2015

A Dynamic Programming Algorithm for Finding the Optimal Placement of a Secondary Structure Topology in Cryo-EM Data

Abhishek Biswas
Old Dominion University

Desh Ranjan
Old Dominion University

Mohammad Zubair
Old Dominion University

Jing He
Old Dominion University

Original Publication Citation

Biswas, A., Ranjan, D., Zubair, M., & He, J. (2015). A dynamic programming algorithm for finding the optimal placement of a secondary structure topology in Cryo-EM data. *Journal of Computational Biology*, *22*(9), 837-843. https://doi.org/10.1089/cmb.2015.0120

JOURNAL OF COMPUTATIONAL BIOLOGY
Volume 22, Number 9, 2015
© Mary Ann Liebert, Inc.
Pp. 837–843
DOI: 10.1089/cmb.2015.0120

A Dynamic Programming Algorithm for Finding the Optimal Placement of a Secondary Structure Topology in Cryo-EM Data

ABHISHEK BISWAS, DESH RANJAN, MOHAMMAD ZUBAIR, and JING HE

ABSTRACT

The determination of secondary structure topology is a critical step in deriving the atomic structures from the protein density maps obtained from electron cryomicroscopy technique. This step often relies on matching the secondary structure traces detected from the protein density map to the secondary structure sequence segments predicted from the amino acid sequence. Due to inaccuracies in both sources of information, a pool of possible secondary structure positions needs to be sampled. One way to approach the problem is to first derive a small number of possible topologies using existing matching algorithms, and then find the optimal placement for each possible topology. We present a dynamic programming method of $\Theta(Nq^2h)$ to find the optimal placement for a secondary structure topology. We show that our algorithm requires significantly less computational time than the brute force method that is in the order of $\Theta(q^N h)$.

Key words: algorithms, dynamic programming, electron cryomicroscopy, error, graph, protein, secondary structure, topology.

1. INTRODUCTION

The knowledge of protein tertiary structures is critical in understanding functional mechanisms of proteins. Electron cryomicroscopy (Cryo-EM) has evolved into a structure determination technique that is particularly suitable for large molecular complexes. A number of important large complexes have been resolved to 3–4 Å resolutions at which the backbone of proteins can be determined (Cong et al., 2010; Zhang, Jin et al., 2010). However, it is still challenging to determine protein structures when the resolution of Cryo-EM density map is worse than 4 Å. A density map is a 3-dimensional (3D) image. At medium resolutions, such as 5–10 Å, the backbone of the protein is not resolved. Only secondary structures such as α-helices and β-sheets are detectable. Various methods have been developed to detect α-helices from a 3D image at medium resolutions (Jiang et al., 2001; Dal Palu et al., 2006; Baker et al., 2007; Zeyun and Bajaj, 2008; Ma et al., 2011; Si et al., 2012; Si and He, 2013; Rusu and Wriggers, 2012). A helix detected from such 3D images is represented as an α-trace (a red stick in Fig. 1A) that corresponds to the central axial line of the helix. The helical nature

Department of Computer Science, Old Dominion University, Norfolk, Virginia.

