

# Hex – Modeling Protein Docking Using Polar Fourier Correlations

Dave Ritchie

Team Orpailleur Inria Nancy – Grand Est

#### Outline

Basic Principles of Docking

Fast Fourier Transform (FFT) Docking Methods

Hex Polar Fourier Correlation Method Explained

The CAPRI Experiment

Demo: Using Hex on Linux

Practical: CAPRI Target 40 – API-A/Trypsin



Biological Importance of Protein-Protein Interactions Protein interactions (PPIs) are central to many biological systems

• Humans have about 30,000 proteins, each having about 5 PPIs

• Understanding PPIs could lead to immense scientific advances



Protein-protein interactions as therapeutic drug targets

• Small "drug" molecules often inhibit or interfere with PPIs



Ínría\_

• A six-dimensional puzzle - do these proteins fit together?



• Yes, they fit!





- Yes, they fit!
- It is mostly a rotational problem: ONE translation plus FIVE rotations...

Ínría



- Yes, they fit!
- It is mostly a rotational problem: ONE translation plus FIVE rotations...
- But proteins are flexible => multi-dimensional space!





- Yes, they fit!
- It is mostly a rotational problem: ONE translation plus FIVE rotations...
- But proteins are flexible => multi-dimensional space!
- So, how to calculate whether two proteins recognise each other?



# ICM Docking – Multi-Start Pseudo-Brownian Search

- Stick pins in protein surfaces at 15Å intervals
- For each pair of pins, find minimum energy (6 rotations for each):
  - $E = E_{HVW} + E_{CVW} + 2.16E_{el} + 2.53E_{hb} + 4.35E_{hp} + 0.20E_{solv}$



• Often gives good results, but is computationally expensive

Fernández-Recio, Abagyan (2004), J Mol Biol, 335, 843-865



# Protein Docking Using Fast Fourier Transforms

• Conventional approaches digitise proteins into 3D Cartesian grids...



• ...and use FFTs to calculated TRANSLATIONAL correlations:

$$C[\Delta x, \Delta y, \Delta z] = \sum_{x, y, z} A[x, y, z] \times B[x + \Delta x, y + \Delta y, z + \Delta z]$$

BUT for docking, have to repeat for many rotations – expensive!
Conventional grid-based FFT docking = SEVERAL CPU-HOURS

Katchalski-Katzir et al. (1992) PNAS, 89 2195–2199



# Protein Docking Using Polar Fourier Correlations

- Rigid docking can be considered as a largely ROTATIONAL problem
- This means we should use ANGULAR coordinate systems



• With FIVE rotations, we should get a good speed-up?



# Some Theory – 2D Spherical Harmonic Surfaces

• Spherical harmonics (SHs) are classical "special functions"



- SHs are products of Legendre polynomials and circular functions:
  - Real SHs:  $y_{lm}(\theta, \phi) = P_{lm}(\theta) \cos m\phi + P_{lm}(\theta) \sin m\phi$
  - Complex SHs:  $Y_{lm}(\theta,\phi) = P_{lm}(\theta)e^{im\phi}$
  - Orthogonal:  $\int y_{lm} y_{kj} d\Omega = \int Y_{lm} Y_{kj} d\Omega = \delta_{lk} \delta_{mj}$
  - <u>Rotation:</u>

nín

$$y_{lm}(\theta',\phi') = \sum_{j} R_{jm}^{(l)}(\alpha,\beta,\gamma) y_{lj}(\theta,\phi)$$



## Spherical Harmonic Molecular Surfaces

• Use spherical harmonics (SHs) as orthogonal shape "building blocks"



- Reals SHs  $y_{lm}(\theta, \phi)$ , and coeffcients  $a_{lm}$
- Encode distance from origin as SH series:

$$r(\theta,\phi) = \sum_{l=0}^{L} \sum_{m=-l}^{l} a_{lm} y_{lm}(\theta,\phi)$$

#### • Calculate coefficients by numerical integration



Good for shape-matching, not so good for docking...

