End to end differentiable protein structure refinement

Xiaoyong Cao[1] and Pu Tian[1,2,*]

1. School of Life Sciences, Jilin University
2. School of Artificial Intelligence, Jilin University, 2699 Qianjin Street, Changchun, China 130012

*. Corresponding author: tianpu@jlu.edu.cn

Abstract

Refinement is an essential step for protein structure prediction. Conventional force fields (both physics and knowledge-based ones) and sampling (molecular dynamics simulations and Monte Carlo algorithms) based methods are computationally intensive on the one hand, and are labor intensive for updating parameters on the other hand. A number of neural network based methods have been developed for structure prediction. However, no differentiable refinement algorithm is available up to date. Based on neural network implementation of the local maximum likelihood approximation of generalized solvation free energy theory, we develop a fully differentiable refinement algorithm with clear physical interpretation. Instead of explicit functions utilized by conventional force fields approach, molecular interactions are described by neural network parameters, which are free of limitation by specific functions and may be readily updated by training. Substitution of configuration sampling by differentiation increases optimization efficiency by many orders of magnitude. Additionally, both global and local conformation restraints are added to further improve the refinement algorithm. More importantly, due to modular separation of coordinates transformation and neural network, the algorithm is amenable to further sophisticated incorporation of various information through redesign of neural network architecture.

1. Introduction

With rapid development of sequencing capability, computational protein structure prediction has become an increasingly important tool for molecular biology [1]. Despite great progress made, a big gap remains between present prediction accuracy and that of high resolution experimental methods [2]. Refinement is meant to partially bridge this tremendously challenging gap [2-6], and has become an independent competition in CASP (Critical Assessment of Structure Prediction). Combination of force fields (FF) and sampling is the dominating framework. Sampling methods include MD (molecular dynamics)-based [7-10] and MC (Monte Carlo)-based approaches [11-13], side-chain rebuilding approaches [14-16] and fragment-based approaches [17]. Major FFs are classified as physics based (e.g. CHARMM [18] and AMBER [19]) and knowledge based ones (e.g. Dfire [20], Dope [21], RW [22], RWplus[22], GOAP[23], Rosetta energy function[24-26]). Since CASP10, Feig et al. have made significant progress towards a more consistent refinement with improved FF, the application of Cα restraints and an ensemble averaging stage with explicit solvent [27]. Since then, MD-based approaches have become the mainstream for refinement [26,29,30]. However, optimization of a typical model (~150 residues) requires about 75000 CPU Hours [28]. All algorithms based on conventional FF and sampling framework share the difficulty in updating of parameters and expensive sampling.

NN （Neural Network） has brought great progress for protein structure prediction [31-33]. In particular, Alpha-fold won the first place in the FM (Free Model) structure prediction in CASP13 [34,35]. Three representative NN based structure prediction schemes are summarized in Fig. 1B. RefineD [36] protocol utilized the integrated classifier based on deep discriminant learning to predict multi-resolution probability constraints, which were furthered converted into scoring terms to guide conformational sampling with Rosetta-based FastRelax [37]. AlQuraishi stated that “hybrid systems using deep learning and co-evolution as priors and physics-based approaches for refinement will soon solve the long-standing problem of accurate and efficient structure

prediction" [33]. Suggesting no differentiable refinement is expected in the near future by the community. In structure prediction, known structures are used as targets for training of neural networks. In refinement, one need to repetitively minimize free energy, for which no targets or labels are available. This presents a key hurdle for development of differentiable refinement algorithm. The other major difficulty is the structural constraints that has to be satisfied during the optimization process.

We overcome the first hurdle by development of the GSFE (generalized solvation free energy) theory [38], and tackle the second issue through combination of coordinates transformation and auto differentiation (back propagation) [39]. With these two key innovations, our GSFE-Refinement substitutes sampling with differentiation, resulting in a speeding up of approximately 1000,000 times (seconds vs. thousands of hours CPU core hours) when compared with MD-based sampling approach. Molecular interactions are specified by parameters of a simple NN based on LMLA (local maximum likelihood approximation) treatment of the GSFE. Updating of these NNFF (neural network FF) parameters take a few hours of training on a 1080Ti GPU, and is dramatically more efficient than updating of conventional FF. More importantly, NNFF are not limited by fixed function forms.

