

# Comparison and Classification of Protein Structures

Akira R. KINJO  
金城 玲

Institute for Protein Research, Osaka University  
&  
Protein Data Bank Japan

# Protein Data Bank (PDB)

## Worldwide Protein Data Bank (wwPDB)

Welcome to the Worldwide Protein Data Bank (wwPDB)

Access the PDB FTP:

- RCSB PDB
- PDBe
- PDB
- Archive Download
- Chemical Component Dictionary
- Biologically Interesting Molecule Reference Dictionary (BIRD)
- New Deposition and Annotation System
- Tutorial
- System Information
- Validation Reports
- Reports
- Server Information
- Deposit Data to the PDB:

  - RCSB PDB
  - PDBe
  - PDB
  - PDBj
  - BMRB

- Search for Structures:

  - RCSB PDB
  - PDBe
  - PDBj
  - BMRB

- PDB Archive Snapshots:

  - RCSB PDB
  - PDB

- Instructions to Journals
- Documentation: Format, Annotation and Policies, Remediation
- Workshops and Task Forces

  - X-ray Validation
  - NMR Validation
  - EM Validation
  - SAS Task Force
  - PDBx/mmCIF Working Group

- Past Symposia
- Intl Year of Crystallography
- wwPDBAC
- EMDB

13-May-2014  
PDB Reaches a New Milestone: 100,000+ Entries

In the weeks leading up to this historic event, wwPDB has looked back at other PDB milestones. (Previously: *Building a Community Resource, The Early Structures, Launching Tools for the Next Generation*)

I DEPOSITED  
at the 100,000+  
PDB Structures

Depositors: Download this image and write the number of structures deposited.

With this week's update, the PDB archive contains a record 100,147 entries.

Established in 1971, this central, public archive has reached this critical milestone thanks to the efforts of structural biologists throughout the world who contribute their experimentally-determined protein and nucleic acid structure data.

Four wwPDB data centers support online access to three-dimensional structures of biological macromolecules to help researchers understand many facets of biology, including function and ecology, from small synthetics to health and disease to biological energy. The archive is quite large, containing more than 249 GBbytes of storage.

more

FULL NEWS

Questions? info@wwpdb.org

1. H.M. Berman, K. Henrick, H. Nakamura (2003): Announcing the worldwide Protein Data Bank. *Nature Structural Biology* 10 (12), p. 980

PDB PROTEIN DATA BANK

PDBe PROTEIN DATA BANK IN EUROPE

PDBj PROTEIN DATA BANK JAPAN

© wwPDB

## Protein Data Bank Japan (PDBj)

日本蛋白質構造データバンク (PDBj)

100693

利用できます。[2014-06-04 00:00 UTC / 09:00 JST]

PDBj

Protein Data Bank Japan

English 日本語 简体中文 繁體中文 한국어

Search pdbj.org

ホーム

トピックページ

統計情報

ヘルプ

FAQ

お問い合わせ

リンク集

PDBアーカイブ

データ登録

ヘルプ

ADIT: PDBへの登録

ADIT-NMR

データ登録について

新フォーマット

PDBx/mmCIFについて

検索

ヘルプ

PDB検索 (PDBj Mine)

PDBx検索

巨大構造エントリー

BMRB検索

Sequence-Navigator

Structure-Navigator

EM Navigator

wwPDB/RDF

SeSaw

Ligand Binding Sites (GIRAF)

最新の公開エントリー

未公開エントリーのステータス

サービスを探す

サービスを表示する

例) モチーフ、分子

リセット

最新情報

ニュース (2014年6月4日)

今年で120回を迎える日本バイオインフォマティクストレーニングコースが、6月17日(火)-20日(金)、韓国済州島にて、開催されます。PDBjから企画地が招待演者として参加し、講義を行います。

ニュース (2014年5月14日)

PDBjエントリー数が10万件を突破しました

ニュース (2014年5月1日)

次世代ペイロードの準備開始と10万件への道のり

ニュース (2014年5月1日)

PDB 初期の登録構造と10万件への道のり

ニュース (2014年4月28日)

2014年5月3日 (土・祝) に、大阪大学いちょう祭にて施設・研究紹介を行います。タンパク質に興味のある方などなたでも歓迎です。

ニュース (2014年4月28日)

5月17日 (土) - 20日 (火) 、韓国済州島で開催される第4回 Asia Pacific Protein Association (APPA) Conference にて、講演およびポスター発表を行います。

ニュース (2014年4月24日)

PDBの構造と10万件への道のり

ニュース (2014年4月17日)

PDBjで提供した画像がオンライン事典に掲載されました。

ニュース (2014年4月17日)

PDBjで提供した画像がオンライン事典に掲載されました。

PDBj

PDB

RCSB PDB

BMRB

PDBe

Legacy

4123 最新 公開 リスト

今月の分子

174 GFP-たんぱく質 (GFP-like Proteins)

今月の分子のリスト

WORLD WIDE PROTEIN DATA BANK

wwPDB

PDB Reaches a New Milestone: 100,000+ Entries

The Road to 100,000 Entries: Launching Tools for the Next Generation

The Road to 100,000 Entries: Building a Community Resource

wwwPDB X-ray Validation reports added to PDB archive

BMRB検索

PDBj-BMRB

Accession number

Deposition code

Search

パートナー

DBCLS Database Center for Life Science

NBDC National Bioscience Database Center

The primary database of biological macromolecular structures

# An example of PDB entries

1goF - 案例 - PDBj Mine PC X

pdbj.org/mine/summary/1goF

**100693**  
件が利用できます (2014-06-04  
00:00 UTC / 09:00 JST)

**PDBj**  
Protein Data Bank Japan

English 日本語 简体中文 繁體中文 한국어

Search pdbj.org

wwwPDB RCSB PDB BMRB PDBe Legacy

ホーム  
統計情報  
ヘルプ  
FAQ  
お問い合わせ  
リンク集  
PDBアーカイブ

概要 構造情報 実験情報 機能情報 相同蛋白質 ダウンロード

**1GOF**

NOVEL THIOETHER BOND REVEALED BY A 1.7 ANGSTROMS CRYSTAL STRUCTURE OF GALACTOSE OXIDASE

**1GOF の概要**

分子名稱 GALACTOSE OXIDASE (E.C.1.1.3.9) (PH 4.5)  
機能のキーワード OXIDOREDUCTASE(OXYGEN(A))  
由来する生産種 Hypomyces rostellus  
ポリマー鎖の合計数 1  
分子量の合計 68785.89  
著者 Ito, N., Phillips, S.E.V., Knowles, P.F. (登録日: 1993-09-30, 公開日: 1994-01-31, 最終更新日: 2011-07-13)  
引用文献 Ito, N., Phillips, S.E., Stevens, C., Ogel, Z.B., McPherson, M.J., Keen, J.N., Yadav, K.D., Knowles, P.F.  
**Novel thioether bond revealed by a 1.7 Å crystal structure of galactose oxidase.**  
Nature, 350:87-90, 1991  
Published online 28 August 1991  
DOI: 10.1038/350087a0  
Import into Mendeley

実験手法 X-RAY DIFFRACTION (1.7 Å)

他の静止画像 (非対称単位)

Copyright © 2013-2014 日本蛋白質構造データバンク

サービス&ソフトウェア  
ヘルプ JV: 3次元構造ビュア 万見 (Yorodumi)

