Amino Acids (2010) 39:417–425
DOI 10.1007/s00726-009-0457-y

ORIGINAL ARTICLE

SANA: an algorithm for sequential and non-sequential protein structure alignment

Lin Wang · Ling-Yun Wu · Yong Wang · Xiang-Sun Zhang · Luonan Chen

Received: 23 September 2008 / Accepted: 19 December 2009 / Published online: 2 February 2010

Abstract Protein structure alignment algorithms play an important role in the studies of protein structure and function. In this paper, a novel approach for structure alignment is presented. Specifically, core regions in two protein structures are first aligned by identifying connected components in a network of neighboring geometrically compatible aligned fragment pairs. The initial alignments then are refined through a multi-objective optimization method. The algorithm can produce both sequential and non-sequential alignments. We show the superior performance of the proposed algorithm by the computational experiments on several benchmark datasets and the comparisons with the well-known structure alignment algorithms such as DALI, CE and MATT. The proposed method can obtain accurate and biologically significant alignment results for the case with occurrence of internal repeats or indels, identify the circular permutations, and reveal conserved functional sites. A ranking criterion of our algorithm for fold similarity is presented and found to be comparable or superior to the *Z*-score of CE in most cases from the numerical experiments. The software and supplementary data of computational results are available at http://zhangroup.aporc.org/bioinfo/SANA.

Keywords Protein structure alignment · Fold comparison · Circular permutation · Functional sites

Abbreviations

PDB	Protein data bank
RMSD	Root mean square distance
AFP	Aligned fragment pair
SANA	Protein structure alignment based on sequence neighborhood alignment

Background

Comparison of protein structures remains a fundamental and largely unsolved problem. Determination of equivalently aligned residues of two protein chains is a necessity in identifying homologous proteins, studying fold changes in evolution of protein structures, and revealing structurally conserved residues for protein function, and so on. Currently most protein structure alignment algorithms are either distance-based or coordinate-based. The distance-based class includes aligned fragment pair (AFP) chaining methods (CE, Shindyalov and Bourne 1998; FATCAT, Ye and Godzik 2003 and Matt, Menke et al. 2008), distance matrix alignment methods (DALI, Holm and Sander 1993 and MatAlign, Aung and Tan 2006), contact pattern alignment method (Vorolign, Birzele et al. 2007) and secondary structure elements alignment methods (SSM,

L. Wang
Computer Science and Information Engineering College,
Tianjin University of Science and Technology,
Tianjin 300222, China
e-mail: juanzi_1982_49@126.com

L. Wang · L.-Y. Wu · Y. Wang · X.-S. Zhang (✉)
Academy of Mathematics and Systems Science,
Chinese Academy of Sciences, Beijing 100190, China
e-mail: zxs@amt.ac.cn

L. Chen (✉)
Key Laboratory of Systems Biology,
SIBS-Novo Nordisk Translational Research Centre
for PreDiabetes, Shanghai Institutes for Biological Sciences,
Chinese Academy of Sciences, 320 Yue-Yang Road,
Shanghai 200031, China
e-mail: lnchen@sibs.ac.cn

