Overview of Quantum Algorithms

Samyak Surti

Quantum Computing at Berkeley

Spring 2022

Why should we care about Quantum Algorithms?

From paper...

Quantum computers are designed to outperform standard computers by running quantum algorithms. - *Ashley Montanaro npj Quantum Information (2016)*

Why should we care about Quantum Algorithms?

From paper...

Quantum computers are designed to outperform standard computers by running quantum algorithms. - *Ashley Montanaro npj Quantum Information (2016)*

- Primary allure: Achieve speedups (or other efficiency improvements) that no classical algorithms can hope to achieve.
- Applications include...

Why should we care about Quantum Algorithms?

From paper...

Quantum computers are designed to outperform standard computers by running quantum algorithms. - *Ashley Montanaro npj Quantum Information (2016)*

- Primary allure: Achieve speedups (or other efficiency improvements) that no classical algorithms can hope to achieve.
- Applications include...
  - Cryptography (i.e. Shor's Factoring Algorithm)
  - Search and Optimization
  - Simulation of Quantum Systems (i.e. Feynman's motivation)
  - Solving linear systems of equations

Why should we care about Quantum Algorithms?

From paper...

Quantum computers are designed to outperform standard computers by running quantum algorithms. - *Ashley Montanaro npj Quantum Information (2016)*

- Primary allure: Achieve speedups (or other efficiency improvements) that no classical algorithms can hope to achieve.
- Applications include...
  - Cryptography (i.e. Shor's Factoring Algorithm)
  - Search and Optimization
  - Simulation of Quantum Systems (i.e. Feynman's motivation)
  - Solving linear systems of equations
  - ...and more!

Topics

- Measuring Quantum Speed-up
- Grover's Search and the Hidden Subgroup Problem
- Search and Optimization
- Quantum Simulation
- Quantum Walks
- Solving Linear Systems of Equations
- NISQ-Era Algorithm Development
- Zero-Qubit Applications?

How do we know Quantum Algorithms are faster?

Defining and Measuring Quantum Speedup

- Look to Computational complexity theory...

How do we know Quantum Algorithms are faster?

Defining and Measuring Quantum Speedup

- Look to Computational complexity theory...
- Some familiar (and not so familiar) complexity classes are defined:

How do we know Quantum Algorithms are faster?

Defining and Measuring Quantum Speedup

- Look to Computational complexity theory...
- Some familiar (and not so familiar) complexity classes are defined:

Computational Complexity Classes for Quantum Algorithms

- P - Deterministic classical computer, poly-time
- BPP - Probabilistic classical computer, poly-time
- BQP - Quantum computer, poly-time
- NP - Deterministic classical computer, check in poly-time
- QMA - Quantum computer, check in poly-time

Topics

- Measuring Quantum Speed-up
- Grover's Search and the Hidden Subgroup Problem
- Search and Optimization
- Quantum Simulation
- Quantum Walks
- Solving Linear Systems of Equations
- NISQ-Era Algorithm Development
- Zero-Qubit Applications?

The Famed Shor's Algorithm

- Shor realized one of the first practical applications of quantum computers $\longrightarrow$ integer factorization

The Famed Shor's Algorithm

- Shor realized one of the first practical applications of quantum computers $\longrightarrow$ integer factorization
- Best classical algorithm: $\exp\left(O(\log N)^{1/3}(\log\log N)^{2/3}\right)^{12}$
- Shor's algorithm: $O(\log N^3)$ (Exponential Speed-up!)

The Famed Shor's Algorithm

- Shor realized one of the first practical applications of quantum computers $\longrightarrow$ integer factorization
- Best classical algorithm: $\exp\left(O(\log N)^{1/3}(\log\log N)^{2/3}\right)^{12}$
- Shor's algorithm: $O(\log N^3)$ (Exponential Speed-up!)
- Why do we care: RSA!