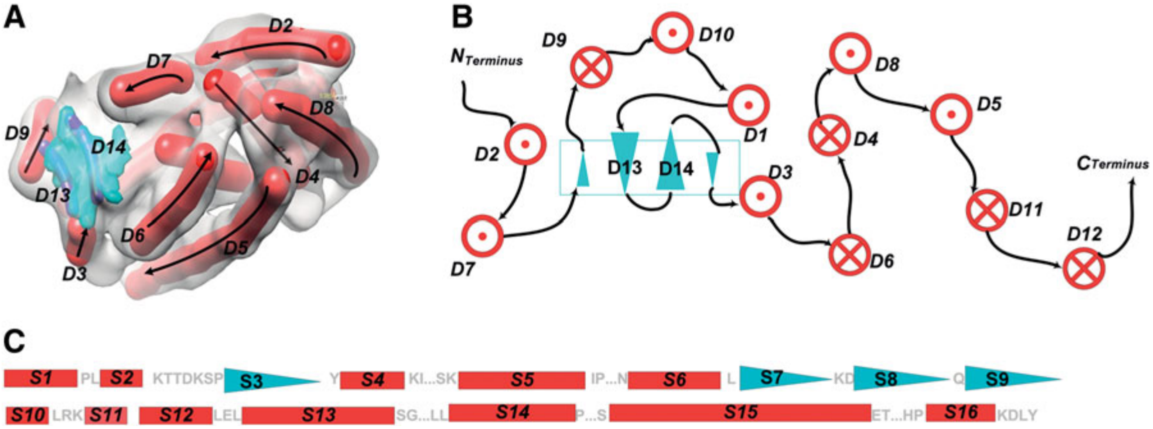

FIG. 1. Secondary structures and topology. **(A)** The density map (gray) was simulated to 10 Å resolution using protein 3PBA from the Protein Data Bank (PDB) and EMAN software (Ludtke et al., 1999). The secondary structures (red sticks, alpha traces; blue, β-sheet; purple sticks, β-traces) were detected using SSETracer (Si and He, 2013) and StrandTwister (Si and He, 2014). The semitransparent surface view was generated using Chimera (Pettersen et al., 2004). For clear viewing, only those at the front of the structure are labeled. Arrows: the direction of the protein sequence at the secondary structure regions. **(B)** The true topology of the secondary structure traces (SSTs) (circles, α-traces; triangles, β-traces). The two directions of the α-traces are represented using a dot and a cross, respectively. **(C)** Secondary structures on the protein sequence (rectangles, α-helix segments; triangles, β-strands; "…," loops longer than two amino acids).

of the backbone is not visible for a helix at such resolutions. A β-sheet appears as a thin sheet (blue in Fig. 1A) and can be detected using automatic or semiautomatic methods (Kong and Ma, 2003; Baker et al., 2007; Si et al., 2012; Si and He, 2013). We recently showed that it is also possible to predict β-strands from β-sheet density by analyzing the twist of a β-sheet (Si and He, 2014). Similar to a detected helix, the location of a β-strand can be represented by a β-trace that corresponds to the central line of the β-strand. Secondary structure traces (SSTs) refer to α-traces and β-traces detected from a 3D image. SSTs provide powerful constraints in protein structure determination.

The connection between SSTs is often ambiguous in density maps at medium resolutions. It is not known which segment of the protein sequence corresponds to which SST in the 3D image. A topology of SSTs refers to their order (with respect to the protein sequence) and the direction of each α-trace or β-trace. For example, the true topology maps SSTs in the order (D_2, D_7, D_9, D_{10}, D_1, D_{13}, D_{14}, D_3, D_6, D_4, D_8, D_5, D_{11}, D_{12}) (Fig. 1B) to sequence segments in the order (S_1, S_2, S_4, S_5, S_6, S_7, S_8, S_{10}, S_{11}, S_{12}, S_{13}, S_{14}, S_{15}, S_{16}) (Fig. 1C). Observe that the two β-strands, S_3 *and* S_9 on the protein sequence, were not detected in the image. Also, note that there are two possible directions when mapping a sequence segment (arrows in Fig. 1A and dot/cross in Fig. 1B).

Secondary structure topology determination is the process of finding the best mapping between the secondary structures detected from a 3D image and those sequence segments predicted from a 1D amino acid sequence. In reality, neither the positions of sequence segments nor the positions of SSTs can be determined with complete accuracy. Long helices are often more accurately detected than shorter helices. Alternative positions are often estimated for individual secondary structures (Fig. 2A, B). Suppose that there are three alternative positions for each sequence segment on the sequence and three alternative positions for each of the SSTs in the image, then there are $3^N 3^N N! 2^N$ possible topologies in the solution space, assuming that N is the number of secondary structures. The first two terms $3^N 3^N$ come from the combinatory nature of the problem when alternative positions of secondary structures are considered. The last two terms correspond to mapping, since there are $N!$ different orders and two possible directions in mapping each secondary structure.

The problem of finding the optimal matching of secondary structures can be broken into two subproblems. One is a matching problem, that is, how to match the secondary structures if a specific set of sequence segments and a specific set of SSTs are given. For this problem, we have shown that there is an effective dynamic programming approach (Al Nasr et al., 2011, 2014a). The other problem is a placement problem, that is, to find an optimal placement of the secondary structures among the alternative positions if the possible orders of the secondary structures are known. We initially proposed a dynamic graph method

