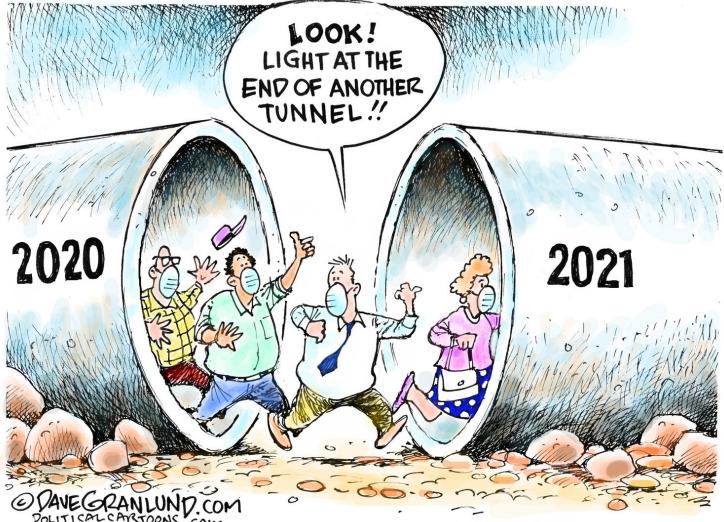


Fast Fourier Transforms (FFTs) applied to docking



How I felt in January...



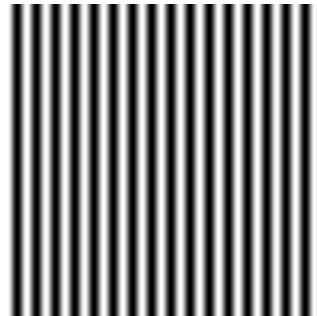
How I feel today...

Shannon Smith
14 April 2021

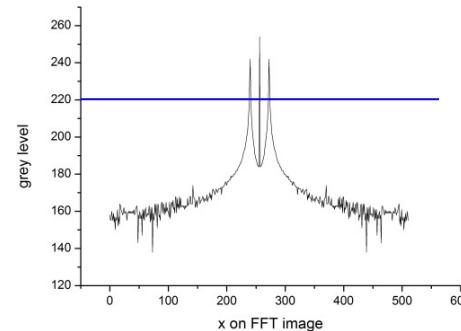
What is a (Fast) Fourier Transform?

- Nomenclature: Fast FT (FFT) is used for rapid implementation of FT on real data
- FFT: spatial domain → frequency domain
- Images are made up of a series of sine and cosine waves of different frequencies and amplitudes
- Filter in frequency domain (deblurring an image, removing background static noise)

