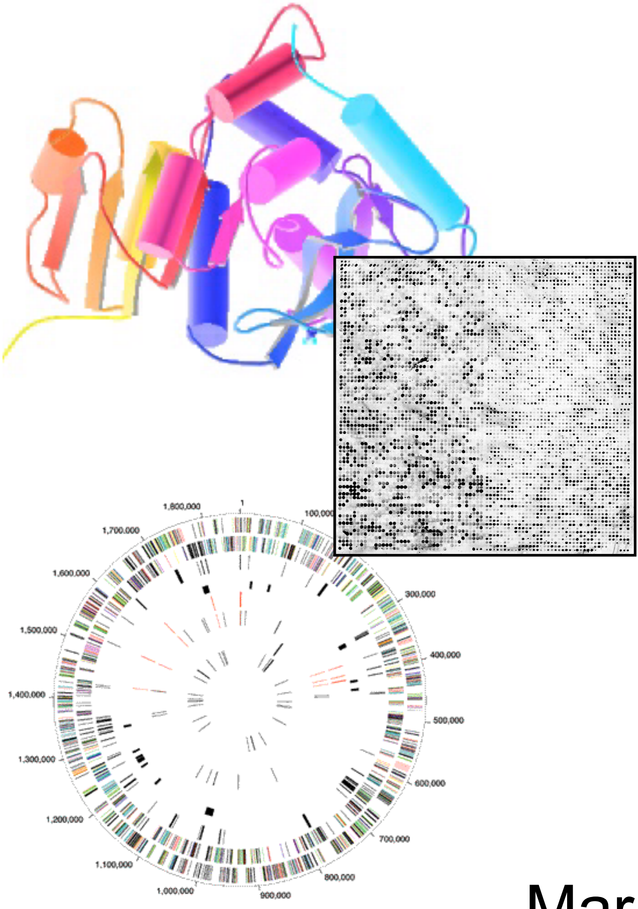
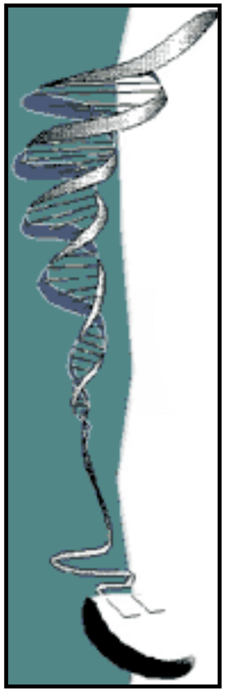
BIOINFORMATICS
Structures



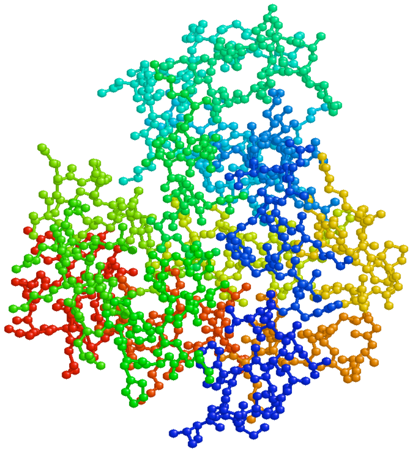


Mark Gerstein, Yale University
gersteinlab.org/courses/452
(last edit in fall '06, includes in-class changes)

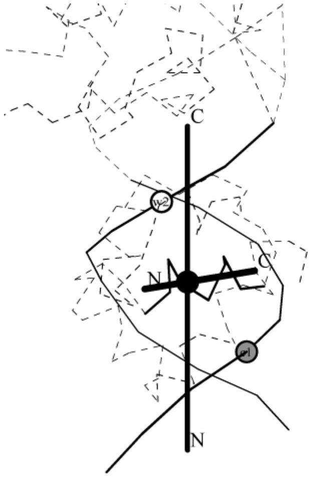
Contents: Structures

- What Structures Look Like?
- Structural Alignment by Iterated Dynamic Programming
  - ◊ RMS Superposition
  - ◊ Rotating and Translating Structures
- Scoring Structural Similarity
- Other Aspects of Structural Alignment
  - ◊ Distance Matrix based methods
- Elaborating structures
  - ◊ Surfaces and volumes

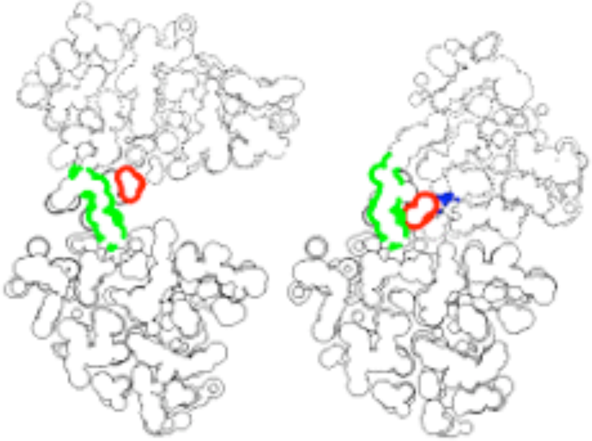
Other Aspects of Structure, Besides just Comparing Atom Positions



Atom
Position,
XYZ triplets



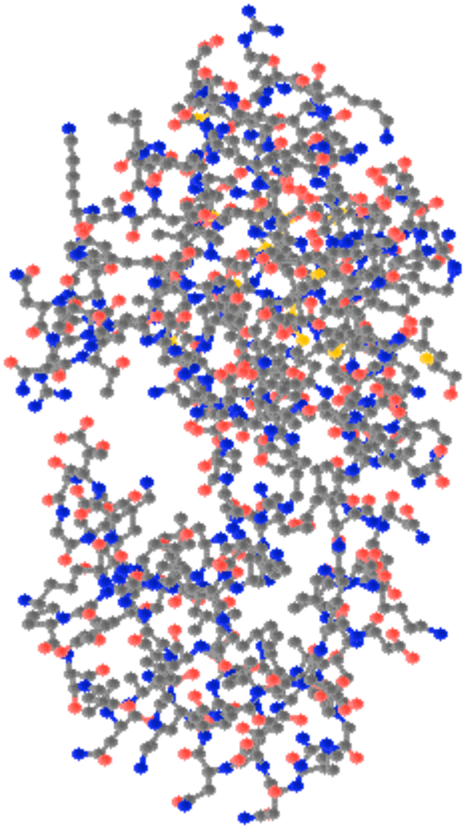
Lines, Axes,
Angles



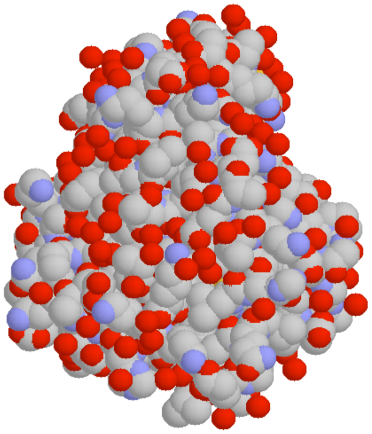
Surfaces, Volumes

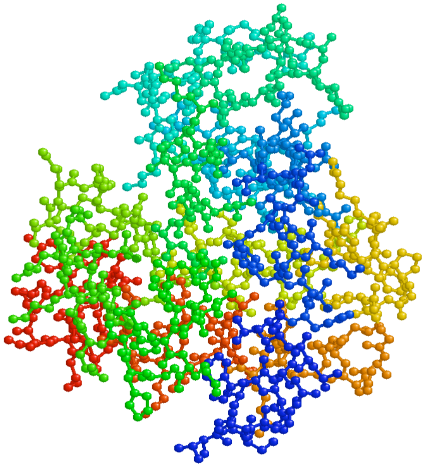
What is Protein Geometry?