Ritchie and Kemp (1999), J. Comp. Chem. 20, 383-395

nnin

#### Docking Needs 3D Polar Fourier Representation

• Special orthonormal Laguerre-Gaussian radial functions,  $R_{nl}(r)$ 

$$R_{nl}(r) = N_{nl}^{(q)} e^{-\rho/2} \rho^{l/2} L_{n-l-1}^{(l+1/2)}(\rho); \qquad \rho = r^2/q, \quad q = 20.$$



## SPF Protein Shape-Density Reconstruction

Interior density: 
$$\tau(\underline{r}) = \sum_{nlm}^{N} a_{nlm}^{\tau} R_{nl}(r) y_{lm}(\theta, \phi)$$



Image	Order	Coeffs
A	Gaussians	-
В	N = 16	1,496
С	N = 25	5,525
D	N = 30	9,455

Ritchie (2003), Proteins Struct. Funct. Bionf. 52, 98-106



#### Protein Docking Using SPF Density Functions



Ritchie and Kemp (2000), Proteins Struct. Funct. Bionf. 39, 178-194



#### Hex SPF Correlation Example – 3D Rotational FFTs

- Set up 3D rotational FFT as a series of matrix multiplications:
- Rotate:  $a'_{nlm} = \sum_{t=-l}^{l} R^{(l)}_{mt}(0, \beta_A, \gamma_A) a_{lt}$
- Translate:  $a''_{nlm} = \sum_{kj}^{N} T^{(|m|)}_{nl,kj}(R) a'_{kjm}$
- Real to complex:  $A_{nlm} = \sum_{t} a''_{nlt} U^{(l)}_{tm}, \quad B_{nlm} = \sum_{t} b_{nlt} U^{(l)}_{tm}$
- Multiply:  $C_{muv} = \sum_{nl} A^*_{nlm} B_{nlv} \Lambda^{um}_{lv}$
- 3D FFT:  $S(\alpha_B, \beta_B, \gamma_B) = \sum_{muv} C_{muv} e^{-i(m\alpha_B + 2u\beta_B + v\gamma_B)}$
- On one CPU, docking takes from 15 to 30 minute...

# Exploiting Proir Knowledge in SPF Docking



• Knowing just one key residue can reduce search space enormously...

• This accelerates calculation and helps to reduce false-positives...



# Docking Very Large Molecules Using Multi-Sampling

• Example: docking an antibody to the VP2 viral surface protein





## The CAPRI Experiment

#### • CAPRI = "Critical Assessment of PRedicted Interactions"

Predictor	Software	Algorithm	T1	T2	T3	T4	T5	T6	T7
Abagyan	ICM	FF			**			***	**
Camacho	CHARMM	FF	*					***	***
Eisenstein	MolFit	FFT	*	*					***
Sternberg	FTDOCK	FFT		*				**	*
Ten Eyck	DOT	FFT	*	*				**	
Gray		MC						**	***
Ritchie	Hex	SPF			**			***	
Weng	ZDOCK	FFT		**					**
Wolfson	BUDDA/PPD	GH	*						***
Bates	Guided Docking	FF	-	-	-				***
Palma	BIGGER	GF	-		-			**	*
Gardiner	GAPDOCK	GA	*	*	-	-	-	-	-
Olson	Surfdock	SH	*			-	-	-	-
Valencia		ANN	*	-	-	-	-	-	-
Vakser	GRAMM	FFT		*		-	-	-	-

 $\ast$  low,  $\ast\ast$  medium,  $\ast\ast\ast$  high accuracy prediction; - no prediction

Mendez et al. (2003) Proteins Struct. Funct. Bionf. 52, 51-67



# Hex Protein Docking Example – CAPRI Target 3

• Example: best prediction for CAPRI Target 3 – Hemagglutinin/HC63



Ritchie and Kemp (2000), Proteins Struct. Funct. Bionf. 39, 178–194 Ritchie (2003), Proteins Struct. Funct. Genet. 52, 98–106



#### Best Hex Orientation for Target 6 – Amylase/AMD9



• CAPRI "high accuracy" (Ligand RMSD  $\leq$  1Å)



## Subsequent CAPRI Targets 8 – 19

Target	Description	Comments
T8	Nidogen- $\gamma$ 3 - Laminin	U/U
Т9	LiCT homodimer	build from monomer – 12Å RMS deviation
T10	TBEV trimer	build from monomer – 11Å RMS deviation
T11	Cohesin - dockerin	U/U; model-build dockerin
T12	Cohesin - dockerin	U/B
T13	SAG1 - antibody Fab	SAG1 conformational change: 10Å RMS
T14	MYPT1 - PP1 $\delta$	U/U; model-build PP1 $lpha  ightarrow$ PP1 $\delta$
T18	TAXI - xylanase	U/B
T19	Ovine prion - antibody Fab	model-build prion

- T15-T17 cancelled: solutions were on-line & found by Google !!
- T11, T14, T19 involved homology model-building step...



