times$, which swaps the state of two qubits and so is equivalent to swapping wires in a classical circuit.

The second disparity presents a more significant problem, because the No Cloning Theorem asserts that quantum states cannot be perfectly copied. Instead of directly copying states, another approach is to initialize a circuit with a number of wires in each state. Thus any previously required copying can be acheived throgh duplication of the applied operations.

The third disparity can be prevented by changing the types of classical gates used. Instead of $2 \to 1$ gates like NAND, one can implement $2 \to 2$ gates which replicate the behavior of the original gate on the lower output wire, and then have some other dummy behavior on the upper output wire. Under this new construction, when given an input of (p, q), the modified NAND gate returns $(0, p \overline{\wedge} q)$.

The final disparity can be solved by just dissallowing any ancilla wires in quantum circuits, but under most formulations of quantum universality, it is assumed that as many ancilla bits are given as are required. This is not so much a mathematical problem as it is a problem with the differing definitions of universality. Because of this, this paper will explore both the case where ancilla bits in the 0 state are allowed.

Fig. 2 shows the new implementation of an XOR gate after making the above changes. It only uses modified $2 \to 2$ NAND gates, and it also shows the current state at various positions along the circuit, logically simplified as much as possible.

2.2 Proof of Classical Universality

The most basic classical gates are the $1 \to 1$ gates: I (identity) and $\neg$ (NOT). However, as neither of these gates can relate one bit to another, neither can perform all binary operations, so it is impossible for either to be universal. Because of this, a universal gate set must have at least one gate with at least two inputs. With this limitation, the simplest possible gate type that could be universal is $2 \to 1$.

Focusing back to the regular classical case, without the modifications for implementing it in quantum circutry, this paper will now present a proof that $\overline{\wedge}$ (NAND) and $\overline{\vee}$ (NOR) are the only $2 \to 1$ universal classical quantum gates.

Theorem 1. *The only $2 \to 1$ universal classical gates are NAND and NOR.*

Lemma 1. *For any universal $2 \to 1$ gate G, $G(x, x) = \neg x$.*

Proof. All universal $2 \to 1$ gates G must be able to create any other possible gate, including the gate $H(x, y) = 1$.

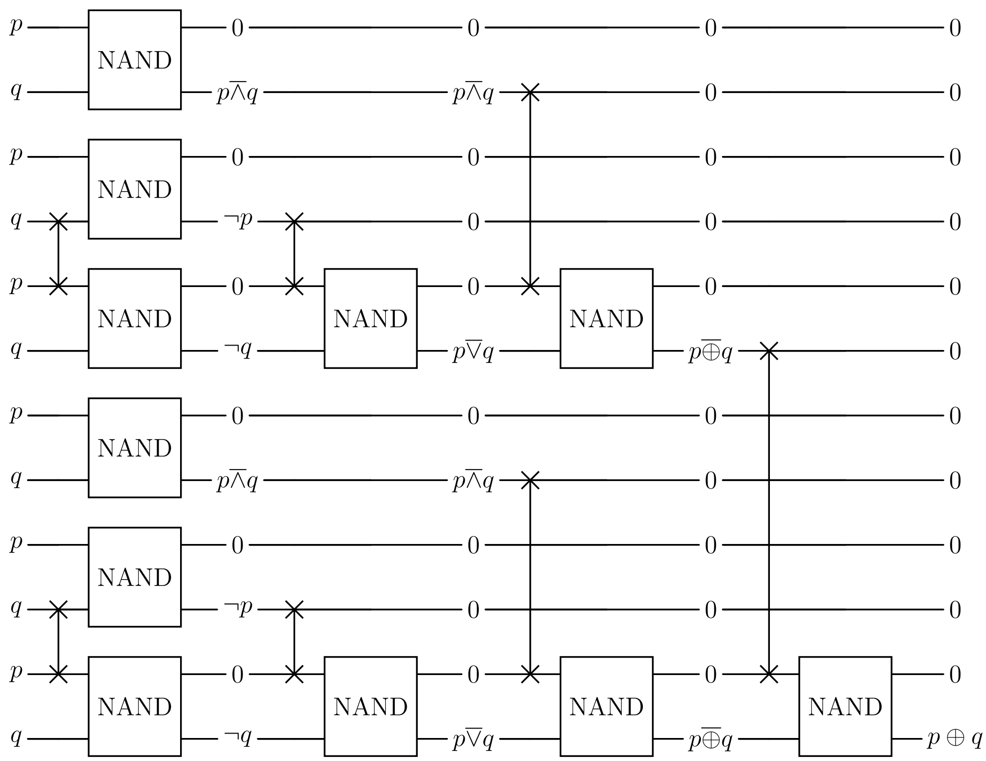

Figure 2: Quantum-Compatible XOR from NAND

$H(0,0) = 1$, so when given an input of $(0,0)$, G must be able to replicate this output in some way. If $G(0,0) \neq 1$, then no matter how many times it is applied or the inputs are copied, all wires will be in the 0 state. There would therefore never be any way to get an output in the 1 state. By contradiction, $G(0,0)$ must equal 1.

By the same logic, when given an input of $(1,1)$, in order to have any wire in the 0 state, $G(1,1) = 0$. Therefore, because $G(0,0) = 1$ and $G(1,1) = 0$, $G(x,x) = \neg x$. $\square$

Corrolary 1. *Any* $2 \to 1$ *universal gate can replicate the behavior of the* $1 \to 1$ *not gate, as any input* x *can be copied to* (x,x)*, onto which* G *can be applied, resulting in* $\neg x$*.*

Lemma 2. *For all universal* $2 \to 1$ *gates* G*,*

$$G(x, \neg x) \neq G(\neg x, x)$$

Proof. Let $T(H)$ denote the number of 1-valued outputs for all possible inputs of H. For example, of all the four possible inputs to an AND gate, only $(1,1)$ returns an output of 1. Thus $T(\wedge) = 1$.

When a NOT gate is applied to either input of a gate H,

$$T(H(p,q)) = T(H(\neg p, q)) = T(H(p, \neg q))$$

This is because applying a NOT gate to an input wire simply changes the order in which each possible input is checked.

When a NOT gate is applied to the output of any gate,

$$T(\neg H) = 4 - T(H)$$

This is because there are four total outputs and $T(H)$ 1-valued outputs, so there are $4 - T(H)$ 0-valued outputs, which will be turned into 1-valued outputs when the NOT gate is applied.

Thus, for a gate H such that $T(H) = 2$, no matter how many NOT gates are applied to the inputs or outputs of H, $T(H\prime)$ remains unchanged.

The identity-1 gate $I_1(p,q) = p$, which returns it's first input, cannot be universal by Lemma 1. $T(I_1) = 2$, so it's $\square$