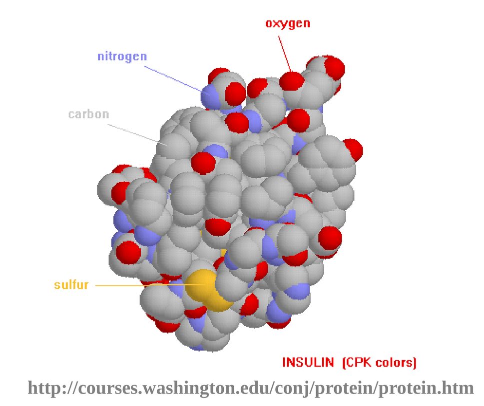
Voronoi Cells of Non-general Position Spheres Using the GPU

Zhongyin Hu[1], Xiang Li[1], Adarsh Krishnamurthy[2], Iddo Hanniel[3], Sara McMains[1]

[1] University of California, Berkeley
[2] Iowa State University
[3] Technion, Israel Institute of Technology

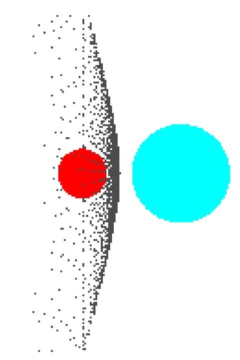
Motivation: Voronoi Diagrams of Spheres

- Not extensively investigated
  - Lack of robust algorithms
  - Especially for non-general position and disconnected Voronoi edges
- Many current applications simplify the problem
  - Assume that all atoms are 3D points
- Example application: Proteins
  - Voronoi diagrams are often used to analyze structural properties of molecules

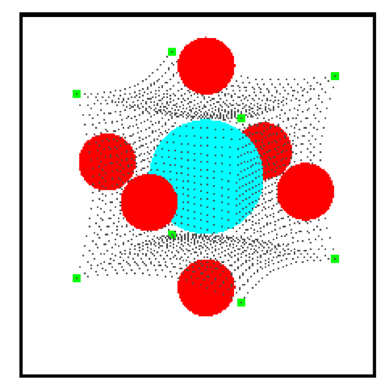


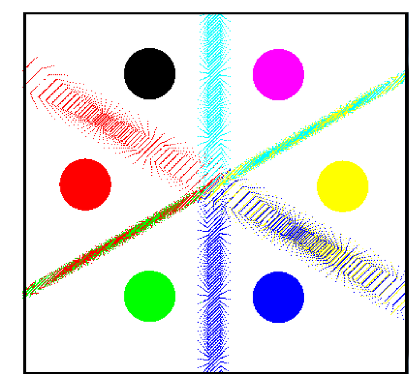
Background

- Bisectors of spheres
  - The locus of points that are equidistant from two spheres
  - Either a plane or a hyperbolic surface
  - Assumption: no completely contained spheres (allow partially intersecting)



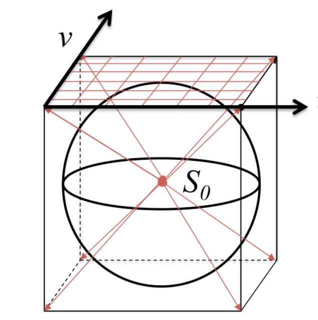
- Voronoi diagrams of spheres in 3D
  - The surface of a Voronoi cell is the lower envelope of its corresponding bisectors
  - In general position, numbers of spheres contributing to: Voronoi vertex: 4; Voronoi face: 3; Voronoi edge: 2; Voronoi cell: 1
  - In non-general positions, there are more contributing spheres





Sampling Rays and Calculating Bisectors

- Sampling rays from spheres
  - Parameterize bounding cube for each sphere (uniformly subdivided domains in u and v)
  - Each face of the bounding cube maps to 1/6 of the sphere
  - Shoot rays from the sphere center through each u-v point
  - Ray function: $\mathbf{r}(t) = \mathbf{r}(x(t), y(t), z(t)) = \mathbf{0} + t \cdot \mathbf{n}$



