Quantum Algorithms 1/31/2023

Lecture 3: Quantum Simulations of Deterministic and Probabilistic Computations

Instructor: Dieter van Melkebeek

Last lecture we saw that simulating deterministic and probabilistic computations on a quantum computer is non-trivial. This lecture we discuss how to do this efficiently. We start with a recap of our model for quantum computing, give some more examples of quantum gates, and list some interesting universal sets of quantum gates.

1 Quantum Computation

Quantum computation is performed on *qubits*. A *qubit* is a quantum system with two basis states. For a single qubit, the standard basis states are denoted $|0\rangle$ and $|1\rangle$. A quantum state composed of m qubits has 2^m basis states. The standard basis states are denoted $|s\rangle$ where the i-th position of the binary string s relates to the i-th qubit. For example, $|001\rangle$, $|101\rangle$, $|100\rangle$ denote some of the standard basis states of a three-qubit system. The state of an m-qubit system is described as a superposition of its basis states:

$$|\psi\rangle = \sum_{s\in\{0,1\}^m} \alpha_s |s\rangle, \text{ with amplitudes } \alpha_s \in \mathbb{R} \text{ (or } \mathbb{C}) \text{ and } \||\psi\rangle\|_2 = \sum_s |\alpha_s|^2 = 1.$$

We view a quantum computation on a given input $x \in \{0,1\}^n$ as a physical process that acts on a quantum system consisting of m qubits and procedes in the three stages: initialization, a sequence of quantum gates, and termination.

Initialization The qubits are initialized to a state representing the input $x \in \{0,1\}^n$, e.g., $|x0^{m-n}\rangle$.

Sequence of quantum gates A sequence of quantum gates are applied to the quantum system, where the sequence is produced by a deterministic Turing machine on input x.

Termination The quantum system is measured, collapsing the quantum system's state $\sum_{s\in\{0,1\}^m} \alpha_s |s\rangle$ into one of the basis states $|s\rangle$, where the probability for a particular $s \in \{0,1\}^m$ is given by $|\alpha_s|^2$. The output of the quantum computation y is then extracted from the string s that was measured.

A computational problem R is a relation between inputs and outputs such that $(x, y) \in R$ if and only if y is a correct output for the valid input x. A quantum computation has error at most ϵ if for every valid input x, $\Pr[(x, y) \in R] \geq 1 - \epsilon$. ϵ is called the error bound.

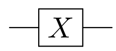
2 Quantum Gates

A *quantum gate*, T, is a local linear transformation with the two following equivalent properties:

- T preserves the 2-norm $\|.\|_2$.

- T is unitary, i.e., $T^*T = I$, where T^* denotes the complex conjugate transpose of T.

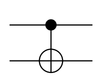
All deterministic gates that are reversible are valid quantum gates. Examples include the following:

- The X gate, a one-qubit gate effecting the logical operation NOT on basis states. The transition matrix is given by
$$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}.$$
When applied to a superposition $\alpha_0 |0\rangle + \alpha_1 |1\rangle$, the effect of the X gate is to interchange the two amplitudes: $\alpha_1 |0\rangle + \alpha_0 |1\rangle$. We will explain later where the name "X" comes from. An X gate is represented by the following circuit diagram:


- The CNOT gate, a two-qubit gate that on basis states effects replacing the second bit by the XOR of the two bits. The transition matrix is given by
$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}.$$
The name "CNOT," short for "Controlled NOT," refers to the alternate interpretation of the action of the gate, where the first bit controls whether or not to apply a NOT to the second bit. The first bit is referred to as the control bit, and the second one as the target bit. The NOT is applied to the target bit if and only if the control bit is set.

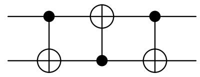
  When the target qubit is in a superposition but the control bit is not, the control bit dictates whether or not the amplitudes of the target qubit are interchanged. When the control qubit is also in superposition, the action is performed independently on the subspace where the control qubit is $|0\rangle$ and on the subspace where the control qubit is $|1\rangle$.