Original image



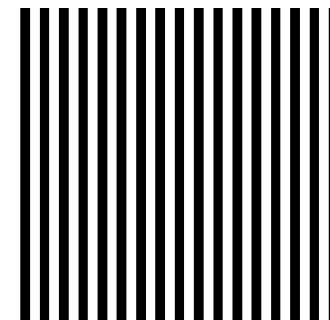
FFT



B→W and W→B

Peaks correlate to the frequency of sharp function changes

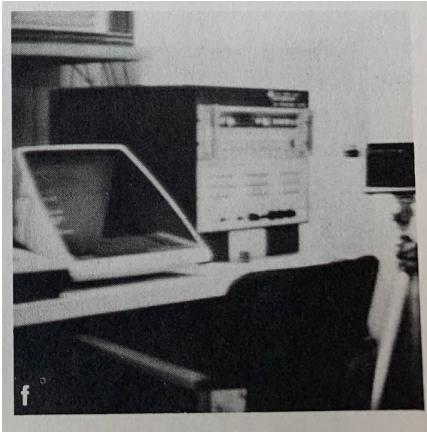
IFFT



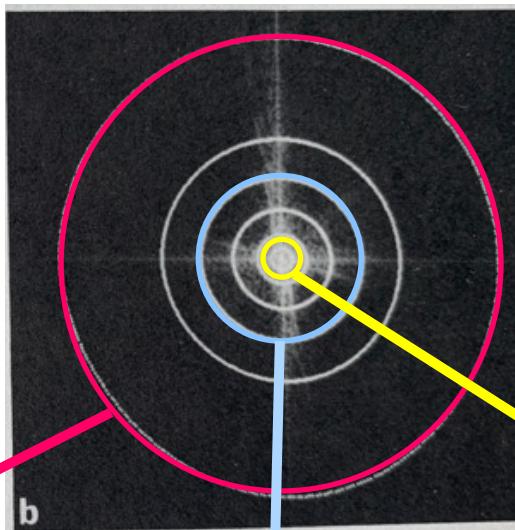
Deblurred image

Data filtering in Fourier Space: (de)convolution

Original Image

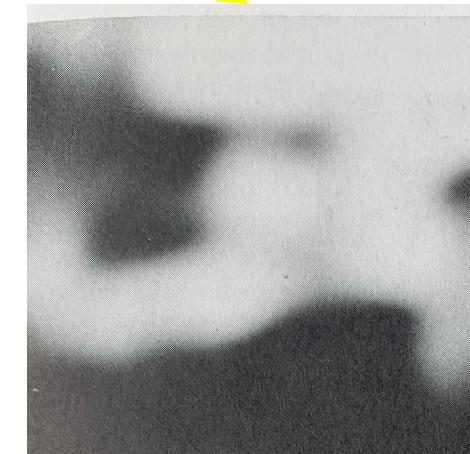
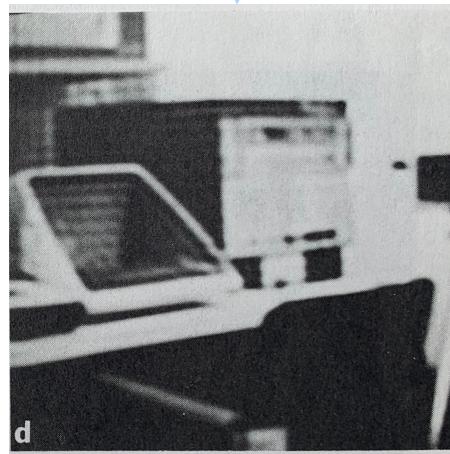
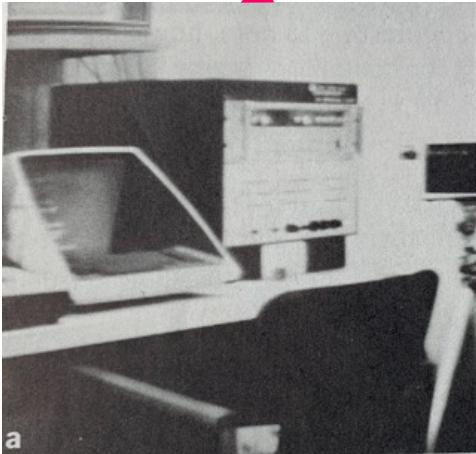


FFT



- Center points = low contrast/low frequency regions
- Outer points = Edges AKA where white meets black

IFFT



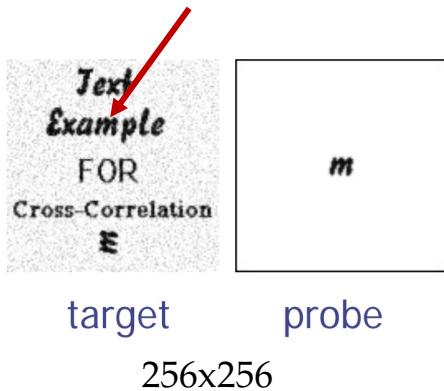
FFT is a way to measure where 2 images correlate

1. Overlay “probe” over “target”
2. Calculate correlation of every pixel in grid $[O(N^2)]$



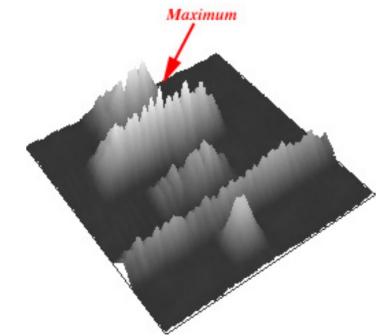
3. Repeat across all rows and columns $[O(N^2)]$

$[O(N^4)] = \sim 4.3 * 10^9$



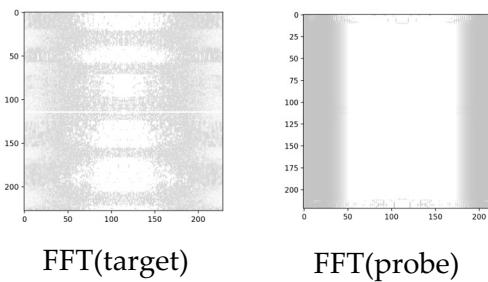
“Regular” 2D correlation

FFT 2D correlation



$[O(N^2 \log N^2)] = \sim 3.2 * 10^5$

1. Calculate FFT of “probe” and “target” individually $[O(\log N^2)]$



2. Pixel-wise multiplication $[O(N^2)]$

Why use FFT?