- Calculating the bisector functions
  - The bisector surfaces between two spheres:

$$\sqrt{(x-C_{x_1})^2+(y-C_{y_1})^2+(z-C_{z_1})^2}-R_1 = \sqrt{(x-C_{x_2})^2+(y-C_{y_2})^2+(z-C_{z_2})^2}-R_2.$$

  - Transforming to implicit hyperbolic surface equation:

$$Ax^2+By^2+Cz^2+Dxy+Exz+Fyz+Gx+Hy+Iz+J=0$$

Calculating Sample Points on Voronoi Faces

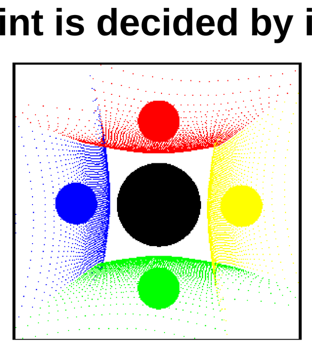
- Combining ray functions and bisector functions
  - For the "base" sphere in each Voronoi cell, test all bisectors corresponding to each ray shot from the base sphere
  - Each ray either shoots to infinity or intersects the lower envelope at a Voronoi face sample point
  - Solve the functions:

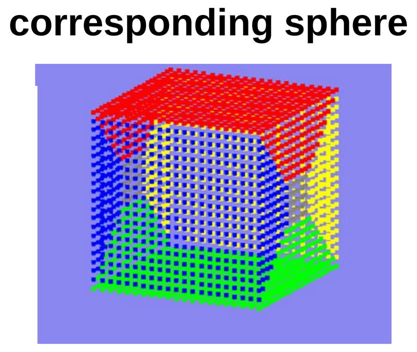
$$\mathbf{r}(t) = \mathbf{r}(x(t), y(t), z(t)) = \mathbf{0} + t \cdot \mathbf{n}$$

$$Ax^2+By^2+Cz^2+Dxy+Exz+Fyz+Gx+Hy+Iz+J=0$$

$$\Rightarrow \quad a \cdot t^2 + b \cdot t + c = 0$$

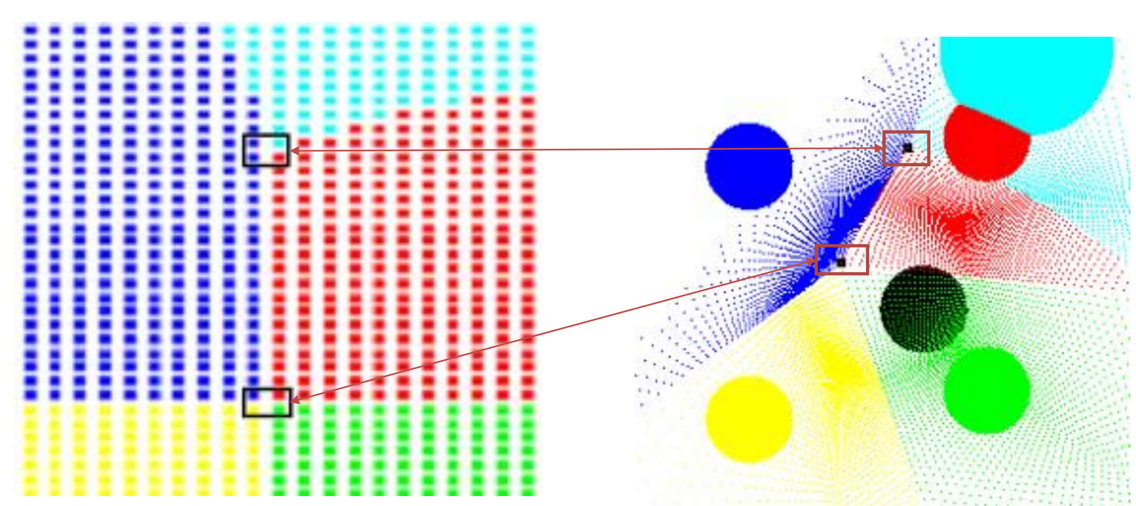
- Visualization of color-coded face sample points per base sphere
  - The color of each sample point is decided by its corresponding sphere
  - Base sphere as black
  - We also get color code on u-v domain (bounding cube)





