Cryo-EM and NMR Structure Determination through Eigenvectors of Sparse Matrices

Amit Singer

Yale University, Department of Mathematics, Program in Applied Mathematics

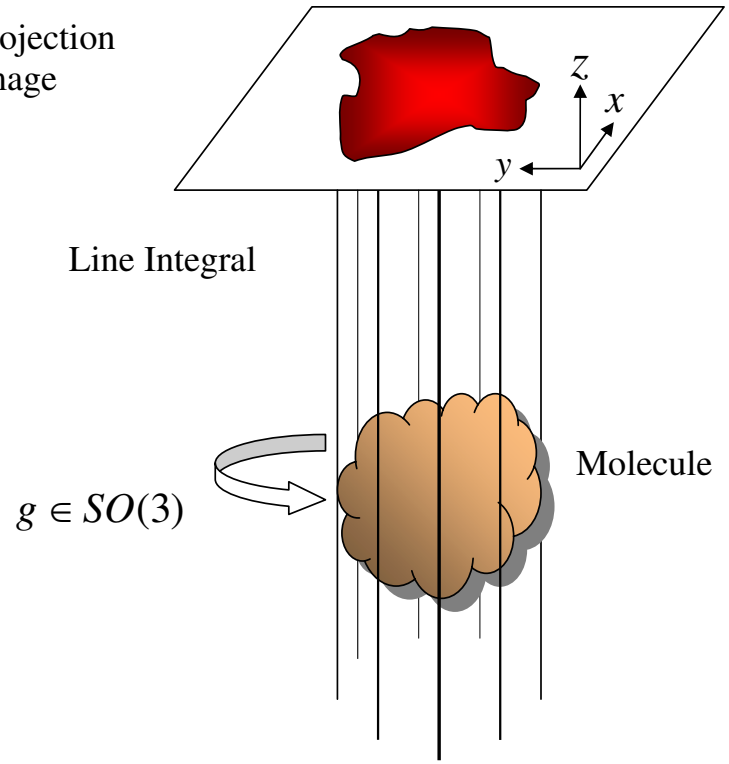
Search and Knowledge Building for Biological Datasets, IPAM, November 2007

Joint work with...

- Ronald Coifman (Yale University, Applied Mathematics)
- Yoel Shkolnisky (Yale University, Applied Mathematics)
- Fred Sigworth (Yale School of Medicine, Cellular & Molecular Physiology)
- Yuval Kluger (NYU, Department of Cell Biology)
- David Cowburn (New York Structural Biology Center)
- Yosi Keller (Bar Ilan University, Electrical Engineering)

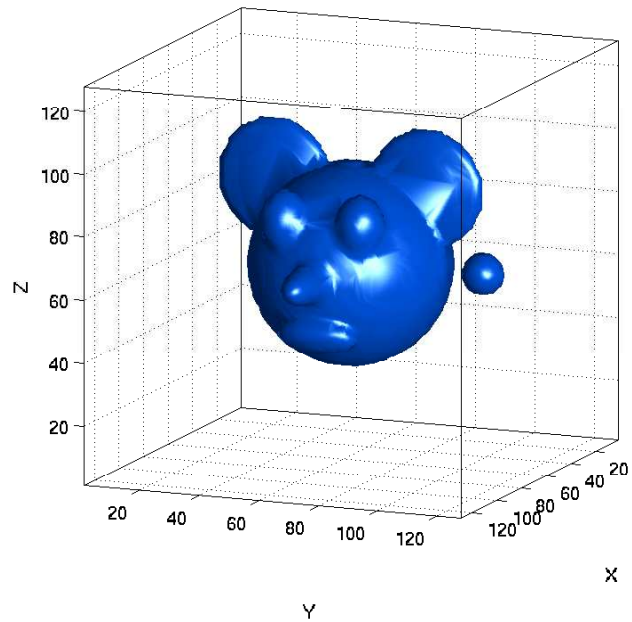
Three Dimensional Puzzle

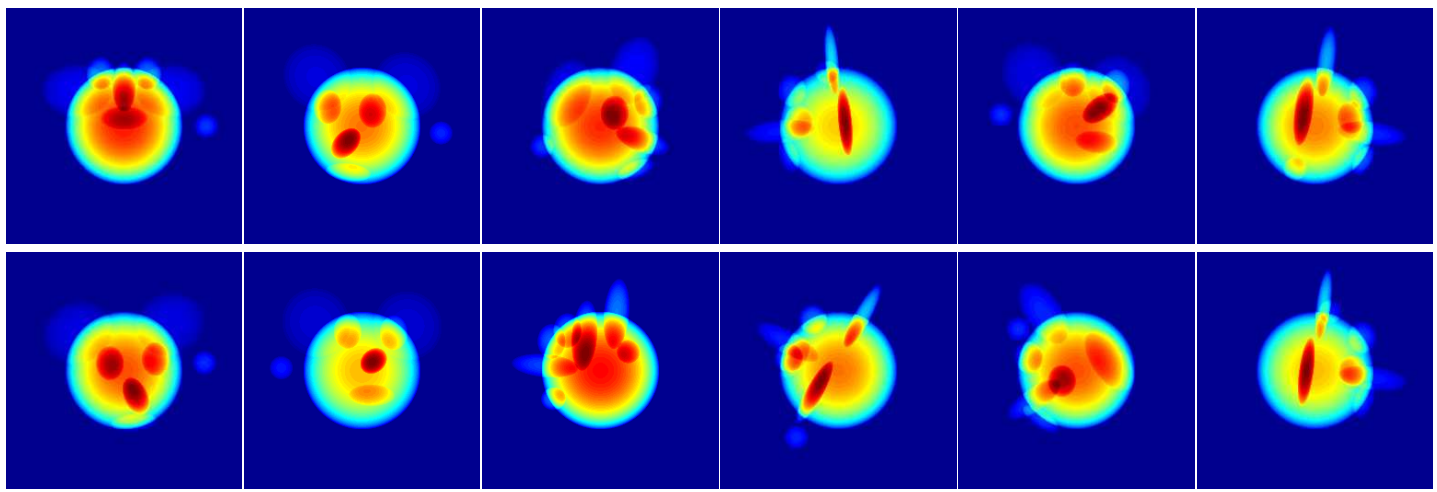
Cryo Electron Microscopy: Projection Images



- The projection image is $P_g(x, y) = \int_{-\infty}^{\infty} \phi_g(x, y, z)\, dz$.
- $\phi(r)$ is the electric potential of the molecule, $\phi_g(r) = \phi(g^{-1} r)$.

Projection Images: Toy Example

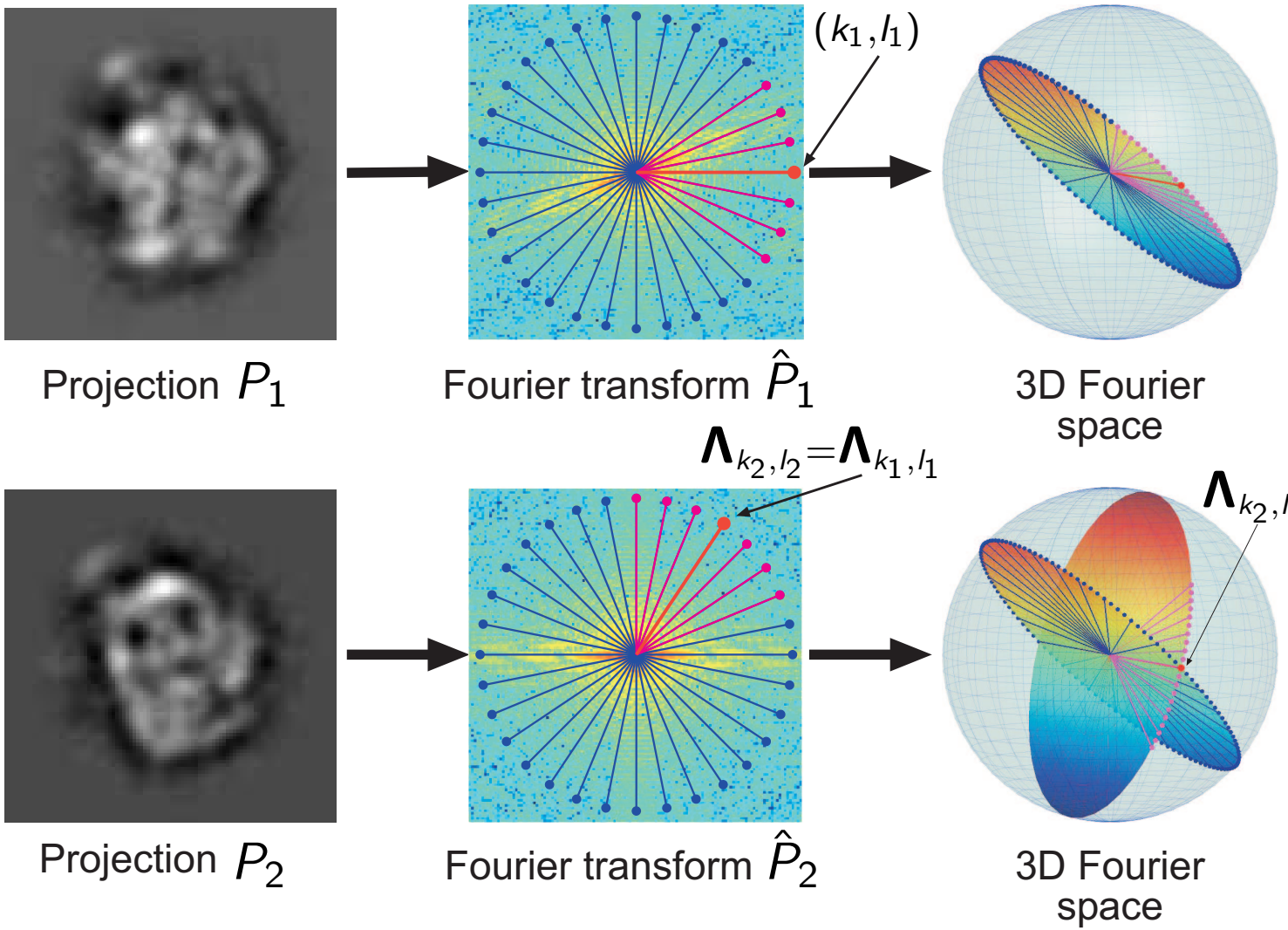




Cryo-EM for Structuring of Proteins

- Almost all protein channels cannot be crystallized.
- Rod MacKinnon was co-awarded the Chemistry Nobel Prize in 2003 for resolving the structure of the Shaker K^+ channel protein by X-ray crystallography.
- Cryo-EM: projection images of "frozen" proteins
- Thousands of images: every image corresponds to a different protein frozen in a different space orientation.
- Orientations are random and unknown.
- Electron beam destroys the imaged protein:
  a single protein can be imaged only once.
- Images are very noisy (low SNR)
- Images are 100×100 pixels.