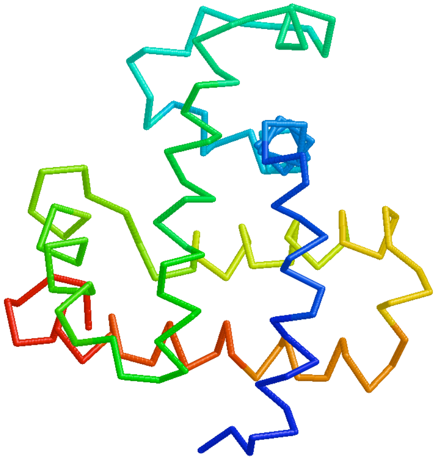
- Coordinates (X, Y, Z's)
- Derivative Concepts
  - ◊ Distance, Surface Area, Volume, Cavity, Groove, Axes, Angle, &c
- Relation to
  - ◊ Function, Energies (E(x)), Dynamics (dx/dt)

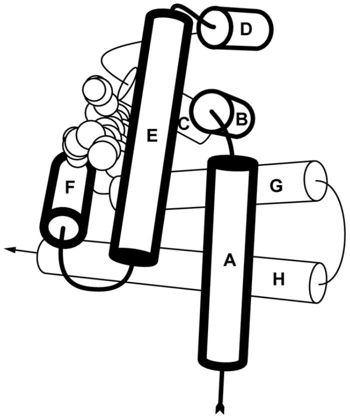


Depicting Protein Structure: Sperm Whale Myoglobin

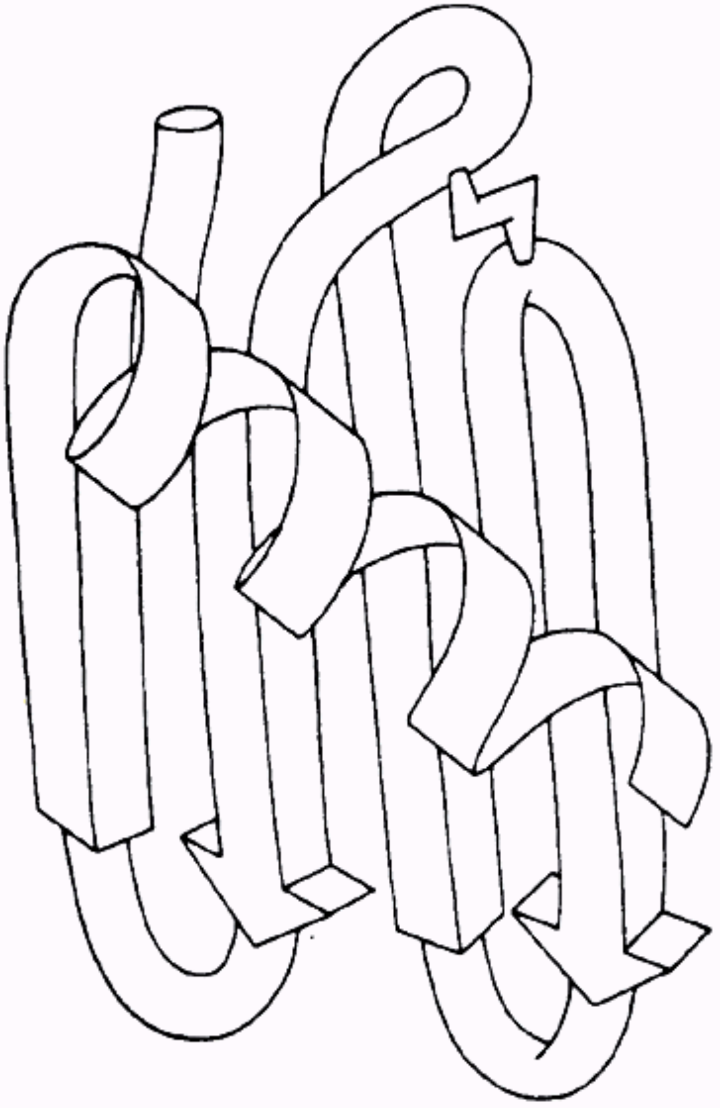








Incredulase



Incredulase

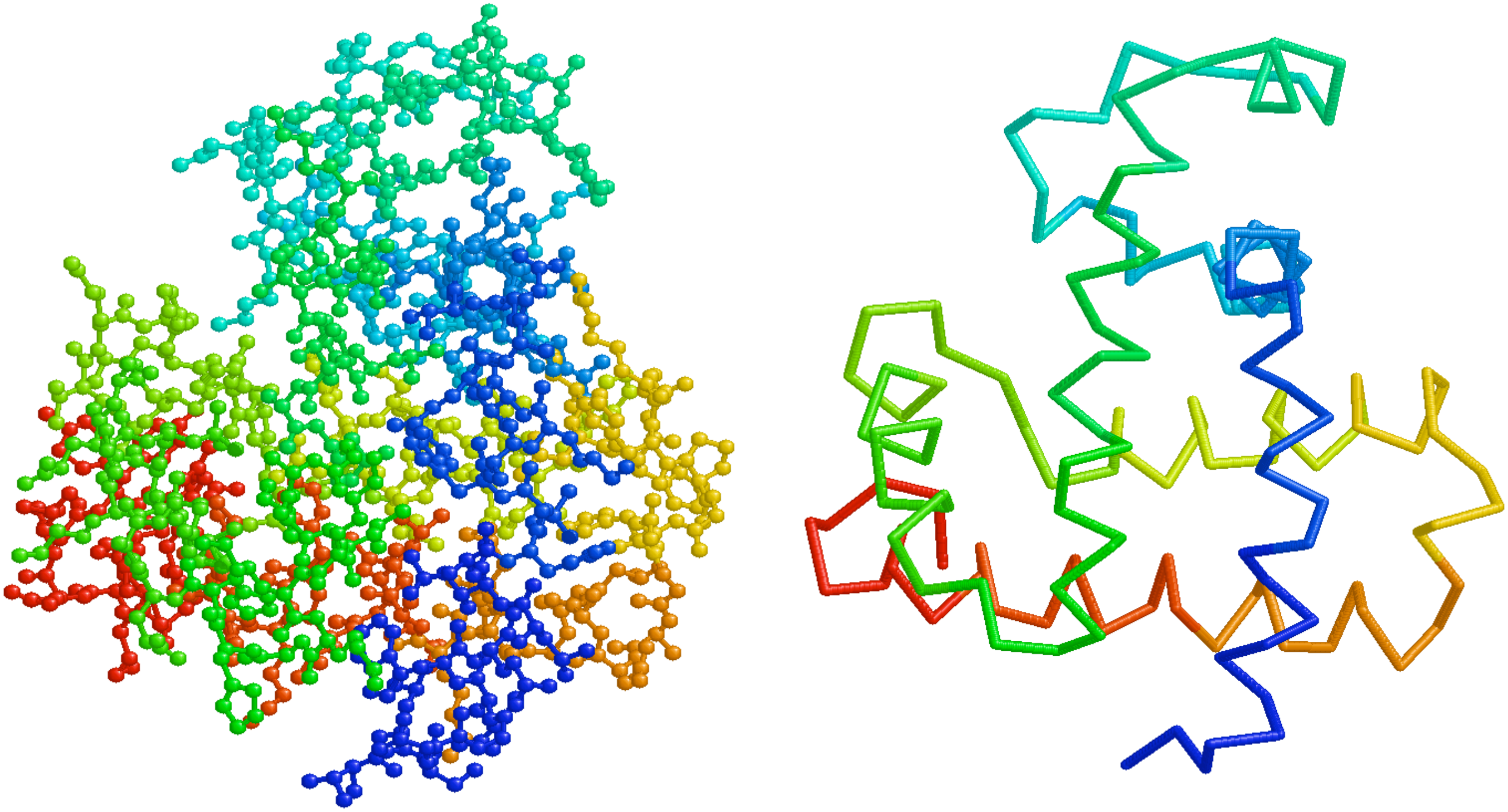
J.S. Richardson and D.C. Richardson, "*Some design principles: Betabellin*", in D.L. Oxender and C.F. Fox (Eds.), "*Protein Engineering*", Alan R. Liss, 1987, p. 149-163

Structure alignment - Method

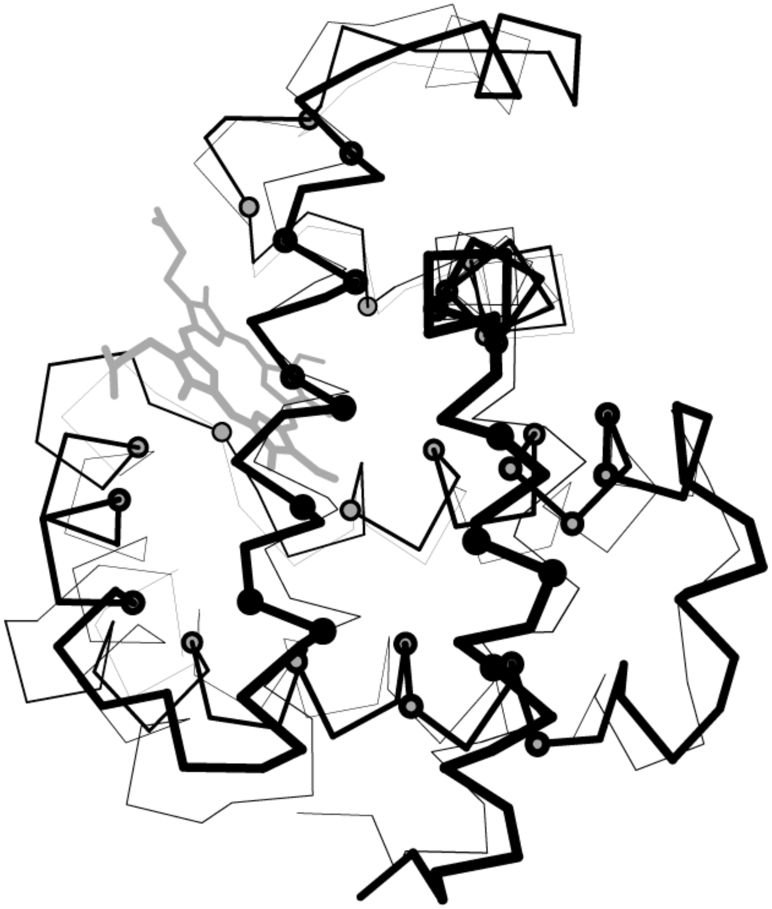
- What Structures Look Like?
- Structural Alignment by Iterated Dynamic Programming
  - ◊ RMS Superposition
- Scoring Structural Similarity
- Other Aspects of Structural Alignment
  - ◊ Distance Matrix based methods
  - ◊ Fold Library
- Relation of Sequence Similarity to Structural and Functional Similarity