Determining Presence of Voronoi Vertices

- U-V grid cells
  - Each group of four neighboring face sample points on the bounding cube
  - "Marching" through the grid cells, check if the four corners have three or more different colors
  - Three or more colors → the corresponding Voronoi faces may intersect in a Voronoi vertex



u-v domain (bounding cube) xyz domain

Calculating Voronoi Vertices' Position

- Newton-Raphson method in 3 variables
  - Calculate xyz coordinates of Voronoi vertex in each 3-color grid cell
  - Equivalent to solving three bisector equations simultaneously
  - Use the average of xyz coordinates of the four corners of the grid cell as start point for iteration

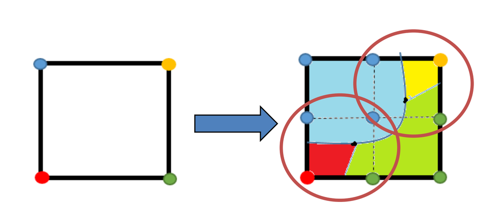
$$P_{k+1} = P_k - J^{-1}(P_k) \cdot F(P_k) \qquad (k = 0, 1, \dots)$$

$$max|f_i(P_k)| < \epsilon$$

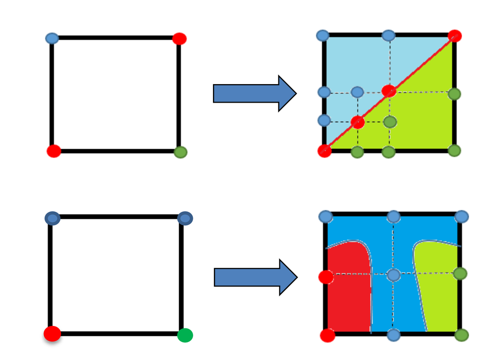
Def: 3x3 Jacobian $f'(x)$

$$J(P_k) = \frac{\partial F(x,y,z)}{\partial P(x,y,z)} = \begin{bmatrix} \frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} & \frac{\partial f_1}{\partial z} \\ \frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y} & \frac{\partial f_2}{\partial z} \\ \frac{\partial f_3}{\partial x} & \frac{\partial f_3}{\partial y} & \frac{\partial f_3}{\partial z} \end{bmatrix}$$