The Famed Shor's Algorithm

- Shor realized one of the first practical applications of quantum computers $\longrightarrow$ integer factorization
- Best classical algorithm: $\exp\left(O(\log N)^{1/3}(\log\log N)^{2/3}\right)^{12}$
- Shor's algorithm: $O(\log N^3)$ (Exponential Speed-up!)
- Why do we care: RSA!

Example (Classical vs. Quantum)

- Classical - 768-bit number, supercomputers over 2 years $\longrightarrow$ 10^20 operations.
- Quantum - 2000-bit number, quantum computer with billion qubits (I know, scary), running for a day

The Famed Shor's Algorithm

The Hidden Subgroup Problem

- Shor's approach based on special case of solution to Hidden Subgroup Problem

Definition (The Hidden Subgroup Problem)

Suppose we're given a group G and a function $f : G$ for some finite set S. Suppose f has the following property: there exists a subgroup $H \subseteq G$ such that f is constant within each coset, and distinct on different cosets. Find this H.

The Famed Shor's Algorithm

The Hidden Subgroup Problem

- Shor's approach based on special case of solution to Hidden Subgroup Problem

Definition (The Hidden Subgroup Problem)

Suppose we're given a group G and a function $f : G$ for some finite set S. Suppose f has the following property: there exists a subgroup $H \subseteq G$ such that f is constant within each coset, and distinct on different cosets. Find this H.

- Shor's algorithm solves the above problem for the case that $G = \mathbb{Z}$.

Topics

- Measuring Quantum Speed-up
- Grover's Search and the Hidden Subgroup Problem
- Search and Optimization
- Quantum Simulation
- Quantum Walks
- Solving Linear Systems of Equations
- NISQ-Era Algorithm Development
- Zero-Qubit Applications?

Definition (The Unstructured Search Problem)

Given we are able to evaluate an arbitrary function of the form $f : \{0,1\}^n \to \{0,1\}$ find x such that $f(x) = 1$, if such an x exists.

Search and Optimization

The Unstructured Search Problem and Grover

Definition (The Unstructured Search Problem)

Given we are able to evaluate an arbitrary function of the form $f : \{0,1\}^n \to \{0,1\}$ find x such that $f(x) = 1$, if such an x exists.

- Grover's Algorithm addresses just this!

Search and Optimization

The Unstructured Search Problem and Grover

Definition (The Unstructured Search Problem)

Given we are able to evaluate an arbitrary function of the form $f : \{0,1\}^n \to \{0,1\}$ find x such that $f(x) = 1$, if such an x exists.

- Grover's Algorithm addresses just this!
- Classical - must evaluate f 2^n times in worst case.
- Quantum - $O(\sqrt{N})$ evalutions of f

Search and Optimization

The Unstructured Search Problem and Grover

Definition (The Unstructured Search Problem)

Given we are able to evaluate an arbitrary function of the form $f : \{0,1\}^n \to \{0,1\}$ find x such that $f(x) = 1$, if such an x exists.

- Grover's Algorithm addresses just this!
- Classical - must evaluate f 2^n times in worst case.
- Quantum - $O(\sqrt{N})$ evalutions of f
  - Slight drawback - Fails with probability ϵ (arbitrary small)

Applications of Grover's Search Algorithm

The Complexity Class NP

- Recall from above...

NP Complexity Class

Problems whose solutions can be checked in polynomial time on a deterministic, classical computer

- Problems usually related to optimization and constraint satisfiability/satisfaction

Applications of Grover's Search Algorithm

The Complexity Class NP

- Recall from above...

NP Complexity Class

Problems whose solutions can be checked in polynomial time on a deterministic, classical computer

- Problems usually related to optimization and constraint satisfiability/satisfaction
- Suppose the existence of a classical checking algorithm $\mathcal{A}$.

Applications of Grover's Search Algorithm

The Complexity Class NP

- Recall from above...