Fourier projection-slice theorem



The Fourier projection-slice theorem

- $\theta \in S^2$ beaming direction, $\theta^\perp$ orthogonal plane.
- The 2D FT of the projection image is the double integral

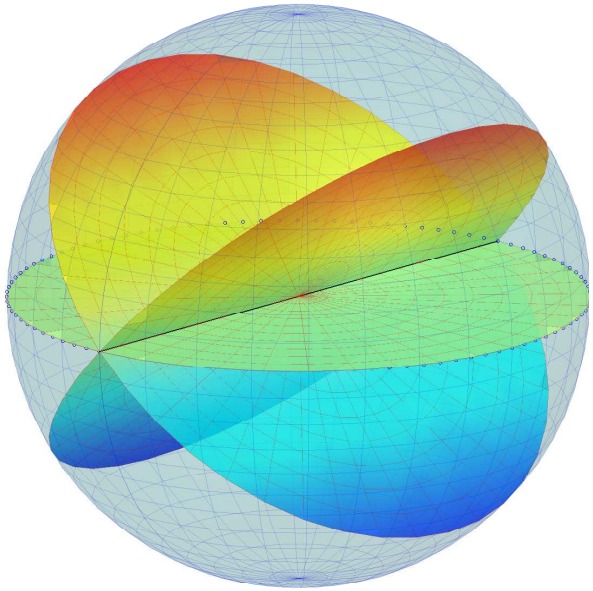
$$\hat{P}_\theta(\xi) = \int_{\theta^\perp} e^{-ir\cdot\xi} P_\theta(r)\, dr.$$

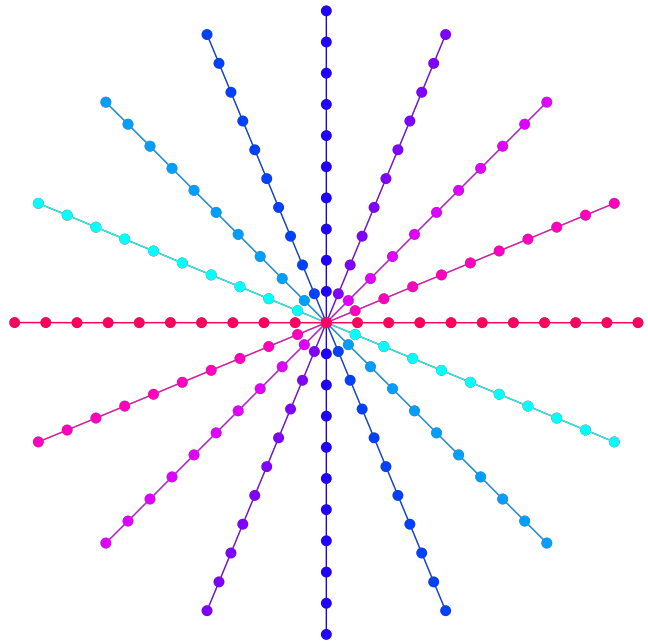
- The 3D FT of the molecule is the triple integral

$$\hat{\phi}(\xi) = \int_{\mathbb{R}^3} e^{-ir\cdot\xi} \phi(r)\, dr.$$

- Slice Theorem: $\hat{P}_\theta(\eta) = \hat{\phi}(\eta), \quad \eta \in \theta^\perp$.

The Geometry of the slice theorem





- Every image is a great circle over S^2.
- Any pair of images have a common line, or
- Any pair of great circles meet at two antipodal points.

Three Dimensional Puzzle

- The radial lines are the puzzle pieces.
- Every image is a circular chain of pieces.
- Common line: meeting point

Spiders: It's the Network

- K projection images
- L radial lines
- We build a weighted directed graph $G = (V, E, W)$.
- The vertices are the radial lines ($|V| = KL$)

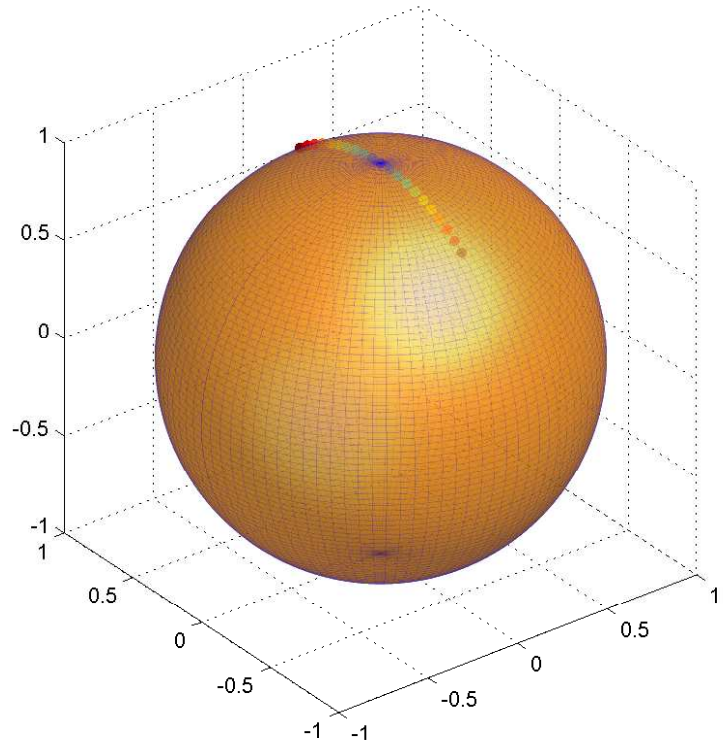
$$V = \{(k, l) : 1 \leq k \leq K,\ 0 \leq l \leq L-1\}$$

$$E = \{((k_1, l_1), (k_2, l_2)) : (k_1, l_1) \text{ points to } (k_2, l_2)\}$$

$$W_{(k_1,l_1),(k_2,l_2)} = \begin{cases} 1 & \text{if } ((k_1, l_1), (k_2, l_2)) \in E \\ 0 & \text{if } ((k_1, l_1), (k_2, l_2)) \notin E \end{cases}$$

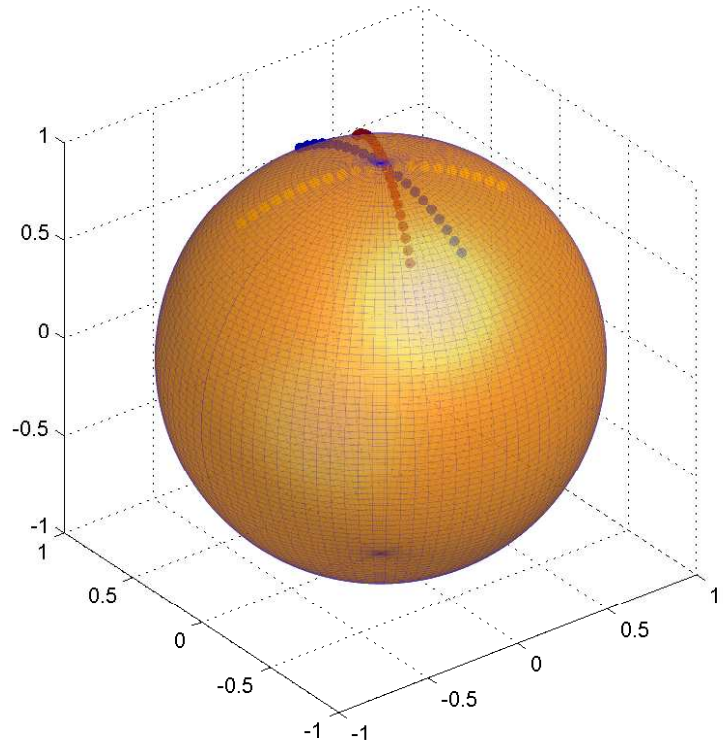
- $\mathbf{W}$ is a sparse weight matrix of size $KL \times KL$

Spider first pair of legs



- Blue vertex (k_1, l_1) is the head of the spider
- Link (k_1, l_1) with $(k_1, l_1 + l)$, $-d \leq l \leq d$ (same image radial lines)
- Weights: $W_{(k_1,l_1),(k_1,l_1+l)} = 1$.

Spider: remaining legs



- (k_1, l_1) and (k_2, l_2) are common radial lines of different images.
- Links: $((k_1, l_1), (k_2, l_2 + l)) \in E \quad \text{for } -d \leq l \leq d.$
- Weights: $W_{(k_1,l_1),(k_2,l_2+l)} = 1.$

Averaging operator

- Row stochastic normalization of $\mathbf{W}$

$$\mathbf{A} = \mathbf{D}^{-1}\mathbf{W}.$$

- $\mathbf{D}$ is a diagonal matrix, $D_{ii} = \sum_{j=1}^{N} W_{ij}$.
- The matrix $\mathbf{A}$ is an averaging operator:

$$(\mathbf{A}\mathbf{f})(k_1, l_1) = \frac{1}{|\{((k_1, l_1), (k_2, l_2)) \in E\}|} \sum_{((k_1,l_1),(k_2,l_2))\in E} f(k_2, l_2).$$

  $\mathbf{A}$ assigns the head of each spider the average of $\mathbf{f}$ over its legs.