  We point out that the CNOT gate can induce *entanglement*. A two-qubit state is entangled if it cannot be written as the tensor product of a state of the first qubit and a state of the second qubit, i.e., there do not exist $\beta_0, \beta_1, \gamma_0, \gamma_1 \in \mathbb{C}$ such that
$$\alpha_{00} |00\rangle + \alpha_{01} |01\rangle + \alpha_{10} |10\rangle + \alpha_{11} |11\rangle = (\beta_0 |0\rangle + \beta_1 |1\rangle)(\gamma_0 |0\rangle + \gamma_1 |1\rangle).$$
Equivalently, the two-qubit state is entangled if the ratios ratios α_{00}/α_{01} and $\alpha_{10}/\alpha 11$ differ. Applying a CNOT gate leaves the first ratio unaffected and inverts the second ratio.

  A CNOT gate is represented by the following circuit diagram, where the horizontal line on top represents the control qubit, and the horizontal line on the bottom the target qubit. The control is marked by a small filled circle and the target qubit by the XOR symbol $\oplus$.


- The SWAP gate:
$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

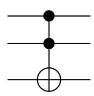
The SWAP gate is used to interchange the states of two qubits. The SWAP gate can be implemented as three subsequent CNOT's with the control qubit alternating:



- The CCNOT gate, also known as the Toffoli gate:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}.$$

  This is a 3-qubit gate where the first two act as controls and the last one as the target. On basis states it performs a NOT on the target if both controls are set. A CCNOT gate is represented by the following circuit diagram, where the control qubits are denoted by the two top most horizontal lines and the target qubit by the horizontal line on the bottom.

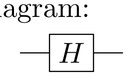


Some other important quantum gates include the following:

- The Hadamard gate, denoted by H:

$$\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}.$$

  The H gate affects an orthonormal basis change from the standard basis $\{|0\rangle, |1\rangle\}$ to the basis $\{|+\rangle, |-\rangle\}$, where $|+\rangle \doteq H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ and $|-\rangle \doteq H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$. Applied to either $|0\rangle$ or $|1\rangle$, the H gate has both similarities and differences with a classical fair coin flip. Analogous to a classical coin flip, both $H|0\rangle = |+\rangle$ and $H|1\rangle = |-\rangle$ have equal probability of yielding $|0\rangle$ or $|1\rangle$ when measured. The H gate differs from a classical coin flip in that if an H gate is subsequently applied a second time without an intermediate measurement, the result is always the initial state. This is because $H^{-1} = H^* = H$ so $HH = I$. An H gate is represented by the following circuit diagram:



- The R_ϕ gates for $\phi \in \mathbb{R}$, also known as the phase shift gates:

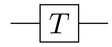
$$\begin{bmatrix} 1 & 0 \\ 0 & e^{i\phi} \end{bmatrix}.$$

  An R_ϕ gate is a one-qubit gate that is used to change the phase difference between the components of the superposition, while preserving their weights.

- The T gate is just a shorthand for $R_{\frac{\pi}{4}}$:

$$\begin{bmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{4}} \end{bmatrix}.$$

The T gate has the property that $T^* = T^7$. A T gate is represented by the following circuit diagram:



3 Universal Sets

An *exactly universal set* is a set of quantum gates which the property that every quantum gate can be computed exactly by a circuit composed of gates from the set. CNOT together with all 1-qubit gates form an exactly universal set.

A *universal set* is a set of quantum gates with the property that every quantum gate U can be approximated by a circuit composed of gates from the set in the sense that for all $\epsilon > 0$ there exists a circuit U' composed of gates in the set such that $\|U' - U\|_2 \leq \epsilon$. For gates of bounded dimension, the precise choice of norm does not matter. We will explain in a later lecture why we choose the 2-norm. An example of a universal set is {CNOT, H, T}.

For any universal set, the Solovay-Kitaev Theorem, stated below, provides an upper bound on how many gates from a universal set we need to approximate any quantum gate with a prescribed precision.

Theorem 1 (Solovay-Kitaev [NC16]). *If a finite universal set is closed under inverses, then any quantum gate can be approximated with 2-norm error at most ϵ by a circuit composed of $O(\text{poly}\log(1/\epsilon))$ gates in the set.*

The Solovay-Kitaev theorem implies that for b bits of accuracy, we only need a number of gates of a universal set that is polynomial in b, provided the universal is closed under inverses.