- Protein Geometry
- Surface I (Calculation)
- Calculation of Volume
- Voronoi Volumes & Packing
- Standard Volumes & Radii
- Surfaces II (Relationship to Volumes)
- Other Applications of Volumes -- Motions, Docking

Sperm Whale Myoglobin



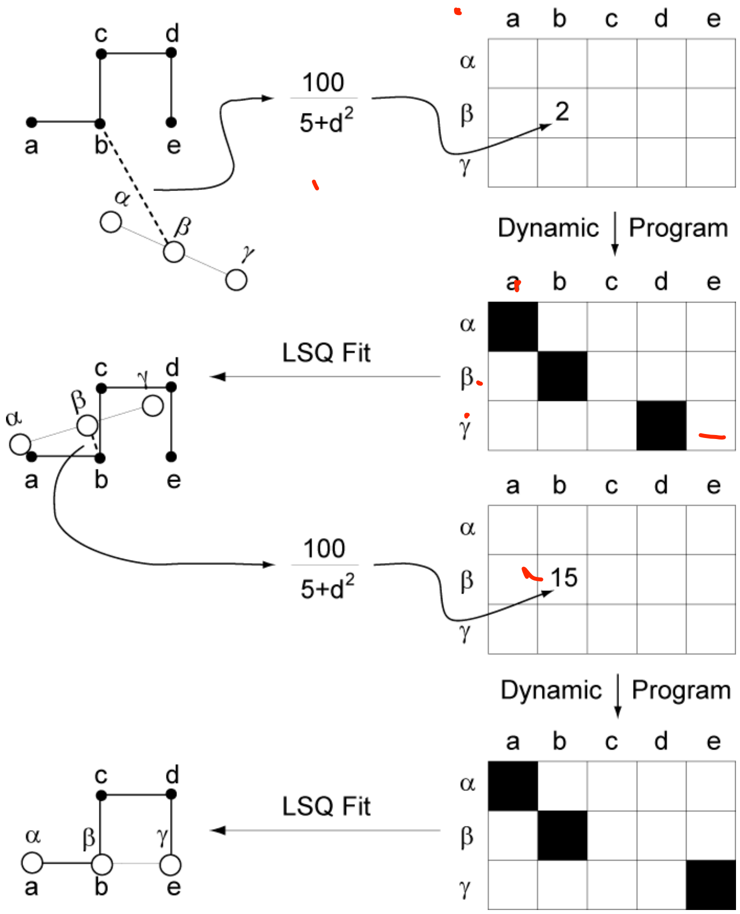
Structural Alignment of Two Globins



Automatically Comparing Protein Structures

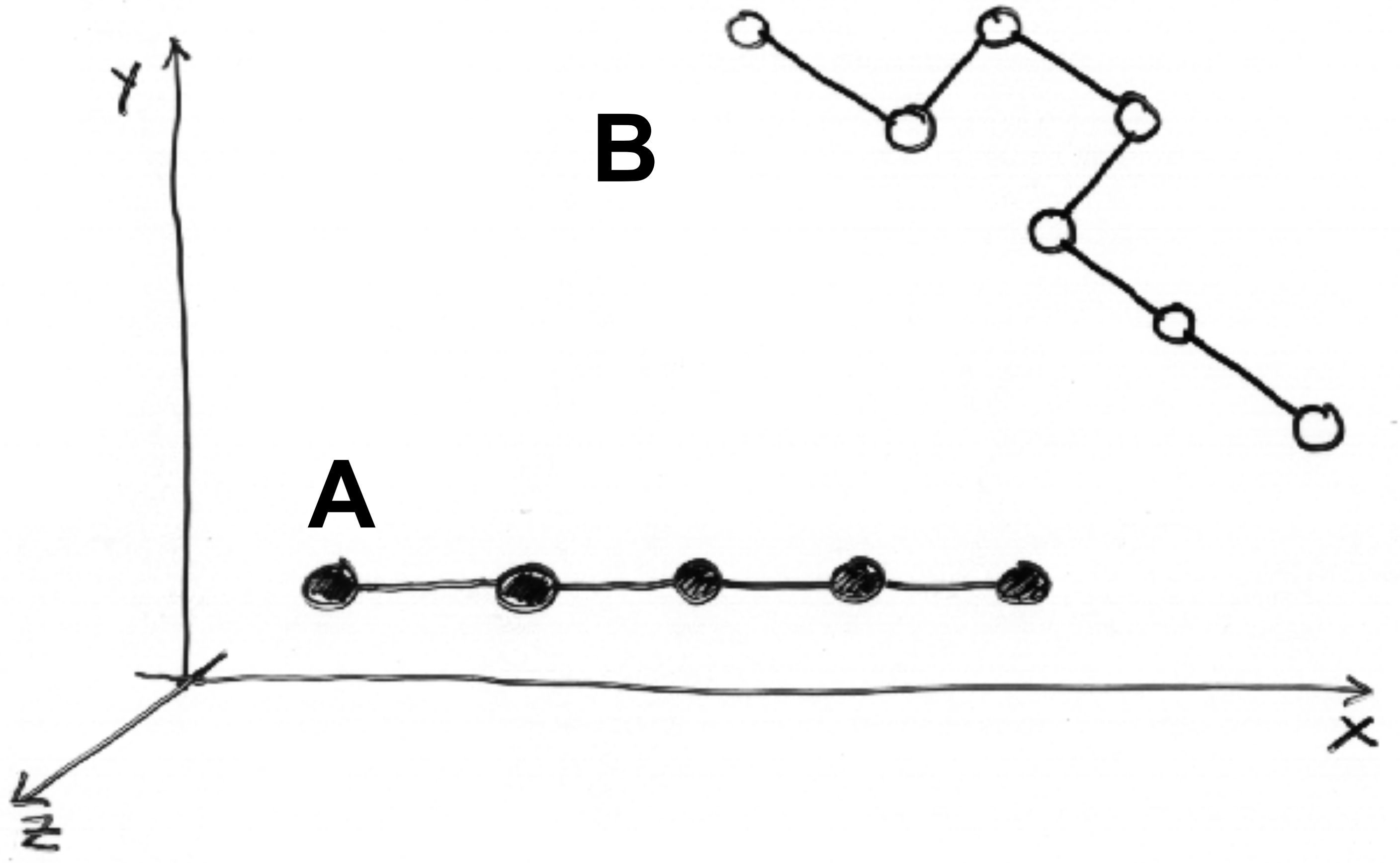
- Given
  2 Structures (A & B),
  2 Basic
  Comparison Operations
  1. Given an alignment optimally **SUPERIMPOSE** A onto B

     Find Best R & T to move A onto B
  2. **Find an Alignment** between A and B based on their 3D coordinates



Cor

RMS Superposition (1)