Speed!!!

- 2D Correlation: $O(N^4)$
- 2D FFT: $O(N^2 \log N^2)$
- 3D Correlation: $O(N^6)$
- 3D FFT: $O(N^3 \log N^3)$
- 6D Correlation: $O(N^{12})$
- 6D FFT: $O(N^6 \log N^6)$

At 100 pixels/dimension:

$$O(100^4) \sim 10^7$$

$$O(100^2 * \log(100^2)) \sim 10^4$$

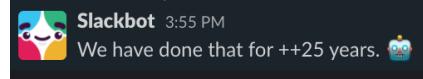
$$\sim 10^{12}$$

$\sim 10^6$ (Translational xyz search)

$$\sim 10^{24}$$

$\sim 10^{13}$ (Translational + rotational search)

This isn't new

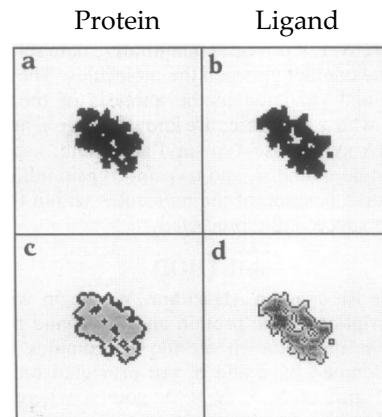


Paper from Katchalski-Katzir (1992)

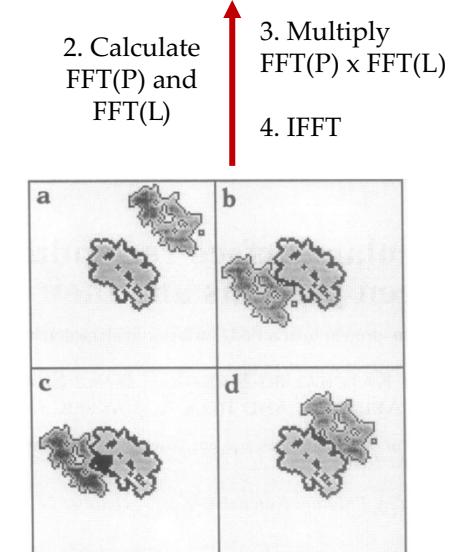
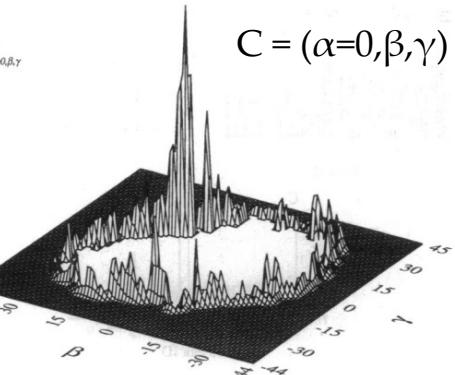
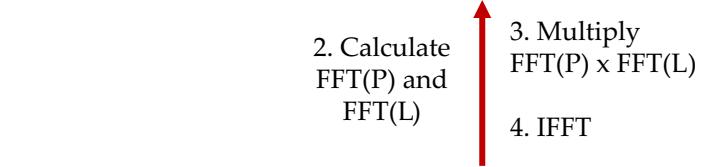
- Goal: Figure out where protein + ligand have most shape complementarity
- Steps:
 1. Convert protein and ligand structures to grids at specific (α, β, γ) orientation
 2. FFT of both protein and ligand grid
 3. $C = \text{FFT}(\text{protein}) \times \text{FFT}(\text{ligand})$
 4. IFFT(C)
 5. Rotate ligand and repeat

$$\text{Protein} = \begin{cases} 1 & \text{on the surface of the molecule} \\ \rho & \text{inside the molecule (-15)} \\ 0 & \text{outside the molecule,} \end{cases}$$

$$\text{Ligand} = \begin{cases} 1 & \text{on the surface of the molecule} \\ \delta & \text{inside the molecule (+1)} \\ 0 & \text{outside the molecule,} \end{cases}$$