2. Methodology

2.1 Introduction to local maximum likelihood approximation of GSFE

In GSFE, each comprising unit is both solute and solvent of its neighbors. For a n-residue protein with sequence X = $\{x_1, x_2, \ldots x_n\}$, the free energy of a given structure is:

$$F = -lnP(Structure|X) \quad (1)$$

With Bayes formula:

$$-lnP(Structure|X) = -lnP(X|Structure)P(Structure)/P(X) \quad (2)$$

For a given sequence X, P(X) is a constant and is dropped for computation. Define R_i (X_i, Y_i) as all local structural regions within selected cutoff distance of X_i, (Y_i being

specific solvent of X_i) we have:

$$P(structure) = \mathrm{P}(\mathrm{R}_1, \mathrm{R}_2 \ldots \mathrm{R}_\mathrm{n}) \prod_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{P}(\mathrm{R}_\mathrm{i}) \quad (3)$$

$$P(X|Structure) \approx \prod_{\mathrm{i}=1}^{\mathrm{n}} P(x_i|R_i) \quad (4)$$

Equation (3) has two terms, $\prod_{i=1}^{n} P(R_i)$ are local priors and $P(R_1, R_2 \ldots R_n)$ is the global correlation term. Equation (4) assumes all influence to each unit is included in its solvent Y_i, this product of n conditional probabilities is the LMLA (local maximum likelihood approximation) of GSFE, which is the basis for our NN training in this work. Incorporation of local priors and global correlations will be tackled in future.

2.2 Training of neural network force fields

The same dataset as Ref 38 is used for training NNFF. For each target residue, 22 neighboring residues are selected as its neighbors (6 upstream, 6 downstream adjacent residues in primary sequence and 10 non-adjacent ones). Features for each target residue include one-hot vectors representing identities of neighboring residues, residue pair distances (Cα-Cα) between target its neighbor residues, and dihedral angles (C_α, C, N, C_β) indicating side chain orientations[38]. Two NNFFs are trained. Input for the first NNFF include a 22-dimensional one-hot vector, 6 dihedral angles (in radians) and the Cα-Cα distance, amount to (22+6+1) * 22 = 638 dimensions. For the other NNFF, each angle θ is converted into sinθ and cosθ, resulting in an input of (22+6*2+1) * 22=770 dimensions. These two NNFFs are denoted as 638-NNFF and 770-NNFF below. A four-layer feed-forward (638/770-512-512-512-21) network architecture is used (Fig. 1A) and trained for 30 epochs with a learning rate of 0.1.

2.3 Refinement process

As shown in Fig.1 C, a given starting structure is first converted from cartesian coordinates of the backbone atoms (C_α, C, N, C_β) into internal coordinates, which is further converted back into cartesian coordinates for feature extraction. Finally, the starting structure is optimized by minimizing the LOSS (approximate free energy plus structural restraints) with the NN model generated in the training of NNFF.

2.3.1 Coordinate transformation

Coordinate transformations are the key steps for constraints of bond lengths and angles. Setting gradient of approximate free energy with respect to backbone dihedrals φ and ψ facilitates automatic differentiation, which is the key technology underlying highly efficient differential optimization. The NeRF (Natural Extension Reference Frame) algorithm [33,43] is used for transformation (See supplementary material for details).