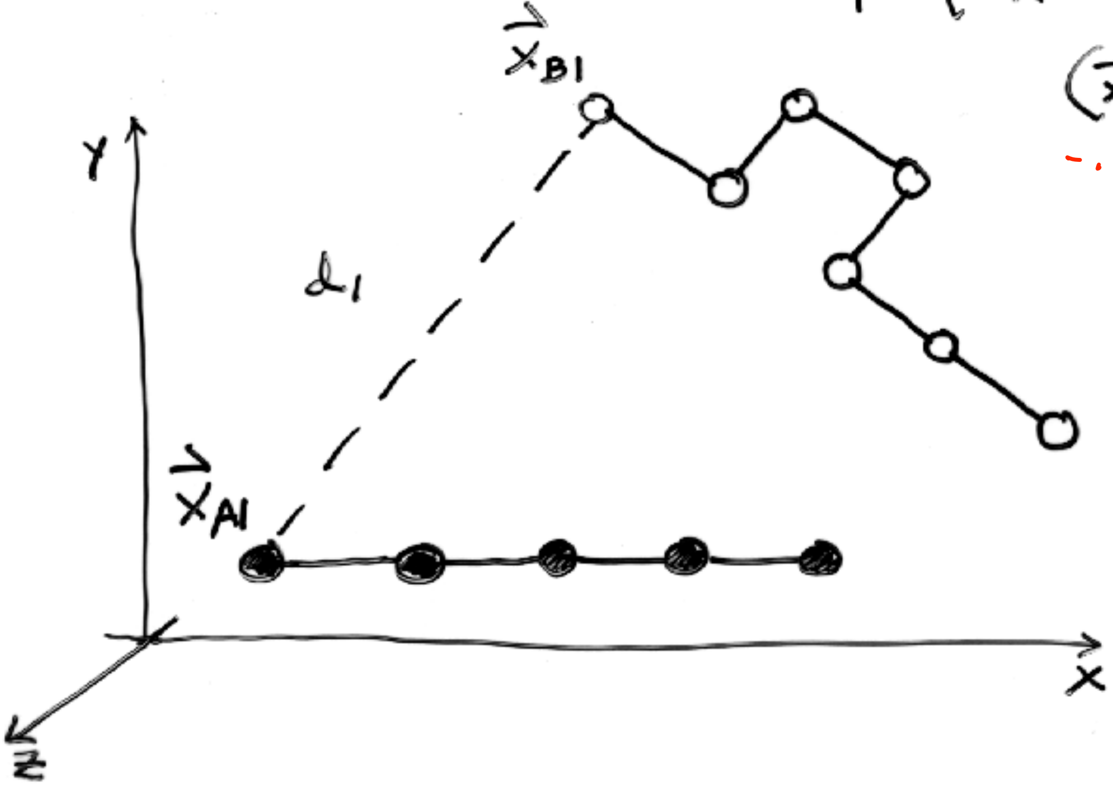


Cor

RMS Superposition (2): Distance Between an Atom in 2 Structures

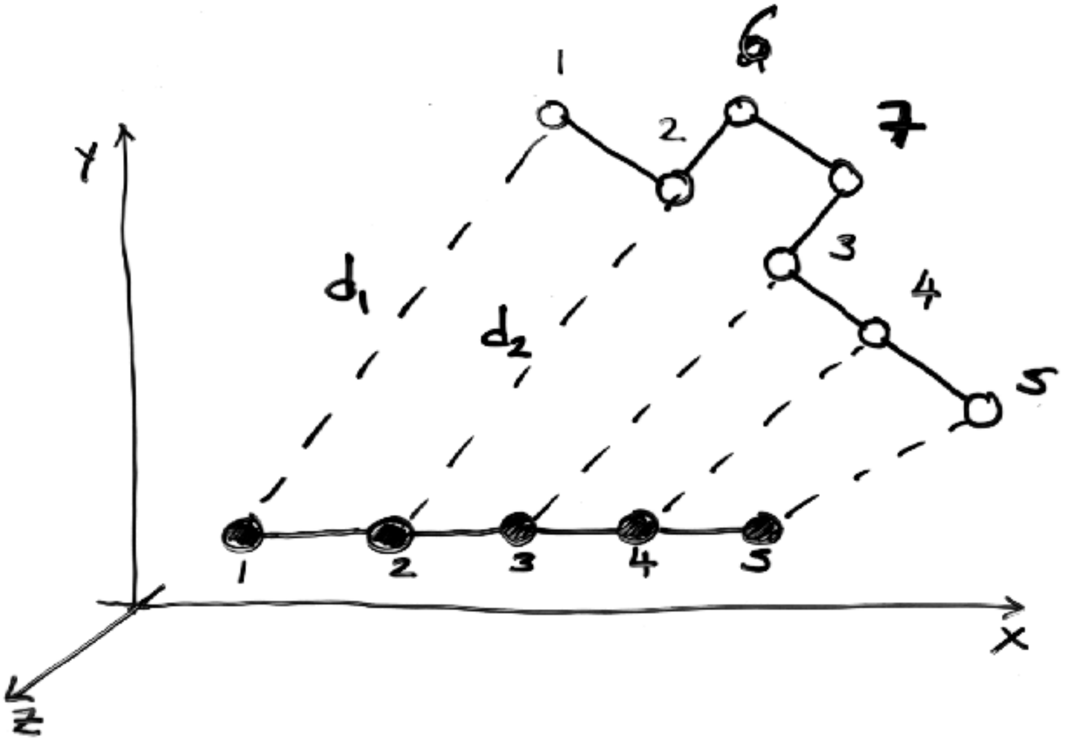
Cor

$$d_1^2 = (\vec{x}_{A1} - \vec{x}_{B1}) \cdot (\vec{x}_{A1} - \vec{x}_{B1})$$



RMS Superposition (3): RMS Distance Between Aligned Atoms in 2 Structures

$$RMS = \sqrt{\frac{\sum_{i=1}^{5} (\vec{x}_{Ai} - \vec{x}_{Bi})^2}{5}} \sim \frac{d_1 + d_2 + d_3 + d_4 + d_5}{5}$$



Cor

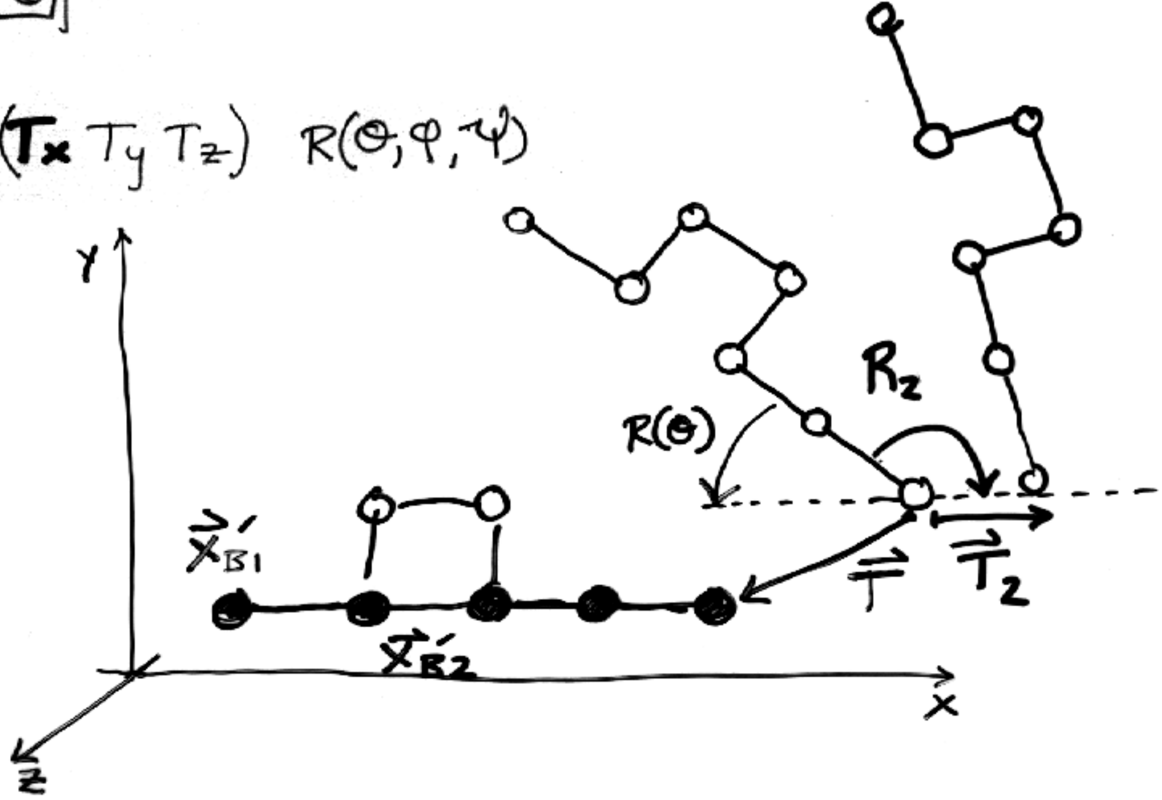
RMS Superposition (4): Rigid-Body Rotation and Translation of One Structure (B)

