Combining Protein Fragment Feature-Based Resampling and Local Optimisation

Trent Higgs, Lukas Folkman, and Bela Stantic

Institute for Integrated and Intelligent Systems, Griffith University, Australia

Abstract. Protein structure prediction (PSP) suites can predict 'near-native' protein models. However, not always these predicted models are close to the native structure with enough precision to be useful for biologists. The literature to date demonstrates that one of the best techniques to predict 'near-native' protein models is to use a fragment-based search strategy. Another technique that can help refine protein models is local optimisation. Local optimisation algorithms use the gradient of the function being optimised to suggest which move will bring the function value closer to its local minimum. In this work we combine the concepts of structural refinement through feature-based resampling, fragment-based PSP, and local optimisation to create an algorithm that can create protein models that are closer to their native states. In experiments we demonstrated that our new method generates models that are close to their native conformations. For structures in the test set, it obtained an average RMSD of 5.09 Å and an average best TM-Score of 0.47 when no local optimisation was applied. However, by applying local optimisation to our algorithm, additional improvements were achieved.

1 Background

A fundamental aspect to modern molecular research is being able to elicit the three-dimensional structure of protein molecules. To date, there are roughly 20 million protein sequences stored in the UniProtKB/TrEMBL databases [1], but approximately only 79,000 of these sequences have available solved structures. Furthermore, it has been demonstrated that even a single amino acid substitution in a protein sequence may result in significant changes in protein stability and structure [2]. This makes it difficult for molecular and cell biologists who need the three-dimensional structure of proteins for their research. Due to so many proteins lacking solved structures, a lot of focus has been placed on improving and developing new computational *protein structure prediction* (PSP) methods.

Computational PSP methods have been historically broken up into three categories. In comparative modelling [3], evolutionary related homologous templates that have a high sequence similarity to the target sequence are identified. Then, the target and templates are aligned to form a three-dimensional structure of the target protein. Finally, this is completed by combining models for loop regions and other segments that do not align properly between the template and target. On the other hand, proteins that belong to different evolutionary classes can

A. Ngom et al. (Eds.): PRIB 2013, LNBI 7986, pp. 114–125, 2013.

have similar structures too. Therefore, threading methods [4] have been developed to allow a query sequence to be mapped directly onto three-dimensional structures of solved proteins. The main motivation here is to recognise folds that are similar to the query even if no evolutionary relationship between the query and the template protein is present. Finally, the last category, *ab initio* [5], is used when the query sequence has no evolutionary related proteins in the template library. This is the most challenging approach, and success is at present limited to small proteins.

PSP has been tackled from numerous angles using one or more of the above methods. Some of the most successful approaches for *ab initio* are techniques that employ a fragment-based search strategy (e.g., Rosetta [6] or I-Tasser [7]). Fragments are derived from protein structures stored in the Protein Data Bank (PDB) based on the likelihood that a segment of the target protein chain will fold into a similar motif that already exists within a structure deposited in the PDB. This fragment-based approach has many benefits, for instance, by using fragments, we can approximate the populated areas of the local potential energy surface for the backbone of the protein structure. This stems from philosophy that when a protein is folding, the local structure will switch between numerous possible local conformations [8]. Therefore, each fragment can be considered a possible candidate for a conformation of the local sequence, which allows an energy function to be used that does not explicitly calculate the local interaction energy (the fragment selection method has already considered local interactions). This simplification is helpful in the PSP process because calculating the interaction energy assumes that a correct potential energy surface is known, which may not be the case. Finally, one of the main benefits of using a fragment-based approach is that we can easily move a protein from one topological isomer to another through a single fragment replacement. This ability can be looked at as moving a protein from one local minimum on the local physical energy surface to another, which is difficult to do in a more continuous based search method like molecular dynamics due to the computational complexity of such a move.