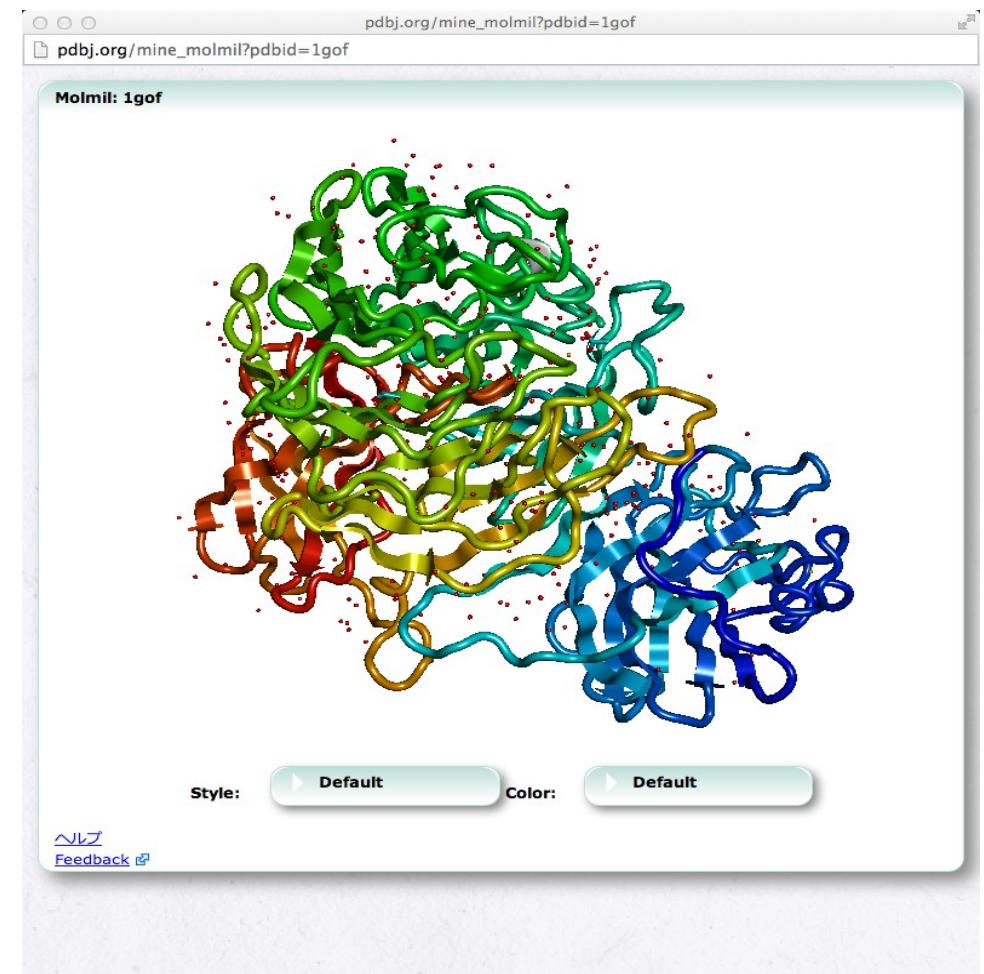
統計情報  
FAQ  
お問い合わせ  
リンク集  
PDBアーカイブ

データ登録  
ヘルプ  
ADIT: PDBへの登録 ADIT-NMR  
データ登録について

新フォーマット  
PDBx/mmCIFについて

検索  
ヘルプ  
PDB検索 (PDBj Mine)  
PDB詳細検索  
巨大構造エントリー  
BMRB検索  
Sequence-Navigator  
Structure-Navigator  
EM Navigator  
wwwPDB/RDF  
SeSAW  
Ligand Binding Sites (GIRAF)  
最新の公開エントリー  
未公開エントリーのステータス

サービス&ソフトウェア  
ヘルプ JV: 3次元構造ビュア 万見 (Yorodumi)



# Protein Structure Comparison

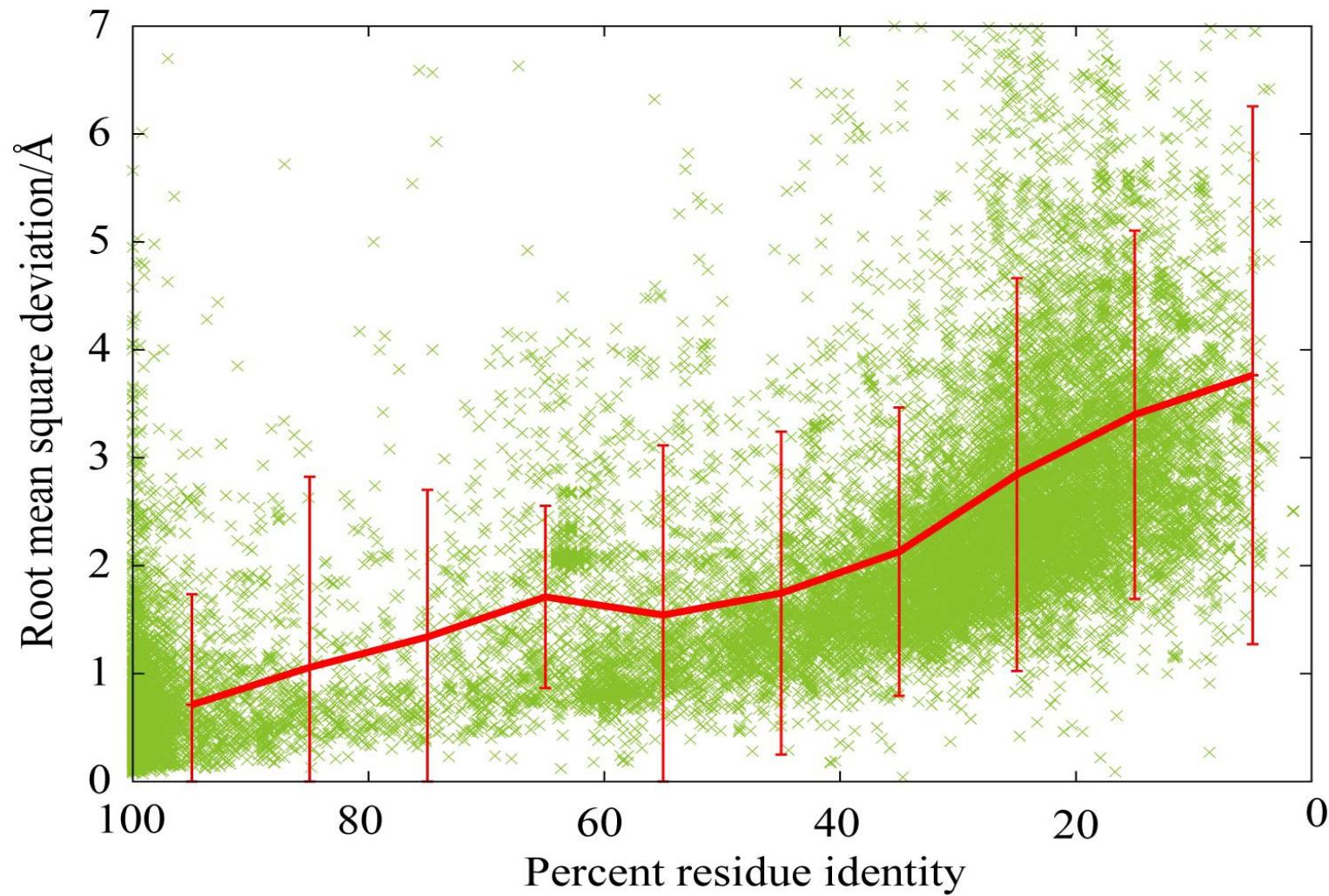
```

[BEGIN ALIGNMENT]
      : E1          H1          E2          H2          E3
SecA : EEEEEEE S HHHHHHHHHHHHHHS SSEEEEEE GGGHHHHHHH TTEEEE
      3 :RTFFFVGGNFKLNGSKQSIKEIWERLNTASIPENVEVVICCPATYLDYSVSLVKKPQVTVG: 62
      * ***** * * * * * * * * * * * * * * * * * * * * * *
      4 :RKFFVGGNWKMNGDKKSLGELIHTLNGAKLSADTEVVCAGPSIYLDFAQRQL-DAKIGVA: 62
SecB : EEEEEEE S HHHHHHHHHHHHHHS TTEEEEEEEE GGGHHHHHHHS- TTSSEE
      : E1          H1          E2          H2          -          E3
      :           H3          E4          H4          H5          E
SecA : ES SSSSSS TT HHHHHHHTT EEEES HHHHHHHS HHHHHHHHHHHHHHHHHHTT E
      63 :AQNAYLKASGAFTGENSDQIKDVGAKWVILGHSERRSYFHEDDKFIADKTFALGQGVG: 122
      *** * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      63 :AQNCYKVPKGAFTGEISPAMIKDIGAAWVILGNPERRHVFGESDELIGQKVAHALAEGLG: 122
SecB : ES SSSSSS TT HHHHHHHTT EEEES HHHHHHTS HHHHHHHHHHHHHHHHTT E
      :           H3          E4          H4          H5          E
      : 5          H6          H7          E6          H8
SecA : EEEEE HHHHHHTT HHHHHHHHHHHHHHHHH S TT EEEE GGGTTTS HHHHH
      123 :VILCIGETLEEKKAGKTLDVVERQLNAVLEEVKDWTNVVAYEPVWAIGTGLAATPEDAQ: 182
      *** * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      123 :VIACIGEKLDEREAGITEKVVFQTKAIADNVKDWSKVVLAYEPVWAIGTGKTATPQQAQ: 182
SecB : EEEEE HHHHHHTT HHHHHHHHHHHHHHHHTT S GGGEEEE GGGTTTS HHHHH
      : 5          H6          H7          E6          H8
      :           H9          E7          E8          H10
SecA : HHHHHHHHHHHHHHH HHHHHH EEEESS TTTGGGGTT TT EEEE SGGGGSTTHHH
      183 :DIHASIRKFASKLGDKAASELRILYGGSANGSNAVTFKDKADVDGFLVGGASLKPETFVD: 242
      * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      183 :EVHEKLRGWLKTHVSDAVAQSTRIIYGGSVTGGNCHELASQHDVDGFLVGGASLKPETFVD: 242
SecB : HHHHHHHHHHHHHHT HHHHHH EEEESS TTTHHHHHTSTT EEEE SGGGGSTTHHH
      :           H9          E7          H10          E8          H11
      :
SecA : HHHHTT
      243 :IINSRN: 248
      ***
      243 :IINAKH: 248
SecB : HHHTT
      :