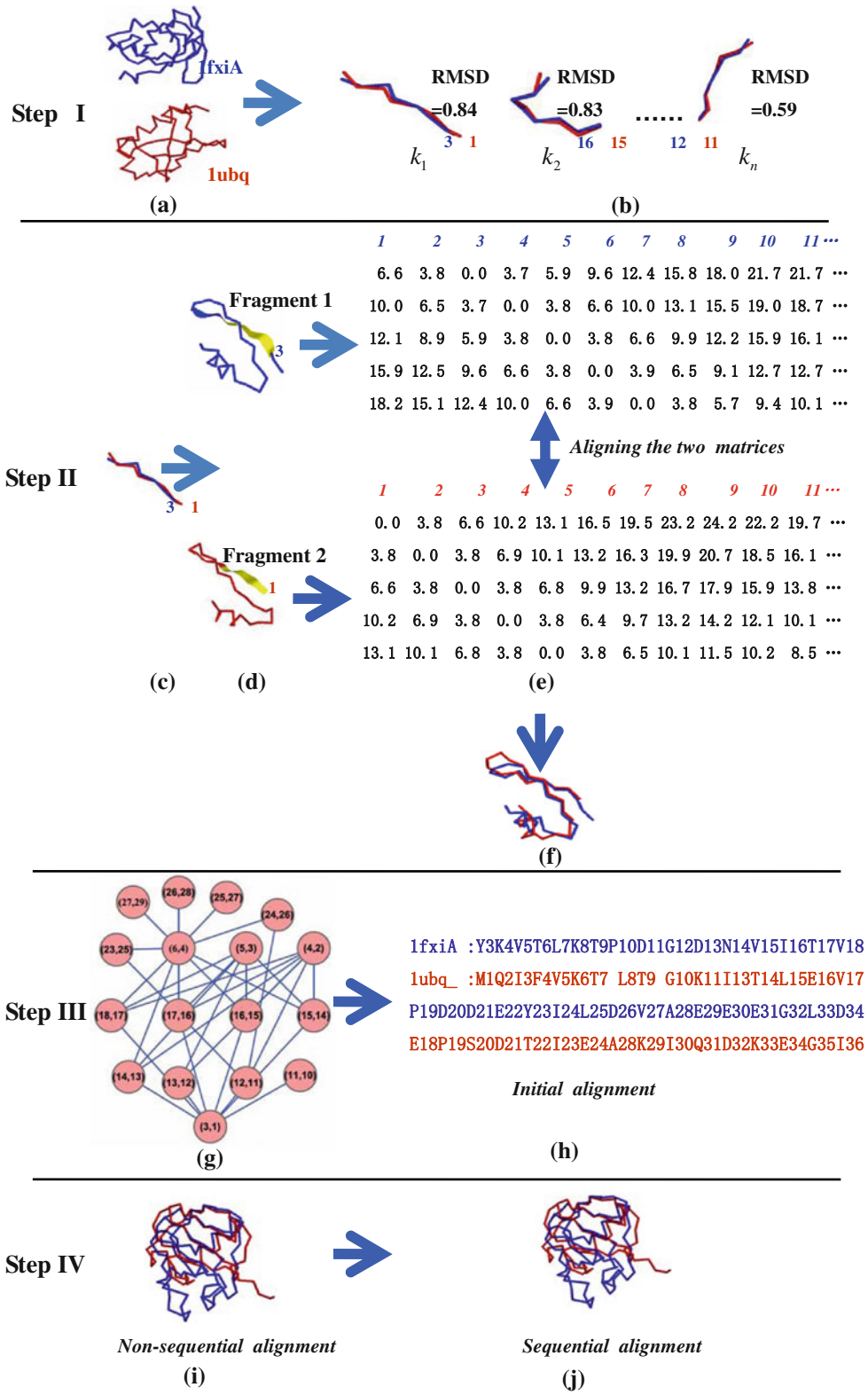
Krissinel and Henrick 2004). The coordinate-based class includes transformation and match iteratively optimizing method (SAMO, Chen et al. 2006) and geometric hashing representation method (C_α-match, Bachar et al. 1993; MultiProt, Shatsky et al. 2004). However, these algorithms sometimes fail to create consistent results for some challenging examples, especially when there are repetitions, indels, local conformational changes and circular permutations. This fact indicates that there is still much room for improvement (Mayr et al. 2007). Here, we provide a new algorithm called SANA (Structure Alignment via sequence Neighborhood Alignment) which has sequential (i.e. consistent with the sequential ordering) and non-sequential (i.e. inconsistent with the sequential ordering of the residues along the backbone) options for protein structure alignment. We show that SANA in the sequential case ($SANA_s$) outperforms other commonly used methods, e.g. DALI and CE, and the recent method MATT in terms of the quality of alignments on several benchmark datasets. More importantly SANA in non-sequential case ($SANA_n$) is adopted to alignment of proteins having circular permutations which cannot be detected by structure alignment methods that keep with the sequence order.

As the first step, SANA uses an AFP chaining method to consider two AFPs' geometric compatibility by comparing aligned residues in two corresponding sequence neighborhood pairs. In each sequence neighborhood pair, the aligned residues are detected by means of dynamic programming. The two AFPs are considered as geometrically compatible if their corresponding sequence neighborhood pairs have several identically aligned residue pairs. Two AFPs are chained to construct a network if they are geometrically compatible, sequential and non-overlapped. Then an algorithm to find connected components is adopted to obtain several connected components according to the number of nodes in a descendent order. The above results are the alignments in core regions. As the second step of SANA, global alignments are computed via a multi-objective optimization method, which can generates both non-sequential and sequential alignment results. The multi-objective programming problem is to maximize the equivalently aligned residues meanwhile minimize the root mean square distance (RMSD), and is converted to a single-objective programming by introducing a trade-off parameter (Chen et al. 2006). At last we select the alignments with the minimal objective function values as the final alignment results for sequential case and non-sequential case, respectively.

Using two well-established benchmark datasets, we compared our algorithm $SANA_s$ to DALI, CE and MATT. We show that $SANA_s$ obtains competitive alignment quality based on evaluating RRMSD values (Betancourt and Skolnick 2001) and Q-score values (Krissinel and Henrick 2004). In addition, we tested $SANA_s$ on a larger dataset which has in total 1,268 pairs of domains, among which two pairs of proteins are sampled from a group of SABmark (Van Walle et al. 2005). SABmark includes superfamily and twilight zone sets which have low to intermediate and low to low similarity, respectively. Compared with CE and MATT, SANA finds better alignments in terms of RRMSD values and Q-score values for most pairs in the dataset.

Results from comparison of proteins with repetitions or indels show that $SANA_s$ can obtain correct and biologically significant alignments for these difficult pairs. Compared to Z-score of CE, $SANA_s$ shows similar or higher accuracy in detecting the protein fold similarity in terms of the percentage of equivalently aligned residues. Finally, as one of the major advantages, we show that $SANA_n$ can detect circular permutations and identify conserved functional sites.

