Search via quantum walk

Ashwin Nayak
University of Waterloo, and
Perimeter Institute for Theoretical Physics

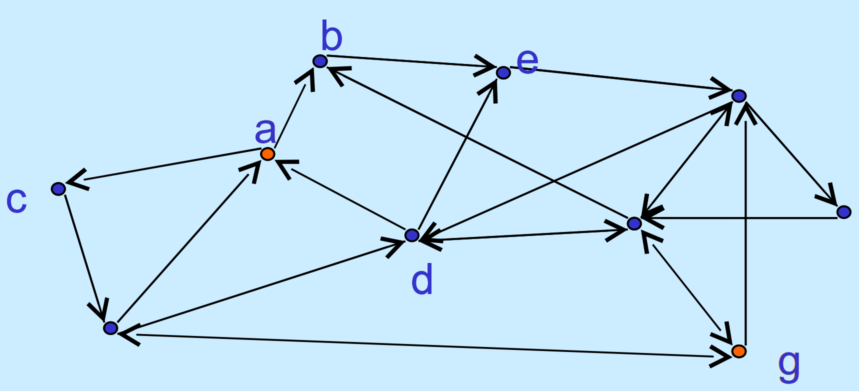
Joint work with

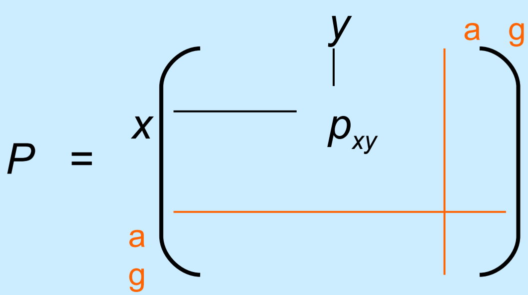
Frédéric Magniez[1], Jérémie Roland[2], Miklos Santha[1]
[1]LRI-CNRS, France, [2]UC Berkeley

Abstract search problem

- Input:
  - Set X = {a, b, c, …}
  - Marked elements M subset of X (say, {a, g})
  - Procedure to answer "x in M ?"
- Output:
  - Some element x in M.

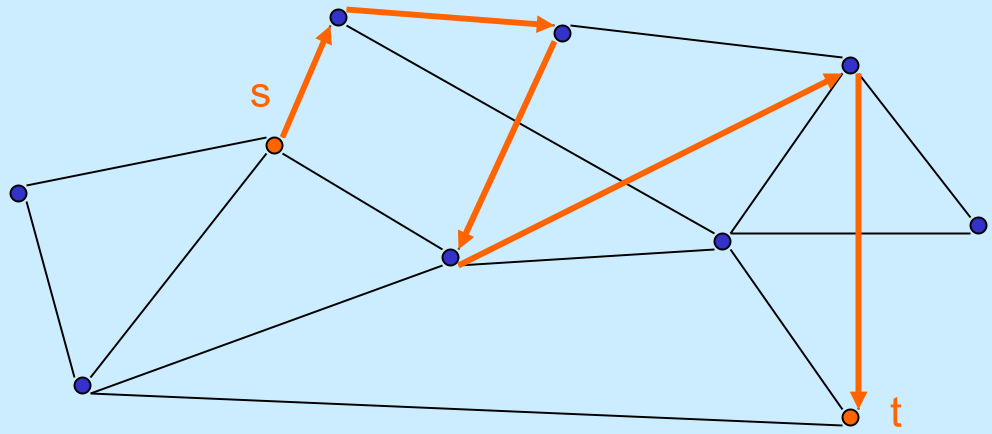
- Additional structure: Markov chain P on X





Random walk for search

- (*s*,*t*)-Connectivity
  - Input: Graph *G* on *n* vertices, two specified vertices *s*,*t*
  - Question: is there is a path from *s* to *t* ?