$$\vec{x}'_{Bi} = R(\theta)\vec{x}_{Bi} + \vec{T}$$

ROTATE & TRANSLATE

$\boxed{6}$ parameters

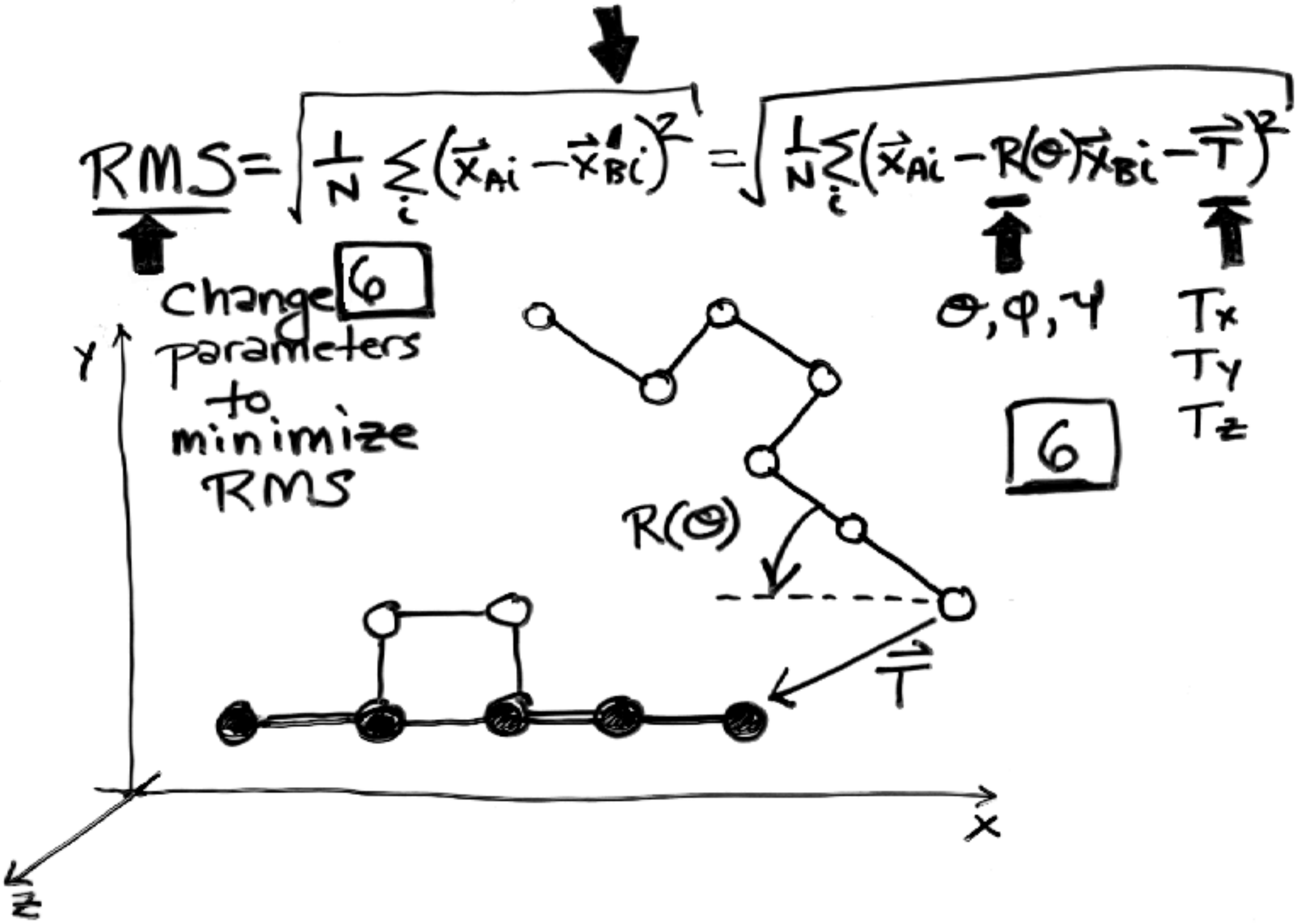
$\vec{T} = (T_x \; T_y \; T_z) \quad R(\theta, \varphi, \psi)$



Cor

RMS Superposition (5): Optimal Movement of One Structure to Minimize the RMS

Cor



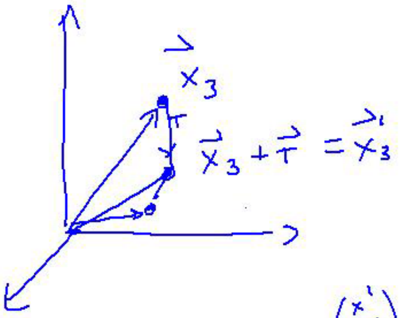
Methods of Solution:

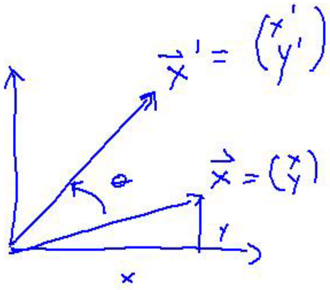
springs (F ~ kx)

SVD

Kabsch

Rotation Matrices

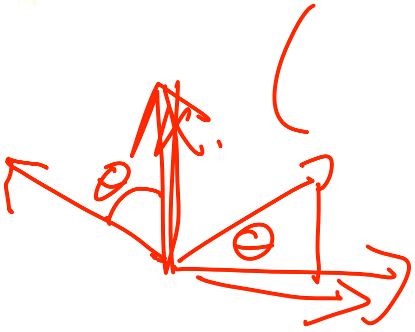


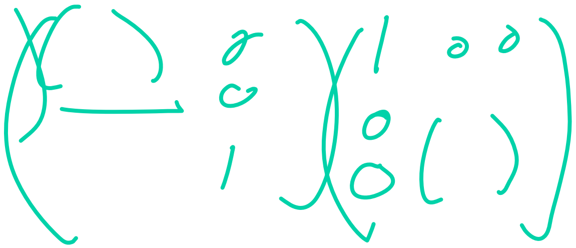


$\begin{pmatrix} x' \\ y' \end{pmatrix} = \left(\quad \right) \begin{pmatrix} x \\ y \end{pmatrix}$

ROTATION MATRIX

$$\boxed{\begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}}$$





$$\hat{z} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

2D

3D

AROUND z

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

$$\begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$\begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

$$\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \quad 0 \quad 0$$

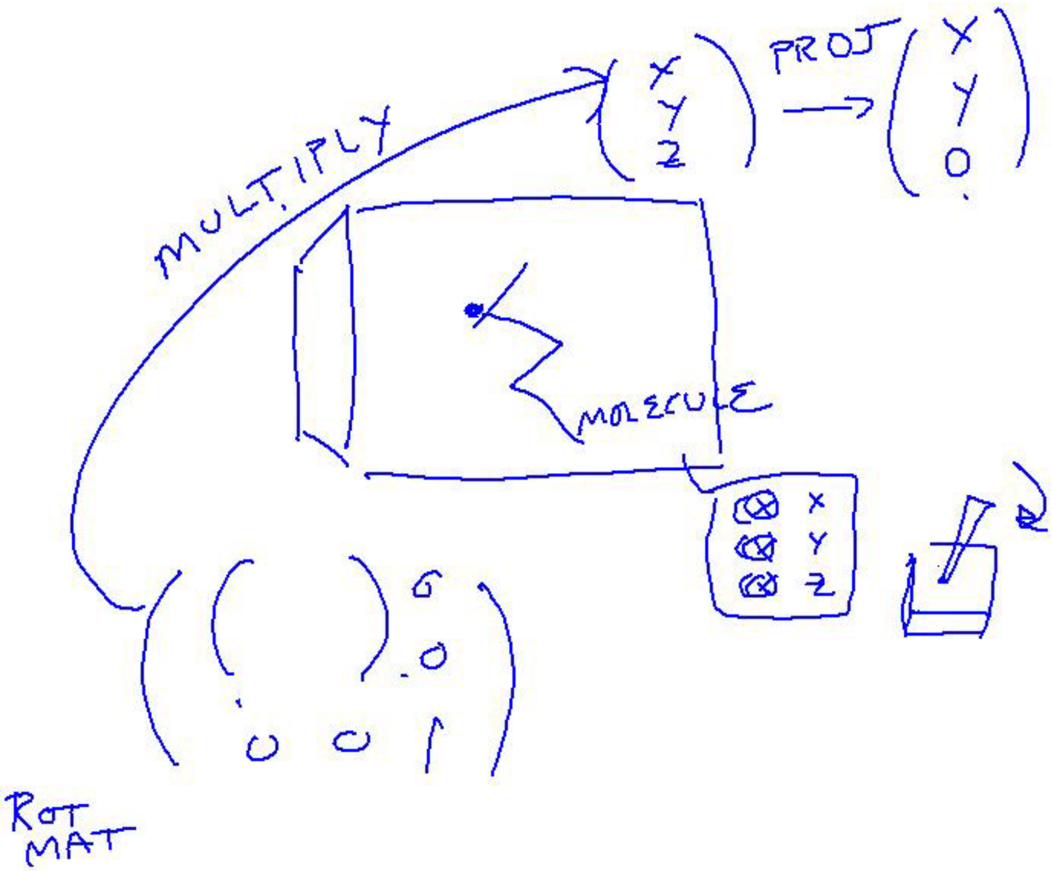
Moving Molecules Rigidly

- $\mathbf{X_i}(t+1) = (x_i(t), y_i(t), z_i(t))$ = coordinates of ith atom in the molecule at timestep t
- Rigid-body Translation of all i atoms
  - ◊ For each atom atom i do $\mathbf{x}_i(t+1) = \mathbf{x}_i(t) + \mathbf{v}$
- Rigid-body Rotation of all i atoms
  - ◊ For each atom atom i do $\mathbf{x}_i(t+1) = \mathbf{R}(\phi,\theta,\psi)\,\mathbf{x}_i(t)$
  - ◊ Effectively do a rotation around each axis (x, y, z) by angles ϕ,θ,ψ (see below)
  - ◊ Many conventions for doing this
    - o **BELOW IS ONLY FOR MOTIVATION**
    - o Consult Allen & Tildesley (1987) or Goldstein for the formulation of the rotation matrix using the usual conventions
  - ◊ How does one do a random rotation? Trickier than it seems