Fig. 1 The schematic illustration of new method SANA. (Step I) ▶ Firstly, for two protein structures 1fxiA (*blue line*) and 1ubq (*red line*) in subgraph (**a**), all AFPs shown in subgraph (**b**) are obtained if their RMSD values are less than a certain threshold (e.g. 1.5 Å). The *numbers* shown on the termini are residue indices. For example, for AFP k_1 we have $s_1(k_1) = 3$, $s_2(k_1) = 1$. (Step II) Secondly, the sequence neighborhoods of each AFP are aligned by dynamic programming. The sequence neighborhoods of AFP k_1 in subgraph (**c**) are shown in subgraph (**d**). The fragments shown as ribbon shape in subgraph (**d**) are the fragments of AFP k_1 1 and 2. For fragment 1 we have neighborhood radius $R = 30$, neighborhood length $l_1(k_1) = 33$, neighborhood starting position $m_1(k_1) = 1$, neighborhood ending position $n_1(k_1) = 33$. Similarly for fragment 2 we have $l_2(k_1) = 31$, $m_2(k_1) = 1$, $n_2(k_1) = 31$. The first five residues in each fragment of AFP k_1 are colored by *yellow*. For each fragment we compute the Euclidean distances of all residues in its sequence neighborhood to the first five residues, respectively. So we obtain two matrices shown in subgraph (**e**), one for each fragment. Then the two matrices are aligned by dynamic programming and the alignment result is used to represent the alignment of the two sequence neighborhoods which is shown in subgraph (**f**). (Step III) Thirdly, a network shown in subgraph (**g**) is constructed by chaining two AFPs if they are sequential, non-overlapped and have several identical aligned residues in their neighborhood alignments. For example, for AFP k_1 and k_2, we have (1) $(s_1(k_1) - s_1(k_2)) \times (s_2(k_1) - s_2(k_2)) = (3 - 16) \times (1 - 15) > 0$, (2) $|s_1(k_1) - s_1(k_2)| = 13 > 8$ and $|s_2(k_1) - s_2(k_2)| = 14 > 8$, and (3) the alignment profiles of k_1 and k_2 are $v_1(1{:}33) = [-1, -1, 1{:}6, -1, 8, 10{:}21, -1, -1, -1, 23, 26, -1, 27{:}29, 31, -1]$, $v_1(1{:}33) = [-1, -1, 1{:}7, 10, -1, 11{:}21, -1, -1, -1, 25, -1, -1, 27{:}28, 30{:}32]$, respectively, there are 20 (>9) identical aligned residues, so we connect the AFPs k_1 and k_2. The label of each node in the network is the product of starting residue indices of two fragments of an AFP. The detailed aligned residues obtained from the connected network are shown in subgraph (**h**). It is the initial alignment with Aligned = 32 and RMSD = 2.38. (Step IV) The initial alignment is refined through a multi-objective optimization method and results in the final non-sequential alignment with Aligned = 70 and RMSD = 2.45 shown in subgraph (**i**). Furthermore, we implement dynamic programming to obtain sequential alignment with Aligned = 61 and RMSD = 2.57 shown in subgraph (**j**) (color figure online)



Methods

In this paper, we describe a new approach to compare two protein structures in both sequence order-independent and sequence order-dependent way using a connected component finding algorithm and a multi-objective optimization method. The framework of our methodology SANA is presented in Fig. 1, which shows the main steps of SANA

with an example of aligning two proteins 1fxiA and 1ubq. We will describe the key steps of our method point by point as follows: notations and symbols, alignment of sequence neighborhoods, choice of initial alignments, refinement of the initial alignments and derivation of order-dependent alignments.

Notations

Given two protein structures, a match of two fragments of fixed size L (e.g. 8), in the two proteins, respectively, is denoted as an AFP if the RMSD of their C_α atoms is less than a certain threshold (e.g. 1.5 Å). Each AFP describes one way of superimposing one protein on the other.

We use the following notations throughout the paper:

- L_i is the length of a protein structure S_i for $i = 1$ or 2.
- $r_t^i \in S_i$ corresponds to the tth residue in the protein S_i where residue index t is from 1 to L_i.
- The residue index of the beginning of the fragment i of the AFP k is denoted as $s_i(k)$.
- A sequence neighborhood of the fragment i of the AFP k is defined as a fragment of sequence which centers on the starting position of the fragment t (=$s_i(k)$) with neighborhood radius R (e.g. 30), i.e. (r_{t-R}, …, r_{t-1}, r_t, …, r_{t+7}, r_{t+8}, …, r_{t+R}).
- $m_i(k)$ and $n_i(k)$ are the residue indices of the beginning and ending of the sequence neighborhood of the fragment i of the AFP k, respectively. Specifically $m_i(k) = 1$ if $(s_i(k) - R) < 1$, and $n_i(k) = L_i$ if $(s_i(k) + R) > L_i$.
- The length of sequence neighborhood of fragment i of the AFP k is denoted as $l_i(k)$.

Aligning sequence neighborhoods of an AFP

For an AFP there are two sequence neighborhoods, one for each fragment. We use the first five residues in each fragment of an AFP as centers and compute the Euclidean distances of all residues in sequence neighborhood of the fragment to the five residues, respectively, i.e. for residues $r_{t_1}^i$, $m_i(k) \le t_1 \le n_i(k)$, we calculate their distances to residues $r_{t_2}^i$, where $s_i(k) \le t_2 \le (s_i(k) + 4)$. So for each fragment i we get a matrix which contains five rows and $l_i(k)$ columns, where each column is the distances of the five residues to a residue in the neighborhood. To align two sequence neighborhoods of an AFP, we consider these two matrices corresponding to the two fragments of the AFP. Two matrices M_1 and M_2 are compared using dynamic programming, and we do not consider the penalty of gaps. Generalizing the function in MatAlign (Aung and Tan 2006) to determine the degree of match between two $C_\alpha - C_\alpha$ distance values, we use function Match(.,.) to determine the degree of match between any two columns $M_1(.,t_1)$ and $M_2(.,t_2)$ which are extracted from two matrices M_1 and M_2, respectively. For convenience, we record $d_1 = M_1(.,t_1)$ and $d_2 = M_2(.,t_2)$.