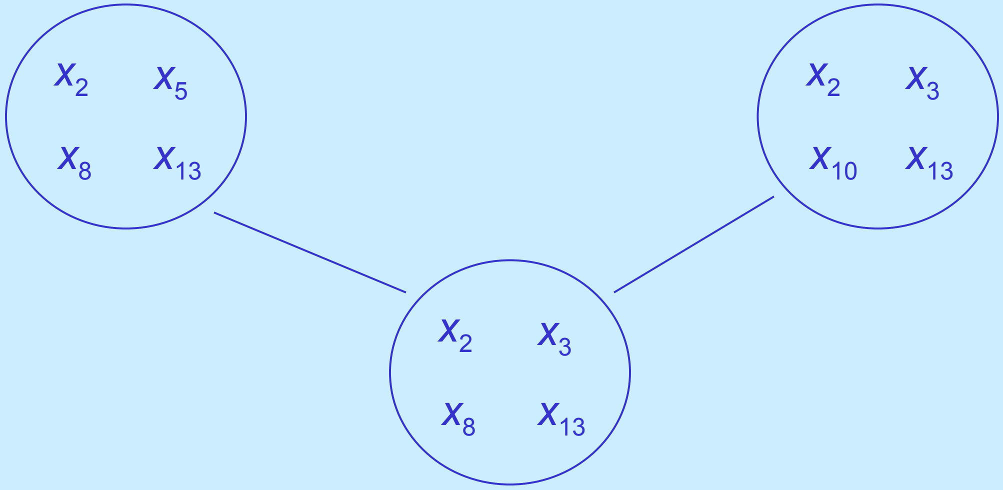
- Algorithm: start at $u = s$, and repeat $O(n^3)$ times
  - Pick a random vertex *v* adjacent to *u*
  - If $v = t$, stop. Else, set $u = v$.

Second example

- Element Distinctness (ED)
  - Input: list of n numbers $\{x_1, x_2, x_3, \ldots, x_n\}$
  - Question: are all the numbers distinct
    (or is there a collision: $x_i = x_j$, $i \neq j$)
- Deterministic Algorithm:
  - Sort elements; check if consecutive numbers are equal
  - Time complexity: $O(n \log n)$
- Not graph search, but can be recast as one.

Element distinctness as graph search

- Johnson Graph (n, r)
  - Vertices: size r subsets of $\{1, 2, \ldots, n\}$
  - Edges: $\{S, T\}$ is an edge iff they differ by 2 elements
- Example: $n = 15$, $r = 4$



- Search for subset with collision

Randomized algorithm for ED

- Start at a random vertex of the Johnson graph

  Pick r indices uniformly at random to form a set S;

  sort the elements x_i for i in S;

  check for collisions.

- Repeat for T_1 steps
  - Perform a random walk on the graph for T_2 steps

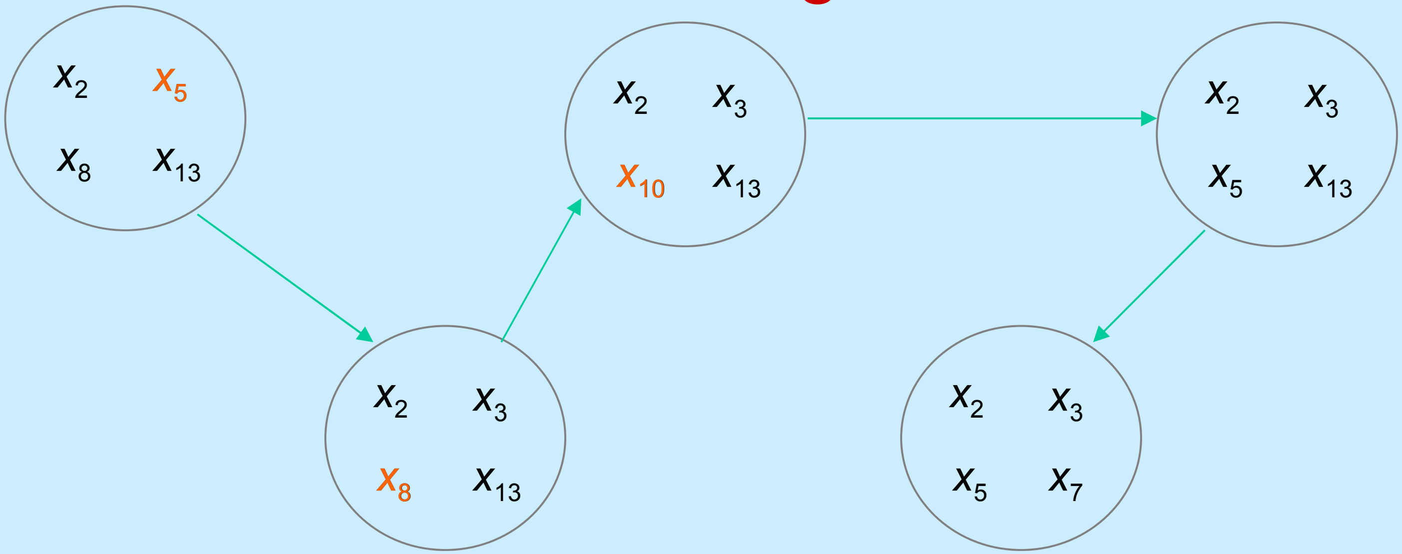
    In each step, swap random element i in S and j not in S;