- 4-color grid cells and special cases
  - Subdivide the grid cell



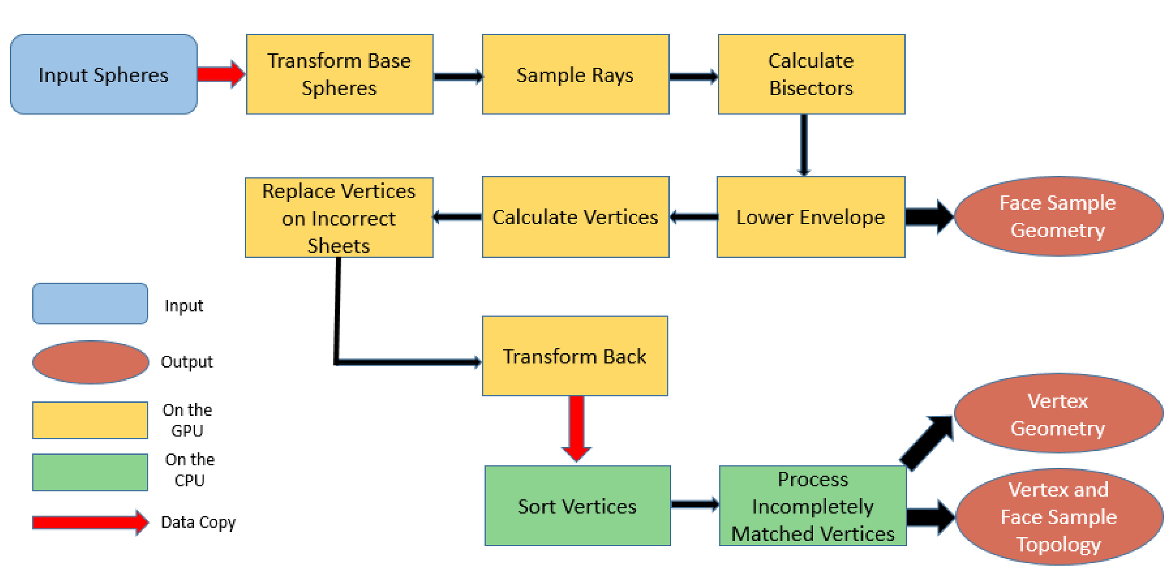
4-color grid cell



Special cases

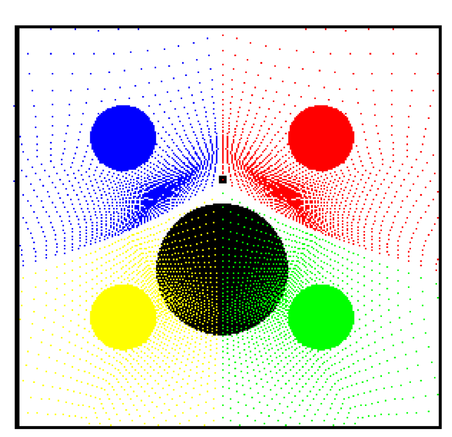
GPU Framework

- Our algorithm is well-suited to the GPU
  - Arithmetic intensity of equation calculations
  - High-density sampling
  - At each sample the calculation is independent
- Design of the GPU framework

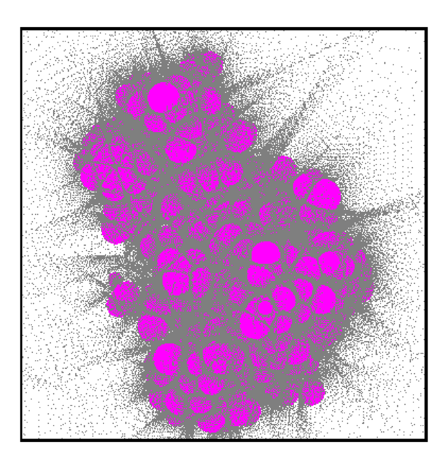


Results

- Handling non-general position and large inputs



A Voronoi cell containing a high-order degree Voronoi vertex



Voronoi diagrams of Protein Data Bank ID "1bh8" consisting of 2161 atoms

- Timing results (seconds) for face samples and Voronoi vertices

Sampling	1al1-PQR (217 atoms)	1crn-PDB (327 atoms)	1crn-PQR (642 atoms)	1bh8-PQR (2161 atoms)	1JD0-PDB (4195 atoms)
30*30	1.30	2.02	3.85	23.3	50.2
60*60	1.92	3.06	5.83	35.4	81.4
80*80	2.29	3.85	7.05	41.2	93.6
100*100	2.45	4.21	7.82	45.9	102

Contributions and Future Work

- A novel approach to compute Voronoi diagrams of spheres
  - Sample based + lower envelope + GPU parallel computing
- Accurately calculate Voronoi vertices' geometry
  - Use the samples to initialize numerical iteration
  - Guarantee the accuracy within user-defined tolerance
- This algorithm is robust for
  - Thousands of input spheres representing actual protein molecules
  - Spheres not in general position, handling Voronoi vertices with degree greater than four
- Future work:
  - On u-v domain (bounding cube), trace Voronoi edges by particular color patterns
  - Apply approach to more complex primitives

We gratefully acknowledge support from National Science Foundation grant 1331352

Computer-Aided Design and Manufacturing Laboratory, Mechanical Engineering Department, University of California, Berkeley