$$\text{Match}(d_1,d_2) = \begin{cases} T \Big/ \left(\sum_{h=1}^{5} |d_1(h) - d_2(h)| + T\right) & \text{if } \sum_{h=1}^{5} |d_1(h) - d_2(h)| < D \\ 0 & \text{otherwise} \end{cases}$$

where T is a score adjusting weight and D is a threshold of the sum of differences between corresponding distances. We take $T = 2$, $D = 4.3$ Å. After executing the dynamic programming to maximize the objective function: $\sum_{t_1=m_1(k)}^{n_1(k)} \sum_{t_2=m_2(k)}^{n_2(k)} \text{Match}(M_1(.,t_1), M_2(.,t_2))$, we get aligned residues in two sequence neighborhoods. For convenience, we represent the alignment as a L_1-dimensional vector v:

$$v(t_1) = \begin{cases} t_2 & \text{if } r_{t_1}^1 \text{ and } r_{t_2}^2 \text{ are aligned residues in two sequence neighborhoods} \\ -1 & \text{otherwise} \end{cases}$$

Constructing network and finding initial alignments

At the next stage we chain two AFPs based on the alignments of the sequence neighborhoods of each AFP. For some AFPs, their corresponding sequence neighborhoods have less aligned residues, so we filter out these AFPs when the number of aligned residues is less than a threshold (e.g. 18). Each remaining AFP is viewed as a node, and two nodes are joined if two AFPs are *sequential*, *non-overlapped*, and *geometrically compatible*, i.e. for two AFPs k_1 and k_2, they satisfy:

1. $(s_1(k_1) - s_1(k_2)) \times (s_2(k_1) - s_2(k_2)) > 0$,
2. $|s_1(k_1) - s_1(k_2)| \ge 8$ and $|s_2(k_1) - s_2(k_2)| \ge 8$,
3. at least N (e.g. 9) residue indices t_i, $i \ge N$, such that $v_1(t_i) = v_2(t_i)$ and $v_1(t_i) \ne -1$.

The condition 1 indicates the initial residues of the two AFPs keep the sequence order. The condition 2 requires the residues in the two AFPs not overlapped. The condition 3 means that the alignment profiles of the two AFPs have at least N identical aligned residues. In this way, a network is constructed. We use the random-mate algorithm (Gazit 1991) to find connected components of the graph, and record at most four connected components according to the number of nodes in a descendent order. Then all AFPs in each connected component are assembled as one initial alignment represented as a vector described above. Because what we find are connected components, in some cases there are initial alignments in which aligned residues are not consistent with the sequence order, we adjust the alignments by removing the non-sequential aligned resides so that the initial alignments are sequential. For example, if $v(t_1) \ge v(t_2)$ and $t_1 < t_2$, we let $v(t_i) = -1$.

Refining initial alignments

Each initial alignment is refined through solving the following multi-objective programming which does not consider sequence-order constraint.

$$\min \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} s_{ij} \left(\left| A + RX_i - Y_j \right|^2 - \lambda^2 \right) \quad (1)$$

$$\text{s.t.} \sum_{i=1}^{n_x} s_{ij} \leq 1 \quad \text{for} \quad j = 1, \ldots, n_y \quad (2)$$

$$\sum_{j=1}^{n_y} s_{ij} \leq 1 \quad \text{for} \quad i = 1, \ldots, n_x \quad (3)$$

where X_i, Y_j are the coordinates of the ith residue and jth residue of two protein structures S_1 and S_2, n_x and n_y are the numbers of residues in the two proteins, and λ is a parameter. Notice that the assignment variables s_{ij} are to determine the correspondence between residues in the two proteins, A and R are, respectively, translation and rotation variables of protein S_1. The objective function aims to maximize the aligned residues meanwhile to minimize the total square distances of the aligned residues, and the parameter λ is used to balance these two objectives. The final protein structure alignment is obtained by assigning each initial alignment to the variables s and iteratively solving the programming until convergence (see detail in Chen et al. 2006). We uniformly set $\lambda = 6.0$ in our experiments. As a result, several non-sequential alignments associated with transformations are generated, and then we choose the alignment which has the minimal objective function value as a final non-sequential alignment.

Deriving sequence order-dependent alignments

After refining each initial alignment through multi-objective programming we also obtain the corresponding solutions for coordinate transformation (translation and rotation). For these transformations, we first substitute each of them to the objective function (1), then implement dynamic programming to get a sequential correspondence of two proteins. In this way, we obtain several sequence order-dependent alignments, and further select the alignment with the minimal objective function value as a final sequential alignment result.

Results and discussion

We assess the alignment quality of SANA by comparing with other structure alignment algorithms. The validation on three benchmark datasets which include around 1,500 pairs of protein structures are reported in "Comparing with popular algorithms on three datasets". "Exploring precise alignments for pairs of proteins containing repetitions or indels" section shows that SANA can find correct alignments for two proteins including internal repeats or indels. In "Detecting fold similarity among proteins" section, we use the percentage ratio of aligned residues to the average length of two proteins to rank the similarity of two structures, and demonstrate that SANA can gain competitive accuracy in terms of fold similarity recognition in comparison with the Z-score from CE in many cases. In "Finding circular permutations and conserved functional sites" section, we test the ability of SANA to detect circular permutations and identify structurally conserved functional sites.