    remove x_i, insert x_j into sorted list
  - check for a collision in S

- If no collision is found, output “no collision”.

  (Less natural algorithm, but adapts well to quantum)

Randomized algorithm for ED



- Intuition:
  - In $T_2 = O(r)$ steps of walk, S is nearly uniformly distributed
  - Pr[collision in random S] $\approx (r/n)^2$
  - So in $T_1 = O((n/r)^2)$ repetitions, a collision will be found

- Runtime: $r \log r + T_1 (T_2 \log r + 1)$

Set up cost — update cost — checking cost

Speed-up via quantum walk

- Quantum analogue of randomized algorithm
- Speeds up both T_1 and T_2 quadratically

[Ambainis ‘04]

- Run time of quantum algorithm for ED

$$r \log r \;+\; (n/r)\,(\, r^{1/2} \log r \;+\; 1\,)$$

$$n^{2/3} \log n \qquad (\text{setting } r = n^{2/3})$$

- *A second* algorithm, for symmetric Markov chains
- Quadratic speed-up in detecting marked elements

[Szegedy ’04]

This talk: New search algorithm

- Quantum walk from any irreducible Markov chain
- Algorithm finds a marked element, if any, from any M
- Run time: set-up + $T_1^{1/2}$ ($T_2^{1/2}$ update + check)

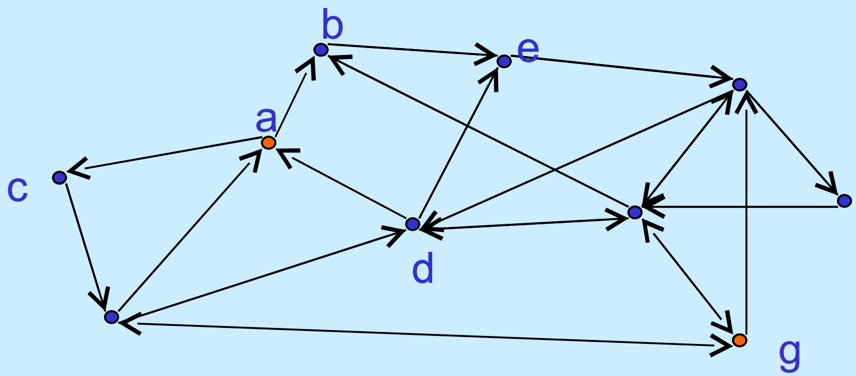
  $T_1^{1/2}$ → $\Pr(M)^{-1/2}$ $T_2^{1/2}$ → singular value gap$^{-1/2}$

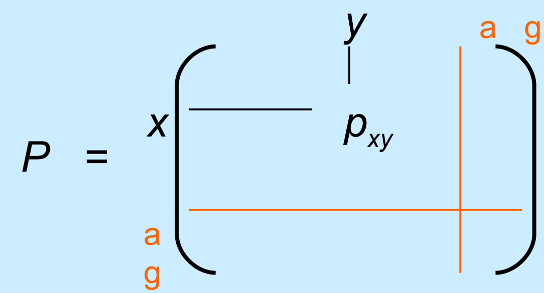
- Simple --- conceptually, and to analyze
- Unifies and improves several applications