NP Complexity Class

Problems whose solutions can be checked in polynomial time on a deterministic, classical computer

- Problems usually related to optimization and constraint satisfiability/satisfaction
- Suppose the existence of a classical checking algorithm $\mathcal{A}$.
  - If answer "yes" $\rightarrow$ have certificate of proof
  - If "no" $\rightarrow$ no certificate

Performance of Grover's Search Algorithm

Circuit SAT

Example (NP-complete Circuit Satisfiability Problem)

- Setup: Electronic circuit comprised of AND, OR, and NOT gates. Take in n bits, output 1 bit.
- Goal: Determine if there exists an input such that the output is 1.

Performance of Grover's Search Algorithm

Circuit SAT

Example (NP-complete Circuit Satisfiability Problem)

- Setup: Electronic circuit comprised of AND, OR, and NOT gates. Take in n bits, output 1 bit.
- Goal: Determine if there exists an input such that the output is 1.

- Best known classical algorithms - Worse-case run time of order 2^n for $ninputvariables$
- Grover's Algorithm - run time of $O(2^{n/2}poly(n))$

Performance of Grover's Search Algorithm

Circuit SAT

Example (NP-complete Circuit Satisfiability Problem)

- Setup: Electronic circuit comprised of AND, OR, and NOT gates. Take in n bits, output 1 bit.
- Goal: Determine if there exists an input such that the output is 1.

- Best known classical algorithms - Worse-case run time of order 2^n for $ninputvariables$
- Grover's Algorithm - run time of $O(2^{n/2}poly(n))$
  - Not exponential, but quadratic speed up achievable.

Generalizing Grover's Algorithm - Amplitude Amplification

The Heuristic Search Problem

Definition (The Heuristic Search Problem)

Suppose we can evaluate a probabilistic guessing algorithm $\mathcal{A}$ and a checking function f. Given probability ϵ of $\mathcal{A}$ outputting w such that $f(w) = 1$, can we output w such that $f(w) = 1$.

Generalizing Grover's Algorithm - Amplitude Amplification

The Heuristic Search Problem

Definition (The Heuristic Search Problem)

Suppose we can evaluate a probabilistic guessing algorithm $\mathcal{A}$ and a checking function f. Given probability ϵ of $\mathcal{A}$ outputting w such that $f(w) = 1$, can we output w such that $f(w) = 1$.

- Classical approach - Run $\mathcal{A}$ many times over and check each output using $f \rightarrow O(1/\epsilon)$ evaluations of f
- Amplitude Amplification - $O(1/\sqrt{\epsilon})$ evaluations of f (Quadratic speed up!)

Applications of Grover's Algorithm and Amplitude Amplification

Example (Applications)

- Finding the minimum of unsorted list of N integers.
- Determining if graph of N vertices is connected.
- Pattern matching
  - Music, speech, bioinformatics, etc...

Adiabatic Optimization

- Alternate approach to quantum combinatorial optimization $\rightarrow$ can solve any constraint satisfaction problem (CSP)

Formulation of Solution

- Prepare quantum state in equal superposition of all solutions to CSP
- Evolve Hamiltonian as slowly as possible with respect to energy gap $\rightarrow$ keeps quantum state in ground state
- Arrive at ground state maximizing most number of constraints $\rightarrow$ solution

Complexity of Quantum Adiabatic Algorithm, Practicality of Application and Drawbacks

- No well-defined or rigorous worst-case upper bound on run-time.
- Effect: Can define problems that take quantum adiabatic algorithm exponential time

Complexity of Quantum Adiabatic Algorithm, Practicality of Application and Drawbacks

- No well-defined or rigorous worst-case upper bound on run-time.
- Effect: Can define problems that take quantum adiabatic algorithm exponential time
- Adiabatic algorithm can be applied on universal quantum computer
  - D-Wave Systems' Quantum Annealers demonstrated outperformance of classical solvers in certain instances.
  - Such systems may not remain in ground-state

Topics

- Measuring Quantum Speed-up
- Grover's Search and the Hidden Subgroup Problem
- Search and Optimization
- Quantum Simulation
- Quantum Walks
- Solving Linear Systems of Equations
- NISQ-Era Algorithm Development
- Zero-Qubit Applications?