CAPRI Results: Targets 8–19 (2003 – 2005)

Software	Т8	T9	T10	T11	T12	T13	T14	T18	T19
ICM	**		*	**	***	*	***	**	**
PatchDock	**	*	*	*	*	-	**	**	*
ZDOCK/RDOCK	**			*	***	***	***	**	**
FTDOCK	*		*	**	*		**	**	*
RosettaDock	-			**	***	**	***		***
SmoothDock	**				***	***	**	**	*
RosettaDock	***	-	-	**	***				**
Haddock	-	-	**	**		***	***		
ClusPro	**				***	*			*
3D-DOCK	**			*	*		**		*
MolFit	***			*	***		**		
Hex				**	***	*	*		
Zhou	-	-		-	***	**	*	*	
DOT					***	***	**		
ATTRACT	**		-	-	-	-	***		**
Valencia	*			*	*	-			-
GRAMM	-	-		-	-	-	**	**	
Umeyama				**	*				
Kaznessis	-	-			***				
Fano	-	-		*					

Mendez et al. (2005) Proteins Struct. Funct. Bionf. 60, 150-169



#### "Hex" and "HexServer"

- Hex: interactive docking (~ 33,000 downloads) http://hex.loria.fr/
- Hexserver (~ 1,000 docking jobs/month) http://hexserver.loria.fr/



Ritchie and Kemp (2000), Proteins 39 178-194

Macindoe et al. (2010), Nucleic acids Research, 38, W445–W449



Inside Hex – High Order FFTs, Multi-threading on GPUs

• SPF approach => analytic <u>translational</u> + <u>rotational</u> correlations:

In particular: 
$$S_{AB} = \sum_{jsmlvrt} \Lambda_{js}^{rm} T_{js,lv}^{(|m|)}(R) \Lambda_{lv}^{tm} e^{-i(r\beta_A - s\gamma_A + m\alpha_B + t\beta_B + v\gamma_B)}$$

- This allows high order FFTs to be used 1D, 3D, and 5D
- It also allows calculations to be easily ported to modern GPUs



- Up to 2048 arithmetic "cores"
- Up to 8 Gb memory
- Easy API with C++ syntax
- Grid of threads model ("SIMT")

BUT – for best results, need to understand the hardware...

Ritchie, Kozakov, Vajda (2008), Bioinformatics 24, 1865–1873 Ritchie and Venkatraman (2010), Bioinformatics, 26, 2398–2405



#### CUDA Device Architecture

• Typically 8–16 multiprocessor blocks, each with 16 thread units



- NB. only a very small amount of fast shared memory is available
- $\bullet\,$  NB. global memory is  $\sim\,$  80x slower than shared memory
- Strategy: aim for "high arithmetic intensity" in shared memory

# CUDA Programming Example – Matrix Multiplication

- Matrix multiplication C = A \* B
- Each thread is responsible for calculating one element: C[i,k]



• Conventional algorithm:

• 
$$C[i,k] = A[i] * B[k]$$

- Thread-block algo uses TILES
- Tiles of 16x16 is just right!
- $\bullet\,$  Threads co-operate by reading & sharing tiles of A & B
- Multi-processor launches multiple blocks to compute all of C
- Executing thread-blocks concurrently hides global memory latency



#### GPU Implementation – Perform Multiple FFTs

• Calculate multiple 1D FFTs of the form:

$$S_{AB}(\alpha_B) = \sum_{m} e^{-im\alpha_B} \sum_{nl} A^{\sigma}_{nlm}(R, \beta_A, \gamma_A) \times B^{\tau}_{nlm}(\beta_B, \gamma_B)$$

- Cross-multiply transformed A with rotated B coefficients
- Perform batch of 1D FFTs using cuFFT and save best orientations



• 3D FFTs in  $(\alpha_B, \beta_B, \gamma_B)$  can be calculated in a similar way...



