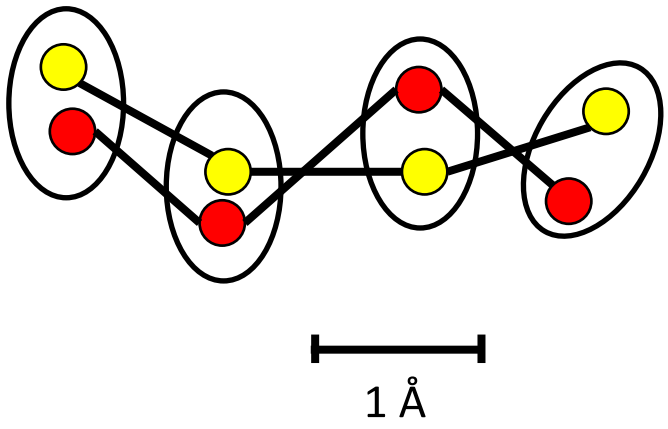
Научно-технологический университет Сириус

Структурная биоинформатика | Лекция 12

Мембранные белки. Каналы

Александр Злобин

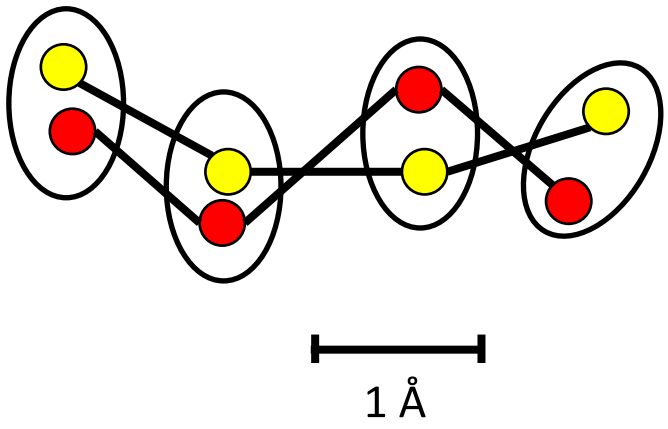
Оценка сходства структур: RMSD



$$\mathrm{RMSD}(\mathbf{v},\mathbf{w}) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\|v_i - w_i\|^2} \qquad (1)$$

$$= \sqrt{\frac{1}{n}\sum_{i=1}^{n}((v_{ix}-w_{ix})^2 + (v_{iy}-w_{iy})^2 + (v_{iz}-w_{iz})^2)} \qquad (2)$$

Оценка сходства структур: eRMSD

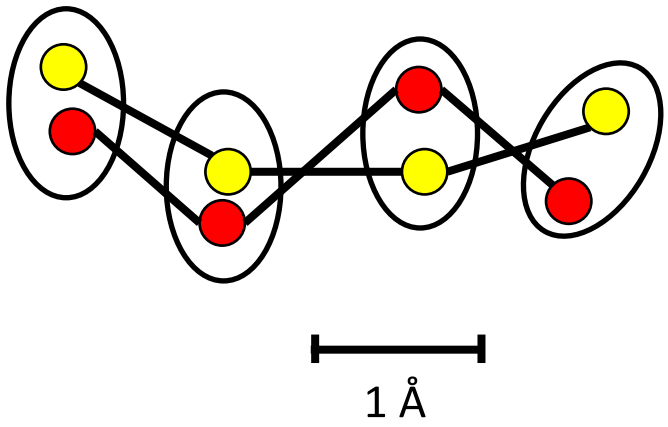


Идея:

- Описать каждую пару оснований внутри структуры как вектор контакта
  - Вектор контакта - вектор между центрами масс оснований
- Оценить сходство структр как разницу векторов

$$eRMSD = \sqrt{\frac{1}{N}\sum_{j,k} |\vec{G}(\tilde{\vec{r}}^{\alpha}_{jk}) - \vec{G}(\tilde{\vec{r}}^{\beta}_{jk})|^2}$$

Оценка сходства структур: TM-score



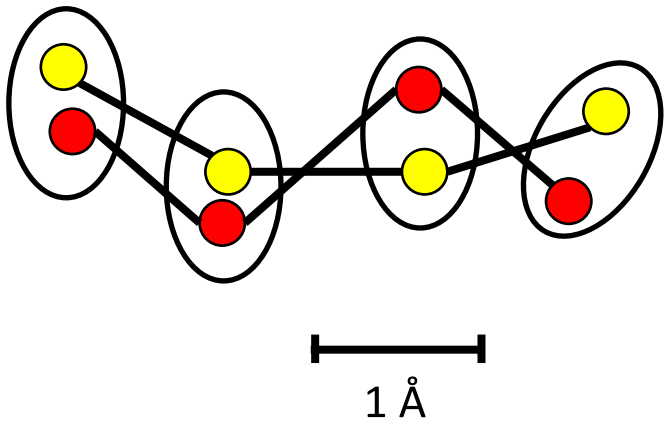
Идея:

- Избавиться от влияния длины

$$\text{TM-score} = \max\left[\frac{1}{L_{\text{target}}}\sum_{i}^{L_{\text{aligned}}}\frac{1}{1+\left(\frac{d_i}{d_0(L_{\text{target}})}\right)^2}\right]$$

$$d_0(L_{\text{target}}) = 1.24\sqrt[3]{L_{\text{target}} - 15} - 1.8$$

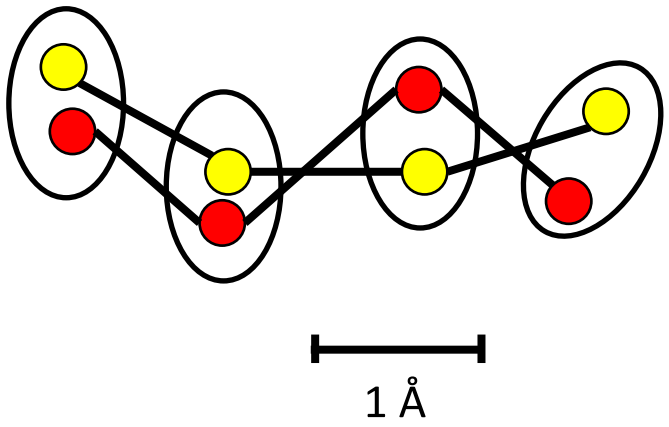
Оценка сходства структур: TM-score



Другие:

- GDT (Global Distance Test)
- MaxSub

Совмещение структур



Целевая функция:

$$\min RMSD \to 0$$

Sipple, Stegbuhner, 1991

1. Совмещаем центры масс:

$$C_j^{B\prime} = C_j^B + \bar{r}$$

$$\bar{r} = \frac{\sum_{k=1}^{L} C_{j_k}^B - \sum_{k=1}^{L} C_{i_k}^A}{L}$$

2. Находим вращение:

2.1. Вычисляем угол поворота вокруг оси X, при котором минимизируется RMSD

2.2. Вычисляем угол поворота вокруг оси Y, при котором минимизируется RMSD

2.3. Вычисляем угол поворота вокруг оси Z, при котором минимизируется RMSD

2.4. Пока углы не стабилизируются.

Инструменты совмещения

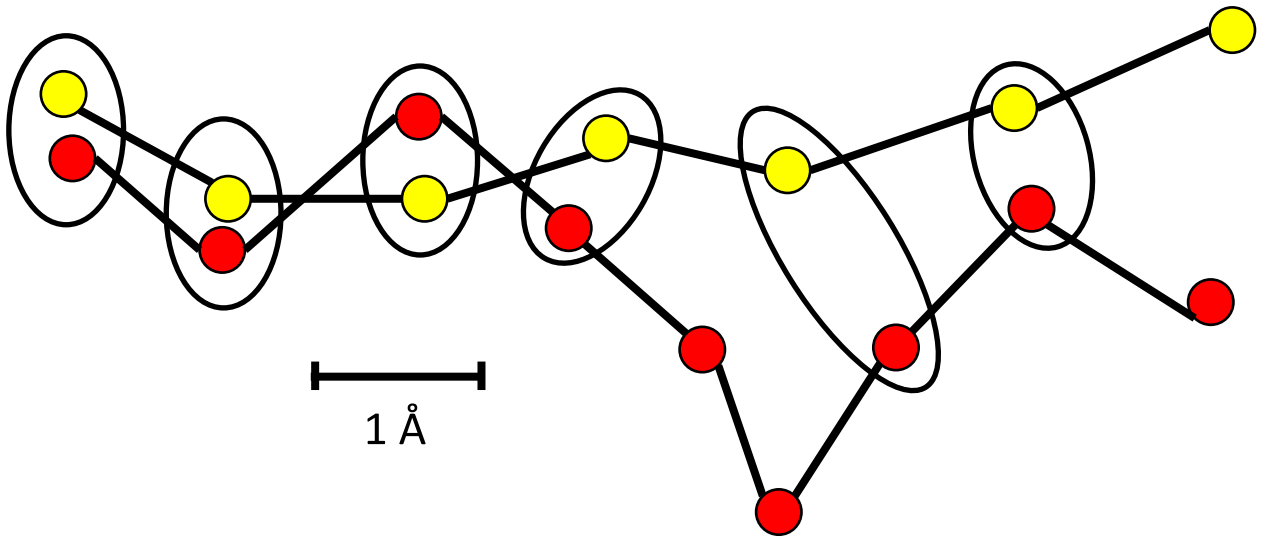
Алгоритмы:

- Kabsh, (Kabsh, 1976)
- QCP (Theobald, 2005)