```



# Sequence & structure similarities

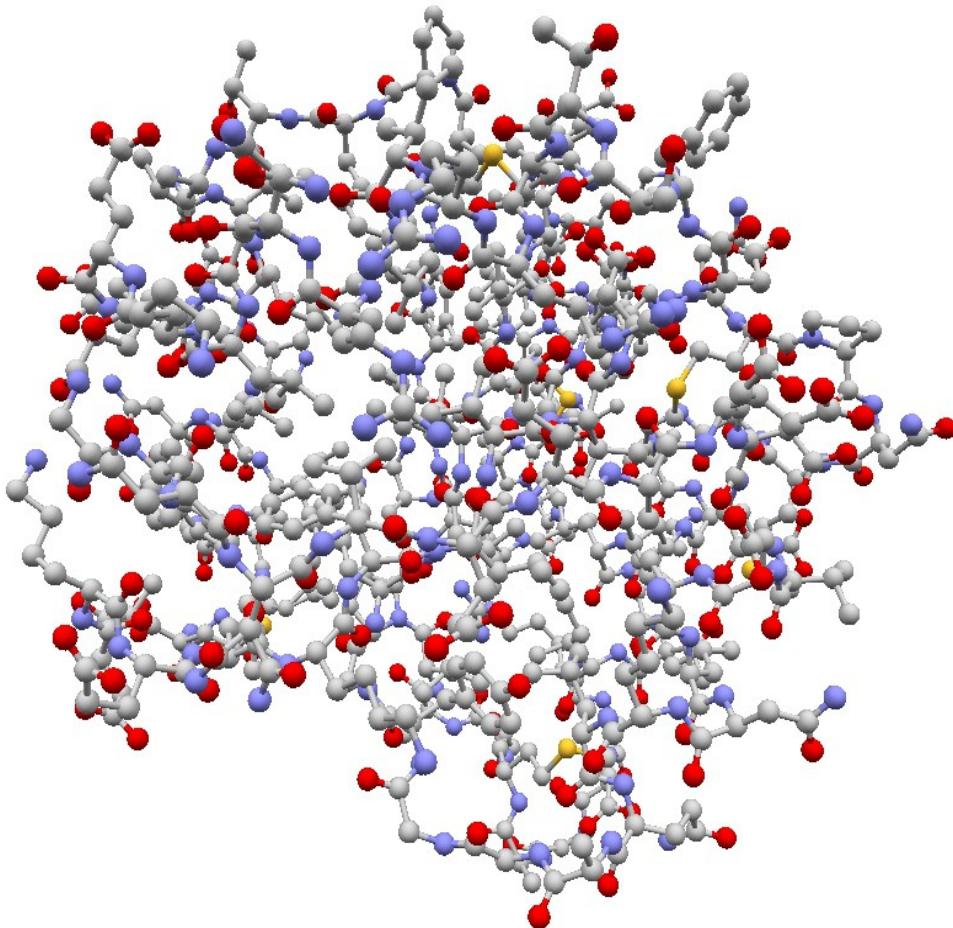


(『タンパク質の立体構造入門』図 4.1 )

# Methods of structure comparison

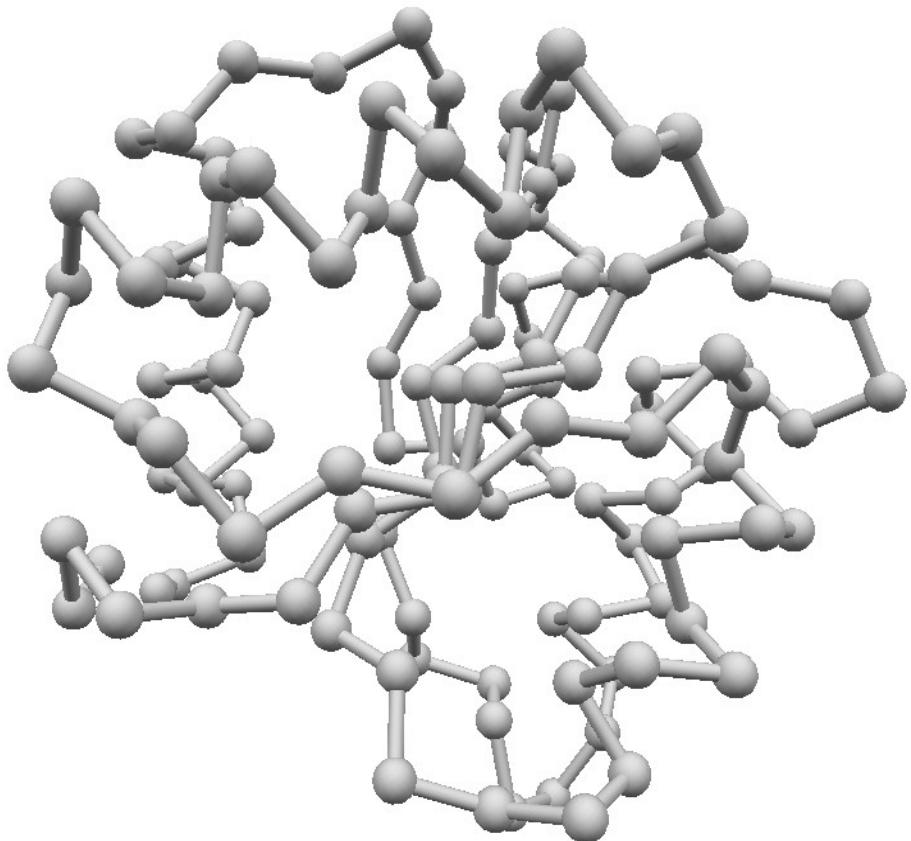
- Visual inspection(!)
  - The “best” method if you are well-trained.
- Algorithms
  - It's an NP-hard problem, so there are a number of approximate methods based on various representations:
    - secondary structure elements (SSE)
    - Amino acid residues
    - Atoms
    - Molecular surface