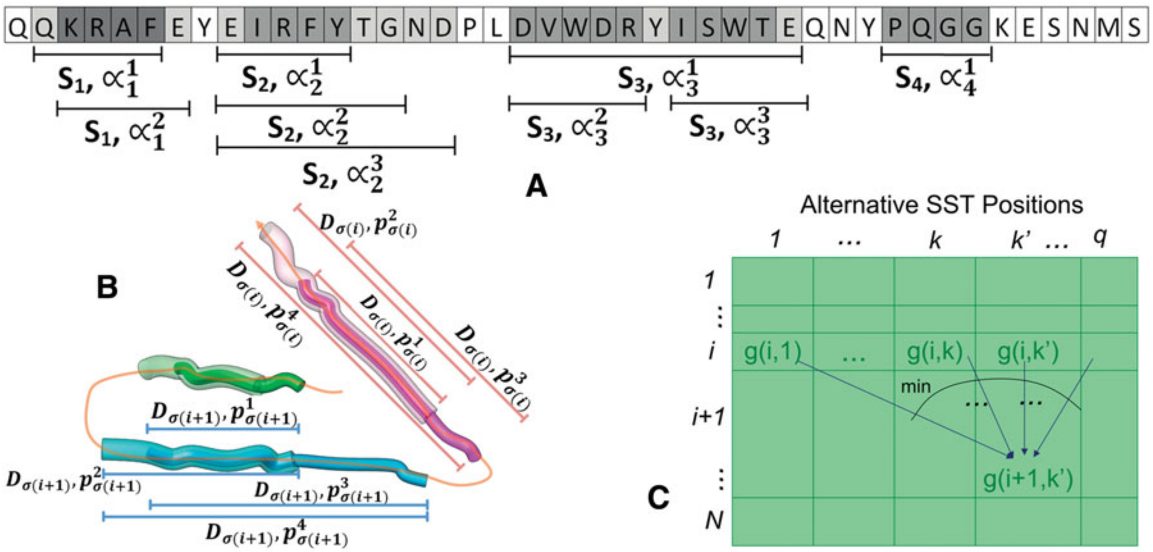
Q Q K R A F E Y E I R F Y T G N D P L D V W D R Y I S W T E Q N Y P Q G G K E S N M S
S_1, α_1^1
S_1, α_1^2
S_2, α_2^1
S_2, α_2^2
S_2, α_2^3
S_3, α_3^1
S_3, α_3^2
S_3, α_3^3
S_4, α_4^1
A
B
$D_{\sigma(i)}, p^2_{\sigma(i)}$
$D_{\sigma(i)}, p^4_{\sigma(i)}$
$D_{\sigma(i)}, p^1_{\sigma(i)}$
$D_{\sigma(i)}, p^3_{\sigma(i)}$
$D_{\sigma(i+1)}, p^1_{\sigma(i+1)}$
$D_{\sigma(i+1)}, p^2_{\sigma(i+1)}$
$D_{\sigma(i+1)}, p^3_{\sigma(i+1)}$
$D_{\sigma(i+1)}, p^4_{\sigma(i+1)}$
Alternative SST Positions
1 … k k' … q
1
i
i+1
N
g(i,1)
…
g(i,k)
g(i,k')
min
g(i+1,k')
C

FIG. 2. Alternative positions of secondary structures and the dynamic programming table. **(A)** Alternative sequence segments. **(B)** Alternative SSTs. The labeling scheme for a sequence segment and an SST can be found in the Methods section. **(C)** The dynamic programming table.

to reduce computational overhead when alternative positions of secondary structures are considered (Biswas et al., 2012). However, more effective methods are needed. In this article, we propose a dynamic programming method to address the placement problem, particularly for the alternative position of SSTs detected from 3D images. We show that, for each given mapping of the secondary structures, the optimal placement problem can be solved using a dynamic programming method. When a small number of possible mappings are derived, such a dynamic programming approach can be used to find the optimal placement for each of the mappings.

