deconSTRUCT: General Purpose Protein Database Search on the Substructure Level – Supplementary Data

Zong Hong Zhang, Kavitha Bharatham,
Westley A. Sherman, and Ivana Mihalek *

Bioinformatics Institute 30 Biopolis Street, #07-01 Matrix, Singapore 138671

---

running title: deconSTRUCT

Algorithmic details

This section lists in more concrete terms the algorithm described in the "Method" section of the main text.

Direction matching

To check efficiently whether the key directions in the two protein structures can be matched, we replace, in each structure, secondary structure elements (SSEs) by direction vectors in space, while keeping the information about the SSE type (α-helix or β-strand). The protein's structure is then"represented" by a set of unit vectors corresponding to SSE directions (relative to each other):

$$X = \{\vec{x}_i : \|\vec{x}_i\| = 1, i = 1, \cdots, N_x\}, \tag{1}$$

for a protein structure of N_x SSEs. The order of the elements is determined by the order in which SSEs appear in the peptide sequence. Each vector represents one of the two types of structural elements that appear in protein structures: α-helix or β-strand. The information about the type is stored as a corresponding set of indicators

$$S = \{s_i\},\ s_i = \begin{cases} +1 & \text{if the element } i \text{ is an } \alpha\text{-helix} \\ -1 & \text{if the element } i \text{ is a } \beta\text{-sheet} \end{cases},\ i = 1..N_x.$$

* Corresponding author.
Email addresses: zhangzh@bii.a-star.edu.sg (Zong Hong Zhang), kavithab@bii.a-star.edu.sg (Kavitha Bharatham), westleys@bii.a-star.edu.sg (Westley A. Sherman), ivanam@bii.a-star.edu.sg (Ivana Mihalek).

Assuming that we are trying to establish whether a rotation exists that matches X to the representation of the other structure, Y,

$$Y = \{\vec{y}_j : \|\vec{y}_j\| = 1, j = 1, \cdots, N_y\},$$

we can calculate for each pair of directions $(\vec{x}_i, \vec{y}_j)$ the following quantity:

$$D_{ij}(R) = e^{-|\vec{y}_j - R\vec{x}_i|^2/\delta^2}. \tag{2}$$

Here, R is the rotation operator, and δ is an adjustable parameter, measuring the tolerance of the directional match (the larger the δ, the larger the mismatch in the directions that still contribute to the matched score). It is precisely through this parameter that the user of the deconSTRUCT server can specify the required precision of the directional match.

The overall quality of the match is scored by

$$F(R; X, Y) = -\sum_{i=1}^{N_x}\sum_{j=1}^{N_y} f(s_i, s_j) D_{ij}(R). \tag{3}$$

The negative sign here is arbitrary, indicating that $F(R; X, Y)$ will be optimized through minimization with respect to rotation R. The function $f(s_i, s_j)$ is a penalty function for matching SSEs of different types. We experimented with $f(s_i, s_j) = s_i s_j$, and have currently settled for $f(s_i, s_j) = \delta_K(s_i, s_j)$ - with δ_K here being a Kronecker δ, not to be confused with the Gaussian width parameter used above.

In the current implementation the optimization is done by Metropolis Monte Carlo sampling. One practical problem, however, is that searching through the space of all rotations is still computationally impractical. Therefore the algorithm only explores the vicinity of a series of initial guesses for the optimizing rotation R. To find the initial guess, all sequential triplets of SSEs are considered in both structures. The attempt is made to find a rotation that will map a triplet from one structure onto a triplet in the other, such that the following conditions are met: (i) the members of triplets in two structures must match in type, and (ii) the RMSD (between two triples) of distances between the geometric center of one element and the line along the other is required to be smaller than some cutoff (currently $3\mathring{A}$). A rotation minimizing the the angle between the members of such triplets is found (1), and maps between triplets sorted according to how closely the angles match. The top N (currently: 30) maps are used as starting Rs to optimize F, Eq. 3.

Sequential order checking

To enforce the same sequential order of the matched SSEs in the two structures, we reinterpret X and Y as two alignable sequences of elements ("letters") labeled i and j. The letters here are SSEs, and their the similarity is given by $D_{ij}(R_{opt})$. This last quantity, given defined in Eq. 2 and evaluated at R_{opt} that optimizes $F(R; X, Y)$ thus becomes a similarity matrix.

One can than use a pairwise sequence alignment algorithm (2), such as Needleman-Wunsch or Smith-Waterman (deconSTRUCT uses the latter, with -1 as the gap opening penalty, and no penalties for endgaps or gap extension). The alignment procedure optimizes the sum of $D_{ij}(R)$ elements over the pairs $(i, M(i))$ matched in the pairwise matching algorithm

$$T(D) = \sum_{i=1}^{N_x} s_i s_{M(i)} D_{iM(i)}. \tag{4}$$

The sum here is to be understood as running only over i for which $M(i)$, that is the mapping to the other structure, is defined. By retaining only the matched pairs which optimize $T(D)$, deconSTRUCT obtains a good orientation match between the pairs of SSEs, that at the same time complies with the sequential ordering in both structures.