Comparing with popular algorithms on three datasets

We benchmark the performance of SANA$_s$ against three algorithms DALI (Holm and Sander 1993), CE (Shindyalov and Bourne 1998) and MATT (Menke et al. 2008) using Fischer's (Fischer et al. 1996) benchmark dataset and Novotny's (Novotny et al. 2004) benchmark dataset, and further show its superior performance on a larger dataset from SABmark (Van Walle et al. 2005).

Fischer's and Novotny's datasets are two popular benchmark datasets for testing protein structure alignment programs, and they contain 68 and 153 pairs of protein structures, respectively. SANA$_s$, DALI, CE and MATT are applied to align the protein pairs in these two datasets to check the performance of SANA. We select the top alignments based on the native scores of each method to compare those structure alignment methods. As shown in Table 1, our method finds better alignment on average, since the average RMSD values and average RRMSD values are considerably better on both two datasets. Here, RRMSD is the RMSD relative to the approximate average RMSD between two random protein fragments of the same length (Betancourt and Skolnick 2001). And it is a measure to evaluate the similarity significance of core regions compared with random fragment pairs of the same length. The smaller the RRMSD value is, the more significant the similarity of the core regions is. To consider the two criteria, e.g. the number of aligned residues and RMSD, simultaneously, we use a Q-score defined as $Q = \text{Aligned}^2/\{[1 + (\text{RMSD}/R_0)^2]L_1L_2\}$ (Krissinel and Henrick 2004) to assess the quality of an alignment, where *Aligned* is the number of aligned residues, R_0 is an empirical parameter of 3.0 Å. Obviously, the higher Q-score value, the better, in general, of the alignment. Compared with DALI, CE and MATT, SANA results in higher average Q-score values on the two datasets. To further investigate the alignment quality of SANA, Table 2 shows the detailed Q-score value differences between

Table 1 Performance comparison on Fischer's and Novotny's datasets, and dataset from SABmark

	aveAligned	aveRMSD	aveRRMSD	ave*Q*-score
Fischer's dataset (68 pairs)				
67 pairs for which DALI has results				
$SANA_s$	150	2.21	0.15	0.37
DALI	155	2.77	0.18	0.33
MATT	152	2.87	0.19	0.33
67 pairs for which CE has results				
$SANA_s$	148	2.20	0.15	0.37
CE	154	2.85	0.19	0.33
MATT	151	2.86	0.19	0.33
Novotny's dataset (153 pairs)				
144 pairs for which DALI has results				
$SANA_s$	163	1.97	0.12	0.45
DALI	175	2.85	0.17	0.42
CE	177	2.67	0.16	0.43
MATT	175	3.08	0.17	0.43
All pairs in the dataset				
$SANA_s$	157	2.04	0.13	0.42
CE	170	2.82	0.17	0.41
MATT	167	3.09	0.17	0.40
SABmark (1,268 pairs)				
$SANA_s$	113	2.21	0.18	0.37
CE	118	2.94	0.23	0.34
MATT	115	2.64	0.21	0.34

SANA and other methods. In each dataset, SANA outperforms other methods for at least 10 pairs by larger than 0.05 *Q*-score points, although the *Q*-score value differences are less than 0.05 for most pairs.

SABmark consists of superfamily and twilight zone sets. The superfamily set contains 425 groups representing structures at the superfamily level of the SCOP hierarchy. The twilight zone set contains 209 groups representing structures at the SCOP fold level which has the lowest structural similarity. All groups consist of at least three structures. For each group, we sample two pairs of structures and there are total 1,268 pairs in the final dataset. The statistical results derived from $SANA_s$, CE and MATT on the dataset are shown in Table 1. Compared with CE and MATT, SANA's average number of aligned residues is lower, but SANA has a considerably better RMSD and RRMSD, and a higher average *Q*-score. Table 2 lists the number of pairs from the dataset which their *Q*-score value differences between SANA and other methods fall into certain intervals. Compared with CE and MATT, in each difference level, SANA has more pairs having higher *Q*-score values, and it shows the superior quality of the underlying alignment generated by SANA.

These implementations from the several datasets show that SANA in sequential case can find similar or better alignment results than DALI, CE and MATT for most pairs.

Table 2 The number of pairs which *Q*-score value differences between SANA and other methods on Fischer's, Novotny's datasets and dataset from SABmark fall into specific intervals