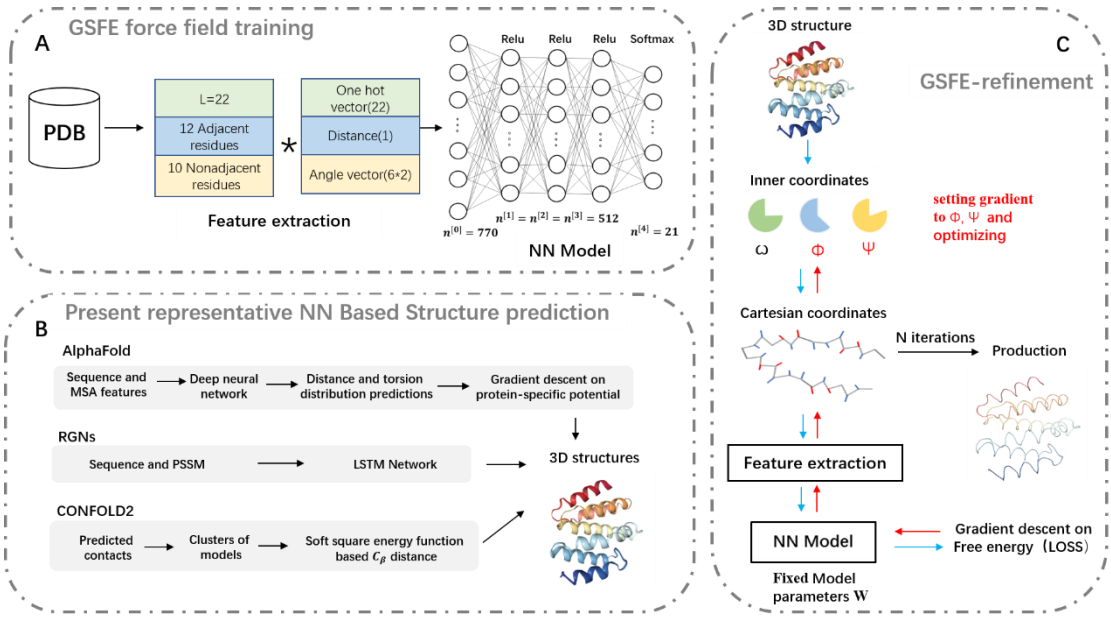

Fig.1 Schematic representation of A) neural network implementation of GSFE (see Ref 38 for details). Amino acid identity is used as labels for training; LMLA approximation is utilized for assessment of structural models. B) major present protein structure prediction schemes based on NN. All networks provide a map from sequence information (or contacts predicted from which) to structure. C) Flowchart of the GSFE-refinement protocol, with "Feature extraction" and "NN Model" being the same as that of A). This scheme provides a map between structure and free energy. Through iterative minimization of free energy by differentiation with respect to structure with the NN model fixed, we realize differentiable structure optimization.

2.3.2 Loss function of the optimization process

The total loss is shown below:

$$LOSS = loss + \lambda loss_{smoothL1} \qquad (5)$$

$$loss = \frac{1}{n}\sum_{i=1}^{n} -w_i y_i log\, p_i \quad (6)$$

$$w_i = 1 + p_i log p_i / log 21, \quad (7)$$

$$p_i = \frac{\exp(x_{ij})}{\sum_j^m \exp(x_{ij})} \quad (8)$$

$$loss_{smoothL1} = \frac{1}{L}\sum_{k=1}^{L} \begin{cases} 0.5 * [dist0 - dist1]^2 & if\ |dist0 - dist1| < 1 \\ |dist0 - dist1| - 0.5 & otherwise \end{cases} \quad (9)$$

Here, λ is the coefficient of $loss_{smoothL1}$, $loss_{smoothL1}$ is designed to limit the conformational search space during optimization. The larger the λ, the stronger the restraint. w_i is the AA (amino acid) site weight parameter, n is the protein chain length, m is the type of AA (m = 21), L is the number of neighboring AA around each target AA (L = 22), and y_i is the label of residue i。$dist0$ is a AA pair distance in the starting structure, and $dist1$ is the corresponding AA pair distance in the updated structure. The *loss* term is the LMLA-GSFE with w_i and $\lambda loss_{smoothL1}$ serve as local and global regularization terms respectively. The total *LOSS* is iteratively minimized during the optimization. Learning rate specifies an effective step size for updating of structures, we use 0.0005 and 0.001 in this study. It is important to note that learning rate in training NNFF specifies magnitude of NNFF updating instead.

2.4 Benchmark dataset

The 3Drobot data set is taken as the source, which contains decoys for 200 native structures. After removing structures with a homology similarity higher than 25% with the training set, 10800 decoys of 36 native structures (300 each) are selected as the benchmark dataset. The RMSD (root mean squared deviation) of 300 decoys from their native structures are approximately evenly distributed in the interval of 0-12Å.