Talk outline

- Classical algorithm
- Quantum walk
- Quantum subroutines
  - Amplitude amplification
  - Phase estimation
- Search algorithm

Classical search algorithm





- Start in some start distribution *s*

- Repeat for T_1 steps
  - Simulate T_2 steps of the Markov chain *P*
  - Check if current state is marked

- If no marked element is found, output “none marked”.

Complexity of classical strategy

- P symmetric (for simplicity), ergodic
- Uniform stationary distribution (1-eigenvector)

- Say we start in s = uniform distribution
- Run-time characterized by
  - Spectral gap $\delta(P) = 1 -$ second largest |eigenvalue|
  - Probability of marked elements $\varepsilon = \Pr(M) = |M| \,/\, |X|$

- Proposition

  Run-time of the classical strategy is

  set-up + $(1/\varepsilon)$ ($(1/\delta)$ update + check)

  $(1/\varepsilon) \rightarrow T_1$, $(1/\delta) \rightarrow T_2$

Talk outline

- Classical algorithm

  Run time = $1/\varepsilon\delta$

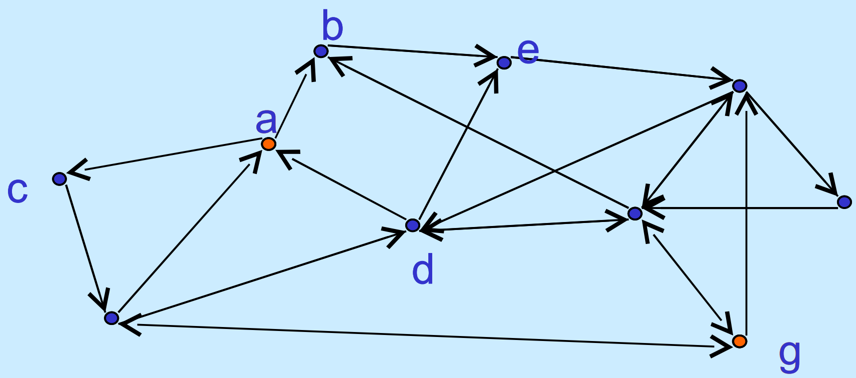
- Quantum walk

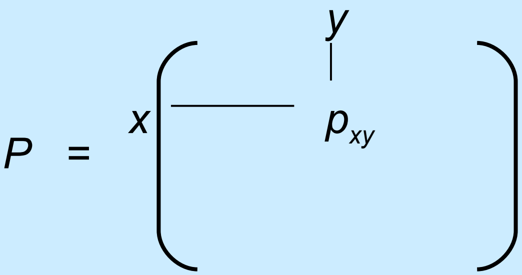
- Quantum subroutines
  - Amplitude amplification
  - Phase estimation

- Search algorithm

The quantum walk

[Watrous ’01, Szegedy ‘04]





- The quantum walk W(P)
  - State space: pairs of neighbouring vertices $|x\rangle\ |y\rangle$
  - Step of walk: diffuse y over neighbours of x, new nbr. y'
    then, diffuse x over neighbours of y'
  - Diffusion: analogous to Grover search operator
    (reflection about state $|x\rangle \sum_y \sqrt{p_{x,y}}\ |y\rangle$, for each x)

Spectrum of W(P)

[Szegedy '04]

- W(P) = product of two reflection operators
- Assume P is symmetric, ergodic
  Has uniform stationary distribution
- Spectrum of W(P) related to that of P
- For every singular value of P, $\sigma = \cos\theta$ in $(0,1)$
  W(P) has eigenvalues $\exp(\pm 2i\,\theta)$
- The remaining eigenvalues are ± 1

Spectral gap

- Largest singular value of $P = 1$, and is unique
  $W(P)$ has unique eigenvalue 1 (in walk subspace)