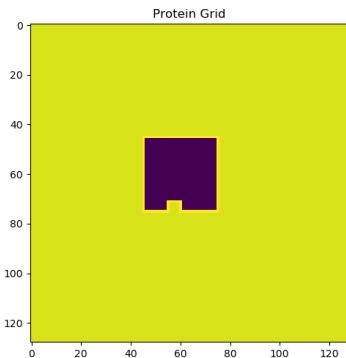


1. Overlay grids in orientation (α, β, γ)

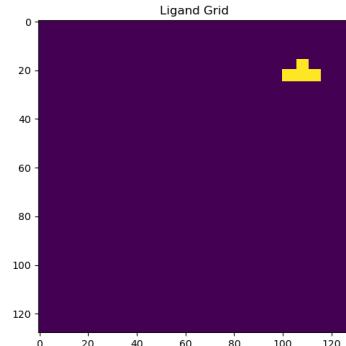
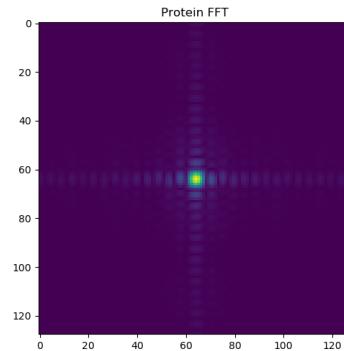


2D tetris example:

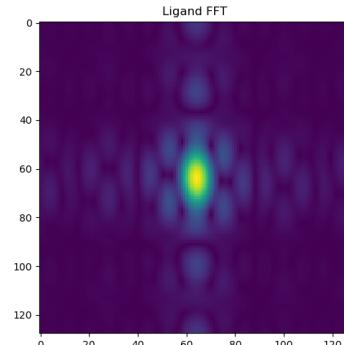
$$P = \begin{cases} +1 \text{ on surface} \\ -15 \text{ in interior} \\ 0 \text{ everywhere else} \end{cases}$$



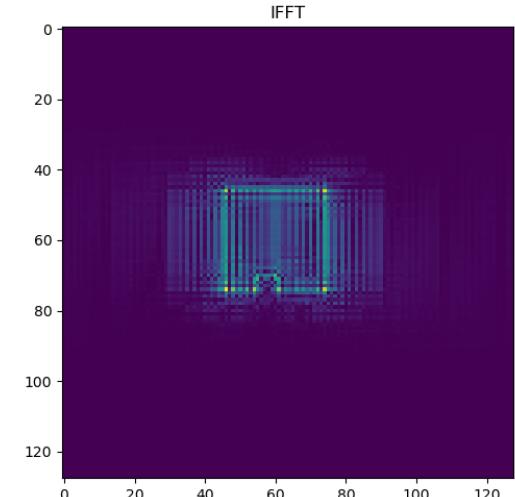
FFT



FFT



Multiply
FFTs



Peaks designate
overlap/shape
complementarity

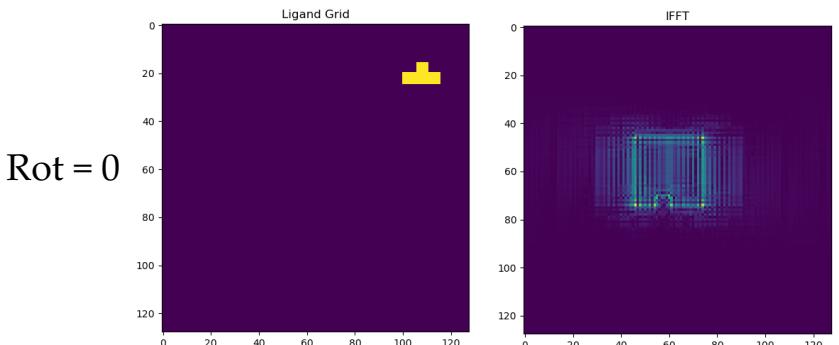
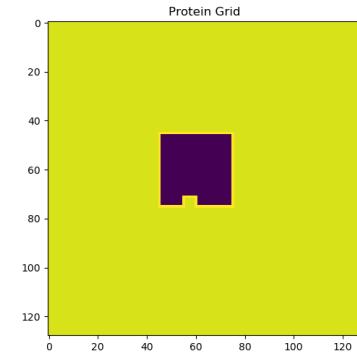
$$L = \begin{cases} 1 \text{ on surface} \\ +1 \text{ in interior} \\ 0 \text{ everywhere else} \end{cases}$$