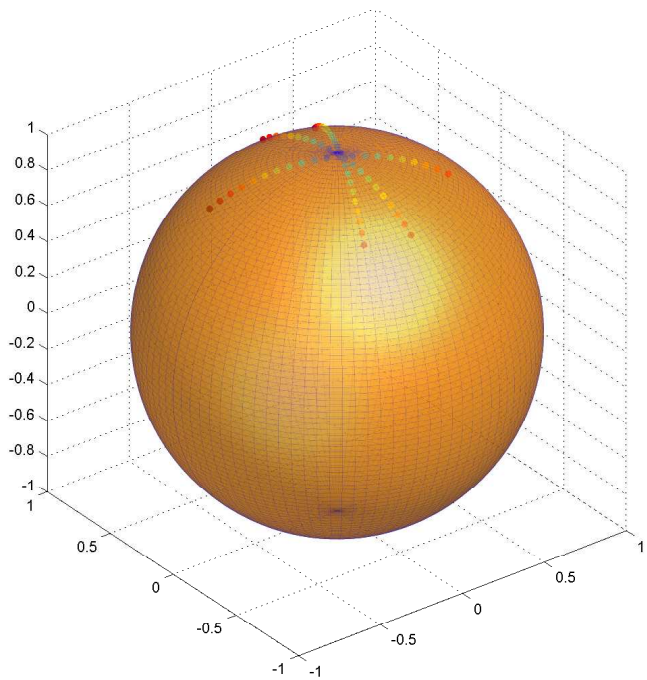
- $\mathbf{A}\mathbf{1} = \mathbf{1}$: trivial eigenvector $\psi_0 = \mathbf{1}$, with $\lambda_0 = 1$.

Coordinate Eigenvectors

- Coordinate vectors **x**, **y** and **z** are eigenvectors:

$$\mathbf{A}\mathbf{x} = \lambda\mathbf{x} \quad \mathbf{A}\mathbf{y} = \lambda\mathbf{y} \quad \mathbf{A}\mathbf{z} = \lambda\mathbf{z}$$

- The center of mass of every spider is beneath the spider's head: any pair of opposite legs balance each other – symmetric weights.



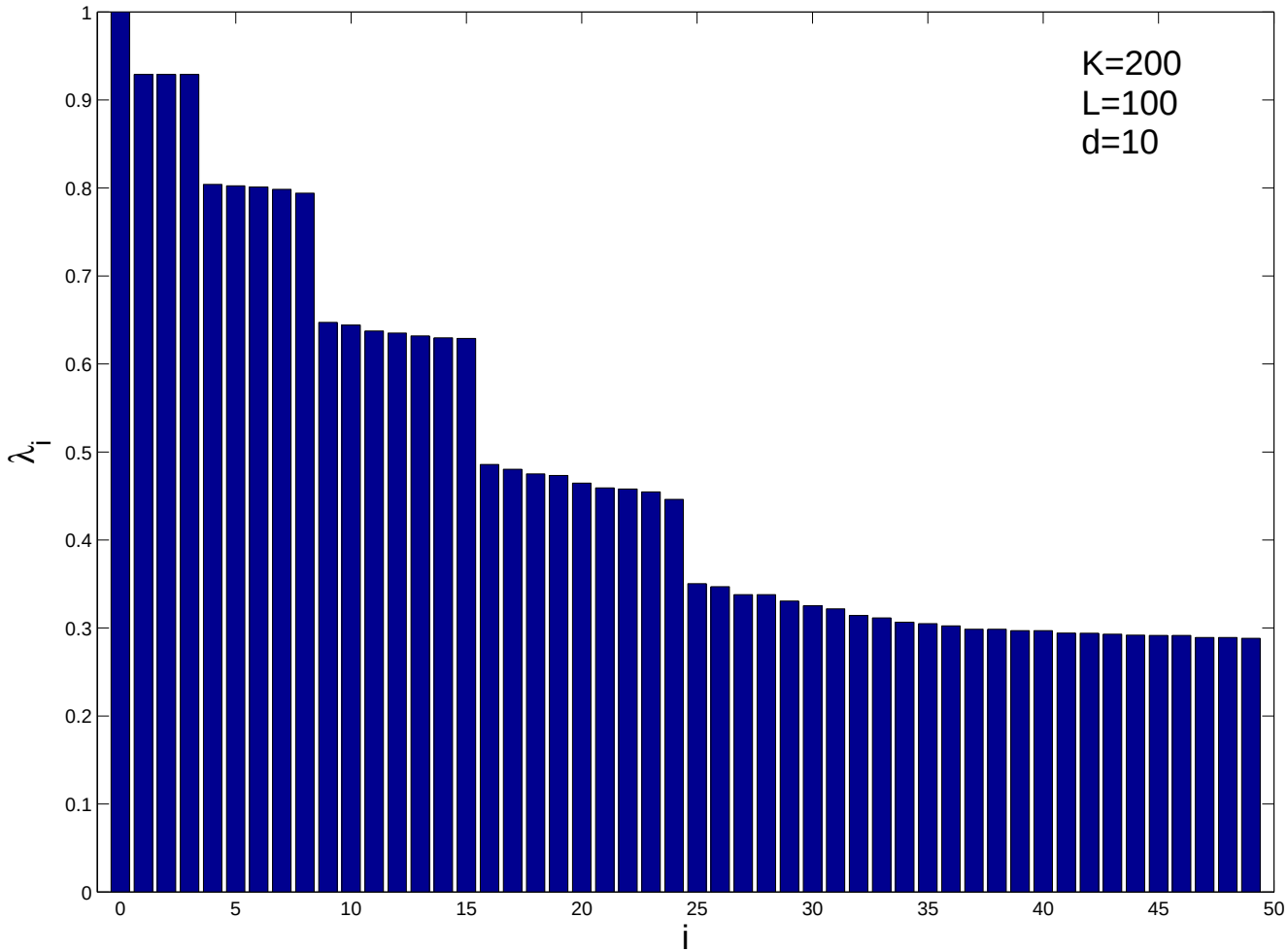
Embedding and algorithm

- Find the common lines for all pairs of images.
- Construct the averaging operator $\mathbf{A}$.
- Compute eigenvectors $\mathbf{A}\psi_i = \lambda_i \psi_i$.
- Embed the data into the eigenspace (ψ_1, ψ_2, ψ_3)

$$(k, l) \mapsto (\psi_1(k, l), \psi_2(k, l), \psi_3(k, l)).$$

- Reveals molecule orientations up to rotation and reflection.
- Final cosmetics:
  PCA same image radial lines and equally space them.

Numerical Spectrum



Spherical Harmonics

- The spherical harmonics Y_l^m are the eigenfunctions of the Laplacian on the sphere

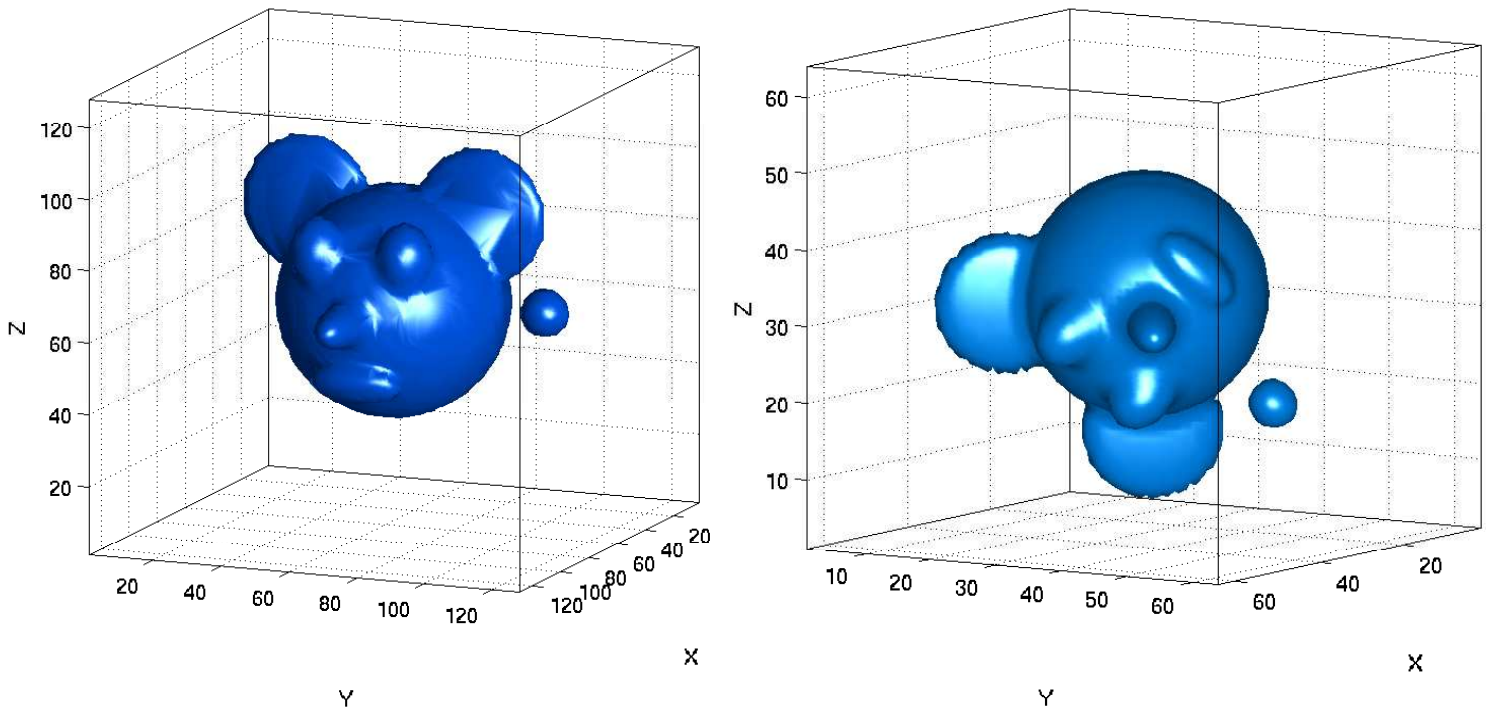
$$\Delta_{S^2} Y_l^m = -l(l+1) Y_l^m, \quad l = 0, 1, 2, \ldots, \quad m = -l, \ldots, l.$$

