# Impact of side chain positioning on the accuracy of discrete models ⋆

Miguel M. F. Bugalho *,1 Arlindo L. Oliveira

*INESC-ID, R. Alves Redol 9, 1000 LISBOA, PORTUGAL*

**Abstract**

Discrete models are important to reduce the complexity of the protein folding problem. However, a compromise must be made between the model complexity and the accuracy of the model.

Previous work by Park and Levitt has shown that the protein backbone can be modeled with good accuracy by four state discrete models. Nonetheless, for ab-initio protein folding, the side chains are important to determine if the structure is physically possible and well packed.

We extend the work of Park and Levitt by taking into account the positioning of the side chain in the evaluation of the accuracy. We show that the problem becomes much harder and more dependent on the type of protein being modeled. In fact, the structure fitting method used in their work is no longer adequate to this extended version of the problem. We propose a new method to test the model accuracy.

The presented results show that, for some proteins, the discrete models with side chains cannot achieve the accuracy of the backbone only discrete models. Nevertheless, for the majority of the proteins an RMSD of four angstrom or less is obtained, and, for many of those, we reach an accuracy near the two angstrom limit. These results prove that discrete models can be used in protein folding for obtaining low resolution models. Since the side chains are already present in the models, the refinement of these solutions is simpler and more effective.

*Key words:* Protein models, discrete state models, side chain positioning, protein folding

⋆ Partially supported by project Biogrid POSI/SRI/47778/2002

* Corresponding author.

*Email addresses:* mmfb@kdbio.inesc-id.pt (Miguel M. F. Bugalho), aml@inesc-id.pt (Arlindo L. Oliveira).

[1] Supported by the Portuguese Science and Technology Foundation by grant SFRH/BD/13215/2003

## 1 Introduction

The ab-initio protein folding problem consists in determining the structure of a protein using only the information of its amino acid sequence. Even extremely simplified versions of this problem have been proved to be NP-Hard[12,6,1,14].

In a protein structure there are several structural constrains. For the atomic angles and bond lengths the variation is small and, thus, the majority of the folding algorithms focus on the dihedral angles. In addition to the structural constrains, the protein structures are defined by the atomic interactions. Although the dihedral angles have optimal values they usually assume different values to allow for interactions between atoms.

In the context of this work, a discrete state model is an all heavy atoms protein model that uses a discrete set for the possible values of the dihedral angles. The atomic bond lenghts and angles are considered fixed at the optimal values. Previous work by[19] has shown that discrete state models with a limited number of states (four) can describe proteins with relatively good accuracy. In that work the authors have also shown that, for the same degree of complexity, off lattice discrete state models are more accurate than lattice models.

In the work of Park and Levitt only the main chain is used and no consideration is made for clashes between atoms. In this work we will analyze the accuracy of discrete state models using an all heavy atoms representation and disallowing atomic clashes. Using all heavy atoms representations requires positioning of the side chains. A simple positioning method based in rotamer libraries is proposed.

### *1.1 Motivation*

Although important studies on the accuracy and application of discrete models, like the referred work of[19] and the work of[21], where published more then 10 years ago, discrete models are still being studied and applied to problems in recently published works. Discrete models are used in studies for ab-initio protein folding[13,17]. Recent works also use discrete models for generating ensembles of structures[18,7]. The study of discrete protein models is, therefore, highly relevant. Although the models presented by Park and Levitt are used in protein folding problems, the models where only shown to be accurate for modeling the backbone. Using only backbone models can produce physically impossible models. Moreover, depending on the scoring function, these models may have a high score and may be chosen as the best model. Therefore, to avoid physically impossible models, we have extended this work by considering side chain position and atomic clashes.

We can divide protein models in two general classes: discrete models and continuous models. For protein folding in continuous models a scoring function, normally an energy function, is used in conjunction with some minimization technique such as simulated annealing, gradient descent or other. In discrete models the problem is reduced to a discrete number of choices for each amino-acid. This reduces the complexity of the problem but makes it unlikely to find the exact model. Consequently, the discrete models are particulary fit to perform high level structure search, since the search space is greatly reduced and very similar structures can be more easily avoided.

The applicability of the discrete models relies on the solution of three difficulties, since discrete models:

- require a scoring function that can ignore the atomic details giving high scores to physically inexact, near native structures.
- need a search technique that can efficiently search the structure space without enumerating all the structures, since the space size is still exponential on the size of the protein.
- need to use a set of dihedral angle values that can accurately model the protein. Modeling the protein means that it should be possible to create a structure with a low root mean square distance to the native one, but also that it should be possible to create feasible structures, which avoid atomic clashes and, when possible, emulate the secondary structure, atomic contacts or other characteristics that define the structure of a protein. This is important since the scoring function must be able to find, in the models, characteristics that are similar to the characteristics of native protein.