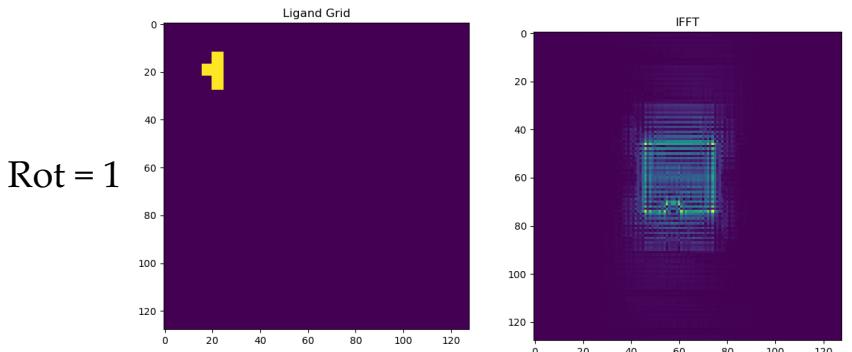
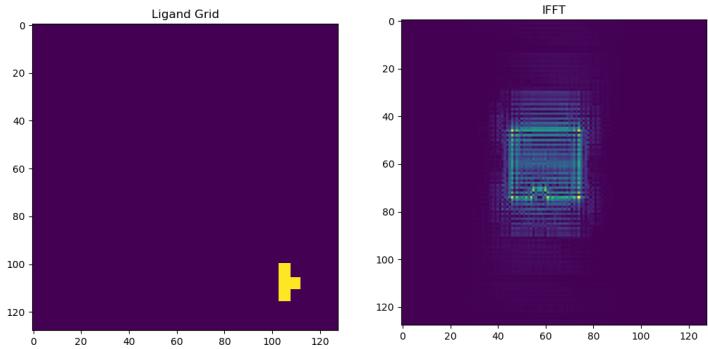
2D tetris example with ligand rotation:

for rotation in 0..3:

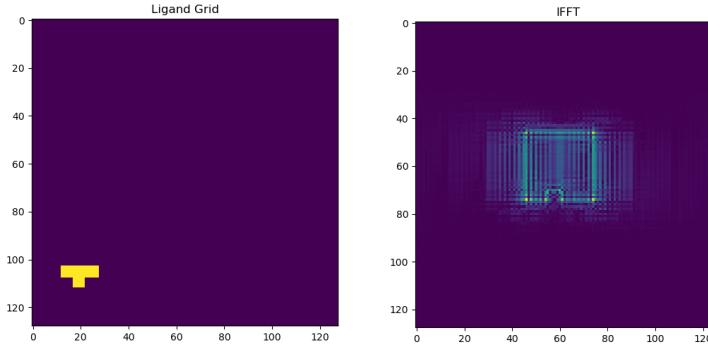
1. Ligand grid = np.rot90(ligand grid, rotation)
2. FFT(protein grid) and FFT(ligand grid)
3. C = FFT(protein grid) x FFT(ligand grid)
4. IFFT(C)



Rot = 3



Rot = 2

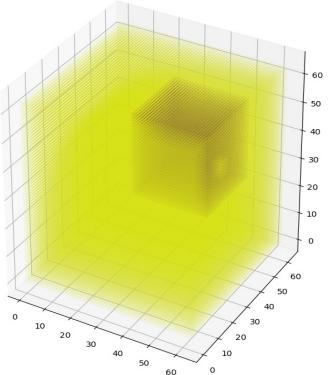


Note:

IFFTs of rotations 1 and 3 are **nearly identical**

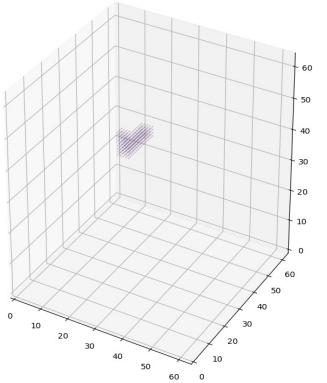
IFFTs of rotations 0 and 2 are close, but **magnitude varies in pocket**