- Funk-Hecke: The spherical harmonics are the eigenfunctions of any integral operator that commutes with rotations:

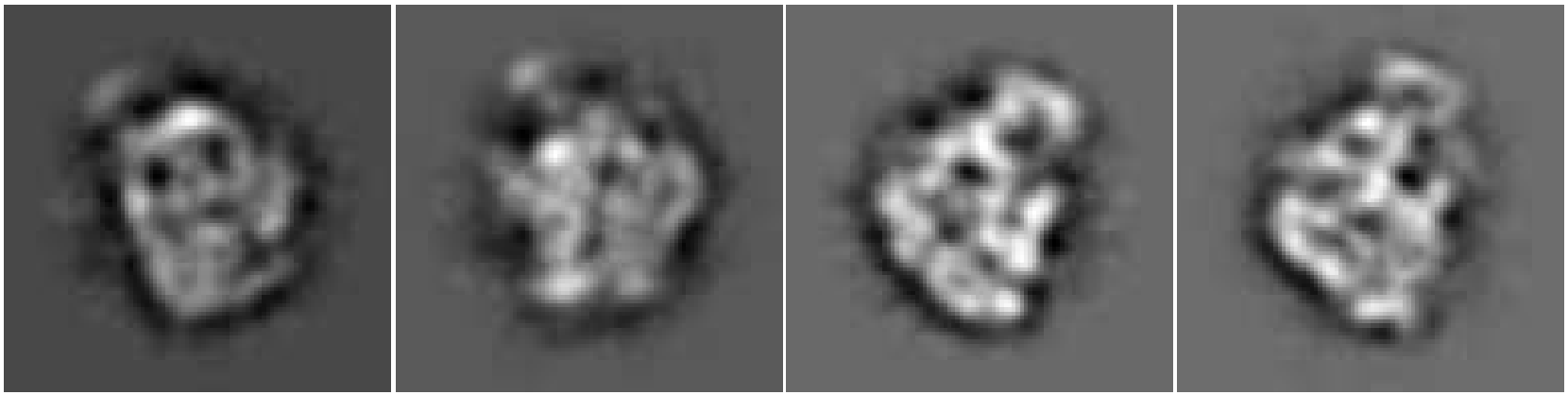
$$\begin{aligned} (\mathcal{K}f)(\beta) &= \int_{S^2} k(\langle \beta, \beta' \rangle) f(\beta')\, dS_{\beta'}, \\ \mathcal{K} Y_l^m &= \lambda_l Y_l^m. \end{aligned}$$

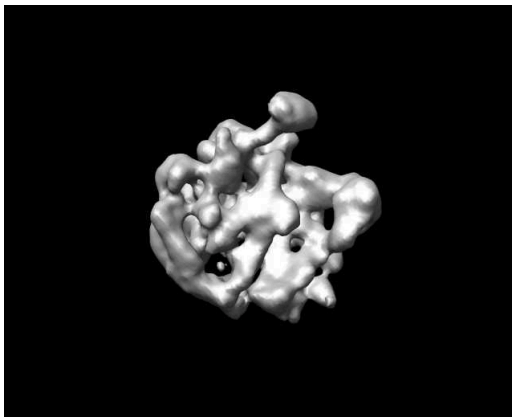
- The spider kernel commutes with rotations only on average, so spherical harmonics are not guaranteed.
- The three linear spherical harmonics are exact eigenfunctions of the spider kernel.

