Quantum Algorithms using the Curvelet Transform

Yi-Kai Liu

Institute for Quantum Information, Caltech

January 8, 2009

Supported by an NSF Mathematical Sciences Postdoctoral Fellowship

Quantum algorithms

- The quantum Fourier transform is a key component
  - On Abelian groups => Shor's algorithm
- Attempts to generalize this approach
  - On the dihedral group => approximating the unique shortest vector in a lattice?
  - On the symmetric group => graph isomorphism?

What about other unitary transforms?

- The curvelet transform
  - A directional wavelet transform on $\mathbf{R}^n$
  - Does this lead to interesting quantum algorithms?

Summary of our results

- Finding the center of a ball in $\mathbf{R}^n$ (approximately)
  - **Quantum:** given a single quantum-sample, can succeed with **constant probability**
    - "Quantum-sample" from a set S = uniform superposition over all points in S
  - **Classical:** given a single classical random sample, can only succeed with **exponentially small probability < $2^{-\Omega(n)}$**
- Finding the center of a radial function on $\mathbf{R}^n$
  - Given oracle access; assume a radial step function
  - **Quantum:** can succeed with probability > $\Omega(1)$, using a **constant number of oracle queries**
  - **Classical:** need **$\Omega(n)$ oracle queries**
- These are polynomial-time quantum algorithms

What is the curvelet transform?

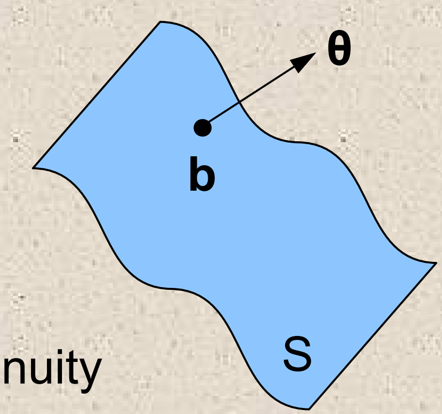
- A directional wavelet transform (Candes & Donoho, 1999, 2002)

f(**x**)
  x in $\mathbf{R}^n$ is a "location"

→

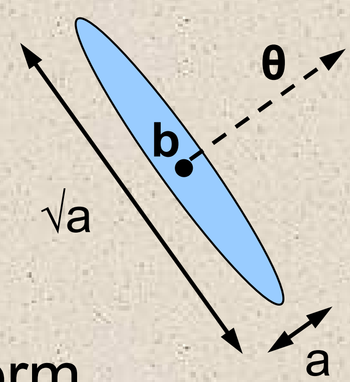
$\Gamma_f(a,\mathbf{b},\boldsymbol{\theta})$
  $0 < a < 1$ is a "scale"
  (0 = fine scale, 1 = coarse scale)
  b in $\mathbf{R}^n$ is a "location"
  θ in $\mathbf{R}^n$, $|\boldsymbol{\theta}| = 1$ is a "direction"

- Intuition:
  - Suppose f is discontinuous along a smooth surface S of dimension n–1
  - Then $\Gamma_f(a,\mathbf{b},\boldsymbol{\theta})$ is "large" whenever: **b** lies on S and **θ** is normal to S at **b**
    - A.k.a., the "wavefront set"
    - a measures the "sharpness" of the discontinuity



What is the curvelet transform?

- Curvelet basis functions look like plane waves, but localized in small regions of space
  - High-frequency oscillations in the **θ** direction, supported on a plate-like region centered at **b** and orthogonal to **θ**



- Note, this is the *continuous* curvelet transform, there is also a finite discrete version

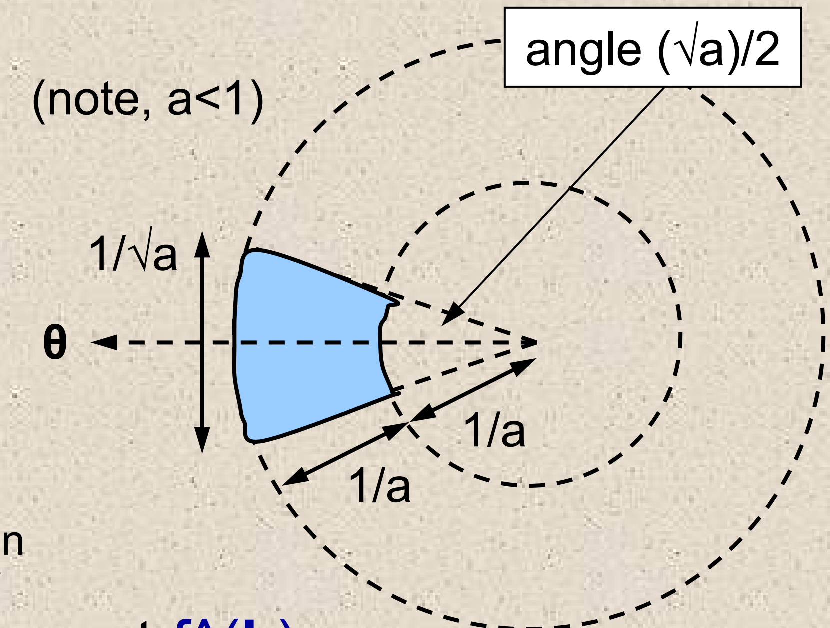
$f(\mathbf{x})$
Location **x** in $(\mathbf{Z}_M)^n$