3D tetris example

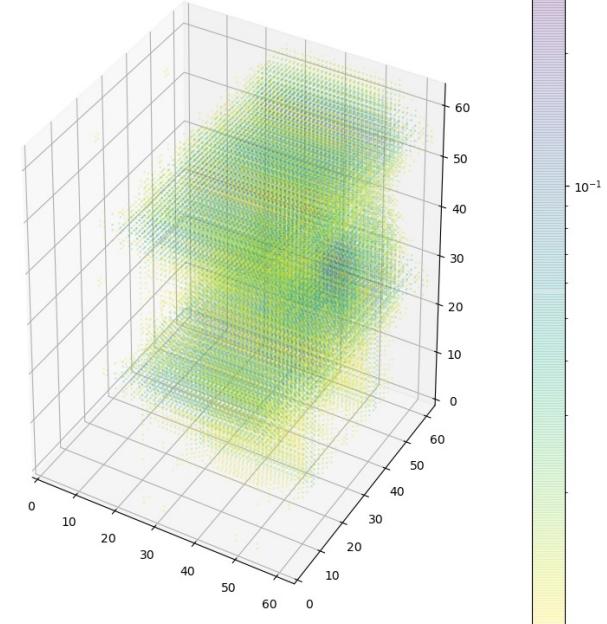


$$P = \begin{cases} +1 \text{ on surface} \\ -15 \text{ in interior} \\ 0 \text{ everywhere else} \end{cases}$$

$\text{IFFT}[\text{FFT}(P) \times \text{FFT}(L)]$



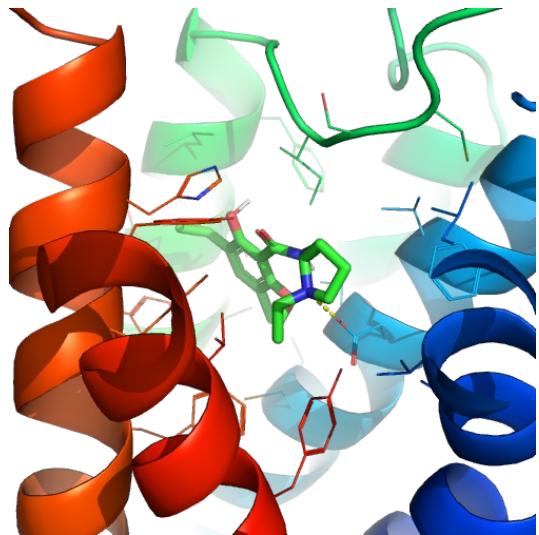
$$L = \begin{cases} 1 \text{ on surface} \\ +1 \text{ in interior} \\ 0 \text{ everywhere else} \end{cases}$$



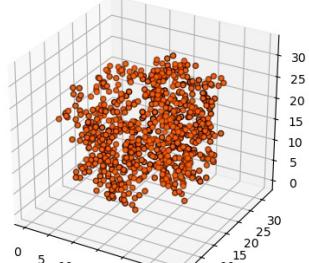
Purple area shows
peak overlap ✓

Now let's try with real things...round 1

1. 30\AA^3 cube around ligand center
2. Make every atom a point with a score (all set to -15)

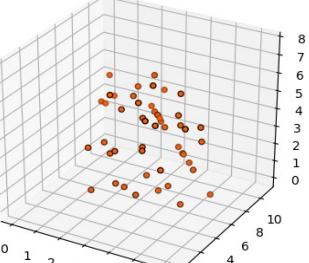


Protein



Protein grid

Ligand

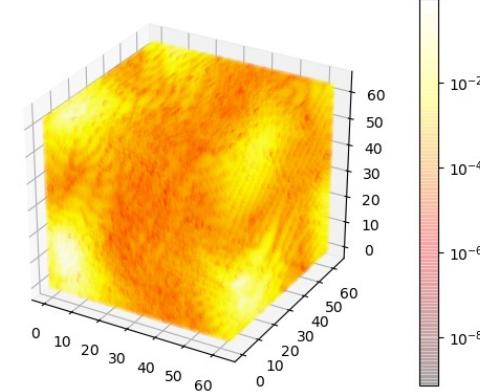


Ligand grid

$$P = \begin{cases} -15 \text{ at atom coordinate} \\ 0 \text{ everywhere else} \end{cases}$$

$$L = \begin{cases} -15 \text{ at atom coordinate} \\ 0 \text{ everywhere else} \end{cases}$$