2. METHODS

To simplify the formulation of the placement problem, let us assume that N secondary structure sequence segments are being mapped to the same number of SSTs in the 3D image. Let $S=(S_1, S_2, \ldots, S_N)$ be a tuple of sequence segments. Let $D=\{D_1, D_2, \ldots, D_N\}$ be a set of the SSTs from the 3D image. In the placement problem, it is assumed that the mapping between S and D is known, and the direction of each SST is known. However, it is not known which alternative position of each secondary structure is to be selected. Given a particular topology of SSTs, let mapping σ be the mapping such that S_i is mapped to $D_{\sigma(i)}$ for $i=1, \ldots, N$. Let us represent the alternative sequence segments and alternative SSTs, respectively, as the following. Let (S_i, α_i^l) be the lth alternative segment for the ith secondary structure on the sequence, where $l=1, 2, \ldots, p$, and $i=1, 2, \ldots, N$. Let $(D_{\sigma(i),}\ p^k_{\sigma(i)})$ be the kth alternative for SST $D_{\sigma(i)}$, where $k=1, 2, \ldots, q$, and $i=1, 2, \ldots, N$. The placement problem is to find a tuple of sequence segments $[(S_1, \alpha_1^{l_1}), (S_2, \alpha_2^{l_2}), \ldots, (S_N, \alpha_N^{l_N})]$ and a tuple of SSTs $\left[\left(D_{\sigma(1),}\ p^{k_1}_{\sigma(1)}\right), \left(D_{\sigma(2),}\ p^{k_2}_{\sigma(2)}\right), \ldots, \left(D_{\sigma(N),}\ p^{k_N}_{\sigma(N)}\right)\right]$, $1 \le l_1, l_2, \ldots, l_N \le p$, and $1 \le k_1, k_2, \ldots, k_N \le q$, such that the score of mapping the two tuples is minimized.

A variety of factors have been considered to score a mapping. The length of a helix can be estimated as $(1.5 \times AA)$ Å for a helix and $(3 \times AA)$ Å for a β-strand, where AA is the number of amino acids involved in the secondary structure. Therefore, the length of the secondary structure can be compared between an SST and a sequence segment during a match. Similarly, the length of a loop between two consecutive secondary structures can also be considered in scoring a mapping. The loop length in the image can be measured along the skeleton image between the two end points (Ju et al., 2007; Baker et al., 2011; McKnight et al., 2013). Additional factors that were considered include the loop score (Lindert et al., 2009) and constraints of β-strands (Al Nasr et al., 2014b). In this article, the scoring function compares the length of secondary structures being matched and the loop length of those calculated from the skeleton trace in the image and those on the sequence (Al Nasr et al., 2014b).

2.1. Dynamic programming algorithm of finding the optimal placement

A naïve way to find the best placement of a topology is to exhaustively score each mapping of a set of alternative sequence segments to a set of alternative SSTs. This will involve $p^N q^N$ possible combinations to find the best placement, where p and q are the number of alternatives for each secondary structure on the sequence and in the image, respectively. However, one can devise a better algorithm to solve the problem using dynamic programming. To simplify the exposition, first consider a subproblem in which the sequence segments are fixed. In other words, there is only one alternative for sequence segments and $p = 1$. However, there are alternative positions for SSTs. In this subproblem, S_i is being mapped to $\left(D_{\sigma(i),}\ p^k_{\sigma(i)}\right)$, for $i = 1, 2, \ldots, N$, and $k = 1, 2, \ldots, q$. Let $g(i, k)$ denote the best cost of placing the first i SSTs using the kth alternative position for $D_{\sigma(i)}$. In other words, the best cost $g(i, k)$ is obtained when sequence segments $\{S_1, S_2, \ldots, S_i\}$ are optimally mapped to SSTs $\{D_{\sigma(1),} D_{\sigma(2),} \ldots D_{\sigma(i)}\}$ using positions $\{p^{k_1}_{\sigma(1)}, p^{k_2}_{\sigma(2), \ldots,} p^k_{\sigma(i)}\}$, respectively. Then for $k' \in \{1, 2, \ldots, q\}$,

$$\begin{aligned} g(i+1, k') = \min_{k \in \{1, 2, \ldots, p\}} & \left(g(i, k) \ + \left|\ l\left(D_{\sigma(i+1),}\ p^{k'}_{\sigma(i+1)}\right)\right.\right. \\ & \left.\left. -\ l(S_{i+1})\right| + \left|\delta\ \left(\left(D_{\sigma(i),}\ p^k_{\sigma(i)}\right), \left(D_{\sigma(i+1),}\ p^{k'}_{\sigma(i+1)}\right)\right)\ -\ d(S_{i,} S_{i+1})\right|\right) \end{aligned} \tag{1}$$