# Representation: all atoms



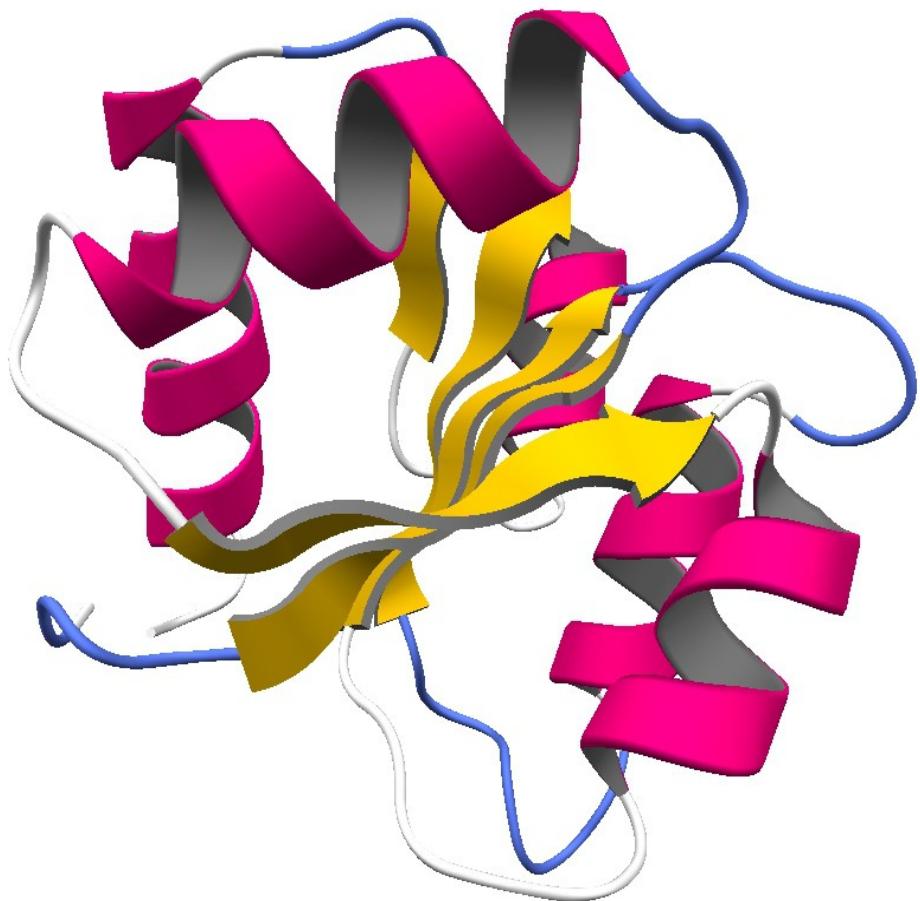
- Dealing with all atoms...
- is difficult. So usually only substructures are treated in this representation.

# Representation: Backbone



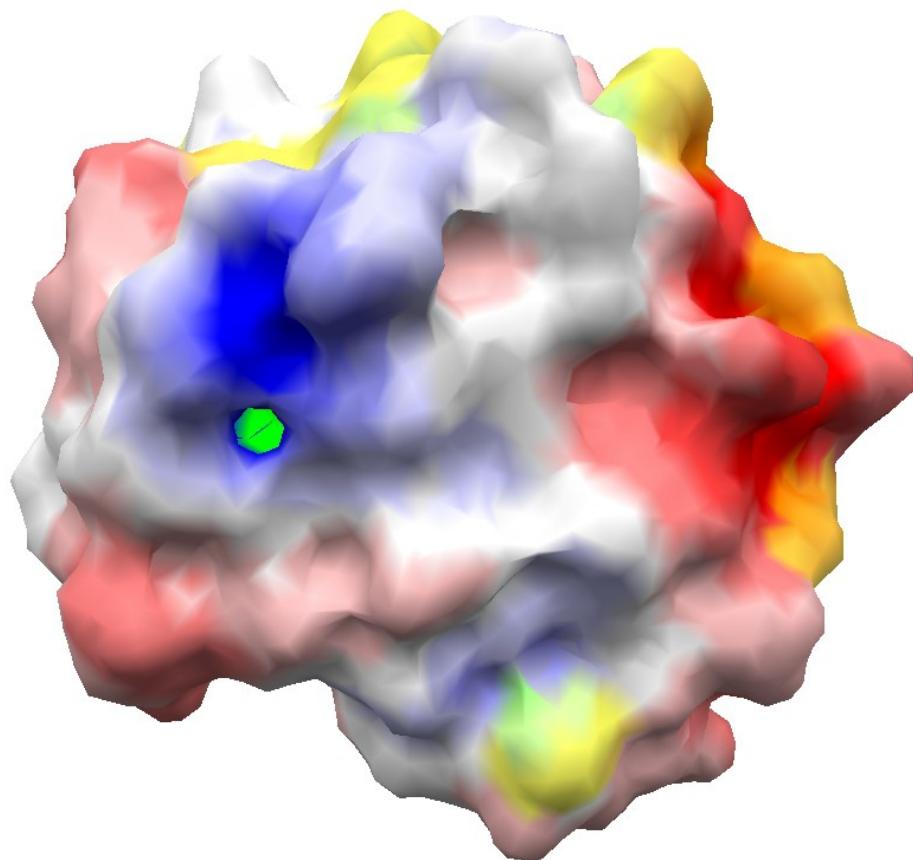
- Using only  $C\alpha$  or  $C\beta$  atoms to reduce computational costs.
- Also compatible with sequence alignment (1 atom / residue)
- Still computationally demanding.

# Representation: 2ndary structures



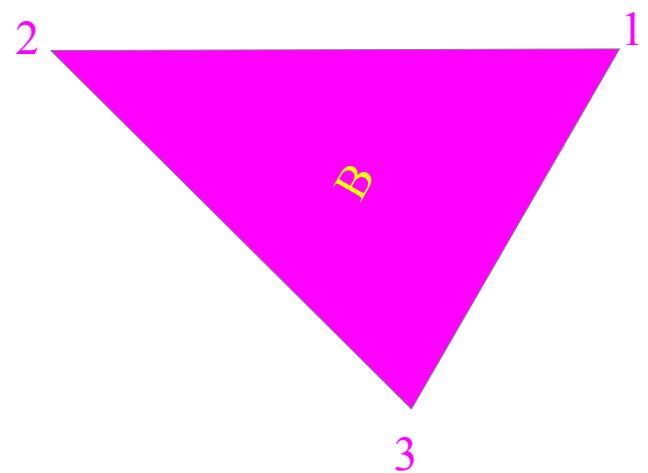
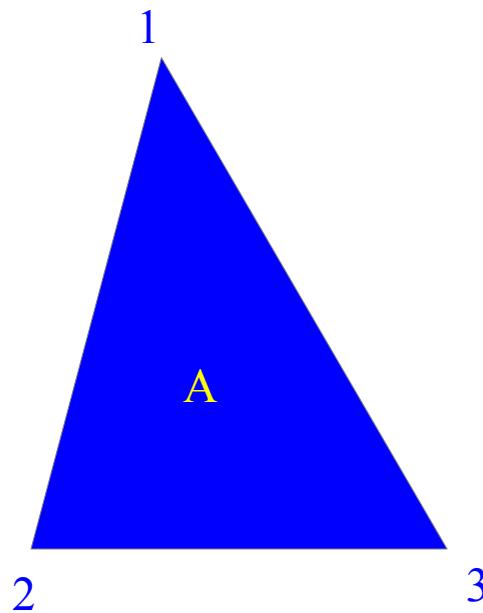
- $\alpha$  helices and  $\beta$  strands as vectors.
- Suitable for finding topological similarities.
- Less cost.