3. Results

3.1 Refinement with LMLA

In order to investigate the efficiency of optimization with back propagation [39] and the performance of LMLA-GSFE, we carried out refinement of the benchmark dataset. With 5 iterations, all target optimization takes less than 30 seconds on one core of a i5-8500 6-core CPU at 3.0GHz, about 1000,000 times more efficient than MD-based optimization, which takes about 75000 core hours [28]. The results for 638-NNFF and 770-NNFF at learning rates of 0.001 and 0.0005 are evaluated by Cα RMSD and GDT-HA (Global Distance Test High Accuracy) as indicators [44] and listed in Table 1 (see detailed results in Table S1-S4 in supplementary materials). For given NNFF, the result with the learning rate of 0.0005 is better than that of 0.001. For the 638-NNFF, the average GDT-HA-num increase from 54.7 to 66.6; For 770-NNFF, it increase from 50.7 to 64.9. There is consistent with the trend for RMSD-num. The best of 5 models improve over the top1 model significantly (Table 1), with 770-NNFF at learning rate of 0.0005 achieved the best performance. We also analyze the optimization of the decoys in the IT1ABV group in the ITASSER data set [22] with similar results (see supplementary materials for details). All these results suggest the limitation of NNFF based on LMLA-GSFE.

Table1. Top 1 and best of 5 models from optimization with LMLA-GSFE

Model	NNFF/lr[a]	GDT-HA-num[b]	Avg ΔGDT-HA[c]	RMSD-num[d]	Avg-ΔRMSD[e]
top 1	638/0.0005	66.6/300	-0.0162	105.7/300	0.0475
	638/0.001	54.7/300	-0.0278	89.7/300	0.1073
	770/0.0005	64.9/300	-0.0187	103.1/300	0.0532
	770/0.001	50.7/300	-0.0351	82.4/300	0.1369
Best of 5	638/0.0005	130.6/300	-0.0009	149.5/300	-0.0141
	638/0.001	110.3/300	-0.0071	140.0/300	-0.0059
	770/0.0005	152.2/300	0.0015	169.3/300	-0.0111
	770/0.001	108.6/300	-0.0071	135.9/300	-0.0084

[a]NNFF is denoted by the corresponding input tensor dimension, lr is the learning rate

[b]The average number of decoys with ΔGDT-HA>0 for all 36 decoy sets

[c]The average value of ΔGDT-HA for all decoys

[d]The average number of decoys with ΔRMSD<0 for all 36 decoy sets

[e]The average value of ΔRMSD for all decoys

3.2 Impact of the Smooth_l1 term

In view of the increasing accuracy of the starting structure generated by protein structure prediction, the imposition of reasonable restrictions has become an important part of refinement [2,40,41,43]. We therefore add the smooth_l1 Loss term ($\lambda = 1.2$) to limit the conformation search space to be in the vicinity of the starting structure (Equations 5 and 9). As shown in Table 2 (see detailed results in Table S5-S8 in supplementary materials), with this restraint, both top 1 and best of 5 models improve. For top 1 models, learning rate of 0.0005 is better than 0.001 for both GDT-HA and RMSD indicators. However, for best of 5 models, the opposite is observed for GDT-HA and RMSD measures. Given learning rate, the optimization results of 638-NNFF and 770-NNFF are similar.

Table2. Top 1 and best of top 5 model with addition of global conformation restraints

Model	NNFF/lr	GDT-HA-num	Avg-ΔGDT-HA	RMSD-num	Avg-ΔRMSD
Top1	638/0.0005	91.3/300	-0.0044	132.3/300	0.0091
	638/0.001	86.4/300	-0.0065	127.4/300	0.0131
	770/0.0005	91.0/300	-0.0048	130.8/300	0.0091
	770/0.001	86.3/300	-0.0071	126.8/300	0.0134
Best of 5	638/0.0005	151.4/300	0.0006	180.5/300	-0.0146
	638/0.001	148.4/300	-0.0011	191.6/300	-0.0241
	770/0.0005	152.5/300	0.0007	182.0/300	-0.0163
	770/0.001	147.6/300	-0.0012	191.6/300	-0.0257