If the inverse of each gate in a finite universal set can be computed exactly by a circuit composed of gates in the set, then we an apply Solovay-Kitaev by first adding the missing inverses to the universal set, and later express those inverses in terms of the original universal set.

Corollary 2. *If the inverse of the each gate in a finite universal set can be computed exactly by a circuit composed of gates in the set, then any quantum gate can be approximated with 2-norm error at most ϵ by a circuit composed of $O(\text{poly}\log(1/\epsilon))$ gates in the set.*

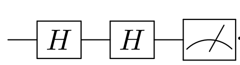
The corollary applies to our example universal set {CNOT, H, T}. Since CNOT* = CNOT, $H^* = H$, and $T^* = T^7$, the inverse of each gate in the set can be computed exactly by a circuit composed of gates in the set.

For the subset of quantum gates that can be expressed as matrices of real elements, the set {H, CCNOT} is a universal set.

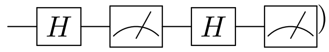
4 Solution to Exercise #1

Recall the statement: Consider a quantum system with 1 qubit. Determine the output distributions of each of the following processes:

1. Start in state $|0\rangle$, apply H, apply H again, and observe, i.e.,



2. Start in state $|0\rangle$, apply H, observe, apply H again, and observe, i.e.,



.

The solution is as follows:

1. For the case where H is subsequently applied twice with no measurement in between, $H^* = H$ and H is unitary so $H^2 = I$ and $H^2 |0\rangle = I |0\rangle = |0\rangle$. Thus, we have a point distribution on $|0\rangle$.
2. For the case where the qubit is measured in between applications of H: $H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, so when the qubit is measured the first time, the result is $|0\rangle$ with probability $\frac{1}{2}$ and $|1\rangle$ with probability $\frac{1}{2}$. So, there are two cases:
   - In the case where $|0\rangle$ was measure, $H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, so when the qubit is measured the second time, the result is $|0\rangle$ with probability $\frac{1}{2}$ and $|1\rangle$ with probability $\frac{1}{2}$.
   - In the case where $|1\rangle$ was measured, $H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$, so when the qubit is measured the second time, the result is $|0\rangle$ with probability $\frac{1}{2}$ and $|1\rangle$ with probability $\frac{1}{2}$.

   Summing over both cases, when the qubit is measured the second time, the result is $|0\rangle$ with probability $\frac{1}{2}$ and $|1\rangle$ with probability $\frac{1}{2}$.

Analogous to a classical fair coin flip, both $H|0\rangle = |+\rangle$, $H|1\rangle = |-\rangle$ have equal probability of yielding $|0\rangle$ or $|1\rangle$ when measured, but the H gate differs from a classical fair coin flip in that if an H gate is subsequently applied a second time without an intermediate measurement, the result is always the initial state (e.g., $HH|0\rangle = |0\rangle$, $HH|1\rangle = |1\rangle$) whereas two subsequent classical fair coin flips yield equal probability of observing either state irrespective of whether there is a measurement in between.

5 Interference

The fact that amplitudes can be both positive and negative allows for destructive interference to happen. Subsequent applications of the H gate discussed above are an example of this phenomenon. Explicitly, $H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \doteq |+\rangle$, and $H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \doteq |-\rangle$. By linearity,

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

The superposition of the $|+\rangle$ and $|-\rangle$ states destructively interfere to completely remove the $|1\rangle$ state from the final superposition, even though when measured in isolation, both the $|+\rangle$ and $|-\rangle$ state have a $\frac{1}{2}$ probability of yielding $|1\rangle$. This interference pattern is illustrated pictorially in Figure 1.

Such destructive interference cannot happen in probabilistic computations, where different computation paths that lead to the same final state $|s\rangle$ can only make the probability of measuring s at the end larger. In contrast, in quantum computations different paths that lead to the same $|s\rangle$