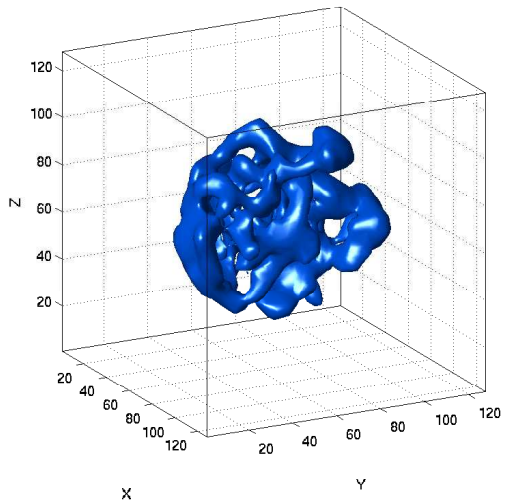
Toy Example

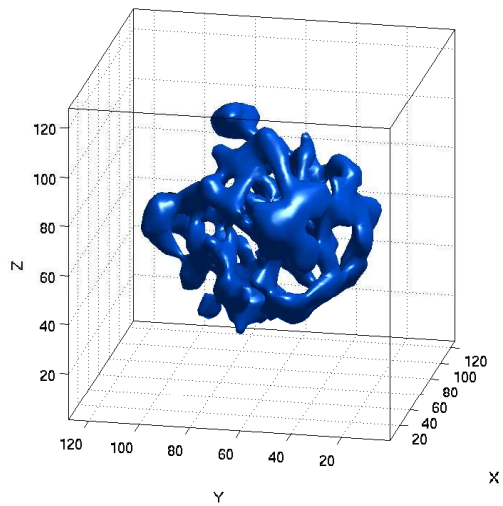

120
100
80
60
40
20
Z
20
40
60
80
100
120
Y
20
40
60
80
100
120
X
60
50
40
30
20
10
Z
10
20
30
40
50
60
Y
60
40
20
X

E. coli ribosome

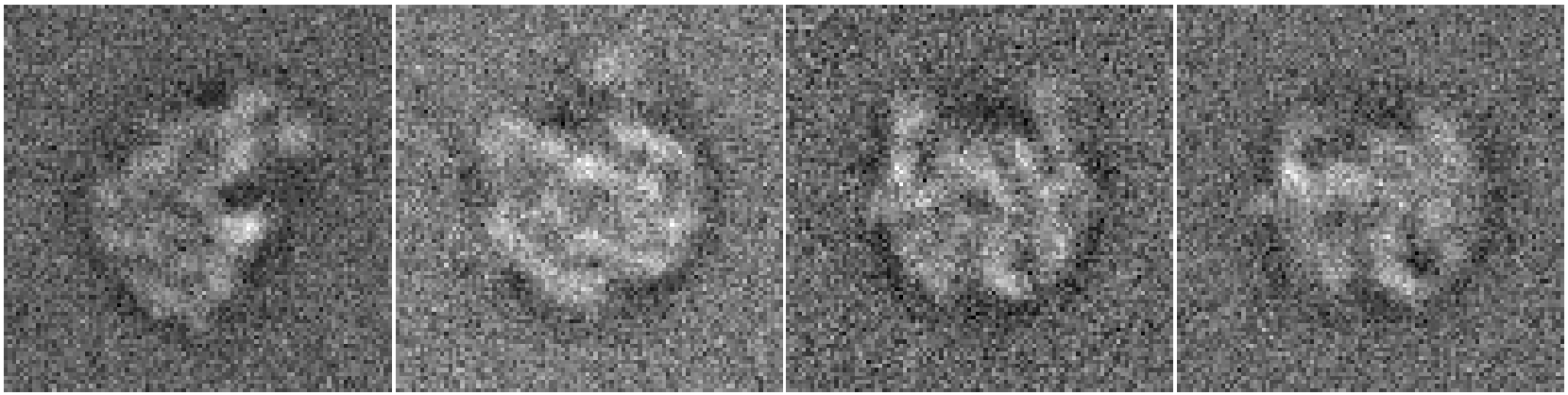


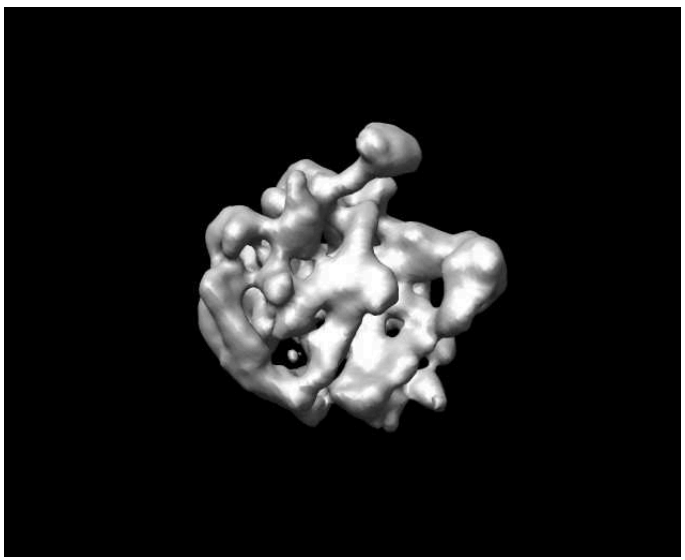


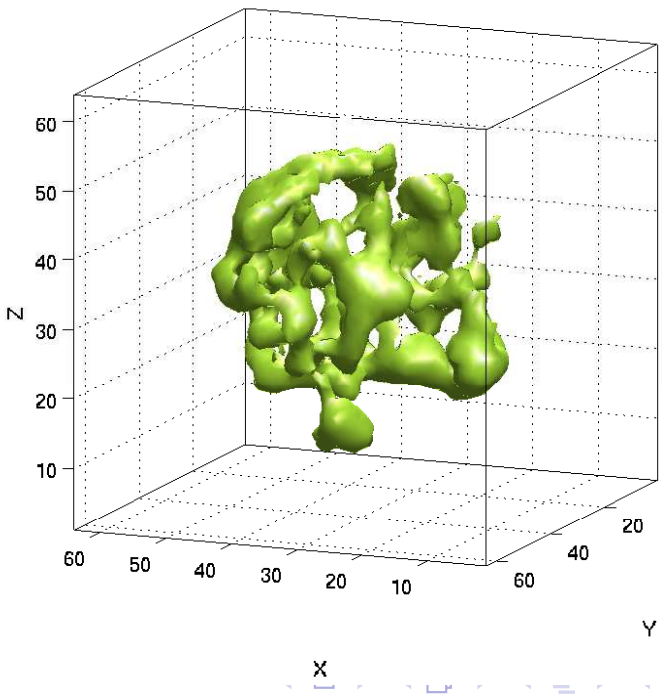




E. coli ribosome







Advantages

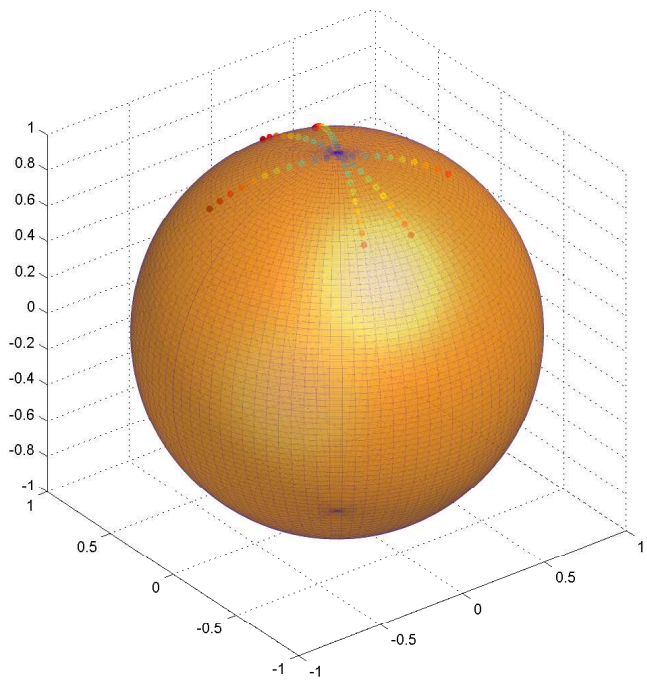
- Global: all radial lines are linked together.
- Fast: linear in data size KL and intersection points $\binom{K}{2}$.
- Averaging: all geometric information is averaged.
- Robust: errors due to false detections of common lines are smoothed out (can be viewed as matrix perturbation).
- Optional: omit uncertain common lines (fewer legs).

Beyond CryoEM: Center of mass averaging operator

- Coordinates $\mathbf{x}, \mathbf{y}, \mathbf{z}$ are eigenfunctions of the averaging operator:

$$\mathbf{A}\mathbf{x} = \lambda\mathbf{x} \quad \mathbf{A}\mathbf{y} = \lambda\mathbf{y} \quad \mathbf{A}\mathbf{z} = \lambda\mathbf{z}.$$

- Sphere has a constant curvature.
  Is there a generalization to Euclidean spaces?



Global Positioning from Local Distances

Problem setup:

- N points $\mathbf{r}_i \in \mathbb{R}^p$ $(p = 2, 3)$.
- Find coordinates $\mathbf{r}_i = (x_i^1, \ldots, x_i^p)$
- Given noisy neighboring distances $\delta_{ij} = \|\mathbf{r}_i - \mathbf{r}_j\|_2 + \text{noise}$.

Solution:

- Build an operator whose eigenfunctions are the global coordinates.
- Efficient eigenvector computation of a sparse matrix.

Applications:

- Sensor networks
- Protein structuring from NMR spectroscopy
  ($1/r^6$ decay of the spin-spin interaction between hydrogen atoms)
- Surface reconstruction and PDE solvers.
- More?

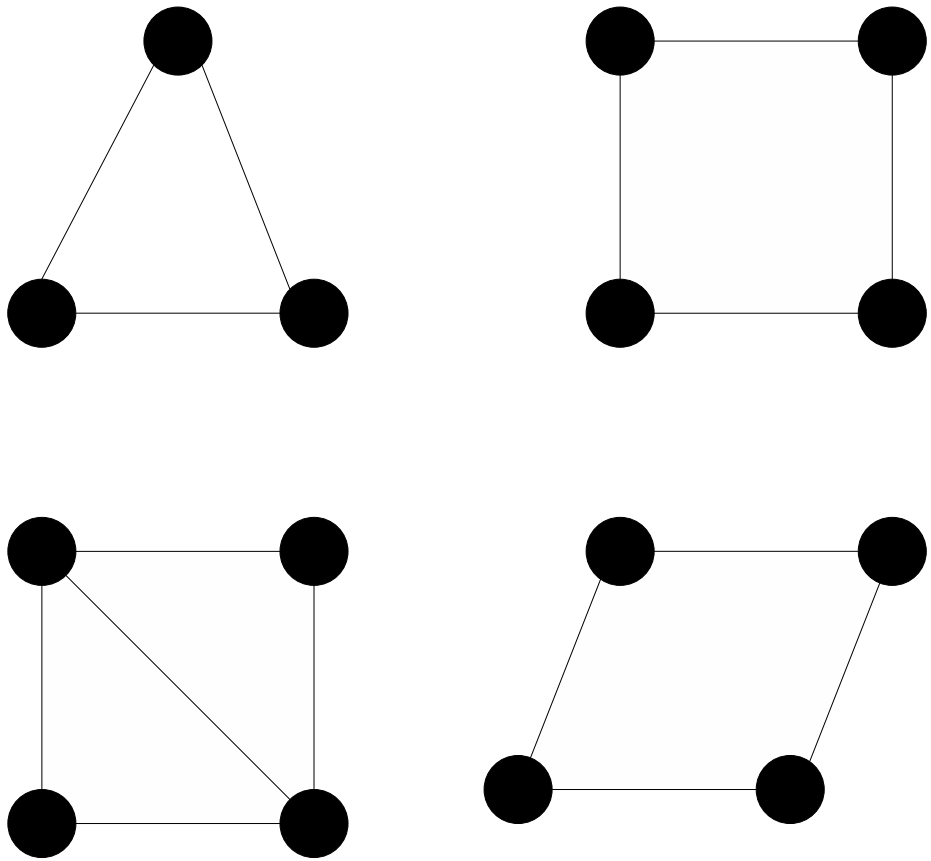
Global Positioning from Local Distances: History

- Multidimensional Scaling (MDS) if all $d_{ij} = \|\mathbf{r}_i - \mathbf{r}_j\|$ are given: law of cosines + SVD of the inner product matrix.
- Optimization: minimizing a variety of loss functions, e.g.

$$\min_{\mathbf{r}_1,\ldots,\mathbf{r}_N} \sum_{i\sim j} \left[\|\mathbf{r}_i - \mathbf{r}_j\|^2 - \delta_{ij}^2\right]^2$$

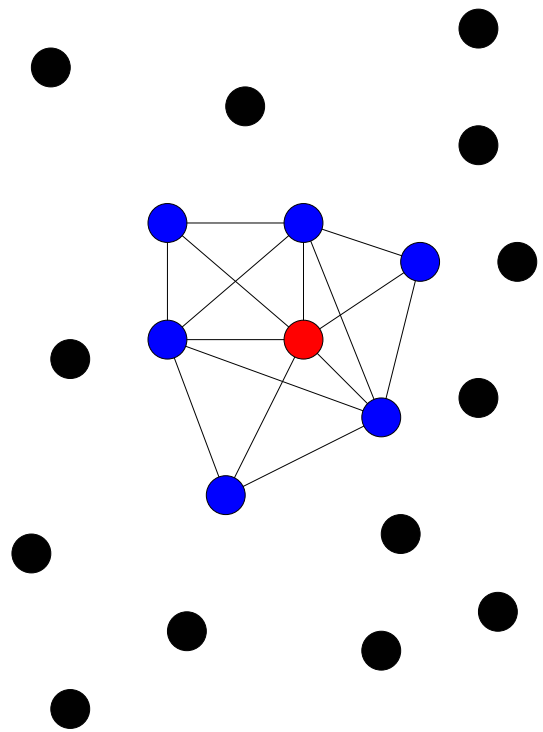
  many variables, not convex, local minima.
- Semidefinite programming (SDP), slow.
- Graph Laplacian regularization (Weinberger, Sha, Zhu & Saul, NIPS 2006).

Rigidity



Locally Rigid Embedding

- Assume local rigidity.
- For each point, embed its k-NN locally (using MDS or otherwise).
- N local coordinate systems mutually rotated and possibly reflected.
- How to glue the different coordinate systems together?



Center of Mass

- Consider the point $\mathbf{r}_i$ and its k-NN $\mathbf{r}_{i_1}, \mathbf{r}_{i_2}, \ldots, \mathbf{r}_{i_k}$.
- Find weights such that $\mathbf{r}_i$ is the center of mass of its neighbors

$$\sum_{j=1}^{k} W_{i,i_j} \mathbf{r}_{i_j} = \mathbf{r}_i$$

and

$$\sum_{j=1}^{k} W_{i,i_j} = 1$$

System of $p+1$ linear equations in k variables; underdetermined for $k > p+1$.

- Weights are invariant to rigid transformations: rotation, translation, reflection.
- In practice we choose the solution with min $\sum_{j=1}^{k} W_{i,i_j}^2$ to keep weights balanced.