Quantum Simulation

Inspiration from Feynman (Also our favorite quote)

Nature isn't classical, dammit, and if you want to make a simulation of nature, **you'd better make it quantum mechanical**, *and by golly it's a wonderful problem, because it's so easy*

Quantum Simulation

Inspiration from Feynman (Also our favorite quote)

Nature isn't classical, dammit, and if you want to make a simulation of nature, **you'd better make it quantum mechanical**, *and by golly it's a wonderful problem, because it's so easy*

Quantum Simulation

Inspiration from Feynman (Also our favorite quote)

Nature isn't classical, dammit, and if you want to make a simulation of nature, **you'd better make it quantum mechanical**, *and by golly it's a wonderful problem, because it's so easy*

- Concretely for Quantum Computing: Given a Hamiltonian H describing a physical system and an initial state $|\psi\rangle$, measure properties of interest of time-evolved state $|\psi_t\rangle = e^{iHt} |\psi\rangle$

Quantum Simulation

Inspiration from Feynman (Also our favorite quote)

Nature isn't classical, dammit, and if you want to make a simulation of nature, **you'd better make it quantum mechanical**, *and by golly it's a wonderful problem, because it's so easy*

- Concretely for Quantum Computing: Given a Hamiltonian H describing a physical system and an initial state $|\psi\rangle$, measure properties of interest of time-evolved state $|\psi_t\rangle = e^{iHt} |\psi\rangle$
- Applications include quantum chemistry, superconductivity, metamaterials, and high-energy physics

The Two Approaches to Quantum Simulation

- Digital Quantum Simulation
  - Assumes possession of practical, fault-tolerant, error-correcting quantum computer and use it to run quantum simulation
- Analog Quantum Simulation
  - Approximate behavior of one physical system by constructing another.

Topics

- Measuring Quantum Speed-up
- Grover's Search and the Hidden Subgroup Problem
- Search and Optimization
- Quantum Simulation
- Quantum Walks
- Solving Linear Systems of Equations
- NISQ-Era Algorithm Development
- Zero-Qubit Applications?

Quantum Walks

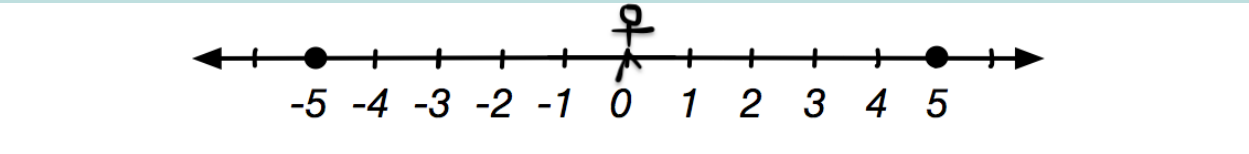
- Classical random walks useful for search and sampling problems.

Quantum Walks

- Classical random walks useful for search and sampling problems.

Example (Coined Random Walk)

Imagine you're standing at 0 of the real line and you flip a coin.



- Heads - Go to the right 1
- Tails - Go the left 1

Quantum Walks

"Quantumness" applied to Random Walks

- Quantum Walk based on quantum evolution of particle moving on graph.