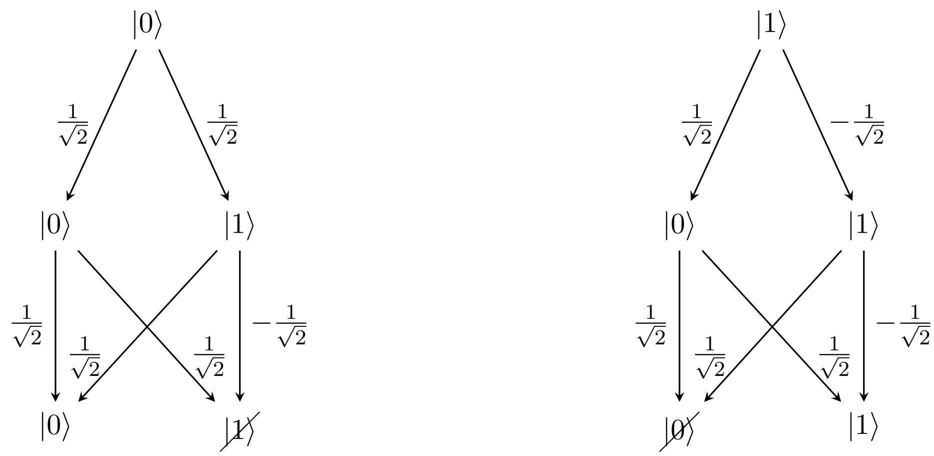


Figure 1: This figure demonstrates destructive interference for two subsequent applications of the H Gate.

can annihilate each other (destructive interference). This phenomenon can be exploited algorithmically by setting things up so that the computation paths that lead to an invalid solution, interfere destructively, whereas the computation paths that lead to valid solutions interfere constructively. If successful, this guarantees that the final measurement yields a valid solution (with certainty if the interference is perfect, or with high probability if the interference is large but not perfect). It is this type of interference (destructive and thus also constructive) that gives quantum computing its power.

We use this opportunity to dismiss a common misconception about the power of quantum computing, namely that it would derive from the fact that quantum systems can be in superposition. In both probabilistic and quantum systems, the state of a system can be described as a superposition of basis states. The key difference is that in quantum systems the coefficients can be both positive and negative, whereas in probabilistic systems they can be positive only. This difference is what makes interference possible in the quantum setting and impossible in the probabilistic setting.

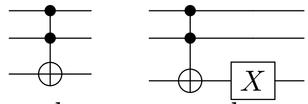
6 Simulating Deterministic Computations

Quantum circuits must be reversible. Not all deterministic computations are reversible, but for every deterministic computation, there exists a reversible transformation that yields the same output as the deterministic computation using some additional auxiliary bits (called “ancillas”). In particular $f : \{0,1\}^k \to \{0,1\}^l$ can be simulated by applying the reversible transformation

$$\tilde{f} : \{0,1\}^{k+l} \to \{0,1\}^{k+l} : (x,y) \mapsto (x, y \oplus f(x))$$

with y (the “ancillas”) initialized to 0^l. Note that $\tilde{f}$ is reversible as $(x, (y \oplus f(x)) \oplus f(x)) = (x,y)$.

If f is binary AND, then $\tilde{f}$ is CCNOT. If f is binary NAND, then $\tilde{f}$ is the composition of CCNOT and an X gate applied to the target qubit of the CCNOT gate. The circuit diagrams of the reversible simulation for binary AND and binary NAND are as follows:



Every deterministic computation can be represented as a sequence of NAND gates, each operating on a subset of the union of the input bits with the set of outputs of previous NAND gates. Therefore, for a deterministic computation that uses n input bits, t NAND gates, and l output bits, a reversible simulation can be achieved with n qubits to store the input bits and t ancillas to store the results of the t NAND gates. For symbolic convenience, we partition those t ancillas qubits into two groups, a group of l qubits that correspond to the output bits, and a group of $t-l$ qubits that correspond to the ancillas that are not output bits, which we call "garbage". Thus, our reversible simulation can be represented as a quantum circuit, call it Q, that realizes the following:

$$|x\rangle\,|0^l\rangle\,|0^{(t-l)}\rangle \mapsto |x\rangle\,|f(x)\rangle\,|\text{garbage}(x)\rangle$$

More generally, deterministic computations that utilize a broader vocabulary of gates than just the NAND gate can be simulated in the same way, where n qubits are used to store the input of size n bits, l qubits are used to store the output of size l bits, and $t-l$ ancillas are used to store the results of the $t-l$ classical gates for which the result in not part of the output.