	≤ -0.1	$(-0.1, -0.05]$	$(-0.05, 0)$	$[0, 0.05)$	$[0.05, 0.1)$	≥ 0.1
Fischer's dataset (68 pairs)						
Q(SANA*s*)–*Q*(DALI)	0	0	2	54	9	2
Q(SANA*s*)–*Q*(CE)	0	0	2	47	15	3
Q(SANA*s*)–*Q*(MATT)	0	0	2	47	12	7
Novotny's dataset (153 pairs)						
Q(SANA*s*)–*Q*(DALI)	0	6	9	104	20	5
Q(SANA*s*)–*Q*(CE)	0	1	28	111	11	2
Q(SANA*s*)–*Q*(MATT)	0	4	16	103	23	7
	≤ -0.2	$(-0.2, -0.15]$	$(-0.15, -0.1]$	$(-0.1, -0.05]$	$(-0.05, 0)$	
SABmark (1268 pairs)						
Q(SANA*s*)–*Q*(CE)	0	4	8	28	99	
Q(SANA*s*)–*Q*(MATT)	0	1	7	16	115	
	[0,0.05)	[0.05, 0.1)	[0.1, 0.15)	[0.15, 0.2)	≥ 0.2	
Q(SANA*s*)–*Q*(CE)	886	191	36	13	3	
Q(SANA*s*)–*Q*(MATT)	886	193	42	6	2	

Exploring precise alignments for pairs of proteins containing repetitions or indels

Repetition is a common feature of a protein structure. Because SANA considers two AFPs' geometric compatibility via comparing the aligned residues in two sequence neighborhood pairs, we examine SANA$_s$ using six pairs of proteins in Table 3 with high degree of repetitions. The results demonstrate that repetitions do not affect the ability of SANA to obtain precise alignment.

The alignment between two P-loop containing NTP hydrolases d1jj7a_ (size = 251) and d1lvga_ (size = 190) requires several indels (Mayr et al. 2007). SANA$_s$ can provide several gaps, especially a large gap to accommodate the subdomain inserted in the d1jj7a_ so that a better alignment is derived. Fig. 2 shows the alignments generated by SANA$_s$ and CE. As illustrated in Fig. 2, the N-terminal P-loops of d1jj7a_ and d1lvga_ which are associated with ADP binding, are not correctly aligned by CE while precisely aligned by SANA.

Detecting fold similarity among proteins

Protein structure alignment algorithms are often used in detecting fold similarity. Novotny et al. (2004) assessed that CE is supreme among the fold similarity recognition algorithms. Here, SANA$_s$ is applied to fold comparison in superfamily level. We consider domains in SCOP1.65, further remove multi-chain domains and finally obtain 4,326 domains as a dataset to be scanned against. We use five proteins in SCOP1.67 but not in SCOP1.65 to search proteins in the dataset that have similar folds in superfamily level. It is noted that using proteins in SCOP1.67 but not in SCOP1.65 is just to make sure that the query proteins are not from the scanned dataset. By adjusting the parameter that plays a trade-off role in the multi-objective optimization method, we find that RMSD for each protein structure alignment is usually less than 4.0 Å. Therefore, using the percentage of aligned residues as a ratio of average number of residues in two proteins to consider fold similarity of two proteins is feasible (Bhattacharya et al. 2007). We calculate it as: $perlen = Aligned/l_{av}$, where *Aligned* is the number of aligned residues, $l_{av} = (L_1 + L_2)/2$,i.e. the average length of two proteins. Table 4 shows the ranking results of SANA$_s$ using the percent of aligned residues, and CE using *Z*-score. As shown in Table 4, we find that the percent of aligned residues for SANA$_s$ is a superior ranking criterion for the fold similarity in most cases.

Table 3 Comparison of pairs of proteins with repetitions

SCOP entries [id1(size)-id2(size)]	SANA$_s$ (Aligned/RMSD/RRMSD/*Q*-score)	DALI (Aligned/RMSD/RRMSD/*Q*-score)
d1thja_ (213)–d1kgqa_ (274)	135/2.11/0.14/0.21	138/3.81/0.25/0.12
d1b5ta_ (275)–d2fzna2(349)	169/2.78/0.16/0.16	183/3.37/0.19/0.15
d1peva1(311)–d1p22a2(293)	260/2.03/0.11/0.51	266/2.34/0.13/0.48
d1peva2(299)–d1b9xa_ (340)	273/2.09/0.11/0.49	275/2.22/0.12/0.48
d1ocxa_ (182)–d2jf2a1(262)	117/1.83/0.13/0.21	123/2.63/0.19/0.18
d1g97a1(196)–d1mr7a_ (203)	98/2.00/0.16/0.17	91/2.23/0.19/0.13