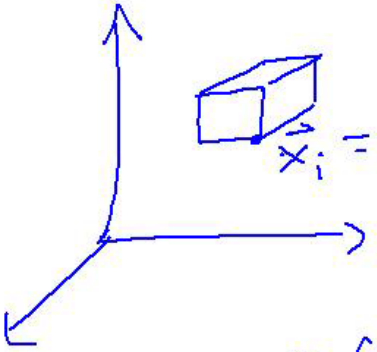
$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \underbrace{\begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix}}_{\text{Finally, rotate by } \theta \text{ around z axis}} \underbrace{\begin{pmatrix} \cos\phi & 0 & -\sin\phi \\ 0 & 1 & 0 \\ \sin\phi & 0 & \cos\phi \end{pmatrix}}_{\text{Second, rotate by } \phi \text{ around y axis}} \underbrace{\begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\psi & -\sin\psi \\ 0 & \sin\psi & \cos\psi \end{pmatrix}}_{\text{First, rotate by } \psi \text{ around x axis}} \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

Computer Graphics Systems: Rotation, Translation, and XY Projection



Worked Example of Rigid Body Movement



$$\vec{x}_i = \begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix} \overset{EX}{=} \begin{pmatrix} 3 \\ 3 \\ 2 \end{pmatrix} = \vec{x}$$

TRANSLATE (UP BY 2)

$$\vec{x} + \vec{T} \overset{EX}{=} \begin{pmatrix} 3 \\ 3 \\ 2 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 2 \end{pmatrix} = \begin{pmatrix} 3 \\ 3 \\ 4 \end{pmatrix}$$

MOST ABSTRACT

$$R\vec{x}_i = \vec{x}_i{}' = \begin{pmatrix} \frac{3\sqrt{3}}{2} - \frac{3}{2} \\ \frac{3}{2} + \frac{3\sqrt{3}}{2} \\ 2 \end{pmatrix}$$

$$\begin{pmatrix} a & b & c \\ d & e & f \\ g & h & i \end{pmatrix}$$

θ, ϕ, ψ

ROT BY 30° AROUND Z

PRACTICAL ABSTRACT

$$\begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

EX

$$\begin{pmatrix} \frac{\sqrt{3}}{2} & -\frac{1}{2} & 0 \\ \frac{1}{2} & \frac{\sqrt{3}}{2} & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 3 \\ 3 \\ 2 \end{pmatrix} =$$

← EX VECTOR

End of class M6
[2006,11.13]
Start of class M7
[2006,11.15]

Alignment (1)
Make a Similarity Matrix
(Like Dot Plot)

	A	B	C	N	Y	R	Q	C	L	C	R	P	M
A	1												
Y					1								
C			1					1		1			
Y					1								
N				1									
R						1					1		
C			1					1		1			
K													
C			1					1		1			
R						1					1		
B		1											
P												1	

Structural Alignment (1b)
Make a Similarity Matrix
(Generalized Similarity Matrix)

- PAM(A,V) = 0.5
  - ◊ Applies at every position
- S(aa @ i, aa @ J)
  - ◊ Specific Matrix for each pair of residues
    i in protein 1 and
    J in protein 2
  - ◊ Example is Y near N-term. matches any C-term. residue (Y at J=2)
- S(i,J)
  - ◊ Doesn't need to depend on a.a. identities at all!
  - ◊ Just need to make up a score for matching residue i in protein 1 with residue J in protein 2

i →

		1	2	3	4	5	6	7	8	9	10	11	12	13
		A	B	C	N	Y	R	Q	C	L	C	R	P	M
1	A	1												
2	**Y**					1			5	5	5	5	5	5
3	C			1					1		1			
4	Y					1								
5	N				1									
6	R						1					1		
7	C			1					1		1			
8	K													
9	C			1					1		1			
10	R						1					1		
11	B		1											
12	P												1	

J ↓

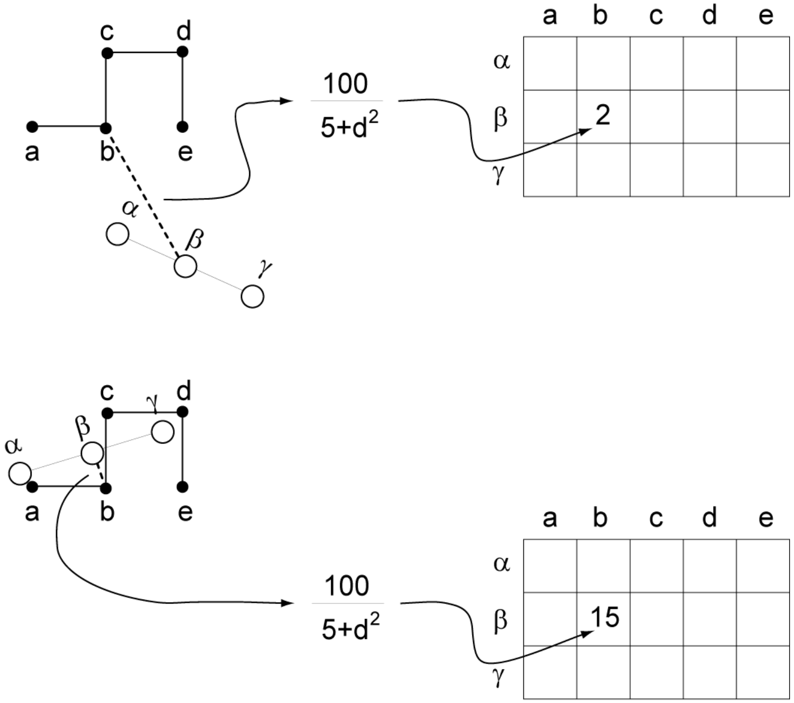
Structural Alignment (1c*) Similarity Matrix for Structural Alignment

- Structural Alignment
  - ◊ Similarity Matrix S(i,J) depends on the 3D coordinates of residues i and J
  - ◊ Distance between CA of i and J

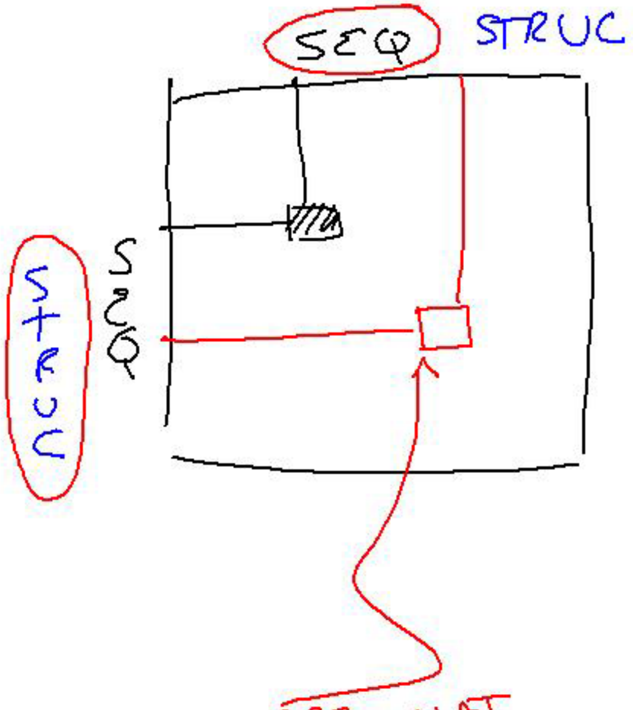
$$d = \sqrt{(x_i - x_J)^2 + (y_i - y_J)^2 + (z_i - z_J)^2}$$

  - ◊ M(i,j) = 100 / (5 + d^2)

- Threading
  - ◊ S(i,J) depends on the how well the amino acid at position i in protein 1 fits into the 3D structural environment at position J of protein 2



Seq. Alignment, Struc. Alignment, Threading



▨ = SEQ IDENTITY FOR SEQ ALIGNMENT

= STRUC COORD SIM. FOR STRUC ALINGMENT

= MATCH OF SEQ TO 3D STRUC FOR THREADING

DEGREE THAT RES i IN TOP SEQ MATCHES STRUC ENVIRON. OF J IN LEFT STRUC

Cor

Alignment (2): Dynamic Programming, Start Computing the Sum Matrix….

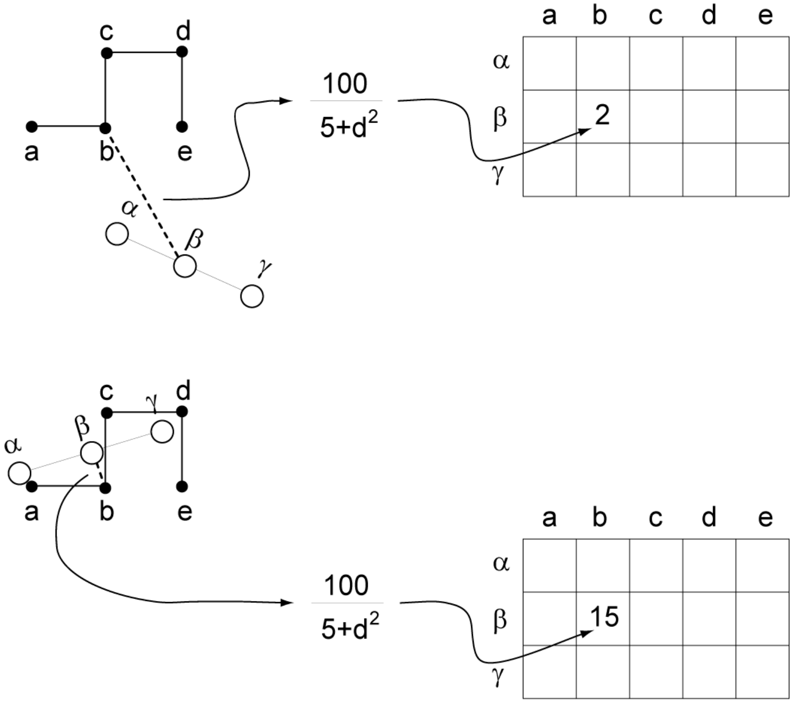
```
new_value_cell(R,C) <=
  cell(R,C)                        { Old value, either 1 or 0     }
  + Max[
        cell (R+1, C+1),           { Diagonally Down, no gaps     }
        cells(R+1, C+2 to C_max),{ Down a row, making col. gap }
        cells(R+2 to R_max, C+2) { Down a col., making row gap }
       ]
```