Note that $l(S_{i+1})$ measures the length of secondary structure segment S_{i+1}, and $d(S_{i,}\ S_{i+1})$ measures the loop length between S_i and S_{i+1} on the sequence. $\delta\ (a, b)$ measures the distance in 3D between SST a and SST b. Intuitively, for any position $p^{k'}_{\sigma(i+1)}$, $g(i+1, k')$ is only affected by the best cost of $g(i, k)$ where $k = 1, 2, \ldots, q$, and the score obtained from the relative positioning of the ith and the $(i+1)$th SST (Fig. 2A, C). The space requirement of the algorithm is Nq, since the implementation of the algorithm uses a table of size Nq to store and reuse the computed $g(i, k)$ (Fig. 2C). The running time of the dynamic programming algorithm is $\Theta\ (Nq^2\ h)$, where h is the time to calculate the 2nd, 3rd, and 4th term of Equation (1). Note that the naïve way to find the optimal placement is $\Theta\ (q^N\ h)$. Our dynamic programming method reduces the computation time from an exponential in N to time that is linear in N. This is a massive improvement.

3. RESULTS AND DISCUSSION

To test the efficiency of the dynamic programming method for the optimal placement problem, an experiment was performed using alternative SST positions while keeping the sequence segments of the secondary structures unchanged. Our dynamic programming method was compared with a brute force method in which all possible placements are calculated. The experimental dataset includes 12 α-proteins and 4 α-β proteins, each contains an image and a known structure in the Protein Data Bank (PDB). α-proteins contain only helices, and α-β proteins contain both α-helices and β-sheets. The proteins in this dataset contain 9–33 helices with sizes ranging from 142 (1FLP) to 585 amino acids (2XVV) in length. The atomic structures were downloaded from PDB and were used to simulate 3D images with 10 Å resolution using EMAN (Ludtke et al., 1999). SSETracer was used to detect the position of α-helices and β-sheets in the 3D images (Si et al., 2013). StrandTwister was used to identify β-traces from the isolated β-sheets.

3.1. Finding optimal placement of α-traces in α-proteins

Although it is possible to detect helices from cryo-EM density maps at medium resolutions, it is almost impossible to detect all the helices with complete accuracy. For each α-trace that was detected using SSETracer and is shorter than 30 Å, five alternatives were produced. Since shorter helices are more error-prone, two shifts (left and right along the central axis) of 10% of the length and two lengths (10% shorter or longer) were created to simulate errors.

The initial mapping between helix traces in the 3D image and sequence segments was performed using the topology graph method DP-TOSS (Al Nasr et al., 2014b). The helix traces detected using SSETracer

Table 1. Run Time Finding the Optimal Placement Among Top 100 Topologies for α-Proteins

Index	*Protein PDB ID*	*No. of true helices*	*No. of sticks detected*[a]	*No. of possible placements*[b]	*Brute force*[c]	*Dynamic programming*[d]
1	1NG6	9	7	3.12×10^{3}	156.87	7.23
2	1FLP	7	7	78.12×10^{3}	198.65	8.71
3	2XB5	13	9	78.12×10^{3}	237.61	11.45
4	2OEV	26	20	1.95×10^{6}	290.78	12.48
5	3ACW	17	14	48.82×10^{6}	641.07	14.2
6	3LTJ	16	12	244.14×10^{6}	3559.81	15.69
7	1Z1L	23	15	244.14×10^{6}	3409.88	14.68
8	3ODS	21	16	3.81×10^{12}	78544.24	17.54
9	2XSI	33	19	3.81×10^{12}	316140.02	24.12
10	2XVV	33	19	3.81×10^{12}	275412.51	26.78
11	3HJL	20	20	3.81×10^{12}	304182.26	24.68
12	1HZ4	21	19	19.07×10^{12}	493718.75	22.35

[a] The number of α-traces detected using SSETracer.

[b] The number of possible placements for α-traces shorter than 30 Å.

[c] The time (in seconds) to find optimal placement for top 100 topologies using brute force.

[d] The time (in seconds) to find optimal placement for top 100 topologies using our dynamic programming algorithm.

and true secondary structure sequence segments were used to generate initial positions in DP-TOSS. DP-TOSS generates a list of possible topologies sorted by topology scores. For each of the top 100 possible topologies, the optimal placement of SSTs was searched. In the case of protein 3ACW (Table 1, row 5), SSETracer detected 14 α-traces out of 17 helices in the true structure. We generated alternative positions for 11 of the 14 detected helices as they were shorter than 30 Å in length and are expected to be error-prone. In this case, the naïve way to find the optimal cost for one topology requires enumerating all the possible 5^{11} placements and finding the best among them. This brute force computation was done for the top-ranked 100 possible topologies. It takes 641.07 seconds for the brute force method to find the optimal placement but only 14.2 seconds for our dynamic programming method. In fact, our implementation of the brute force method used the bitmap technique that is often used in reducing the computation in combination problems. Yet significant difference in run time is shown between the 2 methods for all 12 cases. It takes only about 26 seconds to find the best placement among top 100 topologies for the largest test case 2XVV in which 18 of the 33 helices in the protein have their alternative positions (Table 1).