- Eigenvector of $W(P)$ with eigenvalue 1 is
  $$|\pi\rangle = (1/n^{1/2}) \ \Sigma_x \ |x\rangle|p_x\rangle \qquad \text{where}$$
  $$|p_x\rangle = \Sigma_y \, p_{xy}^{\ 1/2} \, |y\rangle$$

- If $\sigma = \cos\theta < 1$ is second largest singular value, eigenvalue gap of $W(P)$ is
  $$|\, 1 - \exp(\, 2i\,\theta\,)| \ \geq \ 2\,(1 - \sigma\,)^{1/2} \ = \ 2\,\delta(P)^{1/2}$$
  square-root of spectral gap of P

Talk outline

- Classical algorithm

  Run time = $1/\varepsilon\delta$

- Quantum walk

  Spectral gap = $\delta^{1/2}$

- Quantum subroutines
  - Amplitude amplification
  - Phase estimation

- Search algorithm

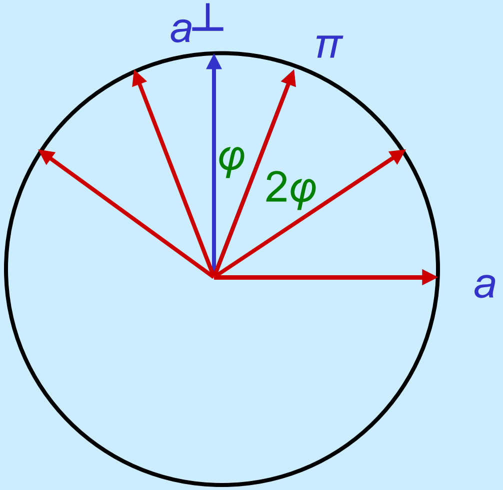
Amplitude amplification

[Grover ’96, BBHT ’98, …]

- Search for *one* out of n states

- Start state: $|\pi\rangle = (1/n^{1/2})\ \Sigma_x\ |x\rangle$
- Desired final state: $|a\rangle$

- Alternately reflect through $|a^\perp\rangle$ and $|\pi\rangle$



Complexity of amplitude amplification

- Angle of rotation $= 2\varphi$ $(\sin \varphi = 1/n^{1/2})$
- Number of iterations $\approx (\pi/2)/(2\varphi) \approx n^{1/2}$
- Required reflection operators have small circuits
- Multiple marked states
  - Fraction of marked states $\varepsilon = m/n$
  - target state $= (1/m)^{1/2} \Sigma_{x \text{ in } M} |x\rangle$
  - Angle of rotation $= 2\varphi$ $(\sin \varphi = (m/n)^{1/2} = \varepsilon^{1/2})$
  - Number of iterations $\approx 1/\varepsilon^{1/2}$
  - Quadratic speed-up over classical

Talk outline

- Classical algorithm

  Run time = $1/\varepsilon\delta$

- Quantum walk

  Spectral gap = $\delta^{1/2}$

- Quantum subroutines
  - Amplitude amplification

    Cost = $1/\varepsilon^{1/2}$

  - Phase estimation

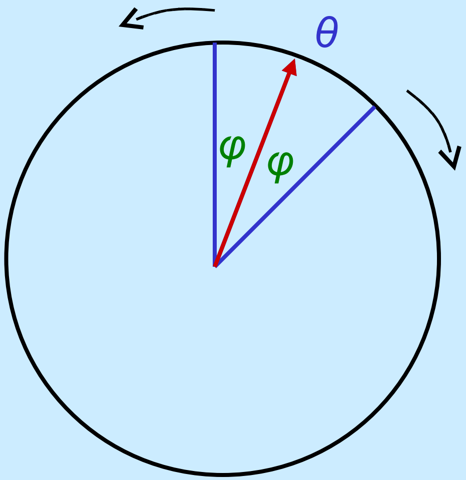
- Search algorithm

Phase estimation

- Input: circuit for unitary U
  superposition $|v\rangle$, eigenvector
  with unknown eigenvalue $\exp(2\pi i\theta)$
- Output: approximation to θ