	A	B	C	N	Y	R	Q	C	L	C	R	P	M
A	1												
Y					1								
C			1					1		1			
Y					1								
N				1									
R						1					1		
C			1					1		1			
K													
C			1					1		1			
R						1					1		
B		1											
P												1	

	A	B	C	N	Y	R	Q	C	L	C	R	P	M
A	1												
Y					1								
C			1					1		1			
Y					1								
N				1									
R						1					**1**		
C			1					1		1			
K													
C			1					1		1			
R						1					*2*	0	0
B	1	2	1	1	1	1	1	1	1	1	1	0	0
P	0	0	0	0	0	0	0	0	0	0	0	1	0

In Structural Alignment, Not Yet Done (Step 6*)

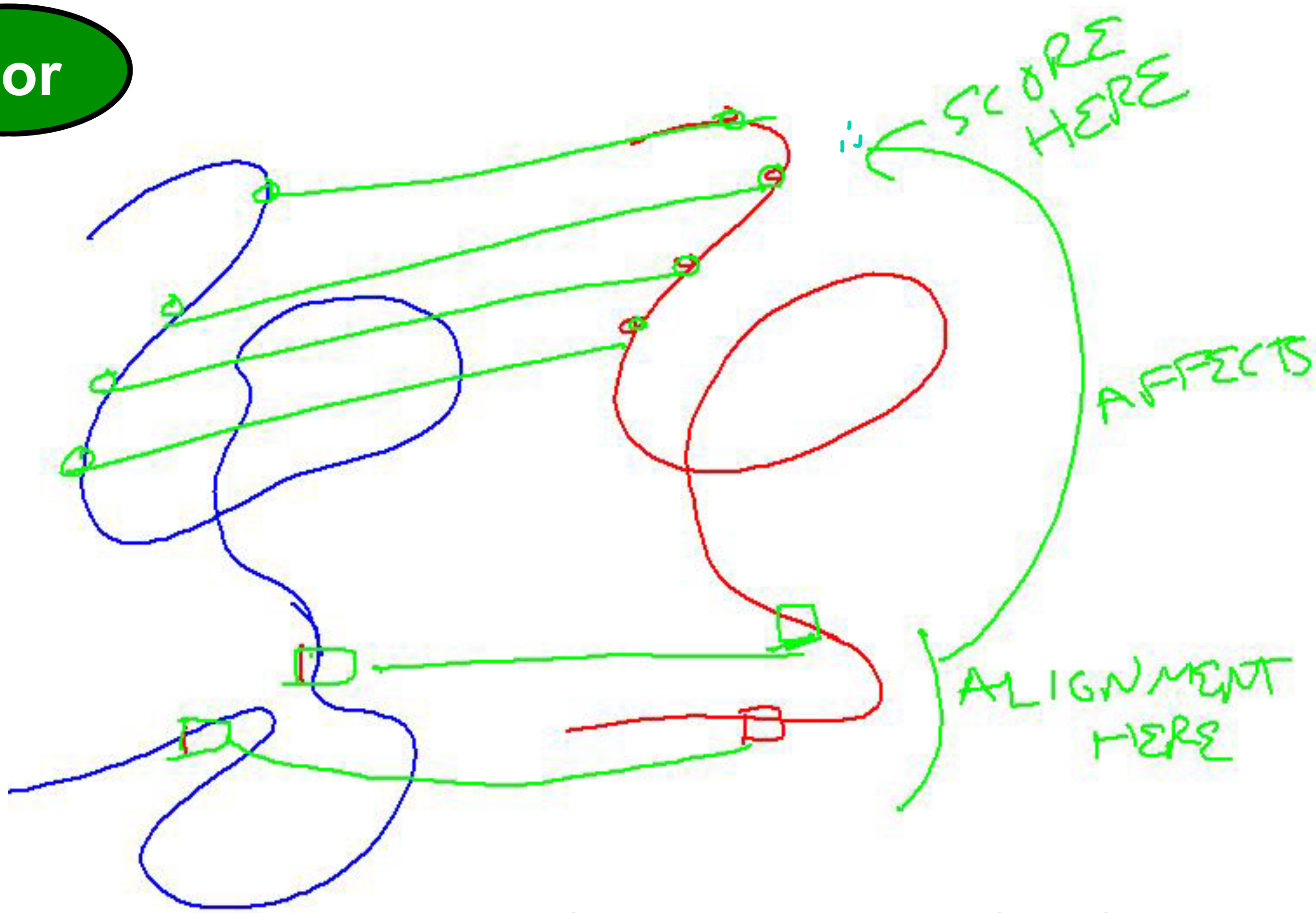
- Use Alignment to LSQ Fit Structure B onto Structure A
    - ◊ However, movement of B will now change the Similarity Matrix
- This Violates Fundamental Premise of Dynamic Programming
    - ◊ Way Residue at i is aligned can now affect previously optimal alignment of residues (from 1 to i-1)



ACSQRP--LRV-SH -R S**E**NCV
A-SNKPQLVKLMTH VK **D**FCV-

How central idea of dynamic programming is violated in structural alignment

Cor



Same theme as the Sec. Struc. alignment

Structural Alignment (7*), Iterate Until Convergence

1 Compute Sim. Matrix

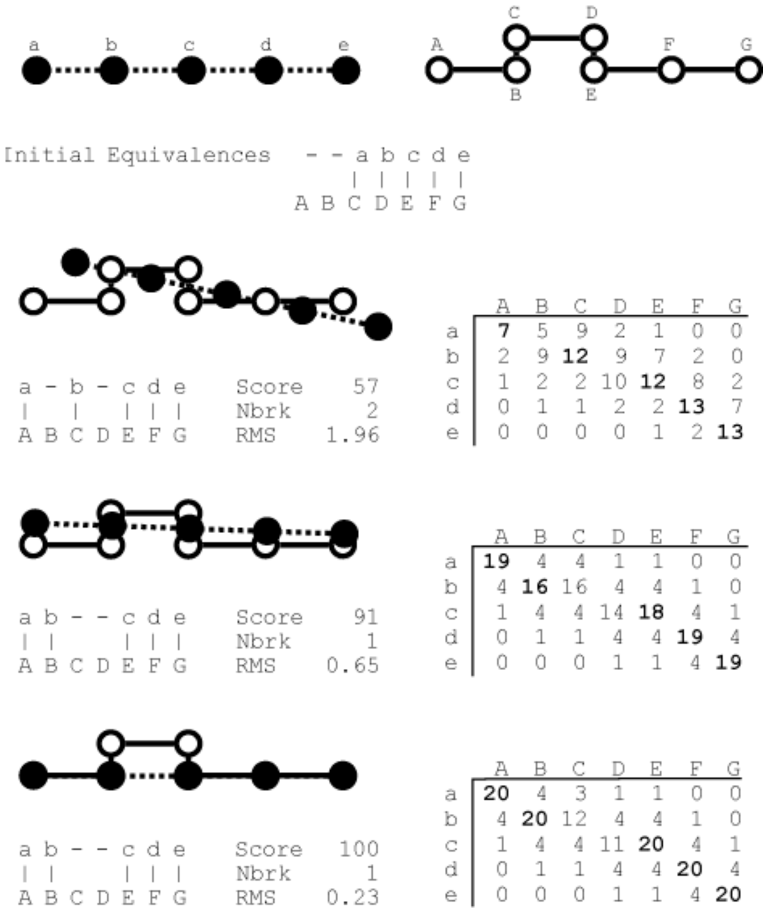
2 Align via Dyn. Prog.