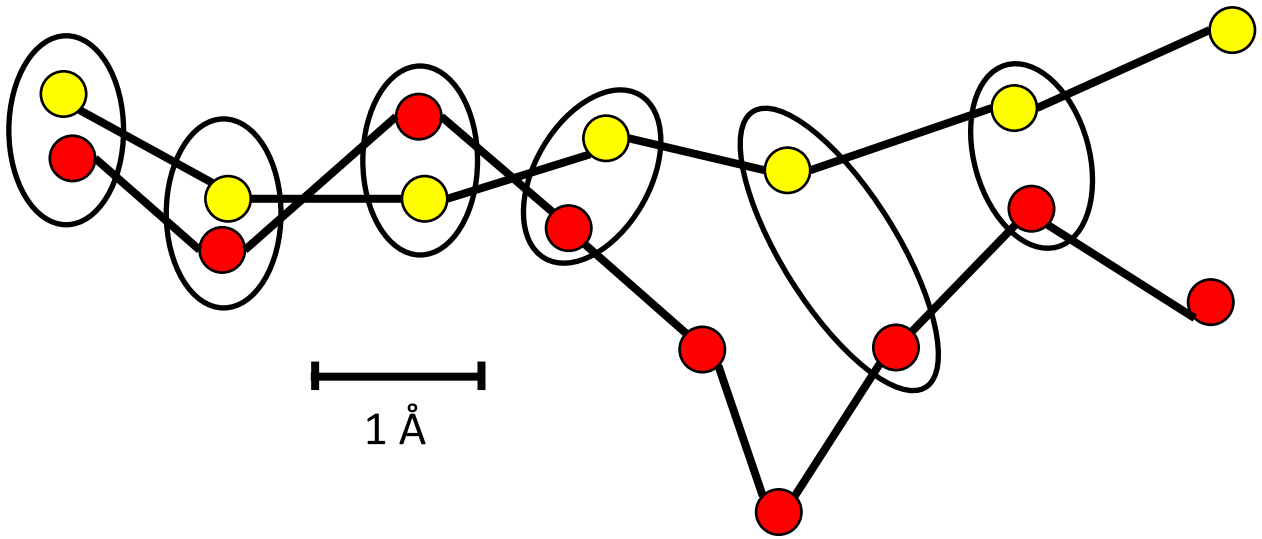
Инструменты:

- PyMOL: fit, super
- pyRMSD
- GROMACS
- Mimiqa
- etc

Поиск соответствия



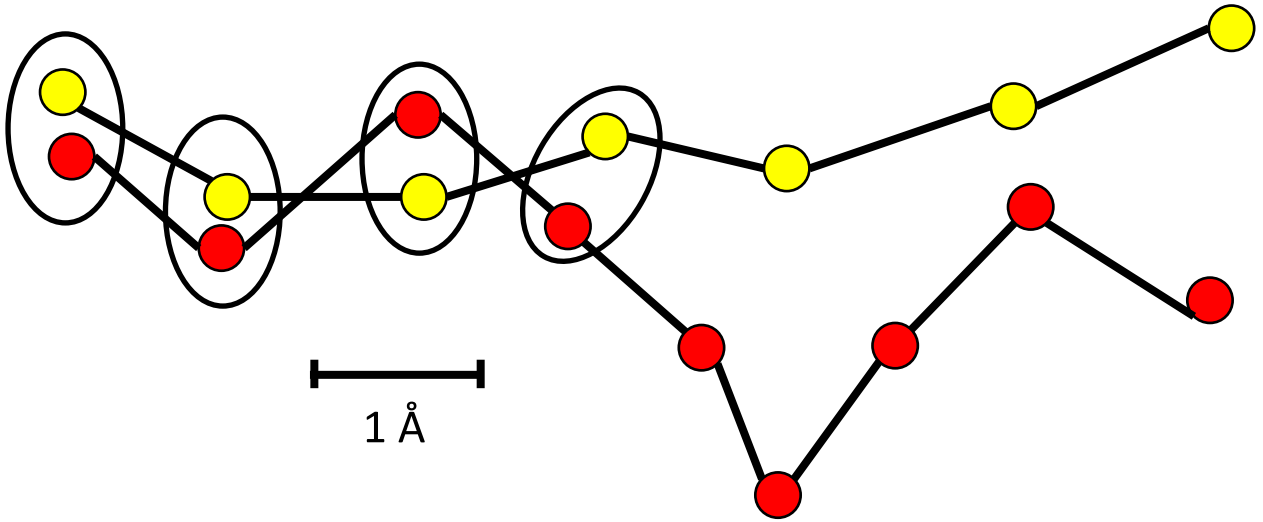
Поиск соответствия



Частный случай:

- Ручное указание
- Парное/множественное выравнивание последовательностей

Поиск соответствия

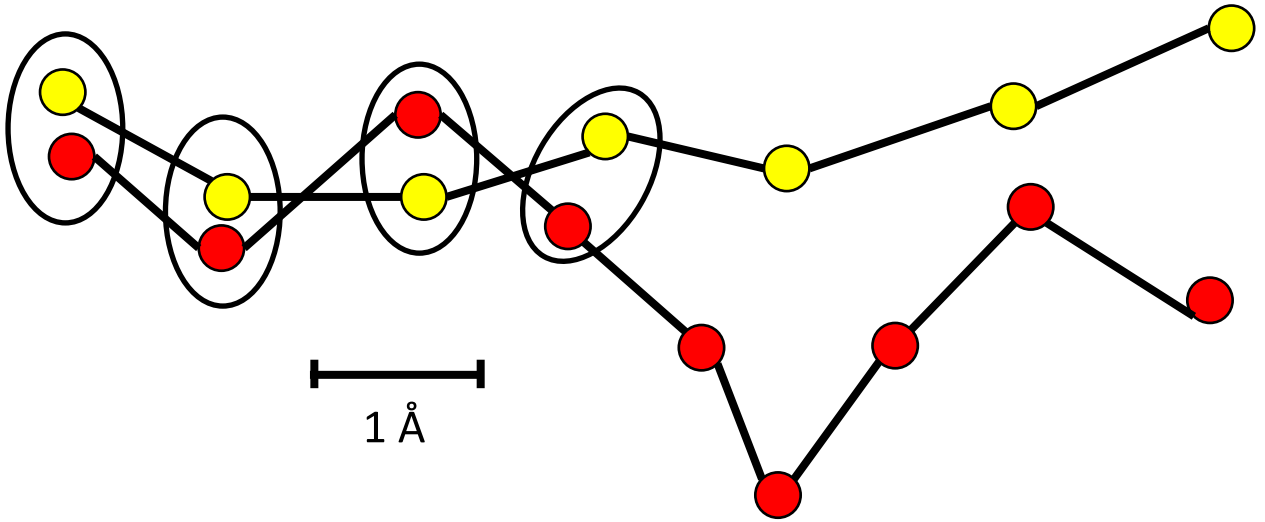


Общий случай: требуется найти такое сопоставление частиц двух молекул, что:

$$L_{matched\ particles} \rightarrow max$$

$$min(RMSD) \rightarrow 0$$

Поиск соответствия



Общий случай: требуется найти такое сопоставление частиц двух молекул, что:

$$L_{matched\ particles} \rightarrow max$$

$$min(RMSD) \rightarrow 0$$

Обычно это противоречит друг другу!

Алгоритм CE (combinatorial extension)

Идея №1: Надо придумать такую меру сходства структур, которая отражала бы некий баланс между нашим требованием к максимизации длины выравнивания и минимизации RMSD.

Конкретно в этом алгоритме эта мера получена так. Авторы взяли много пар структур (какие-то из них похожи, какие-то нет) и построили выравнивания при помощи своего алгоритма. Для каждого выравнивания можно посчитать длину и RMSD. Для каждой длины выравнивания можно получить распределение RMSD по такой случайной выборке. Мерой сходства является Z-score для конкретного выравнивания.

Как видите, тут плохо с формализацией. Алгоритм на обучающей выборке выдает какие-то значения, - те, которые достаточно маленькие считаются достаточно хорошими.

Shindyalov I.N., Bourne P.E. Protein structure alignment by incremental combinatorial extension (CE) of the optimal path. Protein Eng. 1998. 11(9).

Алгоритм CE

Идея №2: Если есть черновое выравнивание, то его можно оптимизировать. А именно, можно:

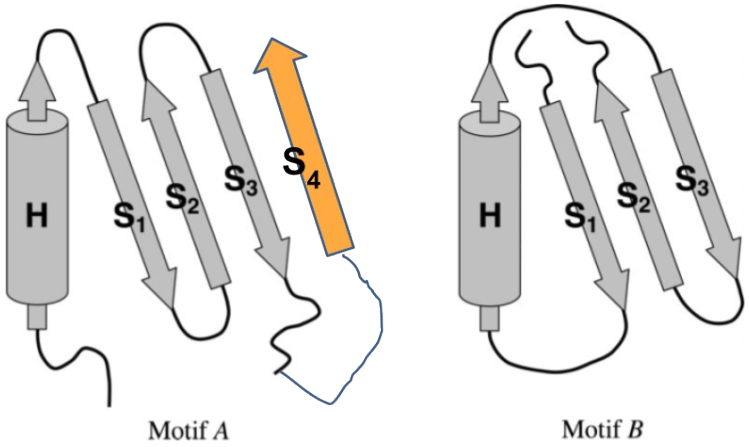
1. Удлинять выровненные участки.
2. Укорачивать их.
3. Сдвигать гэпы.

Каждое такое изменение соответствует некоторому новому выравниванию. И, следовательно, некоторому совмещению. И у него есть длина, RMSD и Z-score. Можно посмотреть, увеличился ли он от каждого возможного изменения.

Алгоритм CE рассматривает последовательности таких изменений, выбираемые случайным образом. Те последовательности, которые ведут к сильному ухудшению Z-score отбраковываются. Остальные – растут дальше.

Shindyalov I.N., Bourne P.E. Protein structure alignment by incremental combinatorial extension (CE) of the optimal path. Protein Eng. 1998. 11(9).

Алгоритм PDBeFold (SSM)



Motif *A* Motif *B*

Это модифицированный пример из оригинальной статьи. Я добавил к первой структуре тяж S_4. Вопрос: какие элементы вторичной структуры тут расположены примерно одинаково и в A, и в B?
Ответ: H, S_1, S_2, S_3.

1. Рассмотрим все пары элементов из первой структуры:
   (H, S_1), (H, S_2), (H, S_3), (H, S_4), (S_1, S_2), (S_1, S_3), (S_1, S_4),
   (S_2, S_3), (S_2, S_4), (S_3, S_4)
   + еще 10 (S_1, H), (S_2, H) и т.д.
2. Рассмотрим все пары элементов из второй структуры:
   (H', S_1'), (H', S_2'), (H', S_3'), (S_1', S_2'), (S_1', S_3'),
   (S_2', S_3')
   + еще 6 (S_1, H), (S_2, H) и т.д.
3. Рассмотрим двойки пар из п. 1 и 2 (например, (H, S_1) + (H', S_3') и проч.)
   Вопрос: сколько всего таких двоек?
   Ответ: 20 x 12 = 240.
   Вопрос: Все ли такие двойки пар соответствуют парам элементов, которые одновременно могут быть выровнены?