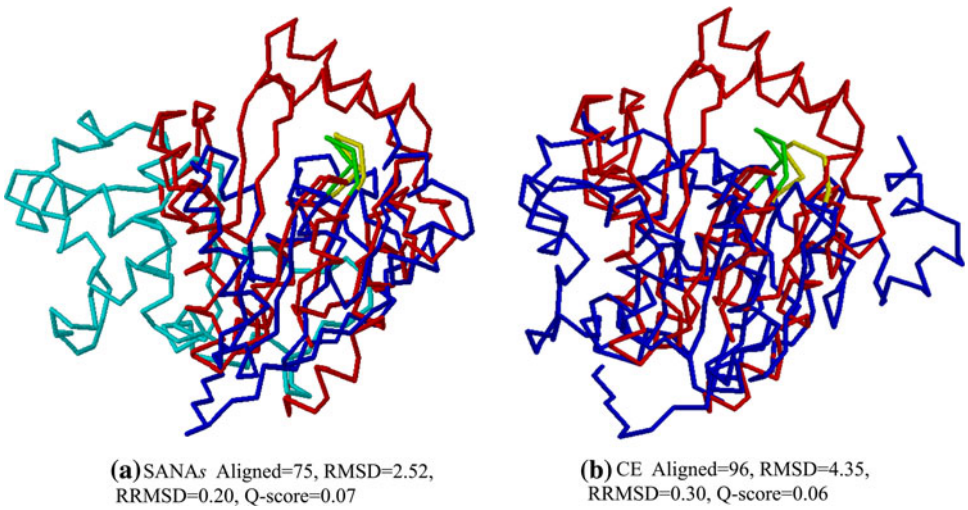


Fig. 2 Alignments between d1jj7a_ (*blue line*) and d1lvga_ (*red line*) by SANA$_s$ and CE. The N-terminal P-loops of d1ljj7a_ and d1lvga_ are marked by *yellow* and *green*, respectively, in two subgraphs, and are correctly aligned by SANA$_s$. Subgraph **a** shows the superposition of these two proteins by SANA$_s$, where the subdomain inserted in d1jj7a_ which aligns with a large gap is marked by cyan. In contrast, subgraph **b** shows the superposition by CE (color figure online)

Table 4 Detecting fold similarity using $SANA_s$ and CE

Query domains	Total number[a]	True positives[b] ($SANA_s$)	True positives[c] (CE)
d1j0wb_	24	23	20
d1uy0a_	25	20	16
d1n4na_	17	13	8
d1rz3a_	98	36	20
d1ixla_	8	8	8

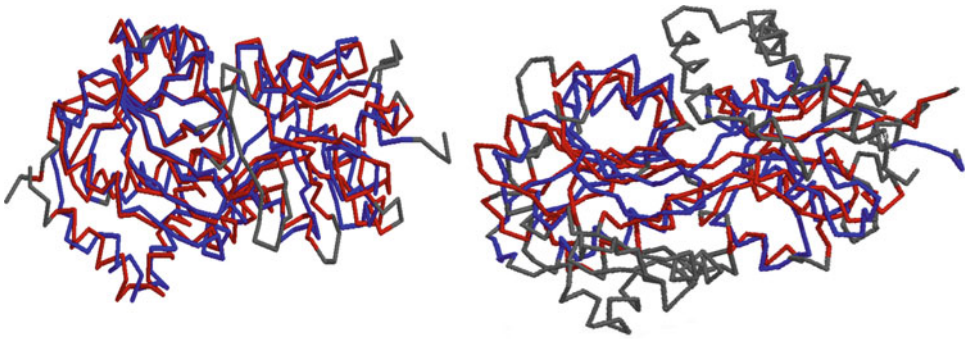
[a] The total number of domains m from the scanned dataset that are in the same superfamily as the query domain

[b] The number of true positives in the prior m domains after ranking all the domains according to the percent of aligned residues in descendent order

[c] The number of true positives in the prior m domains after ranking all the domains according to Z-score of CE in descendent order

Finding circular permutations and conserved functional sites

Circular permutation is a phenomenon of fold changing occurred in evolution of a protein structure that results in the N and C terminus transferring to a different position. Protein structures having circular permutations could exhibit topology independent structural similarity, where polypeptide fragments are inconsistent with the linear order of the protein sequence. Structure sequential alignment-based methods, e.g. DALI, CE and MATT, cannot find good alignment of two proteins related to circular permutation, and one feasible way is to use non-sequential alignment based algorithms, such as C_α-match (Bachar et al. 1993) and MultiProt (Shatsky et al. 2004). In this section, we show that $SANA_n$ can detect circular permutations by examining known examples of circular permutations. Glycine betaine-binding proteins 2b4lA (size = 268) and 1r9lA (size = 309) are related to circular permutation (Lo and Lyu 2008). In contrast with Fig. 3b that from DALI, Fig. 3a shows that $SANA_n$ can identify the two proteins related by circular permutation. The comparison for functional sites marked by green in the alignment generated by $SANA_n$ is shown in Fig. 3c. It illustrates that $SANA_n$ aligns the key residues well, and the aligned functional sites do not keep the sequence order. Furthermore we test $SANA_n$ on other four circular permutation related protein pairs shown in Table 5. Compared with C_α-match and MultiProt, $SANA_n$ tends to find more aligned residues with a higher RMSD.



(a) SANA*n* Aligned=262, RMSD=1.89, RRMSD=0.09, Q-score=0.59

(b) DALI Aligned=188, RMSD=3.52, RRMSD=0.18, Q-score=0.18

2b4lA: SER9 …SER25 GLY26 ILE27 …TRP72 …GLY164
1r9lA : LYS118…TRP140GLY141CYS142…TRP188…LEU305

2b4lA: VAL166…TRP178…TRP225…HIS230…LYS272
1r9lA : LEU3 … ILE17 … TRP65 … HIS69 …ASP116

(c) The matching between functional sites marked by green in two proteins

Fig. 3 The comparison between two proteins 2b4lA and 1r9lA by SANAn and DALI. Subgraphs **a** and **b** show the superposition of two proteins by $SANA_n$ and DALI. The aligned residues in the two proteins 2b4lA and 1r9lA are colored by *blue* and *red*, respectively. Subgraph **c** is the match between functional sites in 2b4lA and 1r9lA which are colored by green. It illustrates that the aligned functional sites do not keep the sequence order (color figure online)