In the test involving four α-β proteins, two images were simulated using the true structure as for the α-proteins and two are experimentally derived cryo-EM density maps that were downloaded from Electron Microscopy Data Bank (EMDB). Each density map corresponds to an atomic structure, and therefore can be used to test the accuracy of our approach (see Table 2). In case of the experimentally derived cryo-EM density maps, we extracted the density component corresponding to chain R of the protein for EMDB_5030

Table 2. Accuracy and Run Time of Finding the Optimal Placement Among Top 1000 Topologies with Maximum 20 Alternate Positions for Each Secondary Structure Trace

PDB ID[a]	*No. of α-helices*[b]	*No. of β-strands*[c]	*No. of α-stk/β-stk*[d]	*Rank*[e]	*Brute force*[f]	*Dynamic programming*[g]
1OZ9	5	5	5/5	76	378.46	28.82
1JL1	4	5	4/5	89	330.88	33.51
3FIN_R (5030)*	3	3	3/3	97	122.90	4.34
3IZ6_K (1780)*	3	5	2/5	6	336.94	18.45

[a] The PDB ID with chain, with asterisk (*) indicating EMDB ID of the experimentally derived Cryo-EM map.

[b] The number of helices in the true structure.

[c] The number of β-strands in the true structure.

[d] The number of α-traces/β-traces detected from the 3D image.

[e] The rank of the true topology using the true sequence position of secondary structures.

[f] The time (in seconds) to find optimal placement for top 100 topologies using brute force.

[g] The time (in seconds) to find optimal placement for top 100 topologies using our dynamic programming algorithm.

and chain K for EMDB_1780 respectively. We derived a maximum of 20 alternative SSTs for each α-helices/β-strand using SSETracer and StrandTwister. Most of the alternatives are produced for β-strands with slightly different orientations and locations, since β-strands are often not detected exactly. The true topology was ranked 97th and 6th for EMDB_5030 and EMDB_1780, respectively.

4. CONCLUSIONS

The inaccuracy in estimating the position of secondary structures on protein sequences and in 3D images at medium resolutions requires the search for an optimal mapping among all possible alternate positions of secondary structures. We propose a dynamic programming algorithm to find the optimal placement for a given secondary structure topology. Our dynamic programming method uses Θ (Nq^2 h) time verses Θ (q^N h) for the brute force method. The test using 12 α-proteins shows that the dynamic programming method uses significantly less time than the brute force method particularly when the number of secondary structures (N) and the number of alternatives (q) are large. The test involving four α-β proteins shows that the dynamic programming method applies to more complicated proteins involving β-strands. In addition to the efficiency, the approach has reasonable accuracy, although it has room to improve. We demonstrated in this article that, for each possible topology of the secondary structures, finding the optimal placement among alternative positions can be addressed using a dynamic programming method.

ACKNOWLEDGMENTS

The work in this article is partially supported by NSF DBI-1356621, M&S Fellowship, and FP3 Fund of the Old Dominion University. We thank Dong Si for the help in SSETracer.

AUTHORS' CONTRIBUTION

A.B. developed and implemented the algorithm under the guidance of D.R., M.Z., and J.H.

AUTHOR DISCLOSURE STATEMENT

No competing financial interests exist.

REFERENCES

Al Nasr, K., Ranjan, D., Zubair, M., et al. 2011. Ranking valid topologies of the secondary structure elements using a constraint graph. *J. Bioinform. Comput. Biol.* 9, 415–430.

Al Nasr, K., Ranjan, D., Zubair, M., et al. 2014a. Solving the secondary structure matching problem in de novo modeling using a constrained K-shortest path graph algorithm. *IEEE Trans. Comput. Biol. Bioinform.* 11, 419–430.