Another technique that has been applied to the PSP problem to help improve prediction accuracy is local optimisation. Local optimisation algorithms use the gradient of the function being optimised to determine which move will bring the function value closer to its local minimum. There are many different methods that have been proposed for this purpose [9]. For example, *linear minimisation* performs a single step based on the gradient, and after a number of recursive invocations, it reaches the local minimum. Compared to other available methods, it is considered rather slow. A variety of *quasi-Newton* methods were proposed in order to tackle local optimisation more efficiently. *Davidon-Fletcher-Powell* and *Broyden-Fletcher-Goldfarb-Shanno* (BFGS) methods are such examples. In both cases, the descent's direction and step is computed according to the gradient and second derivatives of the function. The second derivatives are held in the form of *Hessian matrix* which can be efficiently updated. The extra information accumulated by these methods improves their efficiency, so that they converge faster. Furthermore, inexact search modifications of these methods have also

been proposed. They converge even faster, however, they do not necessarily reach the local minimum. Examples of these are *Armijo rule* and *non-monotone* modifications. In the latter case, the function value can be temporarily increased which may help escape shallow local minima. In another example, the *limited memory* variation of the BFGS method (L-BFGS) [10], instead of storing the whole Hessian matrix, only the vectors which represent the matrix implicitly are held in the memory.

Due to the success that fragment-based techniques have had, and the importance of local optimisation to keep every predicted model at the bottom of its energy basin, we combined both of these concepts to develop a PSP resampling approach that should be able to produce more accurate models. To achieve this, we carried out tests to identify which local optimiser performs the best and incorporated this optimiser into a *fragment feature-based resampling* approach which is discussed in more detail in the next section.

2 Methods

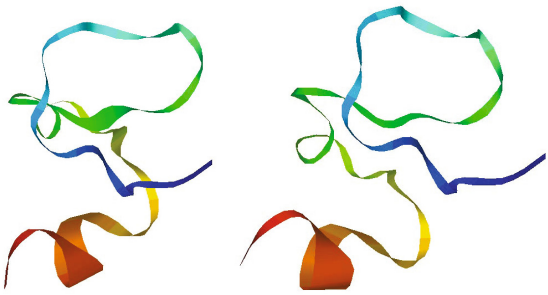
Local optimisation methods can be applied to the prediction process to guarantee that a PSP solution reaches the bottom of its energy basin. To determine the best local optimisation method for the PSP problem, we carried out tests utilising five state-of-the-art algorithms: *Linear Minimisation* (Lin-Min), *Broyden-Fletcher-Goldfarb-Shanno* (BFGS), *BFGS Armijo* (BFGS-A), *BFGS Armijo Non-monotone* (BFGS-A-NM), and *Limited Memory BFGS* (L-BFGS). To supplement these results and gauge the usefulness of local optimisation in the protein structure resampling process, the most promising algorithm was applied to our newly created *fragment feature-based resampling* approach. This new resampling algorithm builds on the concepts of our previous works [11–13].

In the next sections, our approach to analyse local optimisation techniques and our newly developed fragment feature-based resampling algorithm, which was designed to generate good starting points for local optimisation, are explained.

2.1 Local Optimisation

To identify which local optimisation methods perform well on the PSP problem, 128 native protein structures were selected and small random perturbations were applied to them in order to observe how successfully the local optimisers could guide these structures back to their native conformations. These native proteins structures were obtained from the CASP 8 website [14]. The *centroid energy function* [8] was chosen to be the objective function to be minimised using each of the five local optimisation methods (Lin-Min, BFGS, BFGS-A, BFGS-A-NM, L-BFGS). The same energy function was used for the implementation of our fragment feature-based resampling approach.

The general procedure used to test how well a local optimiser performed was by perturbing each structure by a certain amount of residues (between 1 and 3) and degree of movement (between 1 to 15 degrees), applying local optimisation,



(a) Original structure (b) Altered structure

Fig. 1. An example of a protein that had three of its residues perturbed by 15 degrees. Notice that the structure in (b) has several features displaced compared to its original structure in (a). All images were generated using Rasmol [16].

and then, evaluating how much the energy and structural similarity changed. The evaluation was carried out by recording the initial energy of the native structure, then recording the energy and root mean square deviation (RMSD) [15] of the altered structure, and finally, recording the energy and RMSD of the structure after local optimisation. The averages (across the set of all structures) of these values were then used as our final results. An example of one of our perturbations can be found in Figure 1.