Eigenvectors of $\mathbf{W}$

- The $N \times N$ weight matrix $\mathbf{W}$ is sparse:
  every row has at most k non-zero elements.
- W is not symmetric;
  must have negative weights for points on the boundary.
- By construction

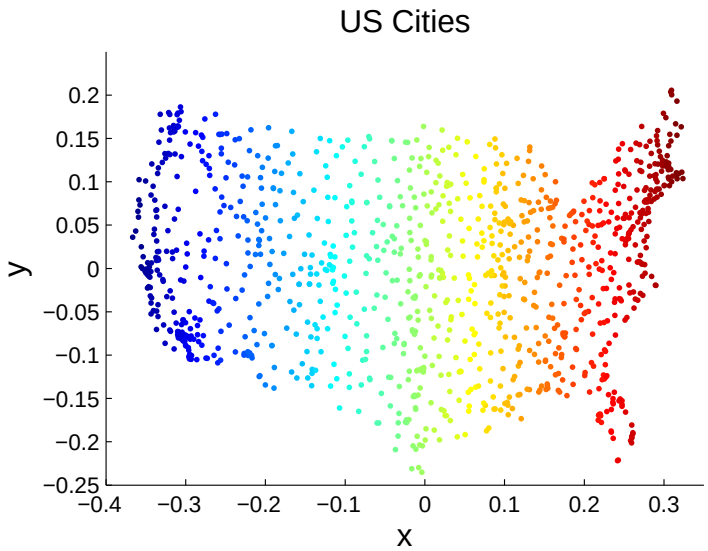
$$\mathbf{W1} = \mathbf{1}, \quad \mathbf{W}\mathbf{x}^1 = \mathbf{x}^1, \quad \ldots, \quad \mathbf{W}\mathbf{x}^p = \mathbf{x}^p,$$

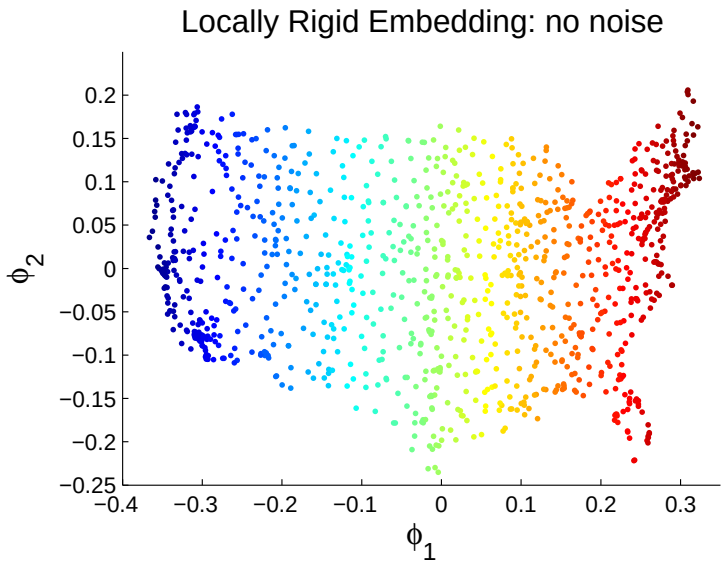
  because

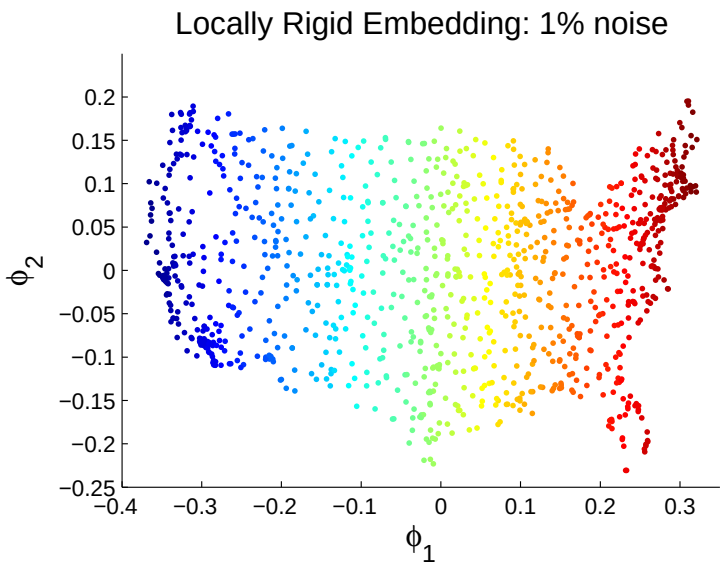
$$\sum_{j=1}^{N} W_{ij} = 1, \quad \sum_{j=1}^{N} W_{ij}\mathbf{r}_j = \mathbf{r}_i.$$

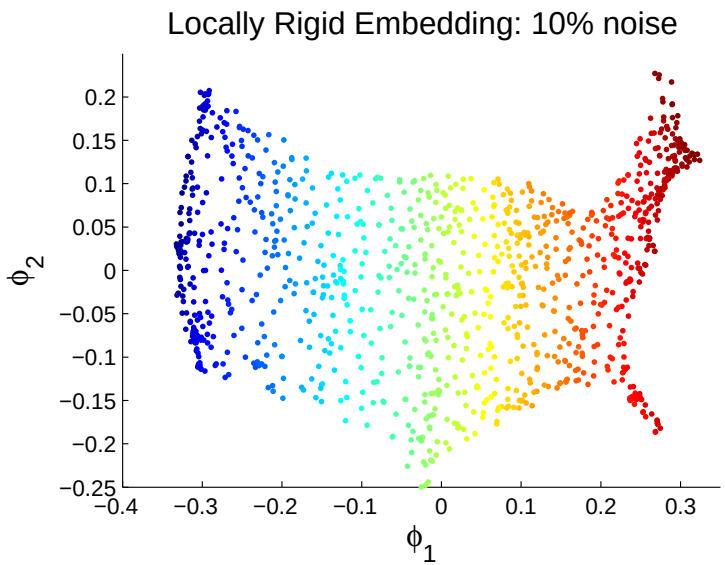
- We are practically done:
  eigenvectors of $\mathbf{W}$ with $\lambda = 1$ are the desired coordinates.
- A little linear algebra is needed to deal with multiplicity and noise.