Krissinel E, Henrick K. Secondary-structure matching (SSM), a new tool for fast protein structure alignment in three dimensions. Acta Crystallogr D Biol Crystallogr. 2004 Dec;60(Pt 12 Pt 1)

Алгоритм PDBeFold (SSM)

Дано: Две структуры.
Задача: Найти такое выравнивание их последовательностей, которое при подстановке в задачу совмещения при заданном выравнивании дает такое RMSD, что величина

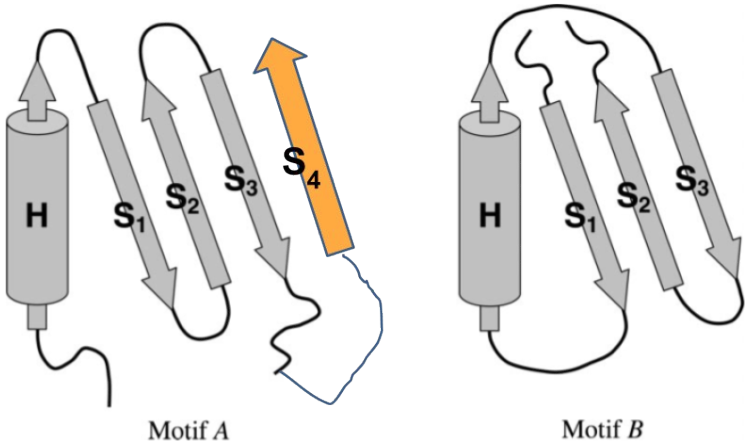
$$Q = \frac{L_{align}^2}{(1+(\mathrm{RMSD}/R_0))L_1 L_2}$$

оказывается максимальной.

Основные этапы:
1. Найти элементы вторичной структуры.
2. Найти такие множества элементов из первой и второй структуры, что эти элементы расположены в пространстве приблизительно одинаково.
3. Построить “черновое” выравнивание последовательностей. И оптимизировать его.

Krissinel E, Henrick K. Secondary-structure matching (SSM), a new tool for fast protein structure alignment in three dimensions. Acta Crystallogr D Biol Crystallogr. 2004 Dec;60(Pt 12 Pt 1)

Алгоритм PDBeFold (SSM)



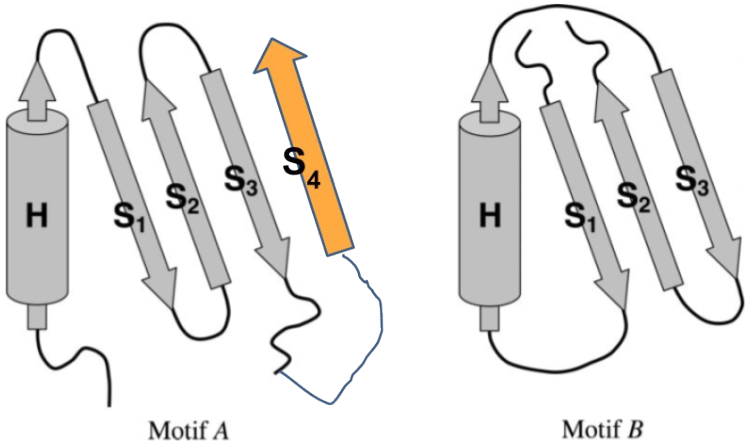
Motif *A* Motif *B*

Это модифицированный пример из оригинальной статьи. Я добавил к первой структуре тяж S_4. Вопрос: какие элементы вторичной структуры тут расположены примерно одинаково и в A, и в B?

1. Рассмотрим все пары элементов из первой структуры:
   (H, S_1), (H, S_2), (H, S_3), (H, S_4), (S_1, S_2), (S_1, S_3), (S_1, S_4),
   (S_2, S_3), (S_2, S_4), (S_3, S_4)
   + еще 10 (S_1, H), (S_2, H) и т.д.
2. Рассмотрим все пары элементов из второй структуры:
   (H', S_1'), (H', S_2'), (H', S_3'), (S_1', S_2'), (S_1', S_3'),
   (S_2', S_3')
   + еще 6 (S_1, H), (S_2, H) и т.д.
3. Рассмотрим двойки пар из п. 1 и 2 (например, (H, S_1) + (H', S_3') и проч.)
   Вопрос: сколько всего таких двоек?
   Ответ: 20 x 12 = 240.
   Вопрос: Все ли такие двойки пар соответствуют парам элементов, которые одновременно могут быть выровнены?

Krissinel E, Henrick K. Secondary-structure matching (SSM), a new tool for fast protein structure alignment in three dimensions. Acta Crystallogr D Biol Crystallogr. 2004 Dec;60(Pt 12 Pt 1)

Алгоритм PDBeFold (SSM)



Рассмотрим двойки пар (например, (H, S_1) + (H', S_3') и проч.) – это гипотезы об одновременном выравнивании пар элементов

Вопрос: Все ли такие двойки пар соответствуют парам элементов, которые одновременно могут быть выровнены?

(H, S_1) + (H', S_3')
Может, т.к. спираль соответствует спирали, а тяж - тяжу

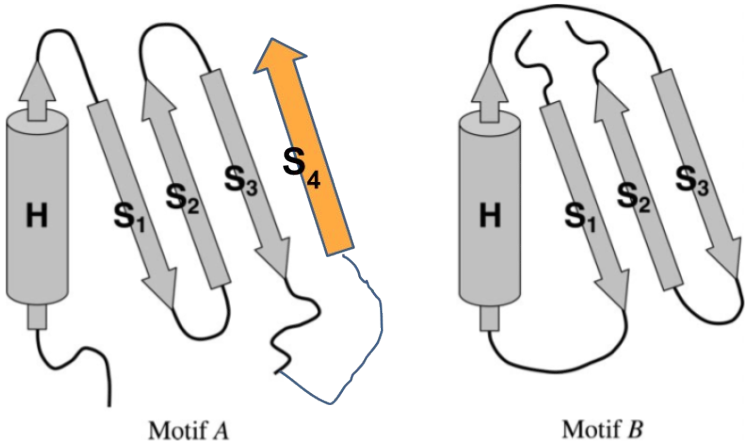
(H, S_1) + (S_1', S_3')
Не может, т.к. спираль не может быть выровнена с β-тяжем

(S_1, S_2) + (S_1', S_3')
Может

Из рисунка видно, что не все эти двойки реально выровнены. Но алгоритм этого пока не знает – это надлежит установить.

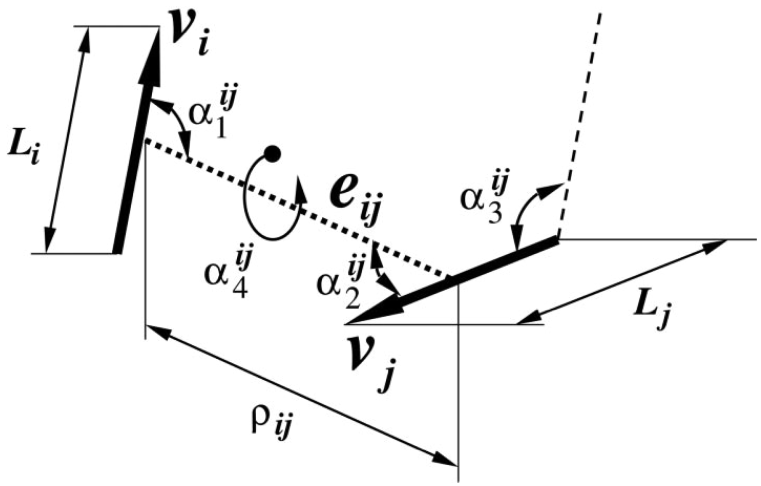
Krissinel E, Henrick K. Secondary-structure matching (SSM), a new tool for fast protein structure alignment in three dimensions. Acta Crystallogr D Biol Crystallogr. 2004 Dec;60(Pt 12 Pt 1)

Алгоритм PDBeFold (SSM)



Рассмотрим двойки пар (например, (H, S_1) + (H', S_3') и проч.) – это гипотезы об одновременном выравнивании пар элементов

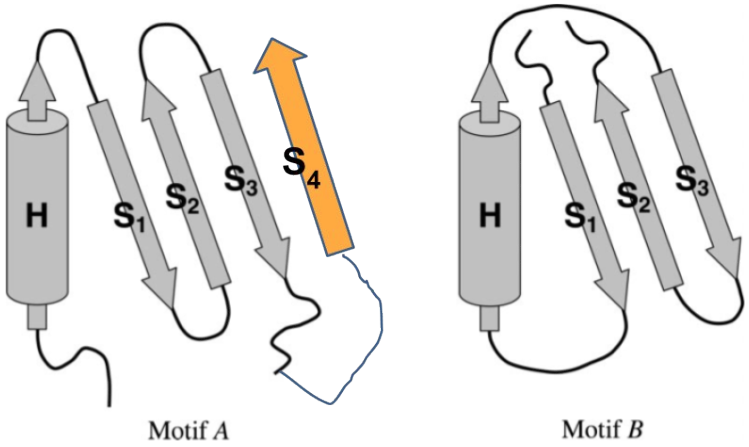
Вопрос: Все ли такие двойки пар соответствуют парам элементов, которые одновременно могут быть выровнены?



Каждая пара элементов аппроксимируется парой векторов. Их взаимное расположение характеризуется длинами векторов, расстоянием между центрами и какими-то углами. На основании сходства таких параметров для двойки пар можно отобрать такие двойки, которые соответствуют парам потенциально выровненных элементов.