2.2 Fragment Feature – Based Resampling

In our previous works on feature-based resampling using a genetic algorithm (GA) [11, 12], we demonstrated that by combining 'native-like' features generated from decoys from other PSP approaches, we could produce structures that were closer to the native conformations. To further this work, we created a *fragment feature-based resampling* algorithm to create 'near-native' starting points for local optimisation.

In our *GA feature-based resampling* algorithm [11, 12], our features were stored as the initial population in the form of decoys outputted from an initial prediction run. Then, crossover and mutation techniques were applied to them throughout the prediction process using energy function for fitness calculations. This was accomplished by using a crossover operator that splices together protein fragments that have 'native-like' features according to the fitness function f. Our GA's crossover operator randomly selected a crossover point (n) where $n \in C_\alpha(S)$ ($C_\alpha(S)$ refers to the set of C_α atoms contained within the structure S). Let $p1$ be parent 1, and $p2$ be parent 2. Everything from n onwards in $p1$ is replaced with everything from n onwards in $p2$, and vice versa. This process produced two offsprings.

In this work, we created a *fragment* feature-based resampling algorithm to overcome some of the limitations that were apparent from our results, the most

obvious being the inconsistencies of the energy function. The *centroid energy function* is not optimised to minimise its energy score in correlation with the RMSD of the structure being predicted, which has been discussed in our previous works [11, 12] and also shown in [17, 18]. This lack of accuracy can heavily affect the GA optimisation process as it relies on the energy function to guide it to more accurate solutions. To combat this, we developed an algorithm that incorporates *random* feature-sampling from a set of 'near-native' fragments.

Our algorithm works by taking a set of protein decoy structures and creating a fragment library from them. Each structure in the library can be broken into numerous fragments of different sizes. Sampling this space is then carried out by randomly selecting a position in the fragment library, randomly picking a fragment size (based on how much of the structure is left to put together), and finally, extracting that fragment based on the position of the structure being processed and the length of the fragment. There are two main constraints that our algorithm imposes on this fragment assembly procedure: (1) no structure can contain more than half of the residues of any given structure within the fragment library (to avoid duplicating any structure that was produced by the PSP suite), and (2) structures must have no collisions between residues.

The assembly process described above is run until 2,000 structures are generated. Based on our initial testing, we concluded that 2,000 structures is a sufficient amount of runs to generate most of the feasible combinations from the set of structures contained in our fragment library. As mentioned above, because we use an exhaustive search process, the energy function is only used to evaluate how well energy function can identify 'near-native' structures generated from our fragment feature-based resampling approach. Evaluation of the final output is carried out by two structural measures: RMSD [15] and template modelling score (TM-Score) [19].

3 Results and Discussion

We carried out two main tests: (1) assessment of which local optimiser performed the best in guiding structures back to their native conformations after random perturbation, and (2) evaluation of our fragment feature-based resampling algorithm with and without local optimisation. In the local optimiser test, 128 native proteins were randomly perturbed using the following criteria: 1 residue by 1 degree, 1 residue between 1–3 degrees, 2 residues between 1–5 degrees, 3 residues between 1–5 degrees, and 3 residues between 10–15 degrees.

For fragment feature-based experiment, the test set contained 14 protein structures. Our fragment library contained 1,000 structures for each prediction, and all fragments were generated from decoys. The local optimiser used for these tests was the one that performed the best in our first experiment. Each protein prediction was run five times, and the best output from each test was averaged for our final results to remove any bias caused by the random fragment assembly process. The best structure was chosen based on its RMSD value to its native conformation.