# Representation: Molecular surface



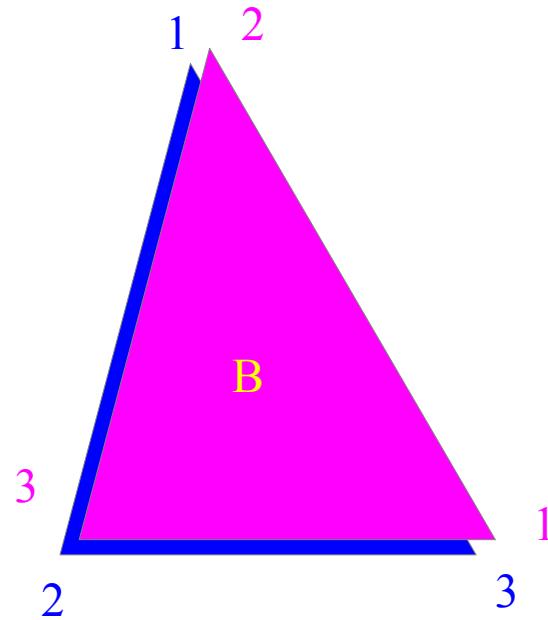
- Protein structure from the view point of a water molecule(?)
- Often used for mapping electrostatic potentials & hydropathy on the structure.

# Basic ideas



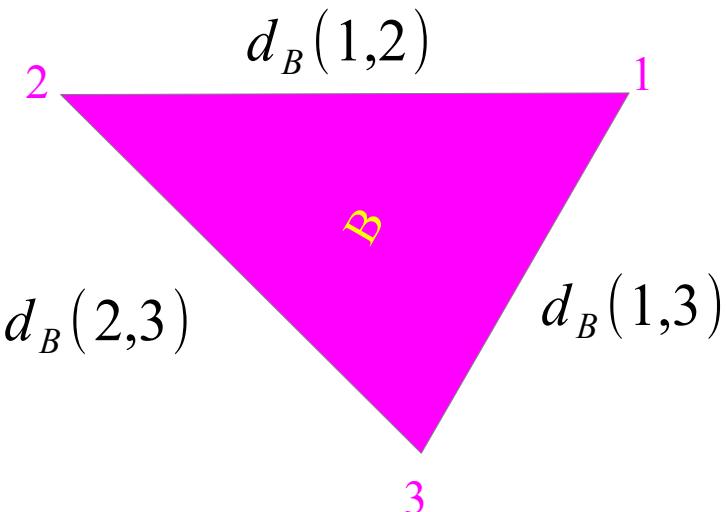
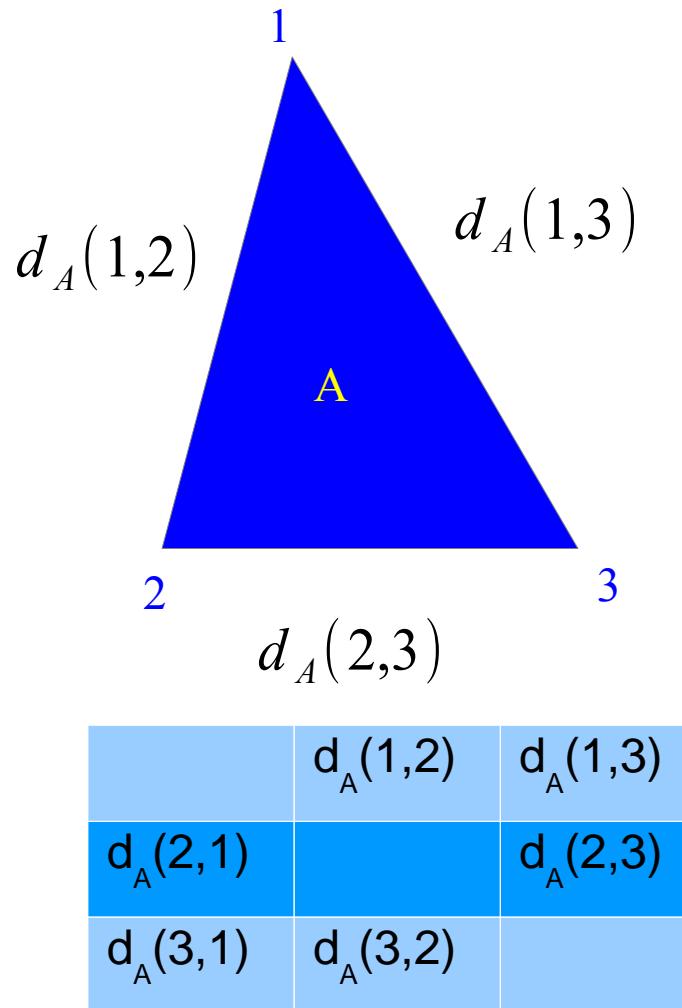
How do you tell the congruence of two triangles?  
(Vertex numbers do not match!)

# Method 1: Coordinate-based



- Actually try to superimpose them!
- Infinite combinations of “translation” & “rotation”.