→

$\Gamma_f(a,\mathbf{b},\boldsymbol{\theta})$
Location **b** in $(\mathbf{Z}_M)^n$
Scale a and direction **θ** from some finite set

The continuous curvelet transform

Window function $\chi_{a\theta}$ is supported on a sector of frequency space:



- Given $f(\mathbf{x})$, a function on $\mathbf{R}^n$
- Take the Fourier transform: get $f\hat{}(\mathbf{k})$
- For each scale a and direction $\boldsymbol{\theta}$, multiply by a smooth window function $\chi_{a\theta}$: get $f\hat{}(\mathbf{k})\,\chi_{a\theta}(\mathbf{k})$
- Take the inverse Fourier transform: get $\Gamma_f(a,\mathbf{b},\boldsymbol{\theta})$

A fast quantum curvelet transform

- Discrete curvelet transform
  - **b** in $(\mathbf{Z}_M)^n$; a, **θ** from some discrete set
  - Not unitary, but an isometric embedding: $|\psi) \rightarrow U(|\psi) \text{ tensor } |0))$
- Quantum curvelet transform
  - Given input state + ancilla: $\Sigma_{\mathbf{x}} f(\mathbf{x}) |\mathbf{x}) |0,\mathbf{0})$
  - Apply QFT: $\Sigma_{\mathbf{k}} f\hat{}(\mathbf{k}) |\mathbf{k}) |0,\mathbf{0})$
  - Prepare superposition: $\Sigma_{\mathbf{k}} f\hat{}(\mathbf{k}) |\mathbf{k}) \Sigma_{a,\boldsymbol{\theta}} \chi_{a,\boldsymbol{\theta}}(\mathbf{k}) |a,\boldsymbol{\theta})$
  - Apply QFT^{-1}: $\Sigma_{\mathbf{b}} \Sigma_{a,\boldsymbol{\theta}} \Gamma_f(a,\mathbf{b},\boldsymbol{\theta}) |\mathbf{b}) |a,\boldsymbol{\theta})$
- Can be computed efficiently
  - Takes time poly(n, log M)
  - Provided that the $\chi_{a,\boldsymbol{\theta}}(\mathbf{k})$ are products of 1-D functions
    - E.g., using spherical coordinates

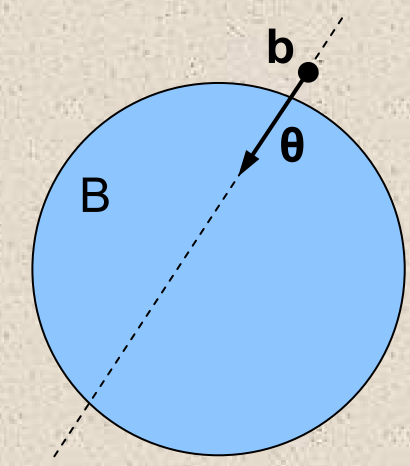
What can we do with this?

Finding the center of a ball

- Let B be a ball in $\mathbf{R}^n$

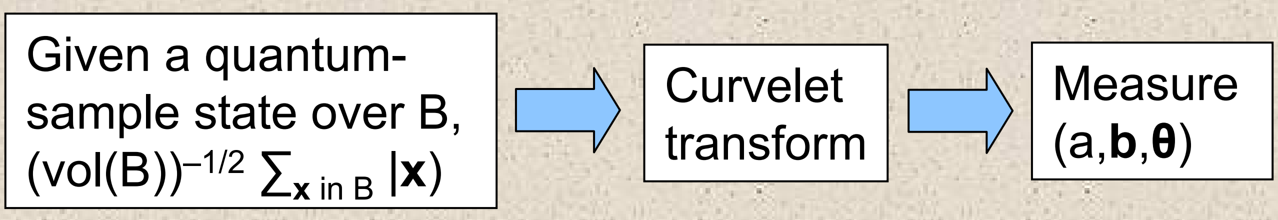
Given a quantum-sample state over B, $(\mathrm{vol}(B))^{-1/2} \sum_{\mathbf{x} \text{ in } B} |\mathbf{x})$ → Curvelet transform → Measure $(a,\mathbf{b},\boldsymbol{\theta})$