Krissinel E, Henrick K. Secondary-structure matching (SSM), a new tool for fast protein structure alignment in three dimensions. Acta Crystallogr D Biol Crystallogr. 2004 Dec;60(Pt 12 Pt 1)

Алгоритм PDBeFold (SSM)



$(H, S_1) + (H', S_1')$

$(H, S_2) + (H', S_2')$

$(H, S_3) + (H', S_3')$

$(S_1, S_2) + (S'_1, S'_2)$

$(S_1, S_3) + (S'_1, S'_3)$

$(S_2, S_4) + (S'_1, S'_3)$

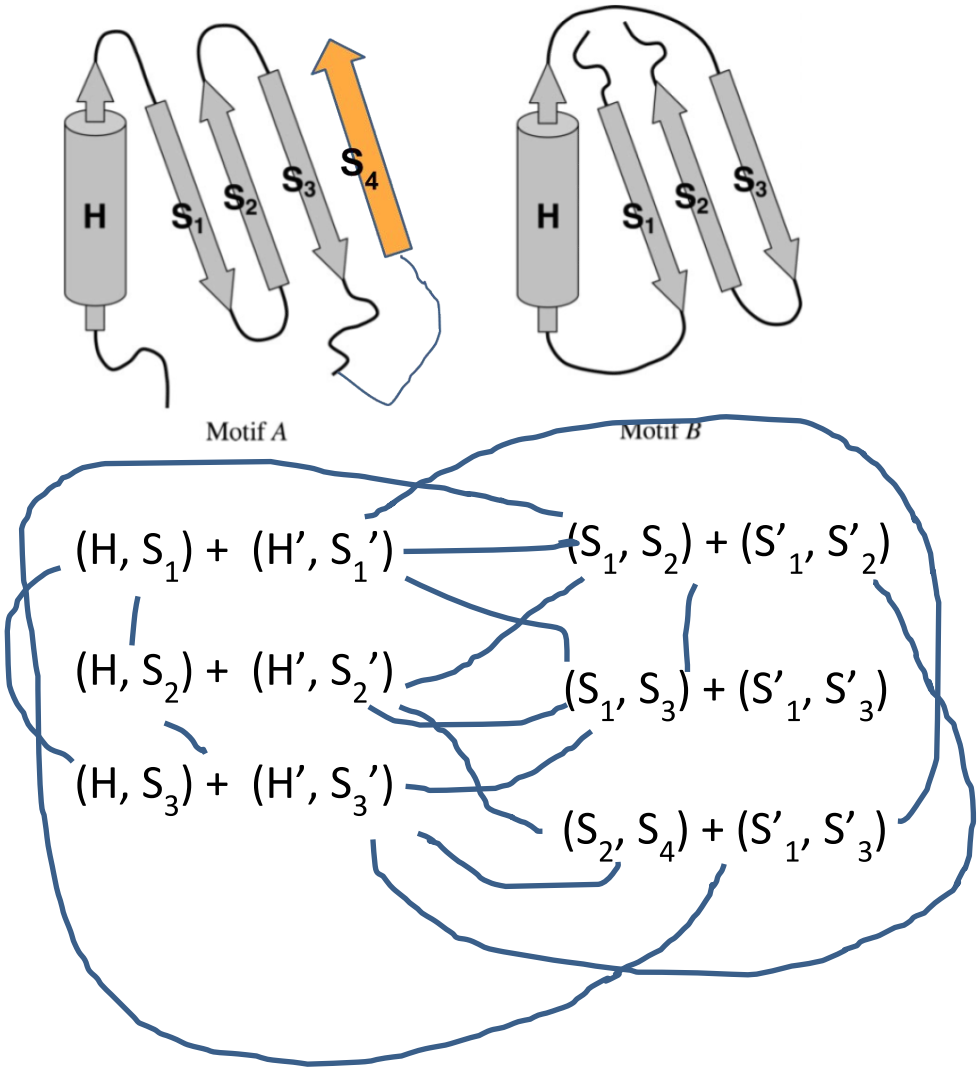
Отберем из всех двоек вида
те, которые потенциально могут быть выровнены. Это будут вершины графа. (На рисунке ниже показаны далеко не все вершины.)

Часть вершин соответствует правильному выравниванию. Часть – нет, но мы этого пока не знаем. Такова, например, двойка $(S_2, S_4) + (S'_1, S'_3)$. Как видно из рисунка в правильном выравнивании такой двойки нет. Но вообще говоря, это две пары тяжей, идущих в одном направлении, примерно с одинаковым расстоянием между ними.

Теперь соединим ребрами те вершины, между которыми нет противоречия.

Krissinel E, Henrick K. Secondary-structure matching (SSM), a new tool for fast protein structure alignment in three dimensions. Acta Crystallogr D Biol Crystallogr. 2004 Dec;60(Pt 12 Pt 1)

Алгоритм PDBeFold (SSM)



Теперь соединим ребрами те вершины, между которыми нет противоречия.

Например, двойки
$(H, S_1) + (H', S_1')$ и
$(H, S_3) + (H', S_3')$
вполне могут быть в одном выравнивании.

А двойки
$(S_1, S_3) + (S'_1, S'_3)$ и
$(S_2, S_4) + (S'_1, S'_3)$
- нет. Действительно, S'_1 не может быть одновременно выровнен
и с S_1, и с S_2.

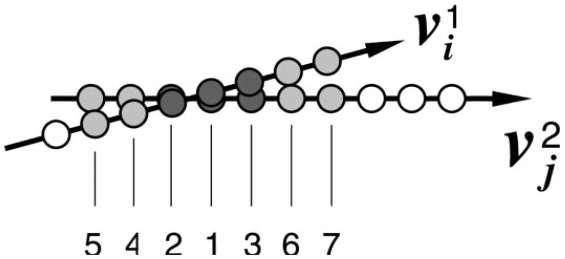
Клика в таком графе соответствует максимальному множеству выровненных элементов.

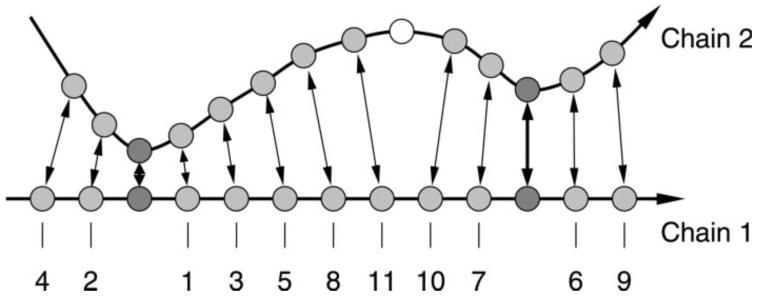
Krissinel E, Henrick K. Secondary-structure matching (SSM), a new tool for fast protein structure alignment in three dimensions. Acta Crystallogr D Biol Crystallogr. 2004 Dec;60(Pt 12 Pt 1)

Алгоритм PDBeFold (SSM)

Далее, заменим каждый SSE на точку (его центр). Поскольку у нас уже есть списки совпадающих элементов, эти точки можно совместить. Получится черновое совмещение структур.

Теперь надо перейти от выравнивания элементов вторичной структуры к выравниванию последовательностей.





Для совпадающих SSE (см. предыдущий этап) выбирает 3-4 аминокислоты, наиболее близких друг к другу. Затем выравнивание расширяется на весь тяж или спираль.

Для всех остальных алгоритм находит взаимно наиболее близкие C_α атомы. Соседние с ними пары атомов также считаются выровненными.

Krissinel E, Henrick K. Secondary-structure matching (SSM), a new tool for fast protein structure alignment in three dimensions. Acta Crystallogr D Biol Crystallogr. 2004 Dec;60(Pt 12 Pt 1)

Алгоритм PDBeFold (SSM)

Далее, заменим каждый SSE на точку (его центр). Поскольку у нас уже есть списки совпадающих элементов, эти точки можно совместить. Получится черновое совмещение структур.

Теперь можно перейти от выравнивания элементов вторичной структуры к выравниванию последовательностей. При этом близко расположенные C_α атомы считаются выровненными и это выравнивание распространяется на их соседей по полипептидной цепочке.

Теперь у нас есть **черновое выравнивание** последовательностей. Ему соответствует некое совмещение структур, длина выравнивания, RMSD и Q-score. Его можно **оптимизировать**: менять разными способами, строить новое совмещение и добиваться увеличения Q. И так много-много раз, пока **решение не стабилизируется**.

Krissinel E, Henrick K. Secondary-structure matching (SSM), a new tool for fast protein structure alignment in three dimensions. Acta Crystallogr D Biol Crystallogr. 2004 Dec;60(Pt 12 Pt 1)

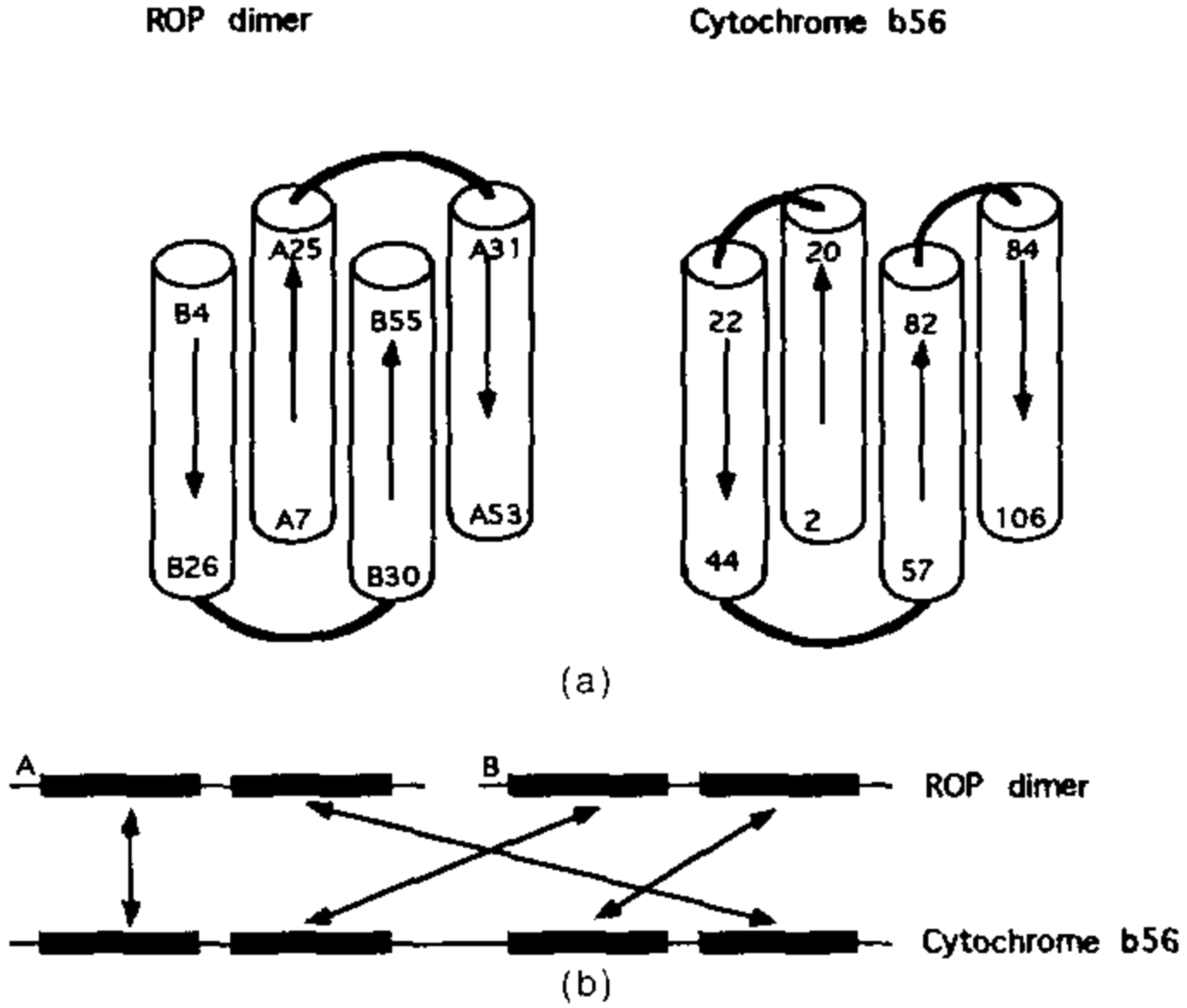
Алгоритм PDBeFold (SSM)