3.1 Empirical Results

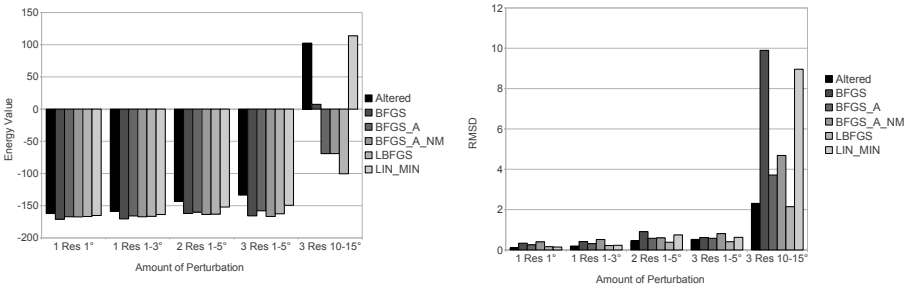
Figure 2 depicts the results that were gained from our perturbation experiments. The x axis is the amount of perturbation, and the y axis is the energy and RMSD values (Figure 2a and 2b, respectively). To complement these results, the local optimiser's ability to guide an altered structure back to its native conformation is visually demonstrated in Figure 3. Table 1 shows the results gained from our fragment feature-based approach. For each protein, the average best energy, RMSD, and TM-Score over the five tests with and without local optimisation are displayed. Finally, Figure 4 depicts the prediction ability of our fragment feature-based resampling by providing some visual comparisons between our models and their native conformations.

3.2 Analysis of Results

Local Optimiser Comparison. In our perturbation experiments, we used 128 protein structures, applied different amounts of perturbation to them, and then, locally optimised these structures. The average results for these experiments can be found in Figure 2. In Figure 2a, for the first four perturbation classes, it can be seen that all the local optimisers minimised the energy values starting from the altered structure. Also, in each of these cases every local optimiser achieved roughly the same energy levels after minimisation. For example, in Figure 2a, for the first perturbation class (1 residue with a perturbation of 1 degree), each optimiser generated models with an average energy between -165 and -171. However, in the last case (3 residues with a perturbation of 10–15 degrees), only BFGS-A, BFGS-A-NM, and L-BFGS minimised the energy significantly when compared to the average altered energy, with L-BFGS being the best. This suggests that the more a structure is altered from its native conformation, BFGS-A, BFGS-A-NM, and L-BFGS are more likely to guide it back to a stable state.

Other than just looking at the minimisation of the energy function to tell us which local optimiser performed the best, their ability to minimise the RMSD value of a structure was also evaluated. This would allow us to know which optimiser could lower the energy of a structure while also guiding it back to its native conformation. The results can be found in Figure 2b. From these results, it is clear that out of all the optimisers, only L-BFGS significantly guided altered structures back towards their native conformations. All the others had some success, but on average, they actually moved structures further away from their native state than the perturbation itself (this can be seen in Figure 2b where all the optimisers in every perturbation class, except L-BFGS, actually have worse RMSD averages when compared to the average altered RMSD).

Analysing the various perturbation classes in Figure 2b, it can be seen that even Lin-Min did well in minimising small perturbations (first two perturbation classes), however, as the structural change increased, its ability to move a structure back to its native state deteriorated, eventually becoming one of the worst out of the five we tested. It was also one of the worst optimisers at lowering the energy after a perturbation was made. L-BFGS, on the other hand,



(a) Energy (b) RMSD

Fig. 2. Results for our local optimiser comparison. In (a), the results of how well each local optimiser minimised the energy function are shown, and in (b), the results how well each optimiser performed in moving the altered structures back towards their native conformation are depicted. Note that these results are averaged from our complete 128 protein set, and the averages for the perturbed structures before local optimisation was applied are also included.