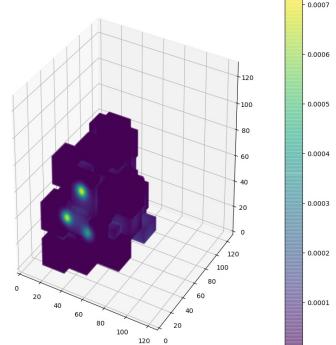
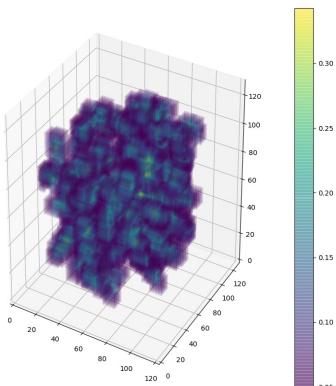
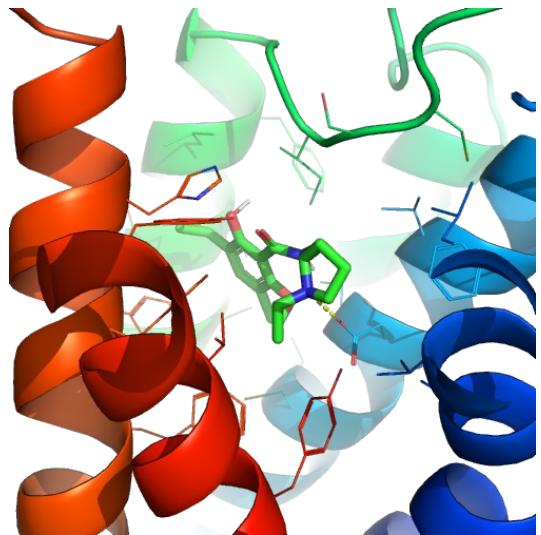
IFFT (I'm sorry—it's difficult to show 3D)



Peaks don't seem to correlate to anything (?)

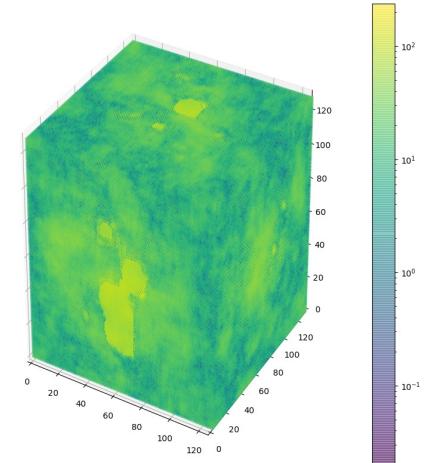
Now let's try with real things...round 2

1. 30\AA^3 cube around ligand center
2. Making initial grids : Created a “surface” based on atomic radii with Gaussian-based smoothing



EDIT:
dimensions
are wrong

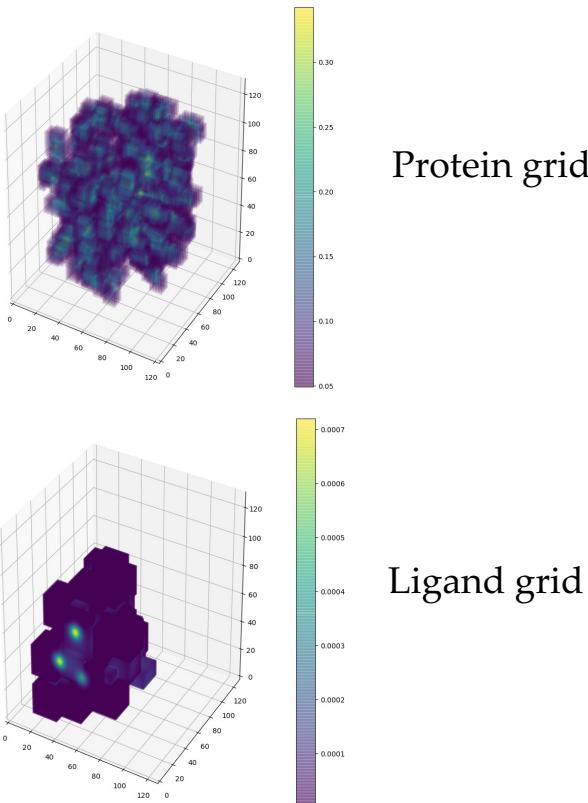
IFFT (I'm sorry—it's difficult to show 3D)