Space layout checking

The problem with the comparison procedure so far is that SSEs laying at completely different positions in space, relative to the remaining bulk of the protein, can still point in very similar directions. To get rid of spurious "matches" of the sort, the deconsTRUCT algorithm next calculates the relative positions of the SSEs in the rotated position. Similarly to the step (ii) in selecting the seed triple of SSEs, now the distances of geometric centers of all SSEs to the to the lines determined by the directions of the seed are calculated. They should be comparable in both structures (the cutoff RMSD is again taken to be $3\mathring{A}$), otherwise the mapping is ignored.

Alignment of matched SSEs on the level of backbone atoms

Up to this point we have associated with SSEs only their unit direction vectors. Here we add to the description a vector describing the direction and the magnitude of translation from the geometric center of all of the SSEs with a match in the other protein to the geometric center of this particualr SSE:

$$\vec{t}_{xi} = \vec{c}_{xi} - \frac{1}{N_x} \sum_{j=1}^{N_x} \vec{c}_{xj}. \tag{5}$$

These vectors, $\vec{t}_{xi}$, are rotated using the same R_{opt} used to optimize the direction match.

To get the initial match on the backbone level, the origin of the system is moved to the geometric center of all of the matched SSEs in X, and the vector corresponding to the position of each C_α in SSE represented by x_i is rotated by R_{opt} and translated by $\vec{t}_{xi}$ (note this vector is different for each SSE.)

Similar operation is performed on the C_αs in Y, only without rotation.

Once the approximate rotation and translation is "felt out," a similarity matrix is con-

structed, with the effective "similarity" between C_α atoms a_1 and a_2 in space measured as

$$\sigma_1(a_1, a_2) = \begin{cases} e^{-d(a_1,a_2)/d_0} & \text{if } a_1 \text{ and } a_2 \text{ belong to matched SSEs} \\ -1 & \text{otherwise.} \end{cases} \quad (6)$$

In this equation d is simply the geometric distance between atoms, and d_0 is a parameter currently set to $5\mathring{A}$. (σ_1 carries the index 1 to suggest there will be another round of the optimization, using σ_2, see below). The closest matching atoms are then determined using dynamic programming, exactly as in looking for $T(D)$, Eq. 4 above (currently, in deconSTRUCT, using Smith-Waterman with gap opening penalty of -0.2, gap extension penalty of -0.1, end no endgap penalty). Once the matching pairs of atoms are known, standard techniques can be used to find the optimal rotation and translation (1).

Alignment extension

It turns out in practice that the transformations which superimpose several of the main geometry defining SSEs also work for the neighboring loops, and perhaps even partially for the neighboring SSEs that could not be matched precisely, in terms of their directions, between the two structures. Therefore, the current implementation of deconSTRUCT attempts the extension of the alignment to these neighboring pieces of structure. A new similarity matrix is constructed $\sigma_2(a_1, a_2)$, this time with a somewhat more involved rule of assignment: $\sigma_2(a_1, a_2) = \exp(-d(a_1, a_2)/d_0)$ if a_1 and a_2 belong to matched SSEs, or to the same type (helix, strand, or, in this case, loop) flanking the matched SSEs by no more than two SSEs on either side. In all of the remaining cases the value of $\sigma_2(a_1, a_2)$ is set to -1. An improved rotation and translation are constructed as before (with the same parametrization), and the following score reported as the alignment score for the two structures:

$$A(\sigma_2) = \sum_{i=1}^{N_x} \sigma_2(a_i a_{M(i)}) = \sum_{i=1}^{N_x} \exp(-d(a_i, a_{M(i)})/d_0), \quad (7)$$

where i and $M(i)$ here stand for the atom index i in one structure, and its match, $M(i)$, in the other. This score is qualitatively similar to the score proposed by Subbiah *et al.* (3),

$$S \propto \sum_{i=1}^{N_x} \frac{1}{1 + \left(\frac{d(a_i, a_{M(i)})}{d_0}\right)^2}, \quad (8)$$

without the gap penalty suggested later (4).

References

[1] Karney, C. (2007) Quaternions in molecular modeling *Journal of Molecular Graphics and Modelling* **25(5)**, 595–604.

[2] Durbin, R. (1998) Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids, Cambridge University Press, .

[3] Subbiah, S., Laurents, D., and Levitt, M. (1993) Structural similarity of DNA-binding domains of bacteriophage repressors and the globin core *Curr. Biol* **3(3)**, 141–148.

[4] Kolodny, R., Koehl, P., and Levitt, M. (2005) Comprehensive Evaluation of Protein Structure Alignment Methods: Scoring by Geometric Measures *Journal of Molecular Biology* **346(4)**, 1173–1188.