- Claim: with significant probability,
  - the scale a is small
  - the line $\mathbf{b}+\lambda\boldsymbol{\theta}$ passes near the center of the ball
- To find the center:
  - Guess some point along the line



Finding the center of a ball

- Let B be a ball in $\mathbf{R}^n$



- To find the center:
  - Guess some u in [-1,1], uniformly at random
  - Let β be the radius of the ball, and let C be some constant
  - Return the point $\mathbf{b} + uC\beta\boldsymbol{\theta}$
- Claim: for some constant $\kappa<1$, with constant probability, this point lies within distance $\beta\kappa$ of the center of the ball
  - Independent of the dimension n

Why is this interesting?

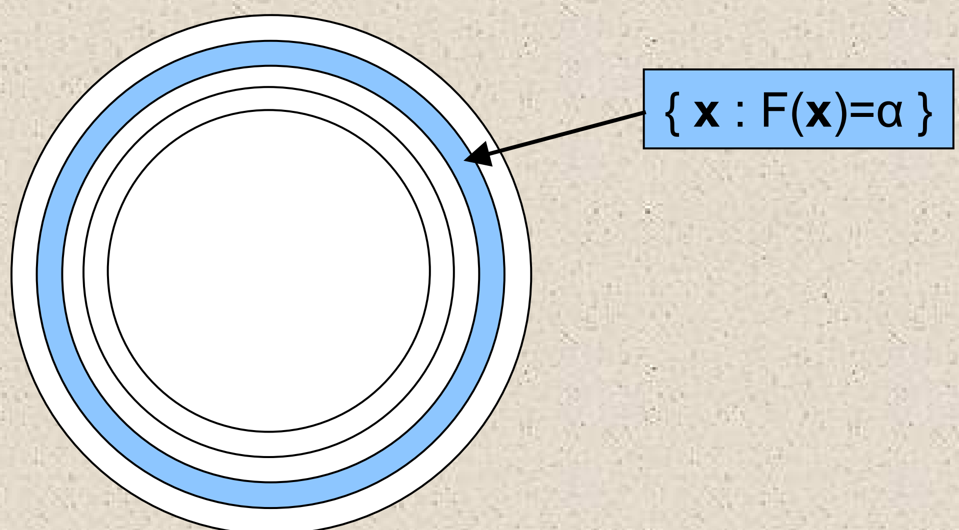
- Can get useful information from just **one** quantum-sample
  - "Single-shot quantum measurement"
  - For any constant $\kappa<1$, algorithm finds a point within distance $\beta\kappa$ of the center, **with probability > $\Omega(\kappa^3)$, independent of n**
  - Compare w/ classical sampling: if we picked a single random point in B, we would succeed **with probability κ^n, exponentially small in n**

- Why? Because volume is concentrated near the surface of the ball
  - Bad for classical sampling, good for the curvelet transform!
  - But advantage disappears if we are given more than one sample – we can take several classical samples and average them

- Caveats
  - To implement this algorithm, have to discretize, and use a slightly different family of curvelets

Finding the center of a radial function

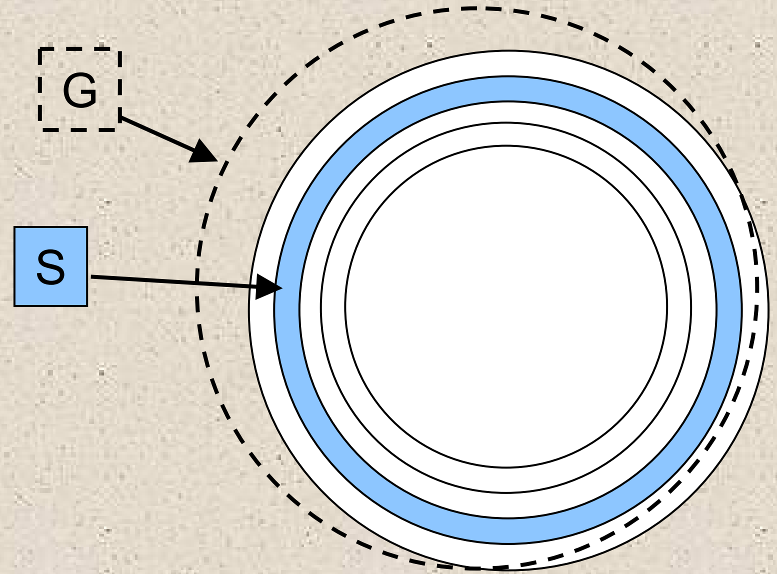
- Let F be a radial function on $\mathbf{R}^n$, centered at some unknown point **c**
  - F can return values in some arbitrary set; assume that the level sets of F are concentric spherical shells of thickness δ
  - We are given oracle access to F, and we are promised that the center **c** lies within distance R of the origin