Quantum Walks

"Quantumness" applied to Random Walks

- Quantum Walk based on quantum evolution of particle moving on graph.
- Advantages of Quantum Walk Algorithms:
  1. Faster Hitting Time (Finding target)
  2. Faster Mixing Time (Spreading across graph)

Quantum Walks

"Quantumness" applied to Random Walks

- Quantum Walk based on quantum evolution of particle moving on graph.
- Advantages of Quantum Walk Algorithms:
  1. Faster Hitting Time (Finding target)
  2. Faster Mixing Time (Spreading across graph)

Graphs for which Quantum Walks have Advantage

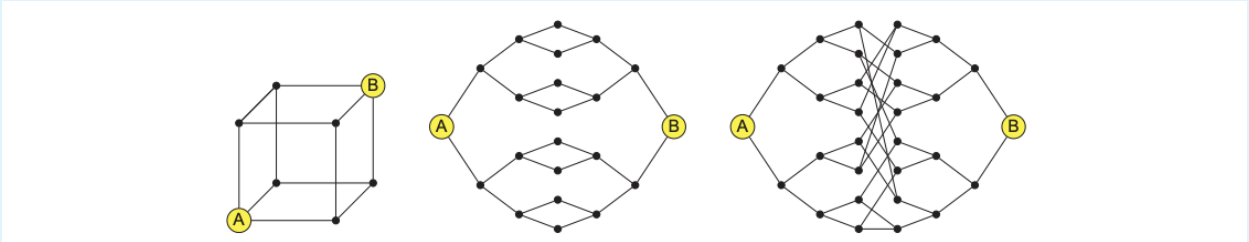


Figure: Quantum walks exhibit faster hitting times when traversing hybercubes, 'glued trees' graphs, and 'glued trees' graphs with a random cycle inserted

Application of Quantum Walks

Example (Evaluation of Boolean Formulae)

Boolean Formulae *on N binary inputs* $x_1, ..., x_N$ *is a tree whose internal vertices represent logical operations that are applied to their children.* N *leaves are labeled with these* N *binary inputs*

- Classical Algorithm - $N^{0.753...}$ worst case
- Quantum Algorithm - $O(N^{1/2})$
- Special case where quantum speed up is realized $\rightarrow$ evaluation of AND-OR trees.

Application of Quantum Walks

Cont.

Example (Algorithms Based on Markov Chains)

Discrete-Time Markov Chain *is a stochastic process whose evolution is determined in terms of a transition matrix*

- Quantum Algorithms have general speed-up over classical algorithms employing Markov Chains

Topics

- Measuring Quantum Speed-up
- Grover's Search and the Hidden Subgroup Problem
- Search and Optimization
- Quantum Simulation
- Quantum Walks
- Solving Linear Systems of Equations
- NISQ-Era Algorithm Development
- Zero-Qubit Applications?

Solving Linear Systems of Equations

- Hugely important in various areas of mathematics and engineering
- Algorithm devised by Harrow, Hassidim, and Lloyd (HHL) set to solve equations of form

$$Ax = b$$

where A is an $N \times N$ matrix, $b \in \mathbb{R}^N$

Solving Linear Systems of Equations

- Hugely important in various areas of mathematics and engineering
- Algorithm devised by Harrow, Hassidim, and Lloyd (HHL) set to solve equations of form

$$Ax = b$$

  where A is an $N \times N$ matrix, $b \in \mathbb{R}^N$
- Constraints:
  1. Sparse matrix
  2. Small condition number (measures numerical instability)

Solving Linear Systems of Equations

- Hugely important in various areas of mathematics and engineering
- Algorithm devised by Harrow, Hassidim, and Lloyd (HHL) set to solve equations of form

$$Ax = b$$

  where A is an $N \times N$ matrix, $b \in \mathbb{R}^N$
- Constraints:
  1. Sparse matrix
  2. Small condition number (measures numerical instability)
- Output is n-dimensional quantum state whose properties of interest we can measure.

Topics

- Measuring Quantum Speed-up
- Grover's Search and the Hidden Subgroup Problem
- Search and Optimization
- Quantum Simulation
- Quantum Walks
- Solving Linear Systems of Equations
- **NISQ-Era Algorithm Development**
- **Zero-Qubit Applications?**