Это весьма приблизительное и свободное изложение алгоритма. Реальность несколько сложнее. Обязательно ознакомьтесь с первоисточником.

В случае с алгоритмами для работы с пространственными (3D) структурами очень часто используется большое количество эвристик и пороговых значений, от которых может зависеть результат. Всегда надо быть очень аккуратными и обращать внимание на все "ручки", которые можно крутить.

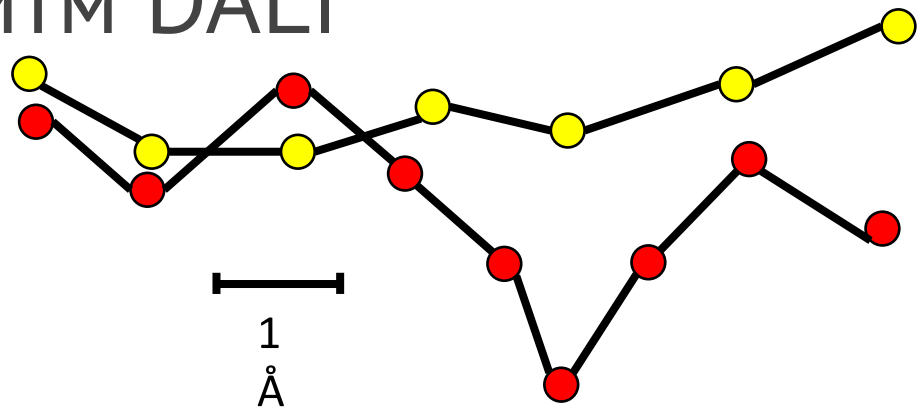
Krissinel E, Henrick K. Secondary-structure matching (SSM), a new tool for fast protein structure alignment in three dimensions. Acta Crystallogr D Biol Crystallogr. 2004 Dec;60(Pt 12 Pt 1)

Алгоритм DALI



Holm L, Sander C. Protein structure comparison by alignment of distance matrices. J Mol Biol. 1993 Sep 5;233(1)

Алгоритм DALI



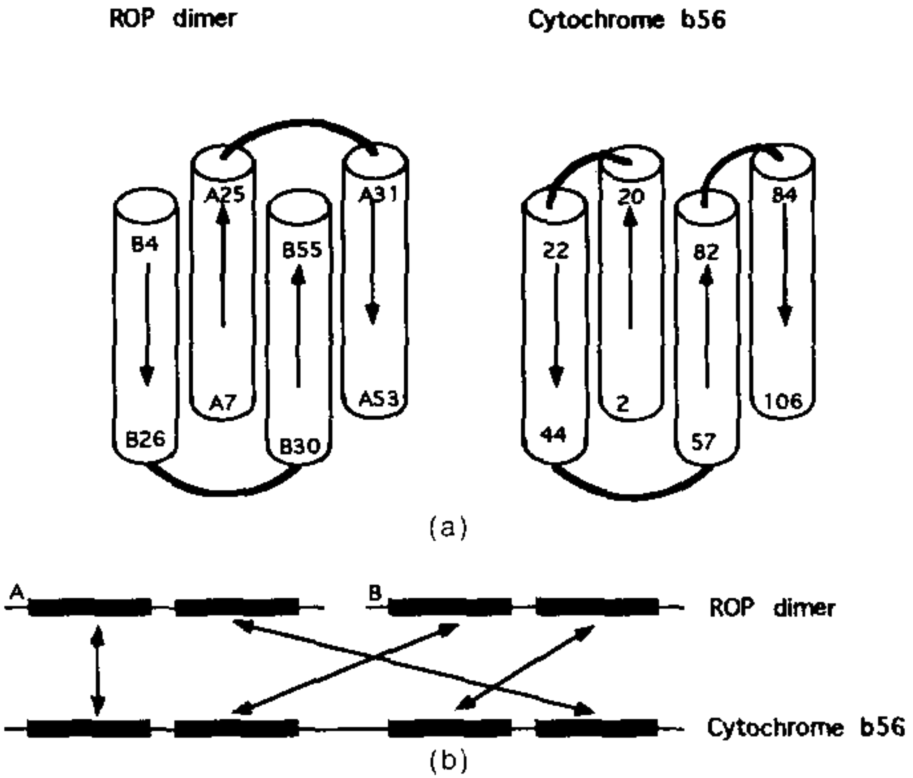
	1	2	3	4	5	6	7
1	0	1	1.9	2.8	3.7	4.5	5.4
2		0	1	1.9	2.8	3.7	4.5
3			0	1	1.9	2.8	3.7
4				0	1	1.9	2.8
5					0	1	1.9
6						0	1
7							0

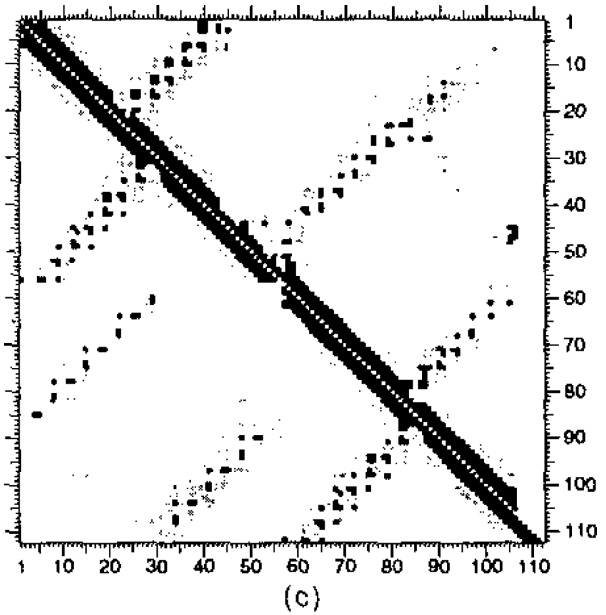
	1	2	3	4	5	6	7	8	9
1	0	1	1.5	2.5	3.2	4.0	4.2	4.6	5.1
2		0	d	d	d	d	d	d	d
3			0	d	d	d	d	d	d
4				0	d	d	d	d	d
5					0	d	d	d	d
6						0	d	d	d
7							0	d	d
8								0	d
9									0

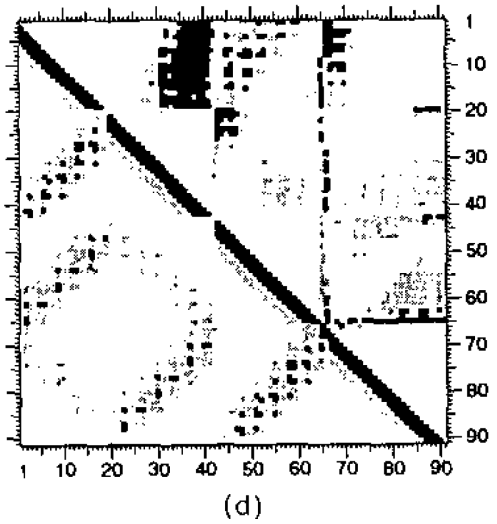
Задача: выбрать выровненные атомы так, чтобы матрицы стали похожими. При этом разрешается удалять строки (и столбцы) из матрицы, а также менять их местами.