Finding the center of a radial function

Prepare a quantum-sample state over a large region G, $(\mathrm{vol}(G))^{-1/2} \sum_{\mathbf{x} \text{ in } G} |\mathbf{x})$

Compute F(**x**) and measure; get a quantum-sample state over a spherical shell S, $(\mathrm{vol}(S))^{-1/2} \sum_{\mathbf{x} \text{ in } S} |\mathbf{x})$



Curvelet transform → Measure (a,**b**,**θ**) → Line $\mathbf{b}+\lambda\boldsymbol{\theta}$ passes near **c**

- To find the center **c**:
  - Do this twice, to find two lines L and L' that pass near **c**
  - Then return the point on L that lies nearest to L'

Why is this interesting?

- Claim: this algorithm finds the center "exactly," when δ is sufficiently small
  - Solution is more accurate when the shell is very thin
  - For any μ, we can find a point within distance μ of the center, provided that $\delta < O(\mu^2/Rn^2)$
- This only requires **O(1) oracle queries**, independent of the dimension n
  - Algorithm succeeds with probability > $\Omega(1)$, independent of n
  - Compare w/ classical case: seems to require **$\Omega(n)$ queries**
- Caveats
  - Our analysis uses an approximation for the spherical shell
  - To implement this algorithm, have to discretize, and use a slightly different family of curvelets

Related work

- Quantum algorithms
  - Estimating the gradient of a function on $\mathbf{R}^n$ (Jordan, 2004)
    - Works when the function is smooth
    - Uses Fourier transform + phase kickback
      – like computing the curvelet transform at a single location
  - Quantum wavelet transform (Hoyer, 1997; Fijany & Williams, 1998)
    - Can be implemented efficiently; any applications?
  - Finding "hidden nonlinear structures" (Childs, Schulman & Vazirani, 2007)
    - Shifted subsets – use the Fourier transform
    - Hidden polynomials – use curvelets?
- The (classical) curvelet transform
  - Image processing, and simulating wave propagation
  - Resolving the "wavefront set" (Candes & Donoho, 2002, 2003)
    - Different formulations of the problem, for general functions, only on $\mathbf{R}^2$

Proof ideas

- Curvelet transform of a radial function
  - Wlog, assume the object is centered at the origin
  - The probability of observing a fine scale element corresponds to the amount of power at high frequencies
    - $\Pr[a \leq \zeta] \approx \int_{|\mathbf{k}| \geq 1/(\lambda\zeta)} |\hat{f}(\mathbf{k})|^2 \, d\mathbf{k}$, we get better accuracy when a is small
    - High-frequency components are due to the discontinuity of f
  - The direction **θ** is uniformly distributed, and the location **b** has expected value **0**
  - We can upper-bound the variance of **b** in the directions orthogonal to **θ**
    - Use Plancherel's theorem to go from spatial to frequency domain
    - Integrate using n-dimensional spherical coordinates
    - Behavior of Bessel functions $J_v(z)$ in the transition regime $z \approx v$
    - Lots of fun…
  - Hence the line $\mathbf{b}+\lambda\boldsymbol{\theta}$ passes near the origin

Conclusions

- The curvelet transform
  - Suppose f is discontinuous along a smooth surface S
  - Then $|\Gamma_f|^2$ is large near the “wavefront set”:
    - Points **b** that lie on S, and directions **θ** that are normal to S
- The quantum curvelet transform
  - Can be computed efficiently, for a “nice” family of curvelets
- Finding the center in $\mathbf{R}^n$
  - Can find the center of a ball (approximately), using 1 quantum-sample
  - Can find the center of a radial function (exactly), using O(1) oracle queries
  - Future work: discretization, different families of curvelets, classical lower bounds better than $\Omega(n)$?

The big picture

- What is the quantum curvelet transform good for?
  - Can it solve some natural class of problems?
    - Like the hidden subgroup problem?
    - Exponential speed-up over classical computation?
  - Where does one get quantum states with “wavefront” features?
    - From quantum walks? Can one extract useful information?

- Generalizing our results on balls and spheres in $\mathbf{R}^n$
  - More complicated objects, e.g., ellipsoids, polytopes?
    - We can efficiently quantum-sample over convex bodies
  - Surfaces over finite fields?
    - These arise in hidden polynomial problems?

Any questions?

Photo by LePetitPoulailler, CC license,
http://www.flickr.com/photos/three_french_hens/2341682479/