1097 US Cities, $k = 18$ NN

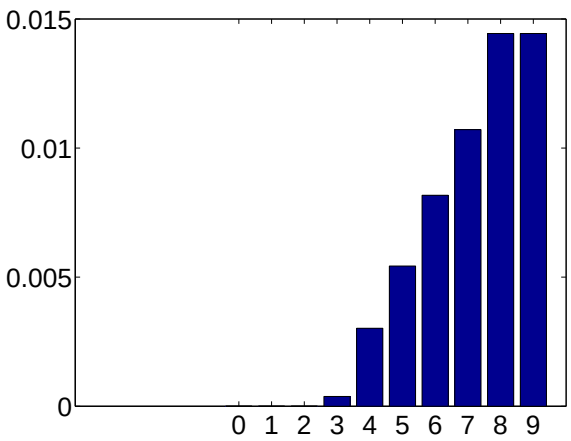


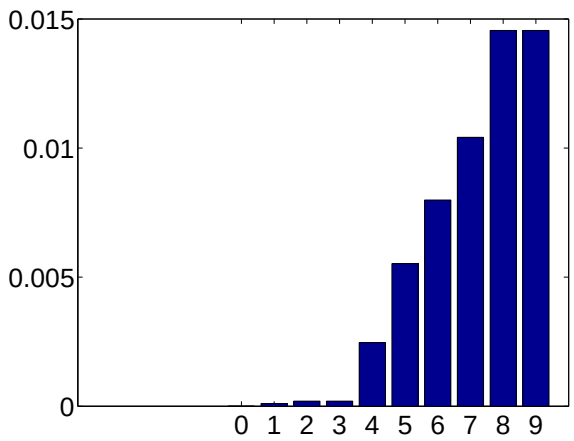


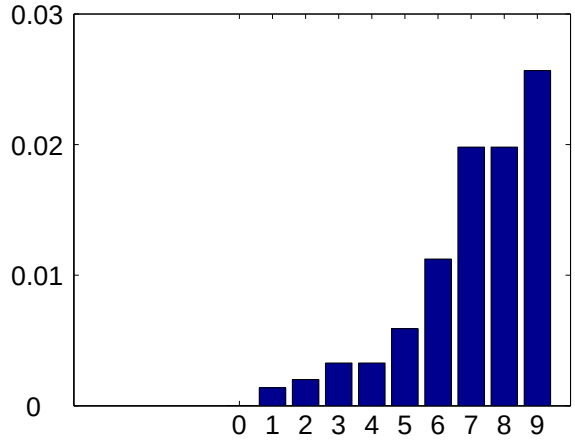




US Cities: Numerical Spectrum

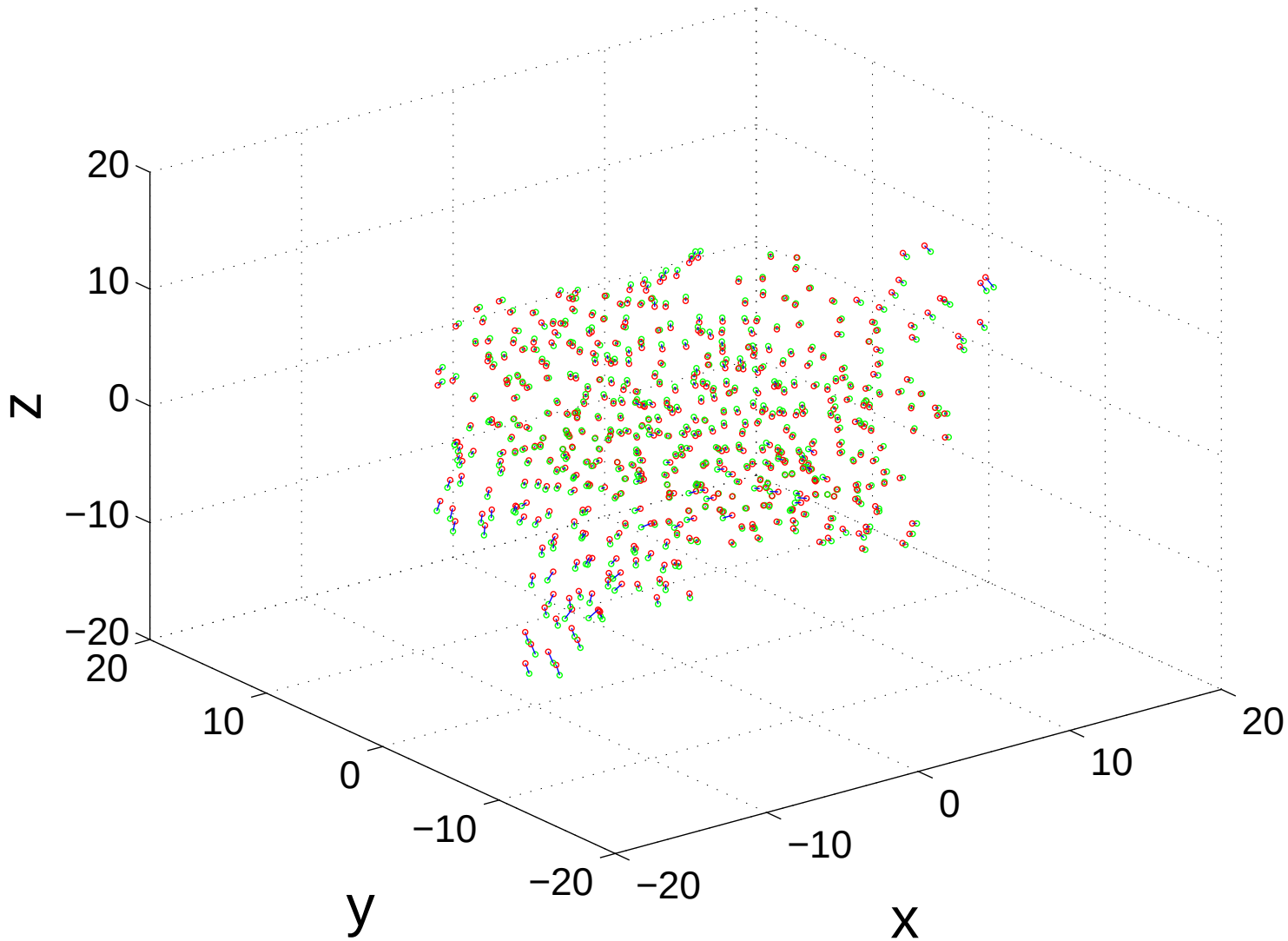




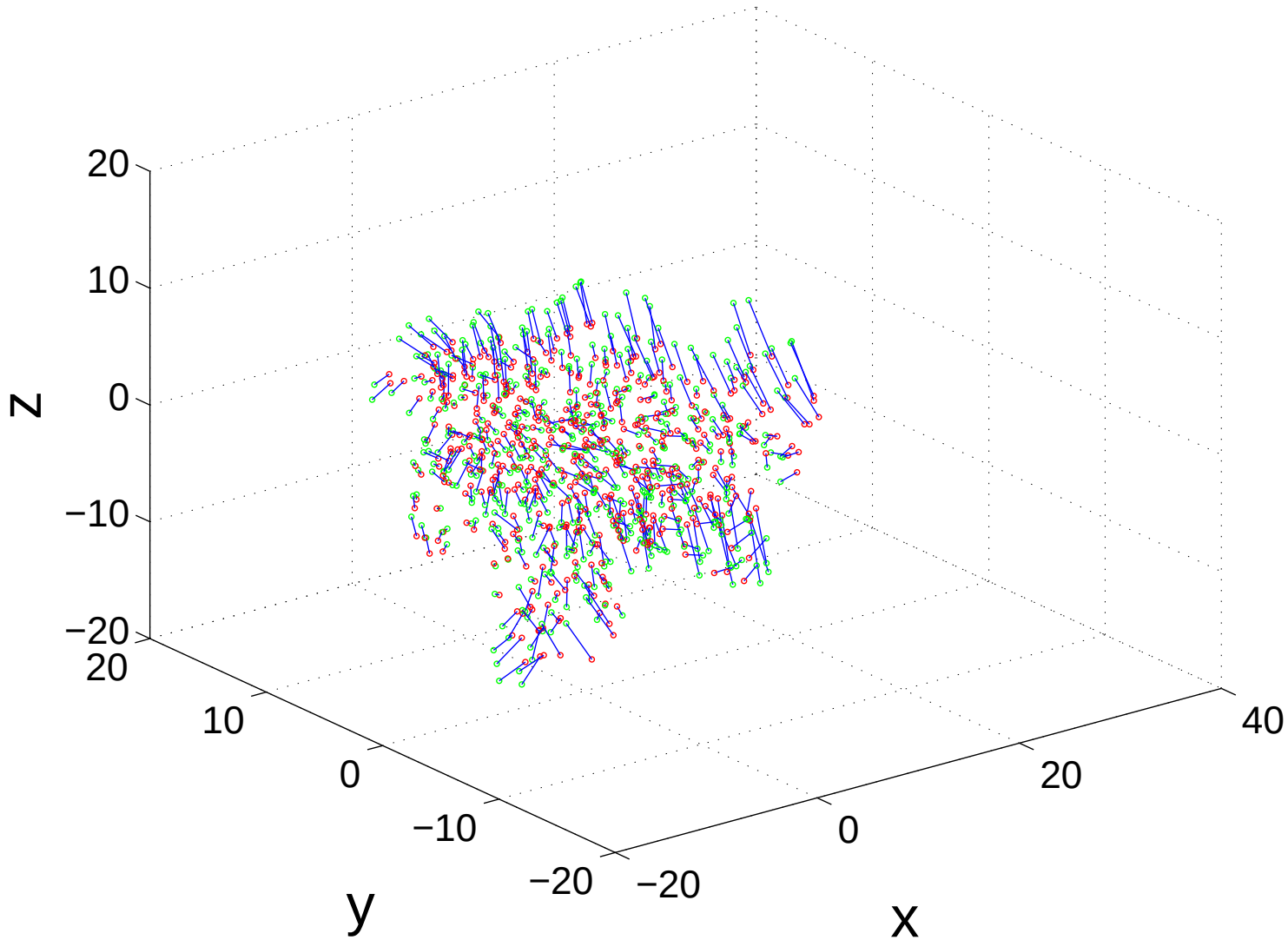


Numerical spectrum of $\mathbf{W}$ for different levels of noise:
clean distances (left), 1% noise (center), and 10% noise (right).

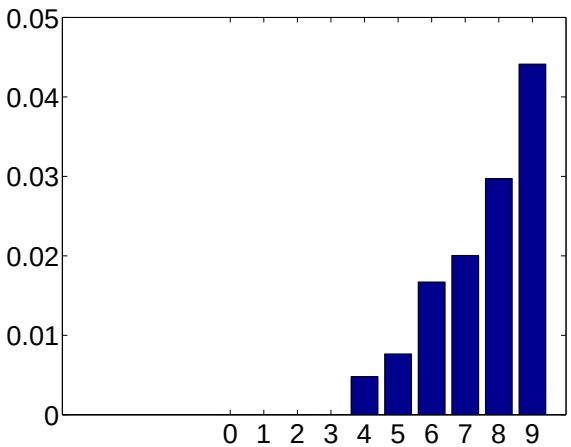
519 hydrogen atoms of 2GGR: 1% noise

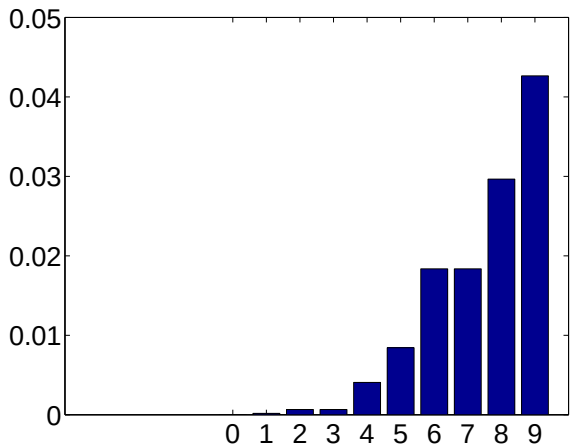


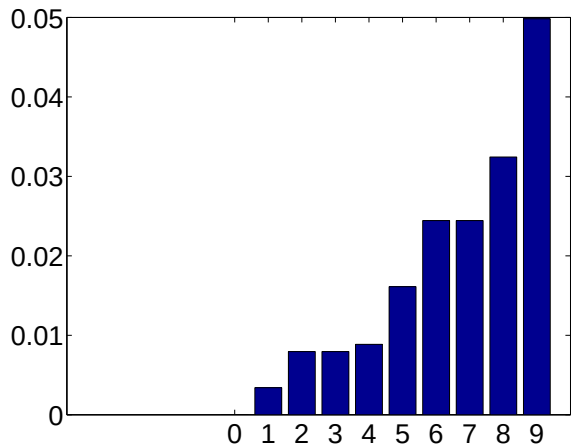
2GGR: 5% noise



2GGR: Numerical Spectrum







Dealing with the multiplicity

- The eigenvalue $\lambda = 1$ is degenerated with multiplicity $p + 1$

$$\mathbf{W}\phi^i = \phi^i, \quad i = 0, 1, \ldots, p.$$

- The computed eigenvectors $\phi^0, \phi^1, \ldots, \phi^p$ are linear combinations of $\mathbf{1}, \mathbf{x}^1, \ldots, \mathbf{x}^p$.
- We may assume $\phi^0 = \mathbf{1}$ and $\langle \phi^j, \mathbf{1} \rangle = 0$ for $j = 1, \ldots, p$.
- We look for a $p \times p$ matrix $\mathbf{A}$ that maps the eigenmap $\mathbf{\Phi}_i = (\phi_i^1, \ldots, \phi_i^p)$ to the original coordinate set $\mathbf{r}_i = (x_i^1, \ldots, x_i^p)$

$$\mathbf{r}_i = \mathbf{A}\mathbf{\Phi}_i, \quad \text{for } i = 1, \ldots, N.$$

- The squared distance between $\mathbf{r}_i$ and $\mathbf{r}_j$ is

$$d_{ij}^2 = \|\mathbf{r}_i - \mathbf{r}_j\|^2 = (\mathbf{\Phi}_i - \mathbf{\Phi}_j)^T \mathbf{A}^T \mathbf{A} (\mathbf{\Phi}_i - \mathbf{\Phi}_j).$$

- Overdetermined system of linear equations for the elements of $\mathbf{A}^T\mathbf{A}$. Least squares gives $\mathbf{A}^T\mathbf{A}$, whose Cholesky decomposition yields $\mathbf{A}$.