- Proposition [Kitaev '95, Cleve, Ekert, Macchiavello, Mosca '98]
  Can compute an approximation to θ within η
  with $1/\eta$ repetitions of U, one copy of $|v\rangle$
  with probability 3/4

Reflection using phase estimation



U	unitary operator
v	isolated eigenvector
φ	spectral gap

Reflection through $|v\rangle$

- Run phase estimation algorithm on the current state, with U
- If approximate phase is “far” from θ, flip sign
- Undo phase estimation

Precision required ≈ $\varphi/2$

Repetitions of U ≈ $1/\varphi$ = 1/ spectral gap

Reflection via quantum walk W(P)

- $|\pi\rangle$ 1-eigenvector of W(P)
- $\delta^{1/2}$ spectral gap of W(P)

- Reflection through $|\pi\rangle$

  Use phase estimation, as described

  Repetitions of W(P) ≈ 1/ spectral gap ≈ $1/\delta^{1/2}$

Talk outline

- Classical algorithm

  Run time = $1/\varepsilon\delta$
- Quantum walk

  Spectral gap = $\delta^{1/2}$
- Quantum subroutines
  - Amplitude amplification

    Cost = $1/\varepsilon^{1/2}$
  - Phase estimation

    Cost = $1/\delta^{1/2}$

- Search algorithm

The search algorithm

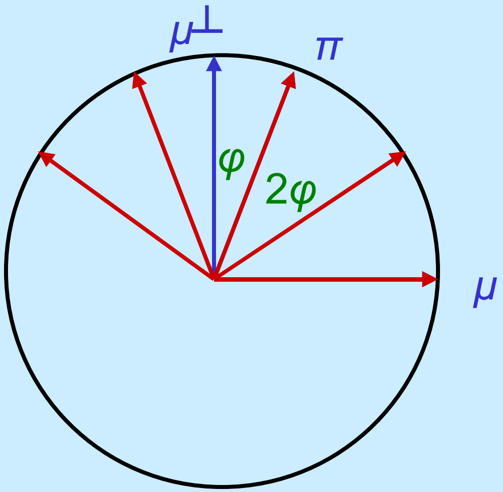
- Start state:

$$|\pi\rangle \quad = \quad (1/n^{1/2}) \;\; \Sigma_x \;\; |x\rangle|p_x\rangle$$

- Desired final state:

$$|\mu\rangle \quad = \quad (1/m^{1/2}) \;\; \Sigma_{x \text{ in } M} \;\; |x\rangle|p_x\rangle$$

- Alternately reflect through $|\mu^{\perp}\rangle$ and $|\pi\rangle$ *à la* Grover



Implementing the reflections

- Reflection through $|\mu^{\perp}\rangle$

  If vertex x in first register is marked,
  and second register is in state $|p_x\rangle$,
  then flip sign

- Reflection through $|\pi\rangle$

  Use phase estimation algorithm, as described

Complexity of the algorithm

- Angle between $|\mu^{\perp}\rangle$ and $|\pi\rangle$:

  $\sin \varphi = (m/n)^{1/2} = \varepsilon^{1/2}$,

  $\varepsilon = \Pr(M)$ = probability of M under stationary distribution
- Number of rotations *à la* Grover: $1/\varepsilon^{1/2}$
- Cost of reflection through $|\mu^{\perp}\rangle$

  check + update cost
- Cost of reflection through $|\pi\rangle$:

  update cost times $1/\delta^{1/2}$

  $\delta^{1/2}$ = spectral gap of $W(P)$
- Complexity

  set-up + $(1/\varepsilon^{1/2})$ ($(1/\delta^{1/2})$ update + check)

Final remarks

- Error due to imperfect phase estimation algorithm handled with a recursive search algorithm *à la* [Hoyer, Mosca, de Wolf ’04]
- Algorithm extends to any irreducible Markov chain
- Unified and improved algorithms for Element Distinctness, Triangle Finding, Matrix Product verification, Group Commutativity
- Better algorithms for applications in which checking cost is higher than update cost