Holm L, Sander C. Protein structure comparison by alignment of distance matrices. J Mol Biol. 1993 Sep 5;233(1)

Алгоритм DALI

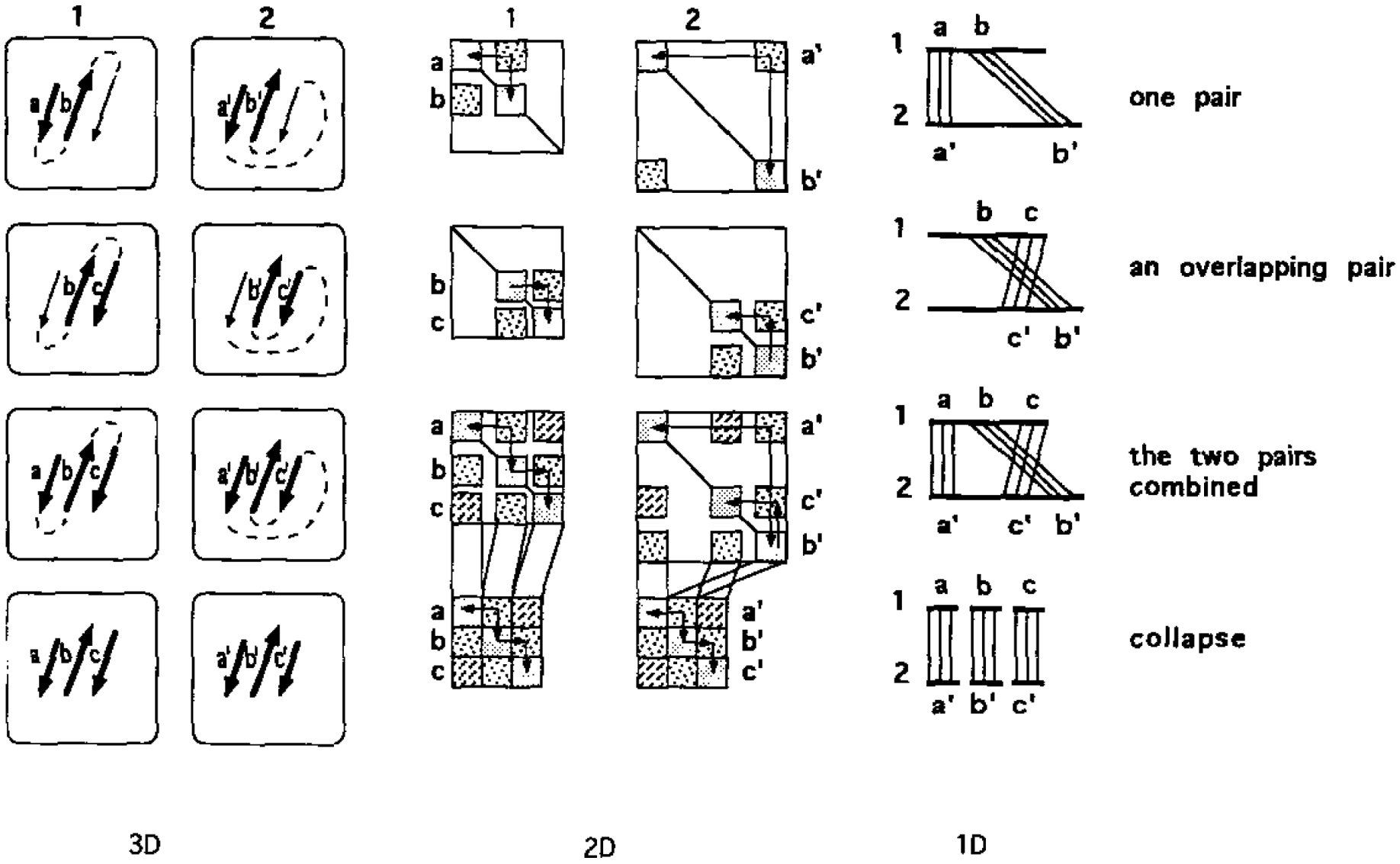






Holm L, Sander C. Protein structure comparison by alignment of distance matrices. J Mol Biol. 1993 Sep 5;233(1)

Алгоритм DALI



Holm L, Sander C. Protein structure comparison by alignment of distance matrices. J Mol Biol. 1993 Sep 5;233(1)

Алгоритм DALI

Целевая функция

$$S = \sum_{i=1}^{L} \sum_{j=1}^{L} \phi(i,j)$$

, где i и j – координаты в матрице (= в выравнивании)

$$\phi^R(i,j) = \theta^R - |d_{ij}^A - d_{ij}^B|$$

$$\phi^E(i,j) = \left\{ \begin{array}{l} (\theta^E - \frac{|d_{ij}^A - d_{ij}^B|}{d_{ij}^*})\omega(d_{ij}^*),\ i \neq j \\ \theta^E,\ i = j \end{array} \right\}, \omega(r) = e^{-\frac{r^2}{400}}$$

Смысл: сильно непохожие расстояния вообще не надо учитывать.

Задача: подобрать такую перестановку строк и столбцов в матрице, чтобы S → max

Holm L, Sander C. Protein structure comparison by alignment of distance matrices. J Mol Biol. 1993 Sep 5;233(1)

Алгоритм гибкого совмещения (FATCAT)

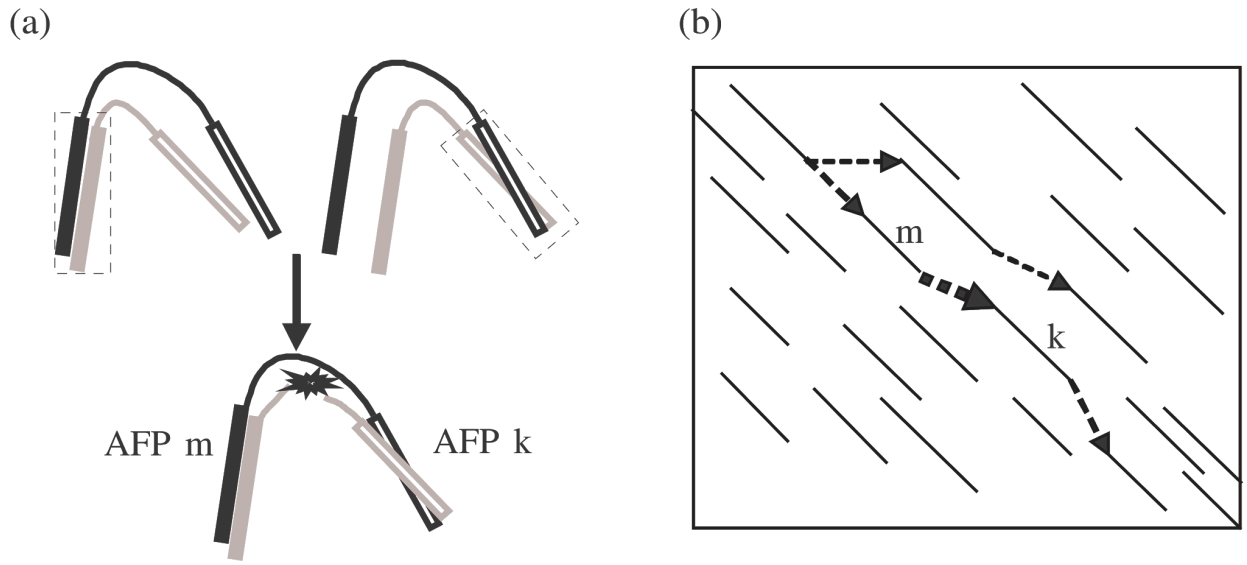
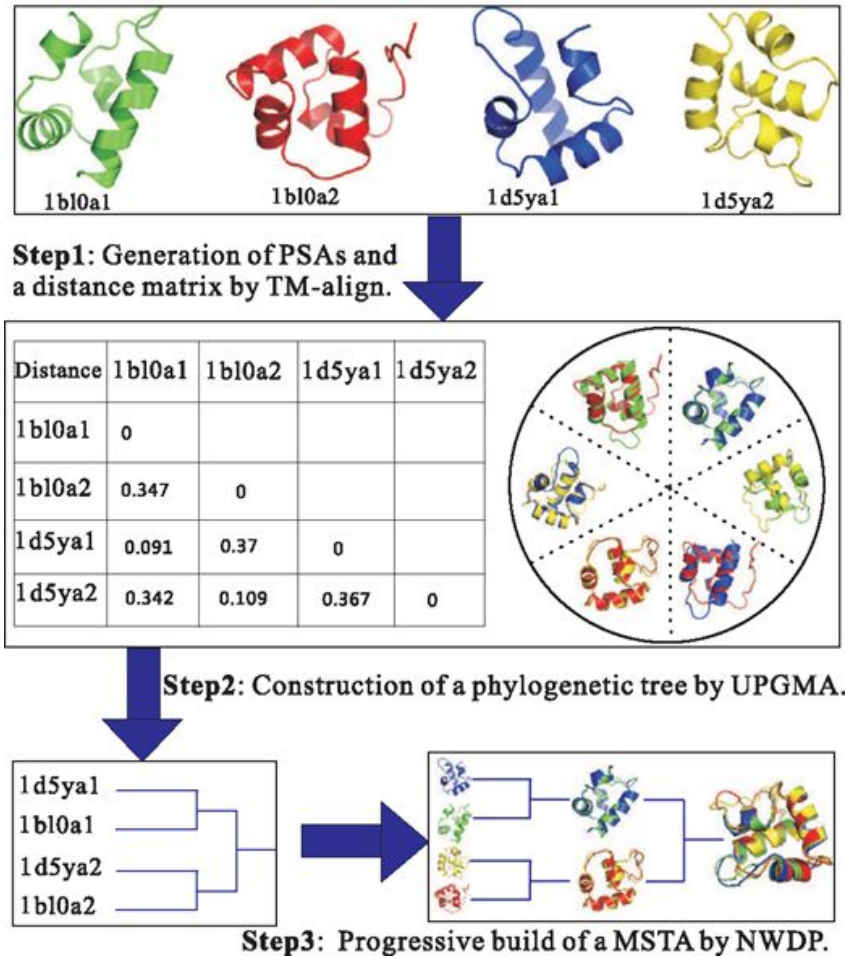


Fig. 2. Structure alignment by AFPs chaining. (a) A twist is introduced in one structure to connect AFP *m* and *k*. (b) Dot matrix of AFPs. Each AFP is shown as a line. A chain linking AFPs corresponds to an alignment between two structures (for clarity, only two chains are shown in the graph).