3.3 Impact of entropy weights on the model

There have been some studies on local restraints of protein structure refinement based on prior knowledge [2,45,46], selection of specific regions [2,15,45], and local structure evaluation [2,22,45]. In this regard, we calculate the entropy of the predicted probability of each AA site as the local optimization weight (Equations 6, 7). Table 3 shows the results after adding local weights (see detailed results in Table S9-S12 in supplementary materials). Compared with the results of Table 2, various evaluation indicators improve further， the RMSD-num in the 638-NNFF is 159.8 / 300 at a learning rate of 0.0005 and 170.2 / 300 at a learning rate of 0.001, more than half of decoy's RMSD decrease after optimization. In particular, Avg.-ΔRMSD for the

combination of 638-NNFF/lr=0.001 is -0.0071. Overall, the optimization results in top1 mode are not optimistic. In contrast, in best of 5 models, GDT-HA-num is between 166.3 / 300 and 171.5 / 300, Avg-ΔGDT-HA is greater than 0, RMSD-num is 203.0 / 300 to 225.0 / 300, and Avg-ΔRMSD is- 0.0156 to -0.0293. All indicators are much better than Top1 model.

Table 3. Top 1 and best of 5 model with both local and global conformation restraints

Model	NNFF/lr	GDT-HA-num	Avg ΔGDT-HA	RMSD-num	Avg-ΔRMSD
Top 1	638/0.0005	100.8/300	-0.0014	159.8/300	0.0032
	638/0.001	93.4/300	-0.0032	170.2/300	-0.0071
	770/0.0005	98.6/300	-0.0019	138.0/300	0.0036
	770/0.001	92.1/300	-0.0036	130.2/300	0.0072
Best of 5	638/0.0005	171.5/300	0.0017	210.7/300	-0.0156
	638/0.001	170.5/300	0.0011	225.0/300	-0.0293
	770/0.0005	166.3/300	0.0015	203.0/300	-0.0158
	770/0.001	166.5/300	0.0008	218.0/300	-0.0293

3.4 GSFE-refinement performance on refineD data set

In order to further investigate the robustness of GSFE-refinement, we test the performance of the model on the refineD dataset at a learning rate of 0.0005 with 770-NNFF. As shown in Table 4 (see detailed results in Table S13 in supplementary materials), in top 1 models, the GDT-HA score of GSFE-refinement is -0.08 and rank fourth. In best of 5 models, GSFE-refinement ranks sixth, its result (0.4322) is better than FastRelax-0.5Å (0.0548), FastRelax-4.0Å (0.0751), FastRelax (-0.1999), ModRefiner-0 (-0.8400), and ModRefiner-100 (0.1491). These results are generated by 5 iterations. With 50 iterations, the result in the best of 50 model is 0.87, and the GDT-HA-num is 122/150. With superb efficiency of our differentiable refinement algorithm, many more (than 50) iterations are easy to achieve, however, our scoring based on LMLA is not always able to successfully identify the best models. This is to some extent expected as on the one hand only backbone and C_β atoms are used, and on the other hand all prior terms are neglected.

Table 4. Average ΔGDT-HA score of GSFE-refinement and other refinement methods on the

refineD benchmark datasets of 150 targets (results for other methods are taken from Ref [36])

Method	Avg. Top 1	Avg. Best of 5	GDT-HA-num
refineD-C	0.6365	1.3109	121/150
refineD-NC	-1.2403	1.5343	104/150
FG-MD	0.5597	0.5597	-
FastRelax	-3.4317	-0.1999	-
FastRelax-0.5 Å	-0.1403	0.0548	64/150
FastRelax-1.0 Å	-0.3411	0.8811	90/150
FastRelax-2.0 Å	-1.2120	0.8223	77/150
FastRelax-4.0 Å	-2.5471	0.0751	67/150
ModRefiner-0	-0.8400	- 0.8400	-
ModRefiner-100	0.1491	0.1491	-
GSFE-refinement	-0.0800	0.4322	98/150