If the qubits that store the input and output are entangled with the garbage qubits, that may prevent the interference necessary for many quantum algorithms, so we must further refine our protocol to ensure that the $t-l$ ancillas are not entangled with the other qubits. Since the quantum circuit Q is reversible, and since the $t-l$ ancillas qubits are not initially entangled with the other qubits, we can remove any entanglement of the $t-l$ ancillas with the other qubits by applying Q^{-1}. However, that application of Q^{-1} would also revert the qubits that store the output to $|0^l\rangle$ so we must first copy those l qubits to l additional ancillas. Then, we can safely apply Q^{-1} to remove any undesired entanglement. The full protocol realizes the following:

$$\begin{aligned}
|x\rangle\,|0^l\rangle\,|0^{t-l}\rangle\,|0^l\rangle &\overset{Q}{\mapsto} |x\rangle\,|f(x)\rangle\,|\text{garbage}(x)\rangle\,|0^l\rangle \\
&\overset{\text{CNOT}^l}{\mapsto} |x\rangle\,|f(x)\rangle\,|\text{garbage}(x)\rangle\,|f(x)\rangle \\
&\overset{Q^{-1}}{\mapsto} |x\rangle\,|0^l\rangle\,|0^{t-l}\rangle\,|f(x)\rangle
\end{aligned}$$

There are more involved ways to simulate determine computations with a quantum computer that require less overhead. The table below lists the quantum time and space requirements of known approaches as functions of the time t and space s of the deterministic computation that is being simulated. The table demonstrates a time-space trade-off.

Table 1: Time and space complexity of quantum simulation of a deterministic computation running in time t and space s.

time	space	
$O(t)$	$O(s+t)$	[folklore]
$\text{poly}(t, 2^s)$	$O(s+\log t)$	[LMT00]
$O(t^{1+\delta})$	$O(\delta 2^{1/\delta} s \log t)$	[Ben89]

Bennett's approach can be understood as divide-and-conquer applied to our approach. Any deterministic computation can be split in two, so we can apply our protocol to the first half of the computation and then to the second half. As the ancillas used for the first half can be reused for the second half, the total number of required ancillas is halved. As the garbage removal for the second part requires running the simulation for the first half one more time (in reverse), the running time gets doubled. Applying the recursion $O(\log t)$ times yields the entry in the table.

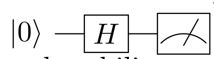
7 Simulating Probabilistic Computations

As mentioned, in both the probabilistic and the quantum setting, the state of a system can be described as a superposition of basis states. There are two differences:

- The normalization condition for a quantum state is that the sum of the squares of the amplitudes of the components sum to 1 whereas the normalization condition for a probabilistic system is that the sum of the (unsquared) amplitudes of the components sum to 1. For example, $\frac{1}{2}(|00\rangle + |11\rangle)$ is a valid probabilistic state but is not a valid quantum state and $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ is a valid quantum state but is not a valid probabilistic state.

- The components of a quantum state can have negative (or even complex) coefficients, whereas the components of a probabilistic system can only have nonnegative real coefficients. For example $\frac{1}{2}(|00\rangle + i\,|11\rangle)$ and $\frac{1}{2}(|00\rangle - |11\rangle)$ are not valid probabilistic states but $\frac{1}{\sqrt{2}}(|00\rangle + i\,|11\rangle)$ and $\frac{1}{\sqrt{2}}(|00\rangle - |11\rangle)$ are valid quantum states. Quantum systems that have identical state probabilities under measurement but that are not identical up to scalar multiplication are said to have phase differences. For example, $|a\rangle = \frac{1}{\sqrt{2}}(|00\rangle + i\,|11\rangle)$ and $|b\rangle = \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle)$ both have equal probability of yielding $|00\rangle$ or $|11\rangle$ when measured, but there is no complex number z such that $|a\rangle = z\,|b\rangle$, so $|a\rangle$ and $|b\rangle$ have phase differences.

Quantum systems can simulate a fair coin flip by applying the H gate to $|0\rangle$ and then observing the state of that qubit. A circuit representation for this procedure is:



Note that such a simulation requires the ability to measure some of the qubits of a system while leaving other qubits unmeasured (*partial measurement*), and also requires the ability to make a measurement that precedes some of the quantum gates rather than occurring after all of the quantum gates are applied (an *intermediate measurement*).