3 RMS Fit Based on Alignment

4 Move Structure B

5 Re-compute Sim. Matrix

6 If changed from #1, GOTO #2



```
Initial Equivalences    - - a b c d e
                            | | | | |
                        A B C D E F G
```

```
a - b - c d e       Score     57
|   |   | | |       Nbrk       2
A B C D E F G       RMS     1.96
```

	A	B	C	D	E	F	G
a	**7**	5	9	2	1	0	0
b	2	9	**12**	9	7	2	0
c	1	2	2	10	**12**	8	2
d	0	1	1	2	2	**13**	7
e	0	0	0	0	1	2	**13**

```
a b - - c d e       Score     91
| |     | | |       Nbrk       1
A B C D E F G       RMS     0.65
```

	A	B	C	D	E	F	G
a	**19**	4	4	1	1	0	0
b	4	**16**	16	4	4	1	0
c	1	4	4	14	**18**	4	1
d	0	1	1	4	4	**19**	4
e	0	0	0	1	1	4	**19**

```
a b - - c d e       Score    100
| |     | | |       Nbrk       1
A B C D E F G       RMS     0.23
```

	A	B	C	D	E	F	G
a	**20**	4	3	1	1	0	0
b	4	**20**	12	4	4	1	0
c	1	4	4	11	**20**	4	1
d	0	1	1	4	4	**20**	4
e	0	0	0	1	1	4	**20**

Structure alignment - Scoring

- What Structures Look Like?
- Structural Alignment by Iterated Dynamic Programming
  - ◊ RMS Superposition
- Scoring Structural Similarity
- Other Aspects of Structural Alignment
  - ◊ Distance Matrix based methods
  - ◊ Fold Library
- Relation of Sequence Similarity to Structural and Functional Similarity

- Protein Geometry
- Surface I (Calculation)
- Calculation of Volume
- Voronoi Volumes & Packing
- Standard Volumes & Radii
- Surfaces II (Relationship to Volumes)
- Other Applications of Volumes -- Motions, Docking

Score S at End Just Like SW Score, but also have final RMS

S = Total Score

S(i,j) = similarity matrix score for aligning i and j

Sum is carried out over all aligned i and j

n = number of gaps (assuming no gap ext. penalty)

G = gap penalty

$$S = \sum_{i,j} S(i,j) - nG$$

Use EVD statistics just like sequence alignment

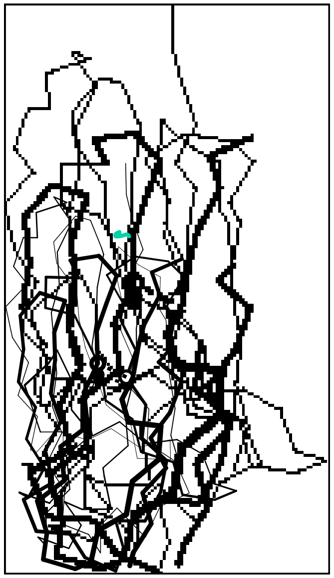
Some Similarities are Readily Apparent others are more Subtle

Easy:
Globins

125 res.,
~1.5 Å

Tricky:
Ig C & V

85 res.,
~3 Å



Very Subtle: G3P-dehydrogenase, C-term. Domain
>5 Å

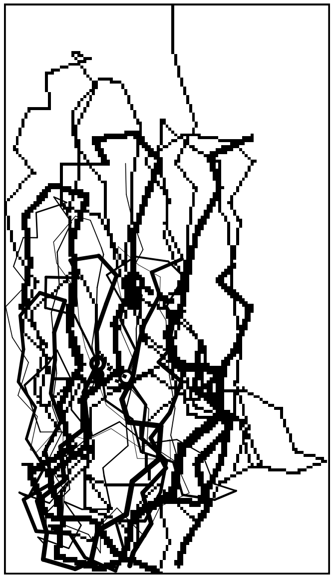
Some Similarities are Readily Apparent others are more Subtle

Easy: Globins

125 res., ~1.5 Å

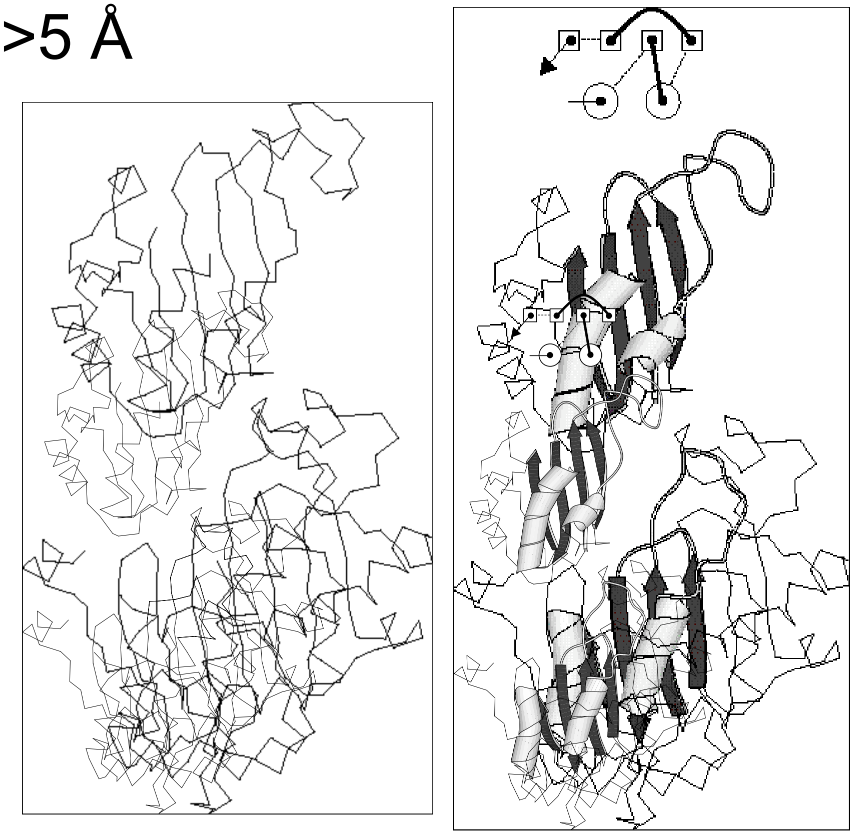
Tricky: Ig C & V

85 res., ~3 Å



Very Subtle: G3P-dehydrogenase, C-term. Domain

>5 Å



Structure alignment - Other methods

- What Structures Look Like?
- Structural Alignment by Iterated Dynamic Programming
  - ◊ RMS Superposition
- Scoring Structural Similarity
- Other Aspects of Structural Alignment
  - ◊ Distance Matrix based methods
  - ◊ Fold Library
- Relation of Sequence Similarity to Structural and Functional Similarity

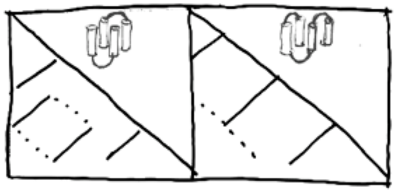
- Protein Geometry
- Surface I (Calculation)
- Calculation of Volume
- Voronoi Volumes & Packing
- Standard Volumes & Radii
- Surfaces II (Relationship to Volumes)
- Other Applications of Volumes -- Motions, Docking

Other Methods of Structural Alignment

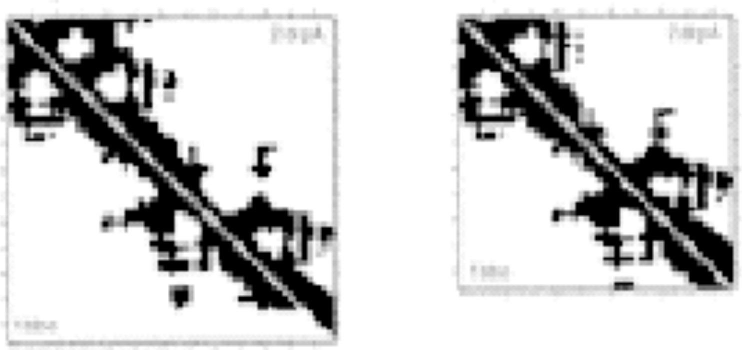
- RMS fitting used universally, but other alignment methods
- Comparison of Distance Matrices
  - ◊ Holm & Sander, DALI
  - ◊ Taylor & Orengo

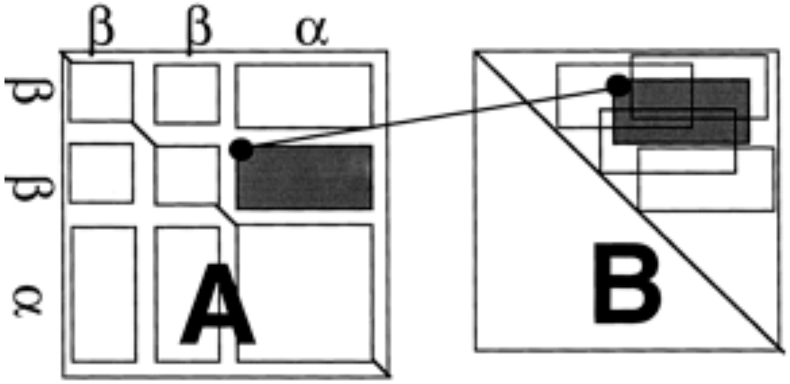
Other Methods

Rossmann
Taylor
Sander x3 } dist. mat.
Barton
Blundell } dist. mat., prop match
Cohen – soap bubble
Artymiuk
Bryant } similar subgraph



Structure Hashing
  Bryant, VAST
  Rice, Artymiuk
Others
  Cohen (Soap)
  Sippl
  Godzik (Lattice)





Multiple Structure Alignment

- Building up a Fold Library (structure clusters)

HMMs