Table 5 Comparison of pairs of protein structures with circular permutations

PDB/SCOP entries [id1(size)–id2(size)]	$SANA_n$ (A/R/RR/Q)	C_α-match (A/R/RR/Q)	MultiProt (A/R/RR/Q)
1nls (237)–2bqpA(228)	220/1.27/0.07/0.76	213/1.23/0.07/0.72	213/1.03/0.06/0.75
1glh (214)–1cpn (208)	207/0.58/0.03/0.93	206/0.53/0.03/0.92	206/0.49/0.03/0.93
1yadA(190)–2duaA(283)	175/2.18/0.13/0.37	130/1.72/0.11/0.24	142/1.92/0.12/0.27
1zbdA(177)–1pujA(261)	146/2.07/0.13/0.31	113/1.49/0.11/0.22	123/1.61/0.11/0.25

A/R/RR/Q aligned/RMSD/RRMSD/Q-score

Conclusions

Our main contribution in this study is that we have developed a novel method (SANA) to chain AFPs which ensures alignments in core regions. The refinement results to two different alignment manners with good alignment quality, i.e. sequential and non-sequential alignments. $SANA_s$ has been found to be competitive to the popular algorithms, e.g. DALI, CE and MATT, in the quality of alignments on the several benchmark datasets. In addition, it can correctly align the protein pairs containing internal repeats or indels, and detect biologically significant regions. Based on the trade-off parameter in the multi-objective optimization method, the RMSD is normally to be constrained below a threshold. We used percentage of equivalently aligned residues for $SANA_s$ to describe fold similarity between two proteins, and found that it is comparable or better than *Z*-score of CE in ranking fold similarity according to the numerical experiments in many cases. Finally, the proposed algorithm $SANA_{\underline{n}}$ is useful in detecting structural similarities with circular permutations and identifying the conserved functional sites.

Proteins may carry out their functions in different conformational states, such as Calmodulin proteins having open, partial open and closed conformations. Because SANA is able to provide several initial alignments which will be refined later, it can obtain distinct solutions for a comparison of two protein structures having significant conformational changes. Therefore, designing techniques to combine the distinct solutions so as to flexibly align the proteins is possible and a future research direction for the improvement of SANA.

Acknowledgments We are grateful to the anonymous referees for many helpful comments that greatly improved the paper. This work is partly supported by the National Natural Science Foundation of China (NSFC) under Key Research Grant No. 10631070, Research Grant No. 60503004, and JSPS-NSFC collaborative project (No. 10711140116). This work is also partially supported by the Chief Scientist Program of Shanghai Institutes for Biological Sciences, Chinese Academy of Sciences with the grant no. 2009CSP002.

References

Aung Z, Tan KL (2006) MatAlign: precise protein structure comparison by matrix alignment. J Bioinform Comput Biol 4:1197–1216

Bachar O, Fischer D, Nussinov R, Wolfson H (1993) A computer version based technique for 3-d sequence independent structural comparison of proteins. Protein Eng 6:279–288

Betancourt MR, Skolnick J (2001) Universal similarity measure for comparing protein structures. Biopolymers 59:305–309

Bhattacharya S, Bhattacharyya C, Chandra NR (2007) Comparison of protein structures comparison by growing neighborhood alignments. BMC Bioinformatics 8:77

Birzele F, Gewehr JE, Csaba G, Zimmer R (2007) Vorolign-fast structural alignment using Voronoi contacts. Bioinformatics 23:e205–e211

Chen L, Wu LY, Wang Y, Zhang SH, Zhang XS (2006) Revealing divergent evolution, identifying circular permutations and detecting active sites by protein structure comparison. BMC Struct Biol 6:18

Fischer D, Elofsson A, Rice DW, Eisenberg D (1996) Assessing the performance of fold recognition methods by means of a comprehensive benchmark. In: Proceedings of 1996 Pacific Symposium on Biocomputing, pp 300–318

Gazit H (1991) An optimal randomized parallel algorithm for finding connected components in a graph. SIAM J Comput 20:1046–1067

Holm L, Sander C (1993) Protein structure comparison by alignment of distance matrices. J Mol Biol 233:123–128

Krissinel E, Henrick K (2004) Secondary-structure matching (SSM), a new tool for fast protein structure alignment in three dimensions. Acta Crystallogr Sect D 60:2256–2268

Lo WC, Lyu PC (2008) CPSARST: an efficient circular permutation search tool applied to the detection of novel protein structural relationships. Genome Biol 9:R11

Mayr G, Domingues FS, Lackner P (2007) Comparative analysis of protein structure alignments. BMC Struct Biol 7:50

Menke M, Berger B, Cowen L (2008) Matt: local flexibility aids protein multiple structure alignment. PLoS Comput Biol 4:1–12

Novotny M, Madsen D, Kleywegt GJ (2004) Evaluation of protein fold comparison servers. Proteins Struct Funct Bioinform 54:260–270

Shatsky M, Nussinov R, Wolfson H (2004) A method for simultaneous alignment of multiple protein structures. Proteins Struct Funct Bioinform 56:143–156

Shindyalov IN, Bourne PE (1998) Protein structure alignment by incremental combinatorial extension (CE) of the optimal path. Protein Eng 11:739–747

Van Walle I, Lasters I, Wyns L (2005) SABmark—a benchmark for sequence alignment that covers the entire known fold space. Bioinformatics 21:1267–1268

Ye Y, Godzik A (2003) Flexible structure alignment by chaining aligned fragment pairs allowing twists. Bioinformatics 19:e246–e255