(a) Native (b) Altered (c) Optimised with L-BFGS

Fig. 3. Visual comparison of the native, altered and optimised structures. (a) is the native structure before perturbation, (b) is the altered structure, and (c) is the structure after L-BFGS optimisation was applied. As it can be seen in (c), once local optimisation was applied on the structure in (b), it moved back to its native structure. All images were generated using Rasmol [16].

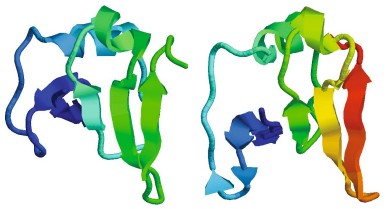
appears to always move the structure back towards its native state. From these findings, we can conclude that out of the five tested local optimisers, L-BFGS was most successful in regards to minimising the energy of structures after being perturbed while at the same time being able to guide the altered structures back towards their native states. To demonstrate the success of the L-BFGS optimiser, Figure 3 allows for a visual comparison of the native conformation, the perturbed structure, and the optimised structure using L-BFGS. It can be seen that the L-BFGS optimiser moved the altered structure back towards its native conformation by shifting the α-helices back into their correct places.

Fragment Feature-Based Resampling. After the perturbation experiment, we performed tests on our new fragment feature-based resampling approach, both with and without the L-BFGS optimiser. The results from these experiments can be found in Table 1. These results indicate that our new algorithm can resample features in such a way that on average 'near-native' models are generated. This is supported by an average best RMSD of 5.09 Å and an average best TM-Score of 0.48 when no local optimisation was applied. Another interesting aspect of these experiments is that the energy for the best scoring models (in terms of RMSD and TM-Score) have quite high energy scores. On average, they are not even in the negatives, meaning that the centroid energy function is rather limited in regards to finding structures that are low in RMSD. This is not to say that the centroid energy function is wrong as it has been proven that it works well in finding compact structures that are roughly close to their native states, but it lacks the accuracy to find models at a finer atomic resolution. A graphical representation of the predictive power of our fragment feature-based algorithm can be seen in Figure 4.

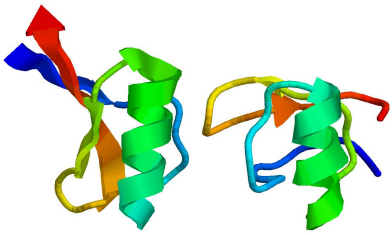
The next set of tests combined our algorithm with the L-BFGS local optimiser, which performed the best in our perturbation tests. In this experiment, we gained an average best RMSD of 5.05 Å and an average best TM-Score of 0.50. This

Table 1. Fragment feature-based resampling without and with local optimisation

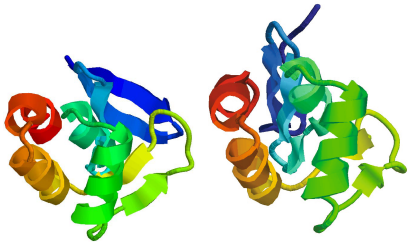
Protein	Without local optimisation			With local optimisation		
	f	***RMSD***	***TM-Score***	***f***	***RMSD***	***TM-Score***
79.1a91A	119.35	5.69 Å	0.37	142.73	5.70 Å	0.37
78.1aoyA	59.83	5.00 Å	0.55	43.26	4.99 Å	0.53
43.1bdsA	115.76	5.85 Å	0.23	89.52	5.61 Å	0.28
99.1bm8A	14.91	7.65 Å	0.29	62.96	7.62 Å	0.29
110.1brsABC	11.29	7.64 Å	0.50	42.45	7.74 Å	0.56
67.1cspA	12.20	2.95 Å	0.65	−18.66	2.75 Å	0.68
54.1enhA	65.25	5.13 Å	0.26	84.43	5.03 Å	0.28
76.1d3zA	−16.75	2.36 Å	0.76	−27.78	2.30 Å	0.76
47.1gptA	20.62	4.94 Å	0.38	75.19	5.03 Å	0.38
74.1kjsA	50.32	3.87 Å	0.55	32.37	3.91 Å	0.53
83.1pgxA	31.98	3.77 Å	0.66	−11.35	3.78 Å	0.66
77.1vccA	26.45	3.11 Å	0.67	12.81	3.19 Å	0.66
107.2pppA	164.93	8.57 Å	0.40	123.08	8.12 Å	0.49
78.2ptlA	39.38	4.70 Å	0.51	−14.94	4.89 Å	0.47