# Method 2: Distance-based

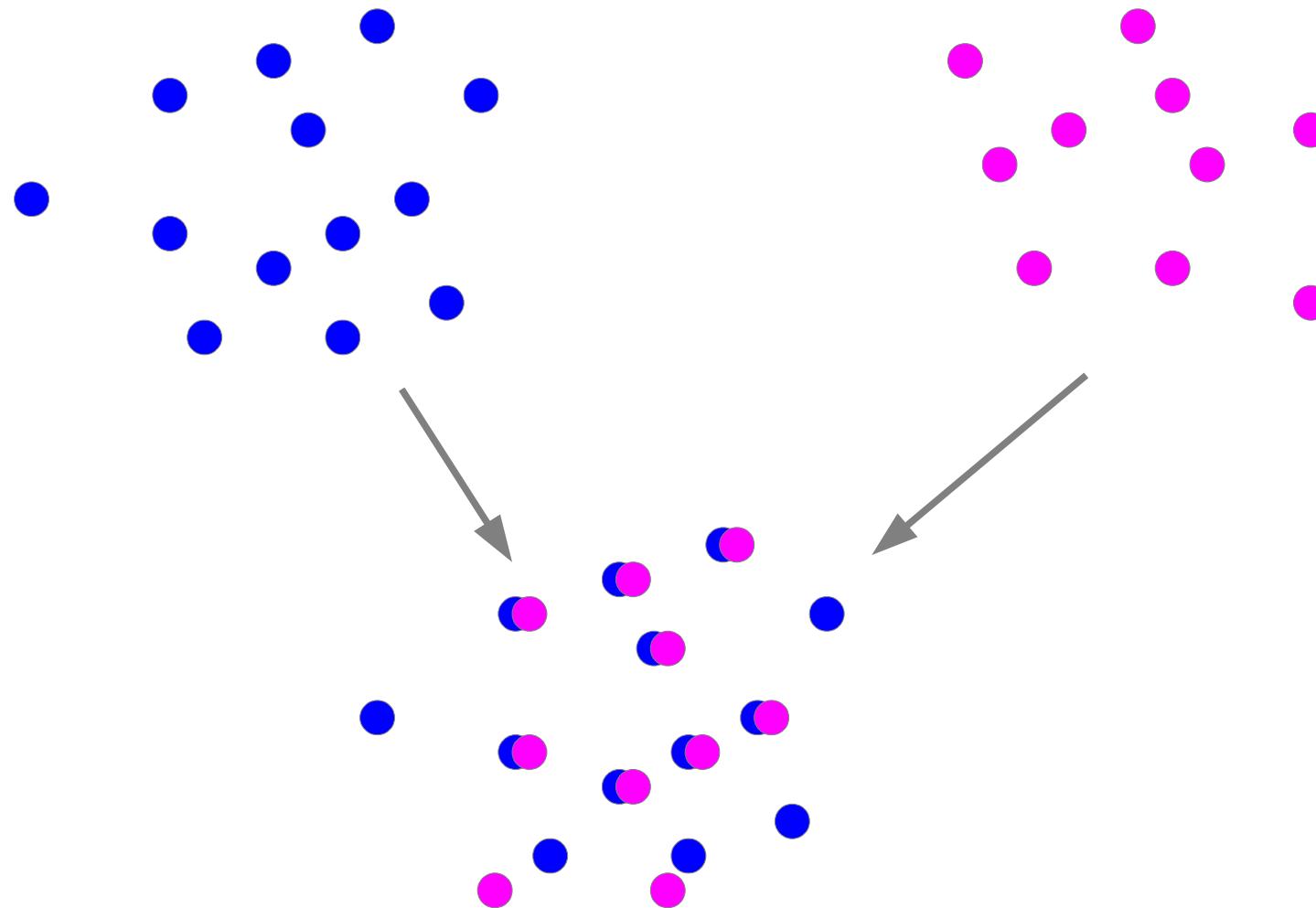


	$d_B(1,2)$	$d_B(1,3)$
$d_B(2,1)$		$d_B(2,3)$
$d_B(3,1)$	$d_B(3,2)$	

Find pairs  $(i,j), (k,l)$  that satisfy  $|d_A(i, j) - d_B(k, l)| = 0$

How many possibilities are there?

# A little more complicated objects



# Summary of comparison methods

- Translation & rotation
  - “Coordinate-based method”
  - Infinite possibilities.
- Comparing the distances between vertices
  - “Distance-based method”
  - Exponentially increasing possibilities.
- In any case, it's a tough problem!

# Coordinate-based method, theory

Let  $A$ ,  $B$  and  $C$  be metric spaces:

$$A = \left( x_1^A, x_2^A, \dots, x_M^A \right) \quad B = \left( x_1^B, x_2^B, \dots, x_N^B \right)$$

$$\begin{aligned} A &\xrightarrow{f} C \\ B &\xrightarrow{g} C \end{aligned}$$

The points in  $A$  and  $B$  are transformed into  $C$ , so the distance between two points, one in  $A$  and the other in  $B$  can be measured in  $C$ .

$$D(A, B) = \sum_{(i, j)} d_C(f(x_i^A), g(x_j^B))$$

The Problem: Find the set of combinations of  $(i, j)$  that minimizes this distance.  
But how do we define  $f$  and  $g$ ?

# Best-fitting problem

Easy case first. Assume the alignment is already known.

$$A = \begin{pmatrix} x_1^A, x_2^A, \dots, x_M^A \end{pmatrix} \quad B = \begin{pmatrix} x_1^B, x_2^B, \dots, x_M^B \end{pmatrix} \quad (\text{the same number of points})$$

$$\text{For all } i=1, \dots, M, (x_i^A, x_i^B) \quad (\text{The } i\text{-th atom in A} \Leftrightarrow \text{The } i\text{-th atom in B})$$

$$\sum_{i=1}^M x_i^A = 0, \sum_{i=1}^M x_i^B = 0 \quad (\text{Both centers of mass are at the origin})$$

$$D(A, B) = \sqrt{\frac{1}{M} \sum_{i=1}^M |x_i^A - R x_i^B|^2} \quad (\text{Rotate B by the rotation matrix R})$$

Now the problem is finding the matrix R (least-square fitting).

This can be solved analytically (Euler angles, singular value decomposition, quaternions)

# Coordinate-based method in practice

- Impossible to try infinite number of transformations
- 3 linearly independent points define a frame.
  - $N!/(N-3)! = N(N-1)(N-2)$
- Consider all combination from two structures
  - $M(M-1)(M-2) \times N(N-1)(N-2)$
  - $M=N=100 \Rightarrow 941,288,040,000$  combinations
- It's finite, but huge!

# Coordinate frame based on 3 points

$$A = \begin{pmatrix} x_1^A & x_2^A & \dots & x_M^A \end{pmatrix}$$

$$\begin{pmatrix} x_i^A & x_j^A & x_k^A \end{pmatrix} \quad \text{3 points}$$

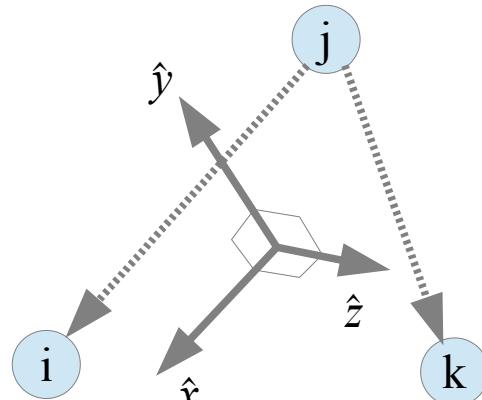
$$\hat{x} = \frac{1}{\|x_i^A - x_j^A\|} (x_i^A - x_j^A) \quad \text{X axis}$$

$$\hat{y} = \frac{1}{\|x_k^A - x_j^A\|} \hat{x} \times (x_k^A - x_j^A) \quad \text{Y axis}$$

$$\hat{z} = \hat{x} \times \hat{y} \quad \text{Z axis}$$

$$O = \frac{1}{3} (x_i^A + x_j^A + x_k^A) \quad \text{Origin}$$

$$\hat{x}_a^A = (\hat{x} \cdot (x_a^A - O), \hat{y} \cdot (x_a^A - O), \hat{z} \cdot (x_a^A - O)), a = 1, \dots, M \quad \text{Transformation}$$