Ye, Godzik, 2003

Множественное структурное выравнивание: mTM-align



Идея:

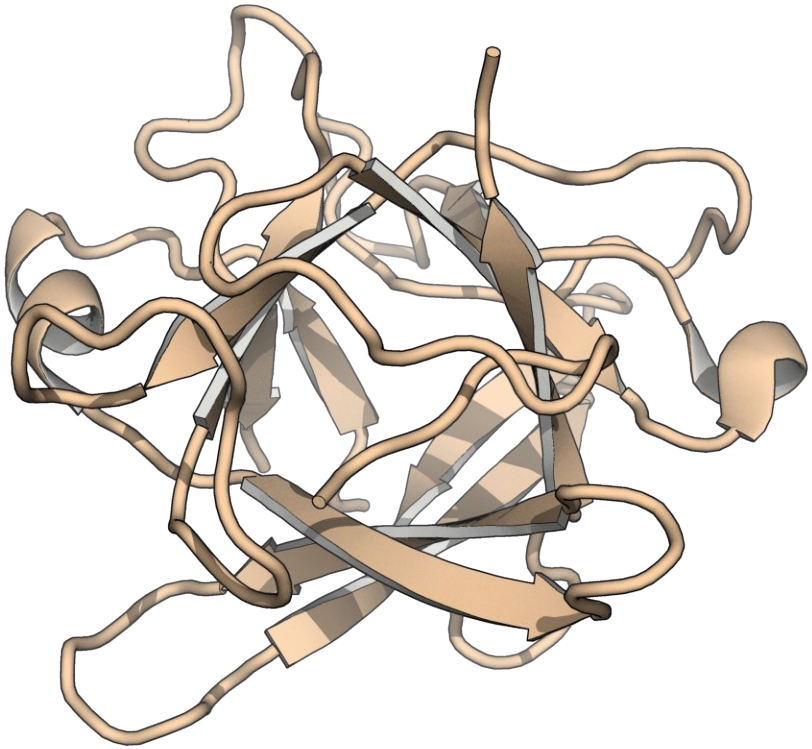
- Есть неплохая парная метрика TM-score
- Давайте позаимствуем иерархические подход у CLUSTALW (MSA)
- Построение UPGMA дерева на матрице попарных расстояний
- Пары начинают собираться по дереву, при помощи Нидлмана-Вунша
  - Используется аугментация данными из структурного выравнивая (скоры, штрафы)
- Таким образом выделяется общее ядро (common core) набора белков
- Строится финальное выравнивание на корне, с использованием обнаруженного общего ядра

Dong, 2018

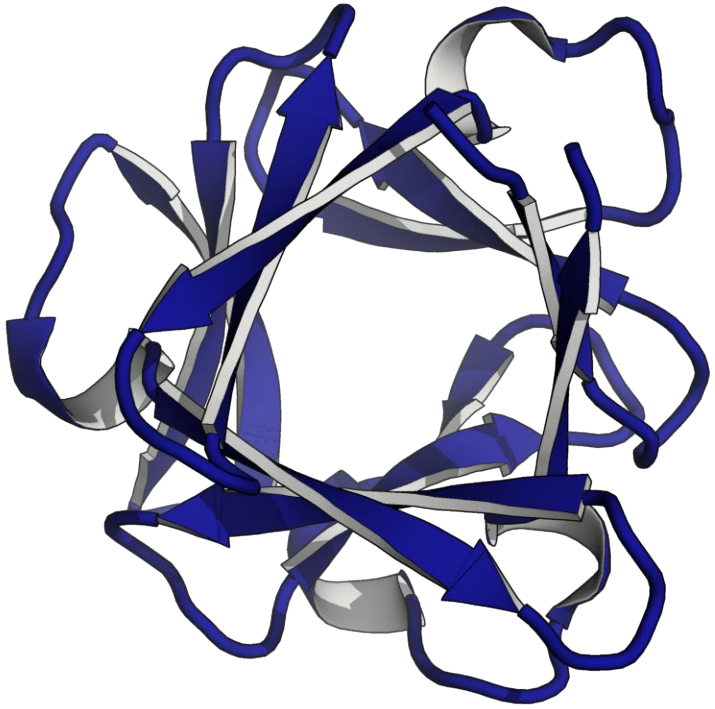
Множественное структурное выравнивание

Другие примеры:

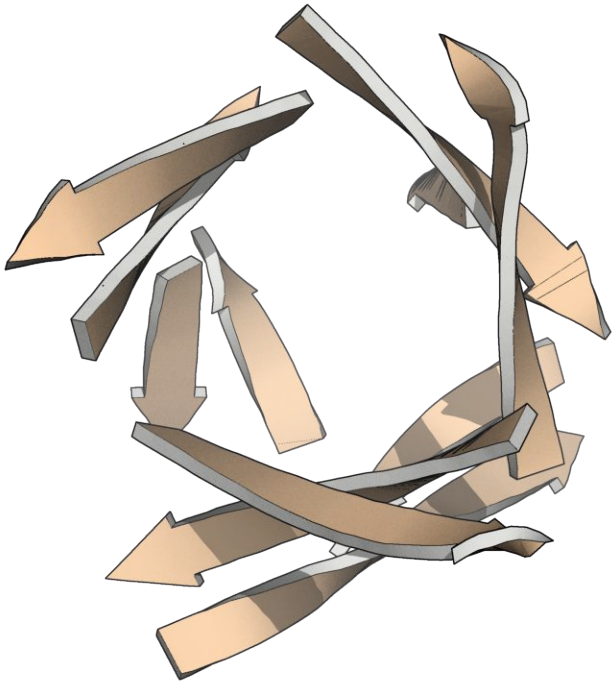
- MUSTANG
- MASS



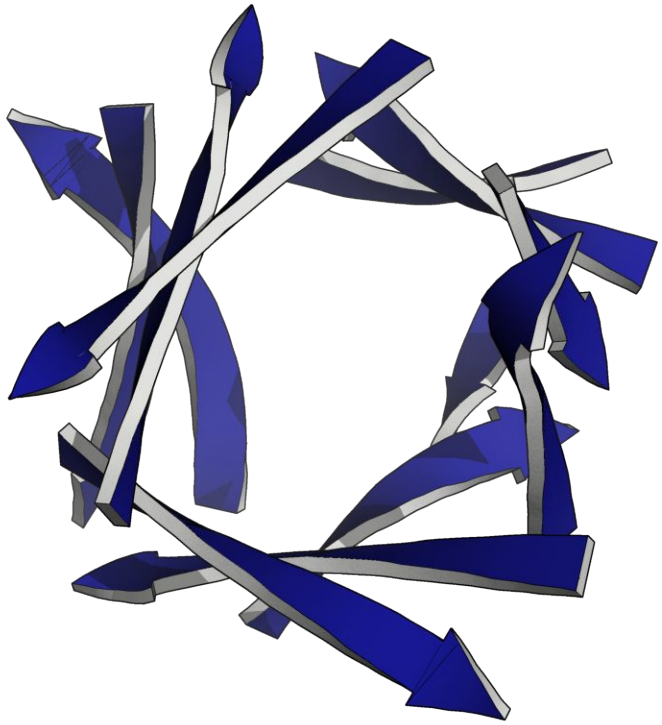
1TIE



4FGF

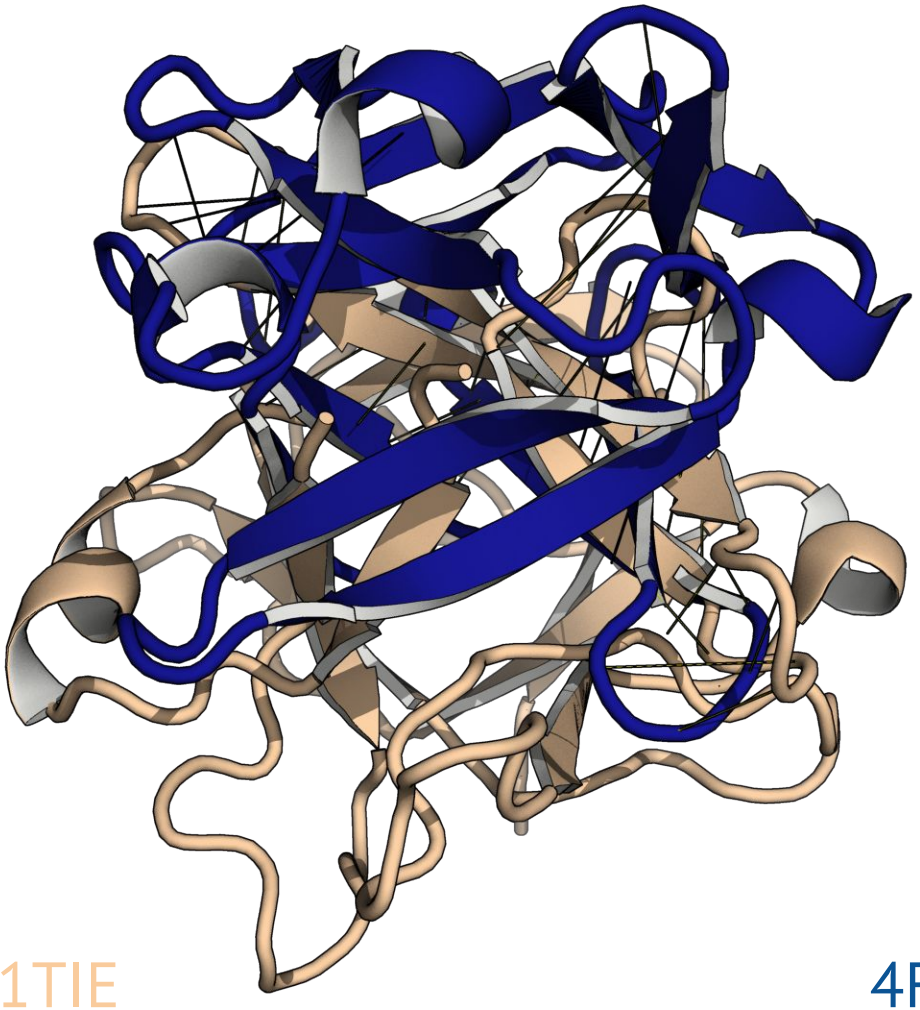


1TIE



4FGF

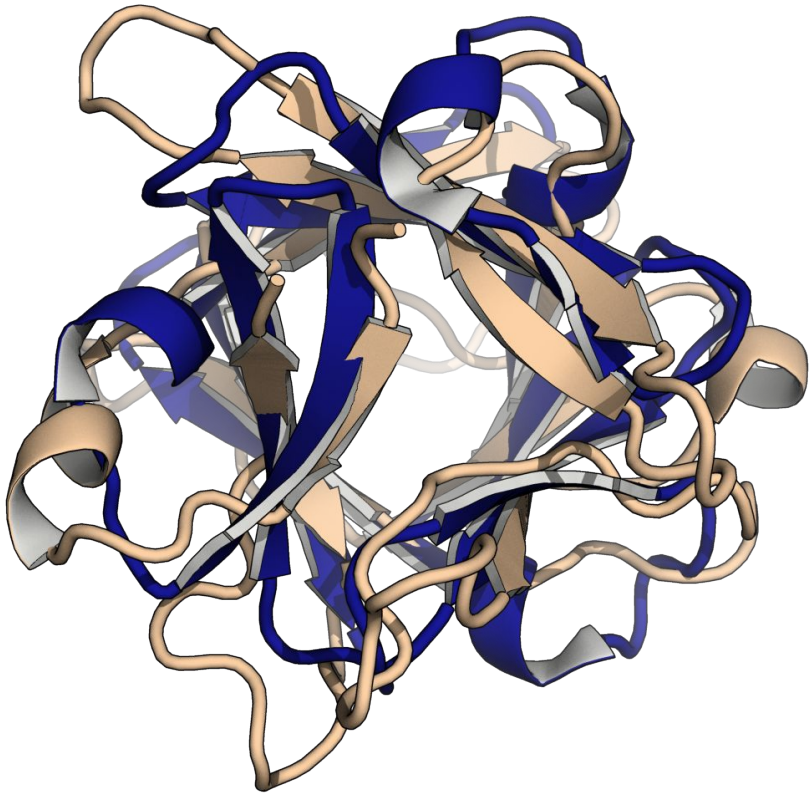
Align

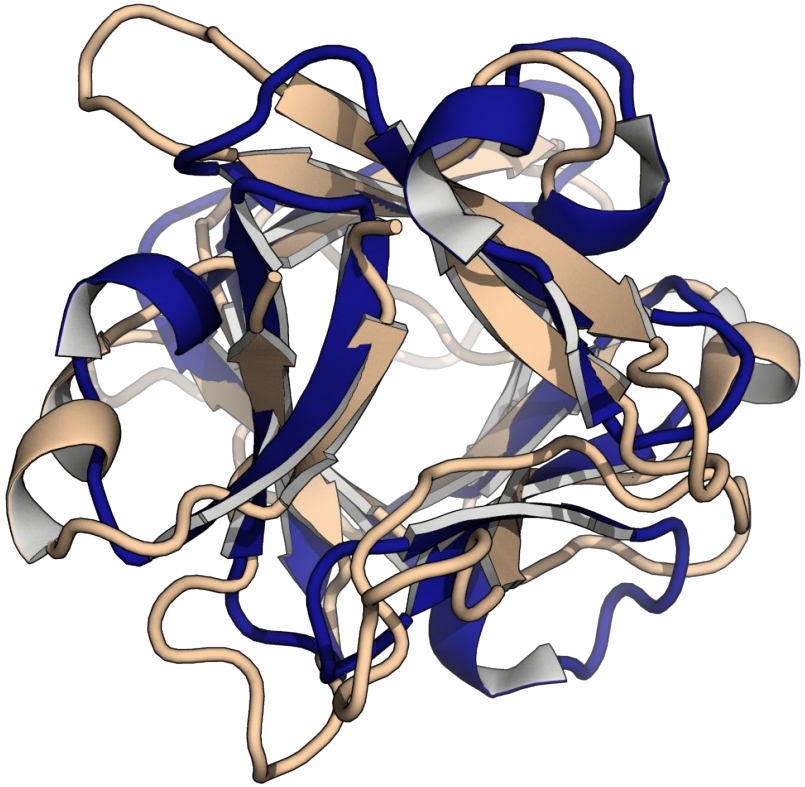


1TIE

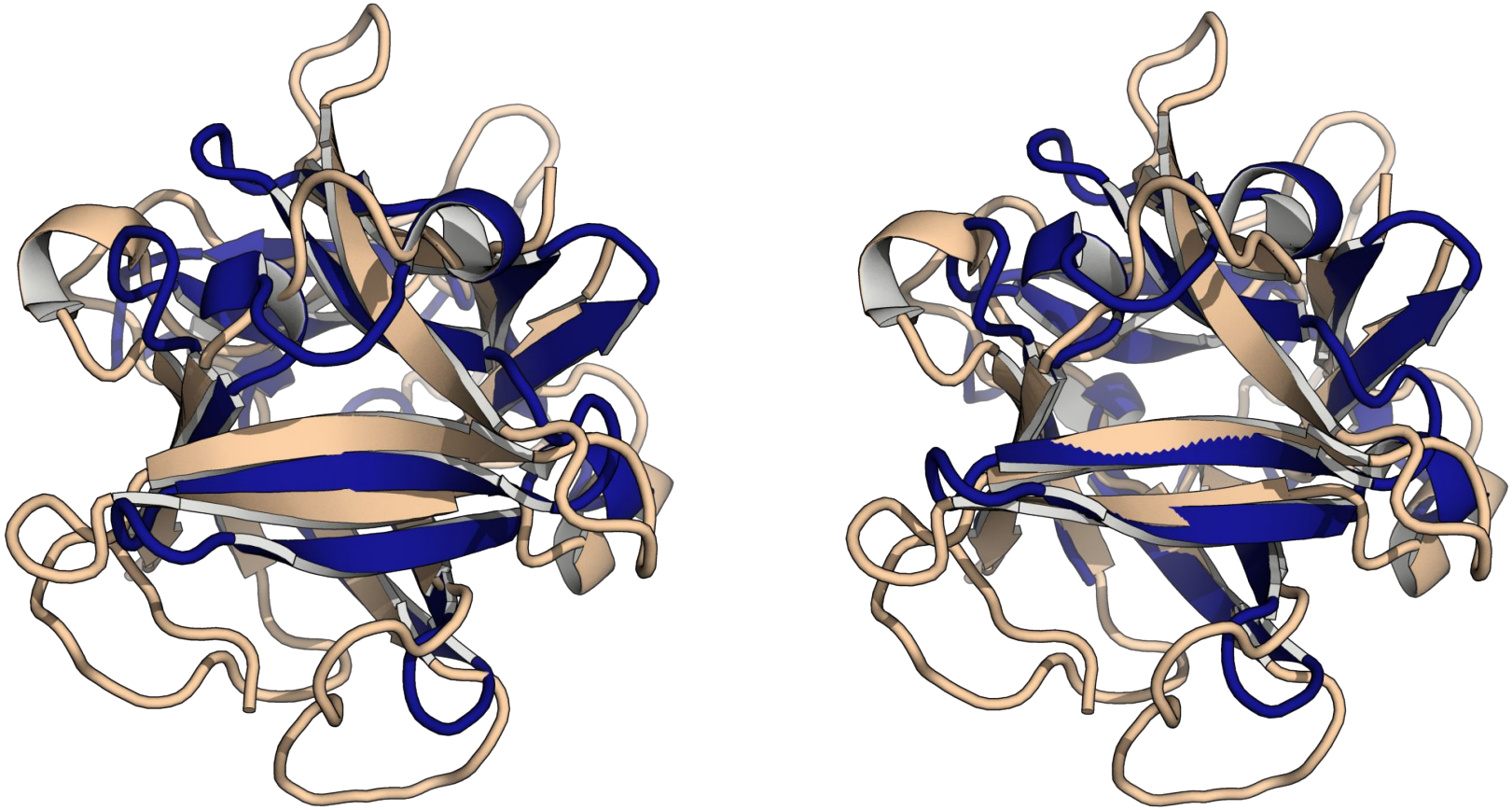
4FGF

Super





Super



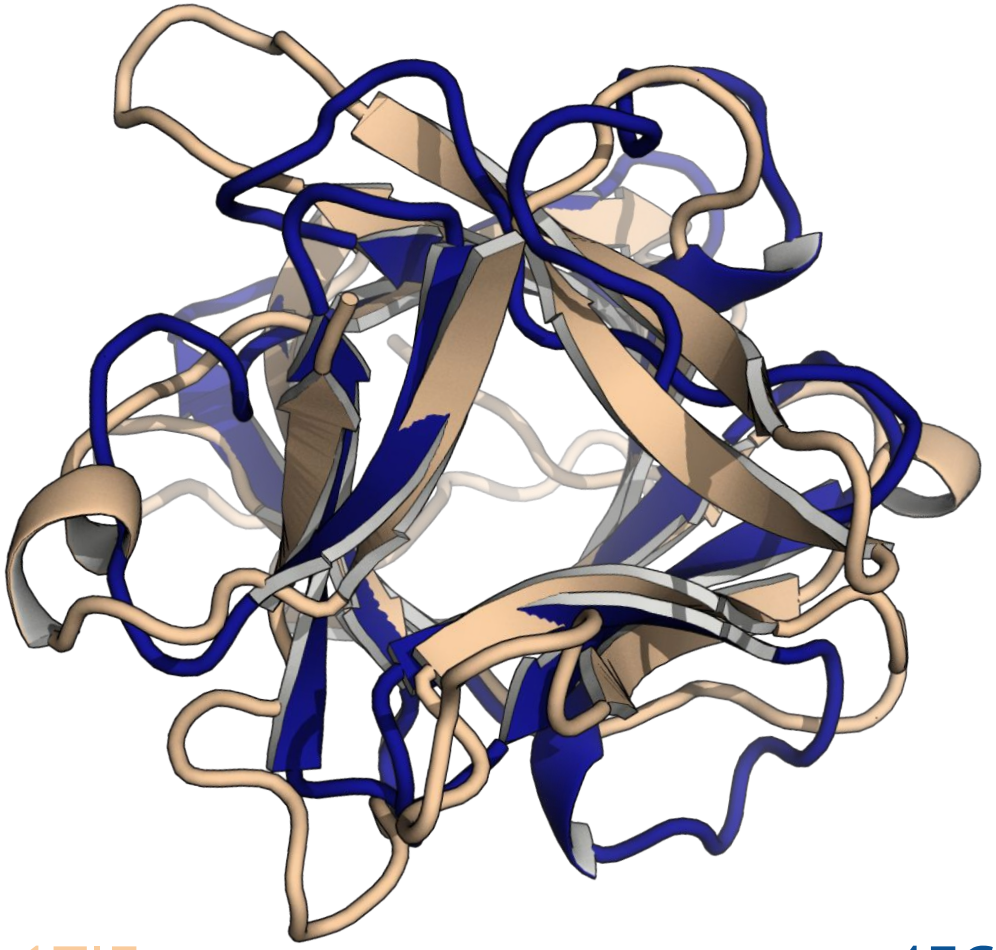
CE (cealign)

1TIE

4FGF

TM-align

(нет в PyMol)



1TIE

4FGF