(a) 1vcc native (b) 1vcc predicted



(c) 1gpt native (d) 1gpt predicted



(e) 1bm8 native (f) 1bm8 predicted

Fig. 4. In (a), (c), and (e), the native conformations for proteins 1vccA, 1gptA, and 1bm8A, respectively, are depicted, and in (b), (d), and (f), the predicted models for these proteins using our fragment feature-based resampling algorithm are shown (note that local optimisation was not used on these structures). All images were generated using Rasmol [16].

means that irrespectively of the measure employed for the comparison, there were additional relative improvements (0.8% and 4.2% in the case of RMSD and TM-Score, respectively). The main reason why local optimisation in this case did not result in higher improvements was that the fragments were obtained from decoys which had already been locally optimised. However, if the algorithm was designed to fold protein structures from just the amino acid sequence, local optimisation would definitely be more useful.

There are aspects to our fragment feature-based approach that could be addressed to obtain further improvements. The first one is the problem of missing features in the fragment library. As features generated by other PSP suites are used in our approach, if the initial decoys do not contain all features necessary

to create the native conformation for a given protein, then, our algorithm will produce poor results. In most cases, given our results, nearly all features were present, however, an example of this problem occurring can be seen in Figures 4c and 4d. In Figure 4d, one of the major β-sheets was predicted incorrectly and also has the wrong orientation, which brings us to the other problem: the orientation of features.

Our approach stitches fragments together until the end of the protein chain is reached, however, it never takes into consideration the orientation these features should have. Figure 4 illustrates that some of the major reasons why we did not obtain better results was due to the orientation of the features. To combat this problem, we could add a move set that rotates the fragments around until their optimal placements are found. This brings up two challenges: firstly, a scoring function that can inform us what the best orientation is for a fragment or a set of fragments, and secondly, how much rotation should be applied. According to the literature, once a compact structure has been obtained it is best to only move fragments slightly (e.g., 1–5 degrees) [8]. If both of these problems were addressed, our algorithm could generate even better models than it already had.

4 Conclusions

Fragment-based protein structure prediction methods have shown a lot of success in predicting the three-dimensional conformations of proteins. In this paper, we combined fragment-based approach and local optimisation techniques. By doing this, we showed that our new *fragment feature-based resampling* algorithm can generate protein models close to native structures. Furthermore, we described the benefits and disadvantages of using local optimisation techniques in conjunction with feature-based resamplig.

To identify which local optimisation methods performed well on the PSP problem, we selected 128 native protein structures to which we applied small random perturbations in order to observe how successfully local optimisation could guide structures back to their native conformations. The five optimisers we tested were: *linear minimisation* (Lin-Min), *Broyden-Fletcher-Goldfarb-Shanno* (BFGS), *BFGS Armijo* (BFGS-A), *BFGS Armijo non-monotone* (BFGS-A-NM), and *limited memory BFGS* (L-BFGS). To supplement these results and gauge the usefulness of local optimisation in the protein structure resampling process, we took the most promising method from our perturbation experiment and combined it with a fragment feature-based resampling approach, which we proposed in this work.

Our new fragment feature-based resampling algorithm works by creating a fragment library from a set of protein decoys. Each structure in the library can be broken up into numerous sized fragments to build up 'near-native' protein models. Sampling this space is carried out by randomly combining fragments together until 2,000 collision-free structures are produced.