For both quantum and probabilistic systems, when a partial measurement is made, all components that are consistent with the measurement are retained with final amplitudes that are proportional to the amplitudes of those components prior to the measurement. For example:

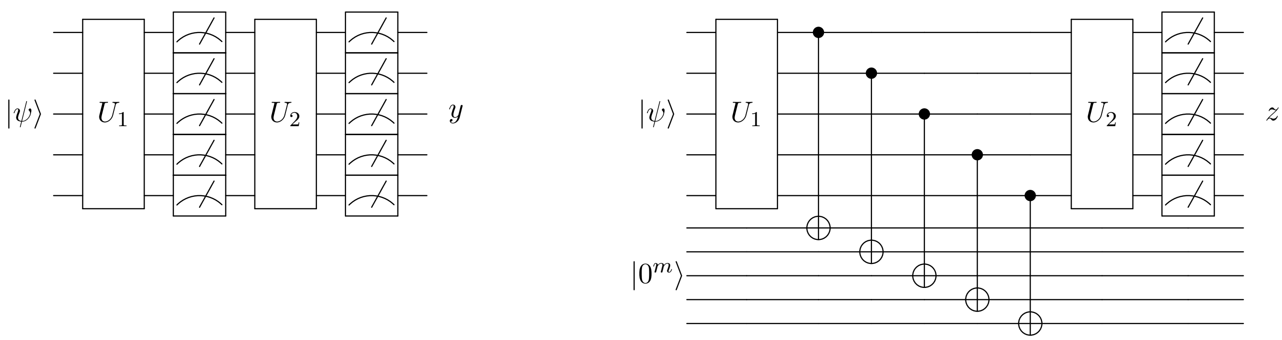
- For a probabilistic system with state $\frac{1}{3}(|01\rangle + |10\rangle + |11\rangle)$, a measurement of the first bit yields 0 with probability $\frac{1}{3}$ and 1 with probability $\frac{2}{3}$. In the case that 0 is observed, the new state is $|01\rangle$, and alternatively, if 1 is observed, than the new state is $\frac{1}{2}(|10\rangle + |11\rangle)$.

- For a quantum system with state $\frac{1}{\sqrt{3}}(|01\rangle + |10\rangle - |11\rangle)$, a measurement of the first bit yields 0 with probability $\frac{1}{3}$ and 1 with probability $\frac{2}{3}$. In the case that 0 is observed, the new state is $|01\rangle$, and alternatively, if 1 is observed, than the new state is $\frac{1}{\sqrt{2}}(|10\rangle - |11\rangle)$.

We use this opportunity to dismiss another common misconception about the power of quantum computing, namely that it would derive from the fact that quantum systems can be in entangled states whereas probabilistic systems cannot. In both the probabilistic and the quantum setting, states may be entangled, meaning that they cannot be decomposed into the product of two states on disjoint parts of the system. The probabilistic and quantum states in the above example are both entangled and behave similarly with respect to the following experiment: First measure the first component and then the second one. If the first component is measured and yields a 0, we know for sure that the subsequent measurement of the second component yields a 1. If the first component is measured and yields a 1, then the subsequent measurement of the second component yields a 1 with only 50% chance.

8 Exercise #2 – Deferring measurements

Consider the following two circuits starting from same pure state $|\psi\rangle := \sum_{s\in\{0,1\}^m} \alpha_s |s\rangle$, where U_1 and U_2 are unitary transformations and y and z are the results of the measurements. In the second circuit, all of the qubits that are not acted on by U_1 are intialized to $|0\rangle$ and the qubits that are acted on by U_1 are initialized to $|\psi\rangle$.



(a) Show that the distributions of y and z is the same.

(b) What if intermediate measurements on the left are removed, or equivalently, the CNOTs on the right are removed?

References

[Ben89] Charles H. Bennett. Time/space trade-offs for reversible computation. *SIAM J. Comput.*, 18(4):766–776, 1989.

[LMT00] Klaus-Jörn Lange, Pierre McKenzie, and Alain Tapp. Reversible space equals deterministic space. *J. Comput. Syst. Sci.*, 60(2):354–367, 2000.

[NC16] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information (10th Anniversary edition).* Cambridge University Press, 2016.