4. Discussions

The superb efficiency of our algorithm is mainly due to the end to end differentiation such that no sampling is performed at all. AlQuraishi developed end-to-end differentiable structure prediction algorithm based on LSTM (long short term memory) architecture that provide a direct map between sequence and structure [33]. Our algorithm, however, is fundamentally different by realizing direct map from structure to free energy. One great feature of GSFE-refinement is that coordinate transformation module is separated from NNFF. Therefore, future modification of neural network is extremely flexible, as we indeed need such flexibility to advance from present AA level LMLA treatment (Equation 4) to incorporate side chain heavy atoms, local priors and global correlations (Equation 3). Another advantage of GSFE is that direct control of each comprising unit is straight forward with well-defined physical interpretation as demonstrated by addition of local restraints (Equations 10, 11).

As each assessment step provides suggestions of preferable AA identity at each position in primary sequence, simple neural network implementation of GSFE is a natural and highly efficient candidate for protein design, which would be one direction

of our future efforts. Therefore, GSFE theory and GSFE-refinement realize unification of NNFF development, structural model assessment, refinement and design within three thousand lines of codes, and provide clear directions of further development.

Each iteration of GSFE-refinement generates one updated structure. If the NNFF is reasonably accurate, then with more iterations, better structural models would be generated. This is indeed the case. In our benchmark data set, best of 5 models is on average significantly better than top1 models. Our experiments with RefindD data set suggest that with 50 iterations, the best generated model is significantly better than best of 5 models. However, picking out the best model is a more difficulty task than generating it in our case as the best models are not necessarily rank first. At present, we utilized the same NNFF for both assessment and refinement. Training separate NNFF for assessment and refinement might be a worthwhile direction of exploration. However, our priority for future work will be first placed on incorporation of side chain information and other forms of high resolution structural information, and on accounting for prior terms. As these directions of exploration have stronger theoretical justifications.

5. Conclusions

In this work, we develop GSFE-refinement, a fully differentiable protein optimization algorithm based on neural network implementation of LMLA treatment of GSFE theory. Training of NNFF may be completed within a few hours on a single GPU card, and optimization of a typical protein structure is accelerated for approximately 1000,000 times when compared with MD-based sampling approach. Therefore, we successfully accomplish our goal to overcome the limitation of fixed functional form in conventional FF, to tackle the difficulty of parameter updating, and to reduce computation intensity in sampling process of protein structure refinement. In terms of accuracy/reliability of interaction description, performance of optimization suggests that our NNFF based on LMLA and backbone atoms (plus C_β), with help of both local and global restraints, are comparable to present state-of-the-art all-atom force fields. There are a lot of potential

improvement to be added, including increasing source experimental data input, addition of side chain heavy atom information, incorporation of the local prior term (Equation 4) and proper treatment of global (both pairwise and higher ordered) correlations (Equation 4). Separation of coordinate transformation and NNFF in our optimization algorithm is very amenable to various possible upgrading of NNFF. We look forward to performing further investigations in these directions.