From our experimentation, we observed that the L-BFGS optimiser performed the best. It was able to both minimise the energy of a structure and bring a

structure back towards its native state. In regards to our fragment feature-based resampling algorithm, we demonstrated that it could generate 'near-native' models. Out of the 14 structures we tried to predict, it obtained an average best RMSD of 5.09 Å and an average best TM-Score of 0.47 when no local optimisation was applied. When we applied local optimisation, additional improvements in both RMSD and TM-Score were recorded.

As mentioned in our results discussion and analysis, there is two avenues to further improve our algorithm. First, being able to ensure that all features which are needed to generate the native conformation are present in the fragment library. However, this may be in some cases rather difficult as we are unsure what features the native model contains, but the probability could be increased if there is a sufficiently large library. And second, finding the correct orientation of the fragments is crucial to allow more accurate models to be produced.

References

1. Consortium, U.: The universal protein resource (uniprot) 2009. Nucleic Acids Research 37, D169–D174 (2009)
2. Folkman, L., Stantic, B., Sattar, A.: Sequence-only evolutionary and predicted structural features for the prediction of stability changes in protein mutants. BMC Bioinformatics 14(suppl. 2), S6 (2013)
3. Sali, A., Blundell, T.: Comparative protein modelling by satisfaction of spatial restraints. Journal of Molecular Biology 234(3), 779–815 (1993)
4. Zhang, Y., Skolnick, J.: Automated structure prediction of weakly homologous proteins on a genomic scale. PNAS 101(20), 7594–7599 (2004)
5. Simons, K.: et al. Prospects for ab initio protein structural genomics. Journal of Moleculer Biology 306, 1191–1199 (2001)
6. Meredith, D.: Rosetta tackles the extreme origami of protein folding. HHMI Bulletin 14, 20–23 (2001)
7. Zhang, Y.: Template-based modeling and free modeling by I-TASSER in CASP7. Proteins 8, 108–117 (2007)
8. Rohl, C., Strauss, C., Baker, D.: Protein structure prediction using rosetta. Methods Enzymology 383, 66–93 (2004)
9. Bonnans, J.: Numerical optimization: theoretical and practical aspects, 2nd edn. Springer (2006)
10. Liu, D., Nocedal, J.: On the limited memory BFGS method for large scale optimization. Mathematical programming 45(1), 503–528 (1989)
11. Higgs, T., Stantic, B., Hoque, T., Sattar, A.: Genetic algorithm feature-based resampling for protein structure prediction. In: IEEE World Congress on Computational Intelligence, pp. 2665–2672 (2010)
12. Higgs, T., Stantic, B., Hoque, T., Sattar, A.: Refining genetic algorithm twin removal for high-resolution protein structure prediction. In: IEEE Congress on Evolutionary Computation CEC 2012, 251–258 (2012)
13. Folkman, L., Pullan, W., Stantic, B.: Generic parallel genetic algorithm framework for protein optimisation. In: Xiang, Y., Cuzzocrea, A., Hobbs, M., Zhou, W. (eds.) ICA3PP 2011, Part II. LNCS, vol. 7017, pp. 64–73. Springer, Heidelberg (2011)
14. CASP8: 8th community wide experiment on the critical assessment of techniques for protein structure prediction (2008), `http://predictioncenter.org/casp8/` (last accessed: July 2012)

15. Carugo, O.: Statistical validation of the rootmeansquaredistance, a measure of protein structural proximity. Protein Engineering, Design and Selection 20(1), 3338 (2007)
16. Sayle, R.: Molecular visualization freeware and rasmol classic site (2009), `http://www.umass.edu/microbio/rasmol/index2.htm` (last accessed: February 2011)
17. Bowman, G., Pande, V.: Simulated tempering yields insight into the low-resolution rosetta scoring functions. Proteins: Structure, Function, and Bioinformatics 74, 777–788 (2009)
18. Shmygelska, A., Levitt, M.: Generalized ensemble methods for de nova structure prediction. PNAS 106(5), 1415–1420 (2009)
19. Zhang, Y., Skolnick, J.: Scoring function for automated assessment of protein structure template quality. Proteins 57, 702–710 (2004)