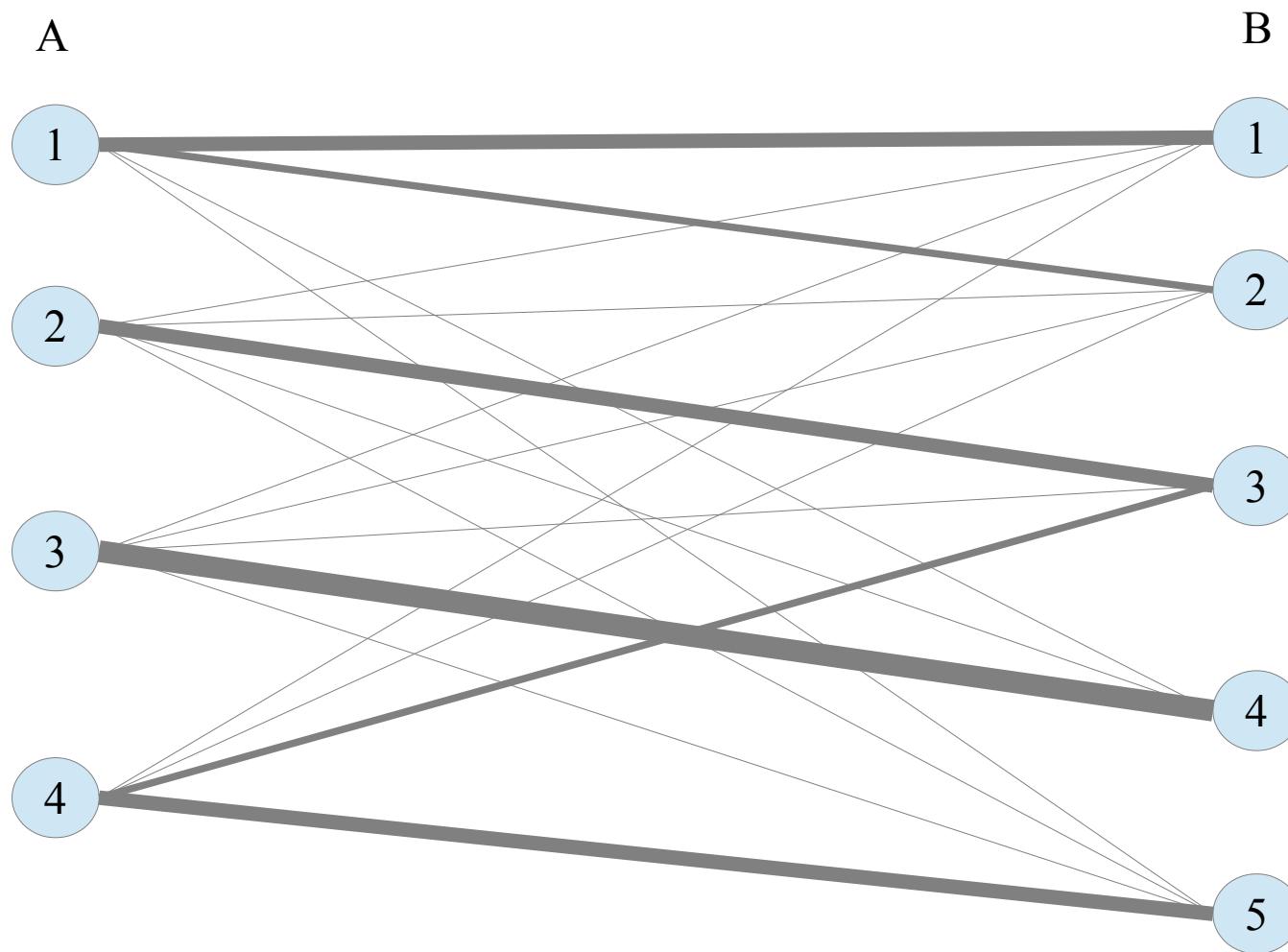


# Simple superposition algorithm

```
Input: Structure A=x(1)..x(M); Structure B=y(1)..y(N)
Output: Best alignment Ali
Ali := {} --- 初期アライメント(空集合)
for (i,j,k) in {1..M} do --- Select 3 points from A
    basisA := make_basis(x(i),x(j),x(k)) --- Make a basis
    for a = 1..M do
        x'(a) := transform(x(a),basisA)
    for (l,m,n) in {1..N} do --- Select 3 points from B
        basisB := make_basis(y(l),y(m),y(n)) --- Make a basis
        S := {} --- Initial (empty) alignment
        for b = 1..N do
            y'(b) := transform(y(b),basisB)

        (* After transformation, count neighboring A,B points *)
        for a = 1..M do
            for b = 1..N do
                if |x'(a) - y'(b)| < delta
                then S := S { (a,b) } --- Add pair to alignment
        if |S| > |Ali| then Ali := S --- Save the best one!
```

# A possible result

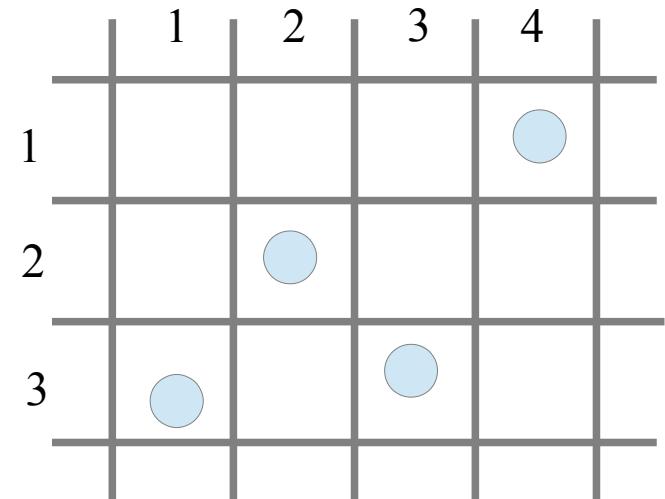


# Geometric Hashing (GH)

- The simple approach is simply too slow.
- Make a dictionary (hash table)  $x' \rightarrow$  basis
- Looking up the dictionary is fast:  $O(1)$ , no loop.

The coordinate after transformed by basisB.

$$(x'(l), y'(l), z'(l)) \rightarrow \text{basisB}$$



# Creating a hash table

Input: Structure B     $y(1) \dots y(N)$

Output: Hash table HB

```
for (l,m,n) in 1..N do
    basisB := make_basis(y(l),y(m),y(n))
    for b = 1..N do
        y'(b) := transform(y(b),basisB)
        HB := HB      (y'(b) => y'(b),basisB)
```

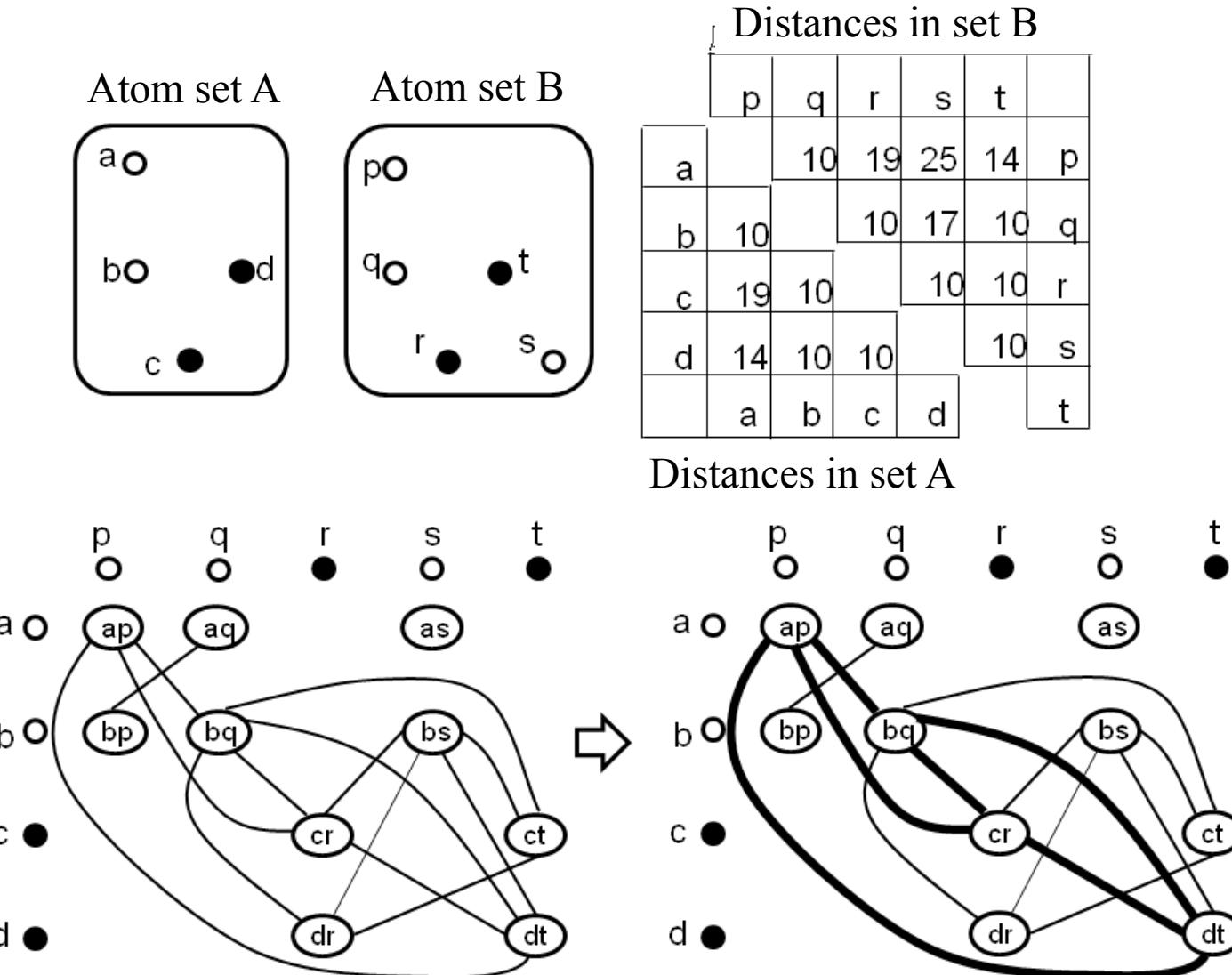
This requires  $N^2(N-1)(N-2)$  steps.