References

1. Zhang Y. Protein structure prediction: when is it useful? [J]. Current opinion in structural biology, 2009, 19(2): 145-155.
2. Feig M. Computational protein structure refinement: almost there, yet still so far to go[J]. Wiley Interdisciplinary Reviews: Computational Molecular Science, 2017, 7(3): e1307.
3. Maccallum J L, Hua L, Schnieders M J, et al. Assessment of the protein-structure refinement category in CASP8[J]. Proteins Structure Function and Bioinformatics, 2009, 77 Suppl 9(S9):66-80.
4. Hovan L, Oleinikovas V, Yalinca H, et al. Assessment of the model refinement category in CASP12[J]. Proteins: Structure, Function, and Bioinformatics, 2018, 86: 152-167.
5. Heo, L.; Park, H.; Seok, C. GalaxyRefine: Protein structure refinement driven by side-chain repacking[J]. Nucleic Acids Res. 2013, 41, 384–388.
6. Adiyaman R, McGuffin L J. Methods for the Refinement of Protein Structure 3D Models[J]. International journal of molecular sciences, 2019, 20(9): 2301.
7. Lee, M. R.; Tsai, J.; Baker, D.; Kollman, P. A. Molecular dynamics in the end game of protein structure prediction[J]. Mol. Biol. 2001, 313, 417−430.
8. Fan, H.; Mark, A. E. Refinement of homology-based protein structures by molecular dynamics simulation techniques[J]. Protein Sci. 2004, 13, 211−20.
9. Chen J, Brooks III C L. Can molecular dynamics simulations provide high-resolution refinement of protein structure? [J]. Proteins: Structure, Function, and Bioinformatics, 2007, 67(4): 922-930.
10. Feig M, Mirjalili V. Protein structure refinement via molecular-dynamics simulations: what works and what does not? [J]. Proteins: Structure, Function, and Bioinformatics, 2016, 84: 282-292.
11. Bhattacharya D, Cheng J. i3Drefine software for protein 3D structure refinement and its assessment in CASP10[J]. PloS one, 2013, 8(7).
12. Park H, DiMaio F, Baker D. CASP 11 refinement experiments with ROSETTA[J]. Proteins: Structure, Function, and Bioinformatics, 2016, 84: 314-322.
13. Bhattacharya, D.; Cheng, J. 3Drefine: consistent protein structure refinement by optimizing hydrogen bonding network and atomic-level energy minimization[J]. Proteins: Struct., Funct., Genet. 2013, 81, 119−31.
14. Heo, L.; Park, H.; Seok, C. GalaxyRefine: Protein structure refinement driven by side-chain repacking. Nucleic Acids Res[J]. 2013, 41, W384−8.
15. Park, H.; Seok, C. Refinement of unreliable local regions in template-based protein models[J]. Proteins: Struct., Funct., Genet. 2012, 80, 1974−86.

16. Park, H.; Ko, J.; Joo, K.; Lee, J.; Seok, C.; Lee, J. Refinement of protein termini in template-based modeling using conformational space annealing[J]. Proteins: Struct., Funct., Genet. 2011, 79, 2725−34.
17. Zhang, J.; Liang, Y.; Zhang, Y. Atomic-level protein structure refinement using fragment-guided molecular dynamics conformation sampling[J]. Structure 2011, 19, 1784−95.
18. Best, R.B.; Zhu, X.; Shim, J.; Lopes, P.E.M.; Mittal, J.; Feig, M.; MacKerell, A.D. Optimization of the Additive CHARMM All-Atom Protein Force Field Targeting Improved Sampling of the Backbone φ, ψ and Side-Chain χ 1 and χ 2 Dihedral Angles[J]. Chem. Theory Comput. 2012, 8, 3257–3273.
19. Maier, J.A.; Martinez, C.; Kasavajhala, K.; Wickstrom, L.; Hauser, K.E.; Simmerling, C. ff14SB: Improving the Accuracy of Protein Side Chain and Backbone Parameters from ff99SB[J]. Chem. Theory Comput. 2015, 11, 3696–3713.
20. Zhou H, Zhou Y. Distance-scaled, finite ideal-gas reference state improves structure-derived potentials of mean force for structure selection and stability prediction[J]. Protein Sci. 2002;11(11):2714–26.
21. Shen M, Sali A. Statistical potential for assessment and prediction of protein structures[J]. Protein Sci. 2006;15(11):2507–24.
22. Zhang J, Zhang Y. A novel side-chain orientation dependent potential derived from random-walk reference state for protein fold selection and structure prediction[J]. PLoS One. 2010;5(10).
23. Zhou H, Skolnick J. GOAP: a generalized orientation-dependent, all-atom statistical potential for protein structure prediction[J]. Biophys. 2011;101(8): 2043–52.
24. Andrew Leaver-Fay, Michael Tyka, Steven M Lewis, Oliver F Lange, Philip Bradley. ROSETTA3: an object-oriented software suite for the simulation and design of macromolecules[J]. Methods Enzymol, 2011, 487(11):545-574.
25 Tyka,M.D.; Keedy,D.A.; André,I.; Dimaio,F.; Song,Y.; Richardson,D.C.; Richardson,J.S.; Baker,D.Alternate states of proteins revealed by detailed energy land scape mapping[J].Mol. Biol. 2011,405,607–618.
26 DiMaio, F.; Tyka, M.D.; Baker, M.L.; Chiu, W.; Baker,D. Refinement of Protein Structures into Low-Resolution Density Maps Using Rosetta[J]. Mol. Biol. 2009, 392, 181–190.
27 Mirjalili V, Feig M. Protein structure refinement through structure selection and averaging from molecular dynamics ensembles[J]. Journal of chemical theory and computation, 2013, 9(2): 1294-1303.
28 Mirjalili V, Noyes K, Feig M. Physics-based protein structure refinement through multiple molecular dynamics trajectories and structure averaging[J]. Proteins: Structure, Function, and Bioinformatics, 2014, 82: 196-207.
29 Critical Assessment of Techniques for Protein Structure Prediction. 13 Abstracts. Available online: http://predictioncenter.org/casp13/index.cgi (accessed on 2 April 2019).
30 Alford, R.F.; Leaver-Fay, A.; Jeliazkov, J.R.; O'Meara, M.J.; DiMaio, F.P.; Park, H.; Shapovalov, M.V.; Renfrew, P.D.; Mulligan, V.K.; Kappel, K.; et al. The Rosetta All-Atom Energy Function for Macromolecular Modeling and Design. J. Chem. Theory Comput. 2017, 13
31 Wang S, Peng J, Ma J, et al. Protein secondary structure prediction using deep convolutional neural fields[J]. Scientific reports, 2016, 6(1): 1-11.
32 Xu J. Distance-based protein folding powered by deep learning[J]. Proceedings of the National Academy of Sciences, 2019, 116(34): 16856-16865.