Dealing with noisy distances δ_{ij}

- Noise breaks the degeneracy and may lead to crossings of eigenvalues.
- Coordinate vectors are approximated as linear combinations of m non-trivial eigenvectors $\mathbf{\Phi}_i = (\phi^1, \ldots, \phi^m)$, with $m > p$ (still $m \ll N$).
- $\mathbf{r}_i = \mathbf{A}\mathbf{\Phi}_i$, with $\mathbf{A}$ being $p \times m$ instead of $p \times p$.
- Replace

$$d_{ij}^2 = \|\mathbf{r}_i - \mathbf{r}_j\|^2 = (\mathbf{\Phi}_i - \mathbf{\Phi}_j)^T \mathbf{A}^T \mathbf{A} (\mathbf{\Phi}_i - \mathbf{\Phi}_j)$$

  with the constrained minimization problem for the $m \times m$ semidefinite positive matrix $\mathbf{P} = \mathbf{A}^T \mathbf{A}$

$$\min \sum_{i \sim j} \left[(\mathbf{\Phi}_i - \mathbf{\Phi}_j)^T \mathbf{P} (\mathbf{\Phi}_i - \mathbf{\Phi}_j) - \delta_{ij}^2 \right]^2, \text{ such that } \mathbf{P} \succ 0.$$

- A small SDP (formulation uses the Schur complement lemma).

Comparison to LLE and Graph Laplacian regularization

- Locally Linear Embedding (LLE) is a non-linear dimensionality reduction method.
- LLE: Construct weights by solving an overdetermined system

$$\min \left\| \sum_j W_{ij} \mathbf{X}_j - \mathbf{X}_i \right\|_2, \text{ where } \mathbf{X}_i \in \mathbb{R}^n.$$

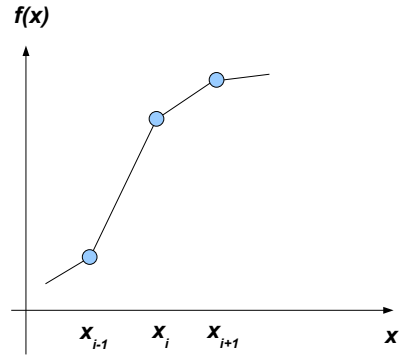
- We have a preprocessing step to reveal vectors ($\mathbf{X}_i$ are not given).
- Graph Laplacian regularization: approximate coordinate by the eigenvectors of $W_{ij} = \exp\left\{-d_{ij}^2/\varepsilon\right\}$ for $i \sim j$, $W_{ij} = 0$ elsewhere.
- LRE requires "locally" rigid subgraphs, graph neighbors can be physically distant.

Technical Remarks

- A sparse symmetric (positive) matrix $\mathbf{W}$ with similar spectral properties can be constructed with the same effort (Dan Spielman)
- The eigenvector computation can be done in parallel.
- The storage of $\mathbf{W}$ is distributed between the N sensors, such that each point stores only k values of $\mathbf{W}$ together with k neighboring values of a given vector.

Numerical Integration

- Find $f = f(x)$ from its derivative $f'(x)$.



- Approximate

$$\begin{aligned} f(x_{i+1}) &= f(x_i) + \frac{1}{2}\left[f'(x_i) + f'(x_{i+1})\right]\Delta x \\ f(x_{i-1}) &= f(x_i) - \frac{1}{2}\left[f'(x_i) + f'(x_{i-1})\right]\Delta x \end{aligned}$$

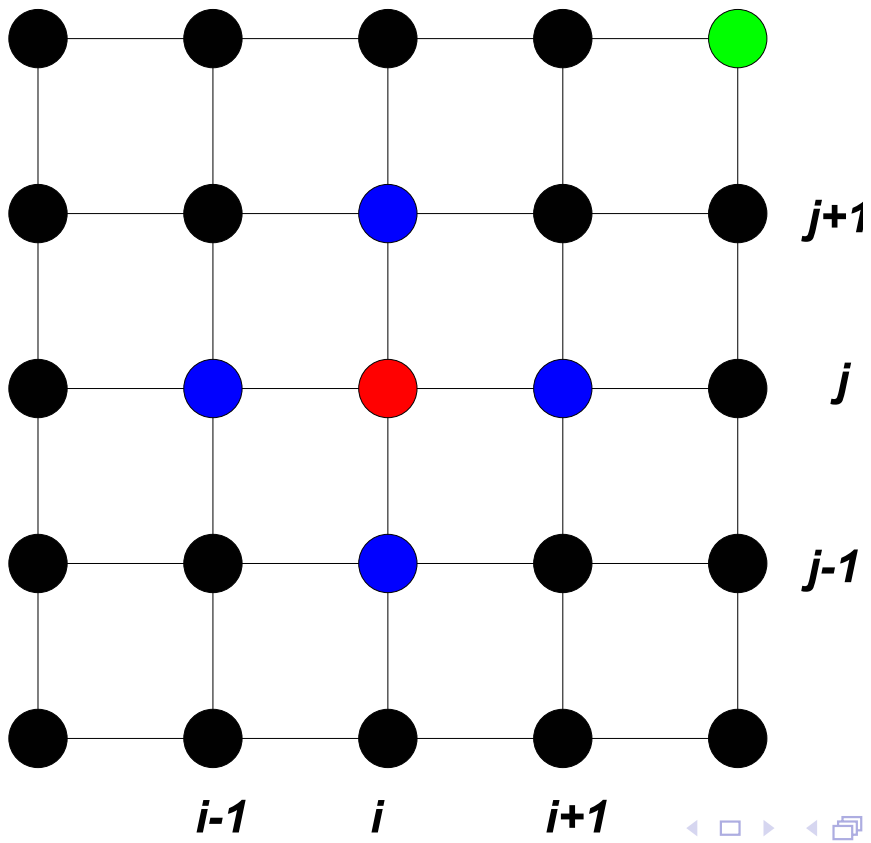
- Find weights $W_{i,i+1}$, $W_{i,i-1}$ such that

$$f(x_{i+1})W_{i,i+1} + f(x_{i-1})W_{i,i-1} = f(x_i), \quad W_{i,i+1} + W_{i,i-1} = 1.$$

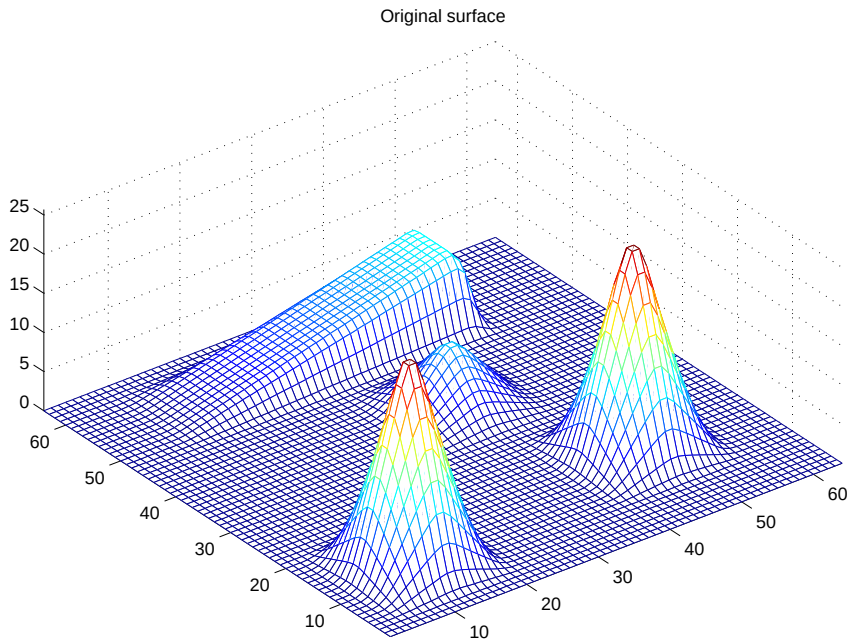
- $\mathbf{f} = (f(x_1), f(x_2), \ldots, f(x_N))$ satisfies $\mathbf{W}\mathbf{f} = \mathbf{f}$.

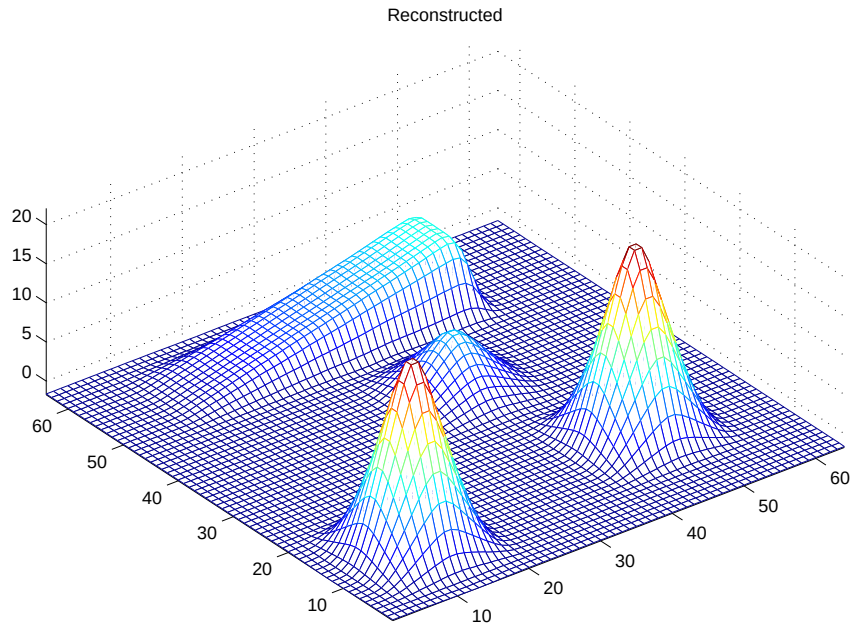
Numerical Integration: Surface Reconstruction

- Find $f = f(x, y)$ from its gradient field $(f_x(x, y), f_y(x, y))$.
- Approximate North, South, East, West and find weights.
- Eigenvector computation $\mathbf{Wf} = \mathbf{f}$ averages over different integration paths between the blue green points.



Numerical Integration: Surface Reconstruction





Thank You!