# Structure comparison by GH

```
Input: Structure A=x(1)..x(M); Structure B=y(1)..y(N)
Output: Best alignment Ali
HB := make_hashtable(B) --- Create hash table
for (i,j,k) in {1..M} do --- Select 3 points from A
    basisA := make_basis(x(i),x(j),x(k)) --- Make a basis
    for a = 1..M do
        x'(a) := transform(x(a),basisA)
        for y'(b),basisB in find_hash(x'(a)) --- Find a B-basis
            P(basisA,basisB) := {(a,b)}      P(basisA,basisB)
            --- Add the atom pair
Ali := Max|P(basisA,basisB)| --- (*) Be careful!
```

The last step (\*) requires a smart data structure!  
Otherwise, this method is as slow as the previous one.

# Distance comparison method



# Basic idea of distance-based method

$$A = (x_1^A, \dots, x_M^A)$$

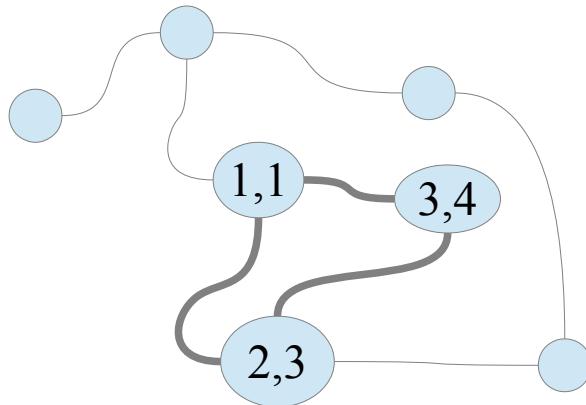
$$B = (x_1^B, \dots, x_N^B)$$

Given A and B, consider all the pairs of A and B points:  $P = \{(x_i^A, x_j^B)\}$

For the pair of pairs (i,j) and (k,l), the two distances (i,k) & (j,l) are similar, draw an edge between the nodes (i,j) & (k,l).

$$\| \|x_i^A - x_k^A\| - \|x_j^B - x_l^B\| \| < \delta$$

Find the subgraph of thus created graph, that is complete and maximum:  
The maximum clique problem



# Algorithm

Bron-Kerbosch (1973)

R := empty  
P := set of vertices  
X := empty

```
BronKerbosch1(R, P, X):  
    if P and X are both empty:  
        report R as a maximal clique  
    for each vertex v in P:  
        BronKerbosch1(R      {v},  P      N(v),  X      N(v))  
        P := P \ {v}  
        X := X      {v}
```

Where  $N(v)$  is the set of vertices connected with "v".

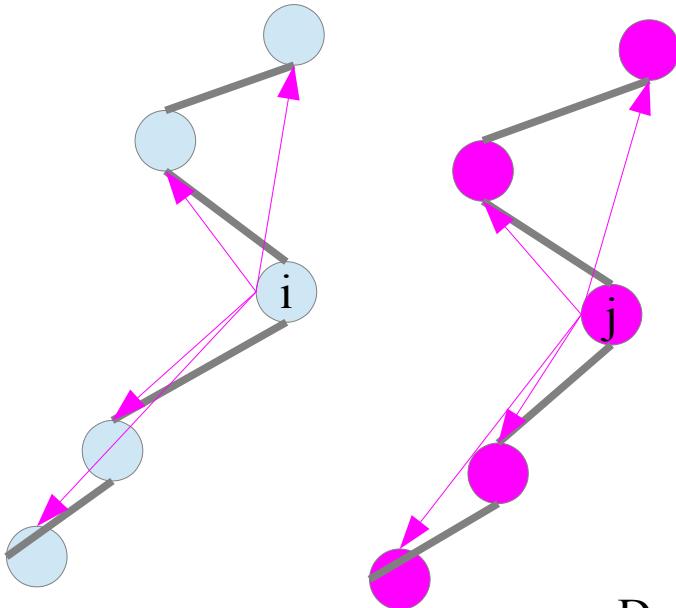
From [http://en.wikipedia.org/wiki/Bron–Kerbosch\\_algorithm](http://en.wikipedia.org/wiki/Bron–Kerbosch_algorithm)

This is an exact algorithm, and may not terminate.

# Double Dynamic Programming

- Distance-based methods are also computationally demanding.
- DDP is a hybrid of coordinate- & distance-based methods
- Applying DP (just as in sequence comparison) in two layers.
- This requires the point set to be ordered.

# DDP: idea



$$A = (x_1^A, \dots, x_M^A)$$

$$B = (x_1^B, \dots, x_N^B)$$

Assume  $(x_i^A, x_j^B)$  is a matching pair of points.

If  $(i, j)$  is really a matching pair, the “scene of A from i” and the “scene of B from j” should look similar.

Define the similarity measure for the “scenes” based on  $(i, j)$ :

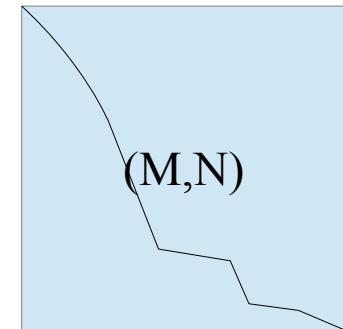
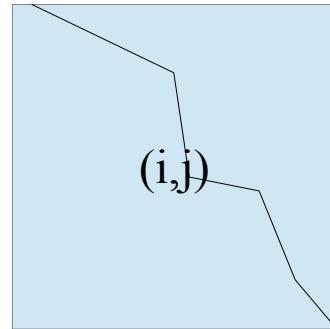
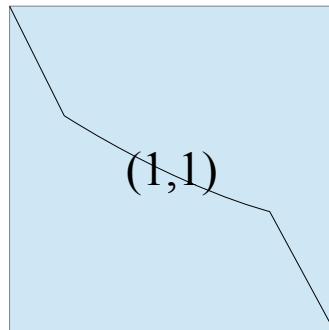
$$s(k, l; i, j) = \frac{1}{|d_A(i, k) - d_B(j, l)| + c}$$

Apply DP by regarding this as a score matrix  $s(k, l)$ , you get the “best” alignment under the assumption that  $(i, j)$  is a matching pair. The score is, say:  $S_1(i, j)$   
( Do this for all possible  $(i, j)$  pairs. )

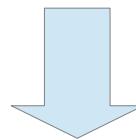
Then using  $S_1(i, j)$  as a score matrix, apply another DP.

This will yield an approximation to the “best” alignment

# DDP procedure



...                    ...



# DDP Algorithm

```
# lower level DP
for i=1..M do
    for j=1..N do
        S(i,j) = DP using s(k,l; i,j) --- details omitted.

# upper level DP
for i= 1..M do
    for j= 1..N do
        D := T(i-1,j-1) + S(i,j)
        V := T(i-1,j) - g
        H := T(i,j-1) - g
        T(i,j) := max(d,v,h)
        if T(i,j) = d then P(i,j) := 'D'           --- diagonal
        else if T(i,j) = v then P(i,j) := 'V'       --- vertical
        else T(i,j) := 'H'                         --- horizontal
    done
done
Score := T(M,N)
--- omitting the rest...
```

# Why DDP works

- If  $(i,j)$  is a truly matching pair
  - $S_1(i,j)$  is a large positive value.
- If  $(i,j)$  is not a truly matching pair
  - $S_1(i,j)$  is a small value.
- The scores of truly matching pairs are amplified along the (sub)optimal alignment.

# Summary

- 2 approaches for structure comparison
  - Coordinate-based
  - Distance-based
- In special cases, dynamic programming can be also used.