33 AlQuraishi M. End-to-end differentiable learning of protein structure[J]. Cell systems, 2019, 8(4): 292-301. e3.
34 AlQuraishi M. AlphaFold at CASP13[J]. Bioinformatics, 2019, 35(22): 4862-4865.
35 Senior A W, Evans R, Jumper J, et al. Improved protein structure prediction using potentials from deep learning[J]. Nature, 2020: 1-5.
36 Bhattacharya D. refineD: Improved protein structure refinement using machine learning based restrained relaxation[J]. Bioinformatics, 2019, 35(18): 3320-3328.
37 Khatib F, Cooper S, Tyka M D, et al. Algorithm discovery by protein folding game players[J]. Proceedings of the National Academy of Sciences, 2011, 108(47): 18949-18953.
38 Long S, Tian P. A simple neural network implementation of generalized solvation free energy for assessment of protein structural models[J]. RSC Advances, 2019, 9(62): 36227-36233.
39 Goodfellow I, Bengio Y, Courville A. Deep learning[M]. MIT press, 2016.
40 Qian B, Raman S, Das R, et al. High-resolution structure prediction and the crystallographic phase problem[J]. Nature, 2007, 450(7167): 259-264.
41 Nugent T, Cozzetto D, Jones D T. Evaluation of predictions in the CASP10 model refinement category[J]. Proteins: Structure, Function, and Bioinformatics, 2014, 82: 98-111.
42 Adhikari, Badri, Cheng, Jianlin. CONFOLD2: improved contact-driven ab initio protein structure modeling[J]. Bmc Bioinformatics, 19(1):22.
43 Parsons J, Holmes J B, Rojas J M, et al. Practical conversion from torsion space to Cartesian space for in silico protein synthesis[J]. Journal of computational chemistry, 2005, 26(10): 1063-1068.
44 Zemla A. LGA: a method for finding 3D similarities in protein structures[J]. Nucleic acids research, 2003, 31(13): 3370-3374.
45 Ishitani R, Terada T, Shimizu K. Refinement of comparative models of protein structure by using multicanonical molecular dynamics simulations[J]. Molecular Simulation, 2008, 34(3): 327-336.
46 Cao W, Terada T, Nakamura S, et al. Refinement of comparative-modeling structures by multicanonical molecular dynamics[J]. Genome Informatics, 2003, 14: 484-485.