Al Nasr, K., Ranjan, D., Zubair, M., et al. 2014b. Sovling the secondary structure matching problem in cryo-EM de novo modeling using a constrained K-shortest path graph algorithm. *IEEE/ACM Trans. Comput. Biol. Bioinform.* 11, 419–429.

Baker, M.L., Abeysinghe, S.S., Schuh, S., et al. 2011. Modeling protein structure at near atomic resolutions with Gorgon. *J. Struct. Biol.* 174, 360–373.

Baker, M.L., Ju, T., and Chiu, W. 2007. Identification of secondary structure elements in intermediate-resolution density maps. *Structure* 15, 7–19.

Biswas, A., Si, D., Al Nasr, K., et al. 2012. Improved efficiency in cryo-EM secondary structure topology determination from inaccurate data. *J. Bioinform. Comput. Biol.* 10, 1242006.

Cong, Y., Baker, M.L., Jakana, J., et al. 2010. 4.0-A resolution cryo-EM structure of the mammalian chaperonin TRiC/CCT reveals its unique subunit arrangement. *Proc. Natl. Acad. Sci. USA* 107, 4967–4972.

Dal Palu, A., He, J., Pontelli, E., et al. 2006. Identification of alpha-helices from low resolution protein density maps. Proceeding of Computational Systems Bioinformatics Conference (CSB), pp. 89–98.

Jiang, W., Baker, M.L., Ludtke, S.J., et al. 2001. Bridging the information gap: Computational tools for intermediate resolution structure interpretation. *J. Mol. Biol.* 308, 1033–1044.

Ju, T., Baker, M.L., and Chiu, W. 2007. Computing a family of skeletons of volumetric models for shape description. *Comput. Aided Des.* 39, 352–360.

Kong, Y., and Ma, J. 2003. A structural-informatics approach for mining beta-sheets: Locating sheets in intermediate-resolution density maps. *J. Mol. Biol.* 332, 399–413.

Lindert, S., Staritzbichler, R., Wötzel, N., et al. 2009. EM-fold: De novo folding of alpha-helical proteins guided by intermediate-resolution electron microscopy density maps. *Structure* 17, 990–1003.

Ludtke, S.J., Baldwin, P.R., and Chiu, W. 1999. EMAN: Semiautomated software for high-resolution single-particle reconstructions. *J. Struct. Biol.* 128, 82–97.

Ma, L., Reisert, M., and Burkhardt, H. 2011. RENNSH: A novel alpha-helices identification approach for intermediate resolution electron density maps. *IEEE/ACM Trans. Comput. Biol. Bioinform.* 99, 1.

McKnight, A., Si, D., Al Nasr, K., et al. 2013. Estimating loop length from CryoEM images at medium resolutions. *BMC Struct. Biol.* 13, S5.

Pettersen, E., Goddard, T., Huang, C., et al. 2004. UCSF Chimera—a visualization system for exploratory research and analysis. *J. Comput. Chem.* 25, 1605–1612.

Rusu, M., and Wriggers, W. 2012. Evolutionary bidirectional expansion for the tracing of alpha helices in cryo-electron microscopy reconstructions. *J. Struct. Biol.* 177, 410–419.

Si, D., and He, J. 2013. Beta-sheet detection and representation from medium resolution Cryo-EM density maps. BCB' 13: Proceedings of ACM Conference on Bioinformatics, Computational Biology and Biomedical Informatics Washington, DC, September 22–25, 2013, pp. 764–770.

Si, D., and He, J. 2014. Tracing beta strands using StrandTwister from Cryo-EM density maps at medium resolutions. *Structure* 22, 1–12.

Si, D., Ji, S., Nasr, K.A., et al. 2012. A machine learning approach for the identification of protein secondary structure elements from electron cryo-microscopy density maps. *Biopolymers* 97, 698–708.

Zeyun, Y., and Bajaj, C. 2008. Computational approaches for automatic structural analysis of large biomolecular complexes. *IEEE/ACM Trans. Comput. Biol. Bioinform.* 5, 568–582.

Zhang, X., Jin, L., Fang, Q., et al. 2010. 3.3 angstrom Cryo-EM structure of a nonenveloped virus reveals a priming mechanism for cell entry. *Cell* 141, 472–482.

Address correspondence to:
Dr. Jing He
Department of Computer Science
Old Dominion University
4700 Elkhorn Ave, Suite 3300
ECS Building
Norfolk, VA 23529

E-mail: jhe@cs.odu.edu