Now let's try with real things...round 2

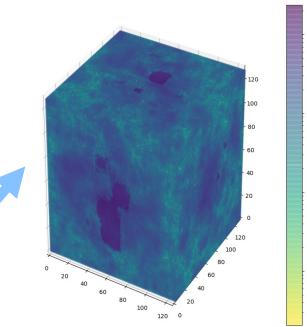
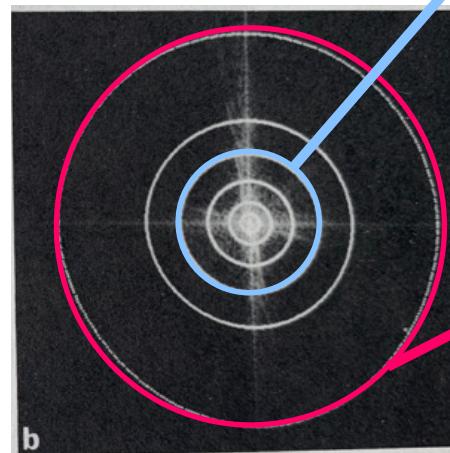
1. 30\AA^3 cube around ligand center
2. Making initial grids : Created a “surface” based on atomic radii with Gaussian-based smoothing
3. Deconvolution (de-blurring) step: Apply filter in Fourier space to extract high frequency areas (AKA areas where surface \leftrightarrow void)

IFFT (I'm sorry—it's difficult to show 3D)

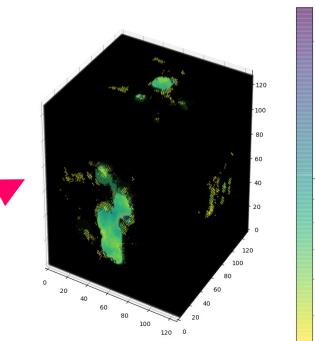


Fourier Space

Filter out low frequency areas (set values inside circle to 0)



Some low frequency parts filtered



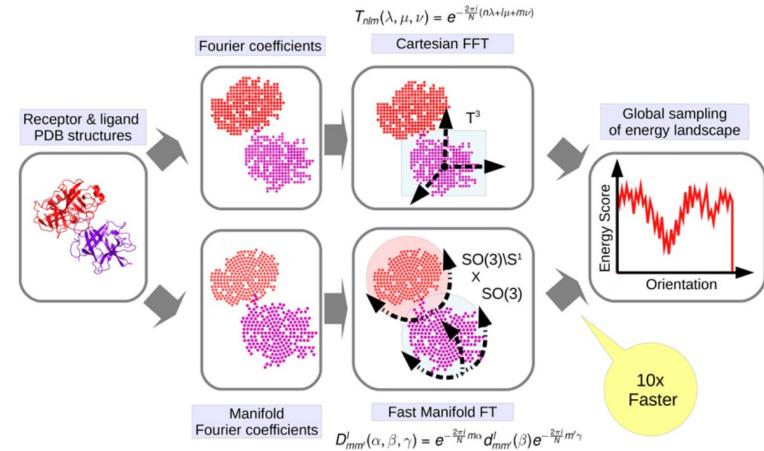
Only very high frequency areas left

Things to try next:

- Add rotational sampling in Fourier Space (Fast Manifold Fourier Transform)
5D FFT, 1D non-FFT

Protein–protein docking by fast generalized Fourier transforms on 5D rotational manifolds

Dzmitry Padhorny^{a,b}, Andrey Kazennov^b, Brandon S. Zerbe^c, Kathryn A. Porter^c, Bing Xia^c, Scott E. Mottarella^c, Yaroslav Kholodov^{b,d,e}, David W. Ritchie^f, Sandor Vajda^c, and Dima Kozakov^{a,g,h,1}



- How can we include different conformations without just looping over them?
Represent conformations on a single grid based on relative density (so basically “occupancy” in crystallography, I think?! @Nina correct me if I’m wrong!)

Thanks!

- Jens
- Sam Schmitz
- Gary Smith (dad)
- StackOverflow responders ☺