#### Results – Multiple GPUs and CPUs

• With Multi-threading, we can use all available GPUs and CPUs



- Best performance: use 2 GPUs alone, or 6 CPUs plus 2 GPUs
- 2 GPUs => 6D docking in about 15 sec important for large-scale!



## Speed Comparison with ZDOCK and PIPER

- Hex: 52000 x 812 rotations, 50 translations (0.8Å steps)
- ZDOCK: 54000 × 6 deg rotations, 92Å 3D grid (1.2Å cells)
- PIPER: 54000 × 6 deg rotations, 128Å 3D grid (1.0Å cells)
- Hardware: GTX 285 (240 cores, 1.48 GHz)

	Kallikrein A / BPTI (233 / 58 residues) $\#$								
	ZDOCK	PIPER <sup>†</sup>	PIPER <sup>†</sup>	Hex	Hex	Hex‡			
FFT	1xCPU	1xCPU	1xGPU	1xCPU	4xCPU	1xGPU			
3D	7,172	468,625	26,372	224	60	84			
(3D)*	(1,195)	(42,602)	(2,398)	224	60	84			
1D	-	-	-	676	243	15			

- What's next ?
  - Better energy functions?
  - Modeling flexibility?
  - Multi-component complexes?
  - Cross-docking?



## Conclusions

- (+) Rigid-body docking on a GPU now takes only a few seconds:
  - This was implemented using only 5 or 6 GPU kernels
- (-) Modeling protein flexibility during docking is still difficult
- SPF approach => high-throughput shape comparison now feasible:
  - All-vs-all docking ?
  - Electron-microscopy density fitting ?
  - Assembling multi-component machines ?
- (?) The next challenge modeling "the structural interactome"



# Thank You!

# Acknowledgments

Vishwesh Venkatraman Lazaros Mavridis

Anisah Ghoorah

Program and papers: http://hex.loria.fr/

Innin-

### Hex Demo – Basic Operations

- Hex web site: http://hex.loria.fr/dist800/
- Loading structures into Hex
- Basic concepts: "receptor", "ligand", "complex" (reference)
- Graphical viewing modes
- Editing the scene (moving structures around)
- Setting docking parameters
- Launching a docking calculation
- Viewing the results
- Saving structures
- ...
- Ask me!

• Disclaimer: please remember, Hex is not "commercial" software!



#### Practical: CAPRI Target 40 – API-A/Trypsin

R Bao at al. (2009), J Biol Chem, 284, 26676-26684

"The Ternary Structure of the Double-headed Arrowhead Protease Inhibitor API-A Complexed with Two Trypsins Reveals a Novel Reactive Site Conformation"

The double-headed arrowhead protease inhibitors API-A and -B from the tubers of *Sagittaria sagittifolia* (Linn) feature two distinct reactive sites, unlike other members of their family. Although the two inhibitors have been extensively characterized, the identities of the two P1 residues in both API-A and -B remain controversial. The crystal structure of a ternary complex at 2.84 Å resolution revealed that **the two trypsins bind on opposite sides of API-A and are 34 Å apart.** The overall fold of API-A site sides of API-A belongs to the  $\beta$ -trefoil fold and resembles that of the soybean Kunitz-type trypsin inhibitors. The two P1 residues [on API-A] were unambiguously assigned as **Leu87** and **Lys145**, and their identities were further confirmed by site-directed mutagenesis...

• The CAPRI challenge: blind prediction of the two binding modes...



#### CAPRI T40 Results

#### X-ray solution

#### **Our predictions**



• Using Hex + MD refinement gave NINE "acceptable" solutions

Ínría

#### Practical Activities

- Download the structures from: http://hex.loria.fr/emmsb/t40.tgz
  - t40\_a.pdb (Trypsin 1)
  - t40\_b.pdb (Trypsin 2)
  - t40\_c.pdb (API-A)
  - t40\_abc.pdb (solution)
  - t40.col (Hex colour file)
- Load the structures C+A or C+B as "receptor" and "ligand"
- Experiment with different graphical viewing options
- Use the "edit mode" to try docking by hand
- Load the solution structure as "complex" and try again by hand
- Load the color file to highlight the key residues
- Does this help?
- Finally, place the API-A key residue near the trypsin site
- Set up and run a focused docking calculation (45 deg on each)
- View and analyse by eye the solutions generated