The first difficulty can be solved using a statistical scoring function. The second is the final step towards finding a near native structure and a number of techniques have been proposed. However, the search will only work if a good discrete model is available. The third difficulty is, therefore, the focus of this work.

As referred before, the model must be able not only to approximate the atomic positions of the native structure, but also to avoid unfeasible structures, *e.g.* structures with atomic clashes or unrealistic side chain conformations. Allowing such errors to occur would provoke too much noise in the scoring function, since it would be possible to pack residues in conformations that are not allowed in the real fold process. Therefore, we will analyze the accuracy of known discrete models, while considering clashes and side chain positioning.

## 2 Side Chain Positioning

One of the most used methods for side chain positioning is based on the use of rotamers libraries. In this work we used the Dunbrack Backbone-Dependent Rotamer Library[8,9,10,11] which contains information about each amino acid side chain. For each amino acid, the library has the side chain dihedral angle values indexed by the backbone $\phi/\psi$ pairs in slots of 10 degrees. The library also contains the observed frequency of the side chain dihedral angles in the particular slot. To reduce the number of possible conformations we used the frequency information contained in the library to prune less probable configurations. Unless otherwise stated, a 0.04 (4%) frequency cutoff was used.

The decision to use a rotamer method was made for two reasons:

- The method creates a discretized set of highly probable configurations. This avoids the usage of continuous minimization methods.
- Since the rotamers are indexed by $\phi$ and $\psi$ backbone angles, there is no need to verify clashes between atoms of the same amino acid.

In this paper we present a simple positioning algorithm that tests the possible rotamer conformations until a valid conformation is found. The frequency information is used to establish the testing order. Rotamers that occur more frequently in known proteins will be tested first. Notice that we only want to verify if there is a possible side chain configuration and not to set the best one. The best side chain conformation can be set in the end during refinement using, for example, the SCWRL program[5], which sets the side chain using a graph theory based algorithm and an energy function. Since SCWRL does not change the backbone it is essential that sufficient space is left for the side chains.

After choosing a possible configuration, the algorithm continues to construct the protein model by extending the backbone configuration. However, this particular choice of the side chain configuration can be changed latter. The side chain is modified if, afterwards, that particular side chain configuration prevents other side chains from being positioned. When a clash between side chains is detected (no changes in previous backbone atoms are allowed) the algorithm sets the new side chain in the configuration with less conflicts and tries to reconfigure the old side chain. If a new side chain conflict is found the process is restarted until a predefined threshold for the number of side chain changes is reached. Figure 1 shows a example of successful side chain reconfiguration.

If an unresolvable conflict is found, like a side chain versus backbone conflict, or the backtrack threshold is reached, the algorithm considers that there is not enough space for the side chain and reports a clash. To avoid loops the

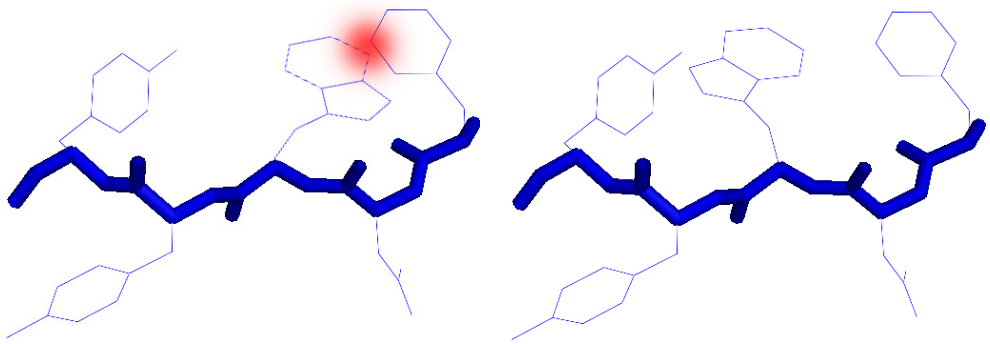


Fig. 1. Example of a successful side chain reconfiguration. Both the backbone, stronger lines, and the side chains are represented. The red sphere shows the side chain conflict between the new side chain and a previously set side chain. A new configuration is chosen for the older side chain by rotating it to the opposite side of the new side chain position.

algorithm also does not allow for the same side chain to be changed twice.

## 3 Discrete State Model

Previous work by[19] has tested various discrete state models and presented some reasonably accurate sets of states. The test made by Park and Levitt consisted in fitting the discrete models into the backbone of the known structure. The side chains, and possible clashes between atoms, were not considered in the fitting problem. In this section we will analyze some of the models described by Park and Levitt in a test platform that considers side chains and atomic clashes.

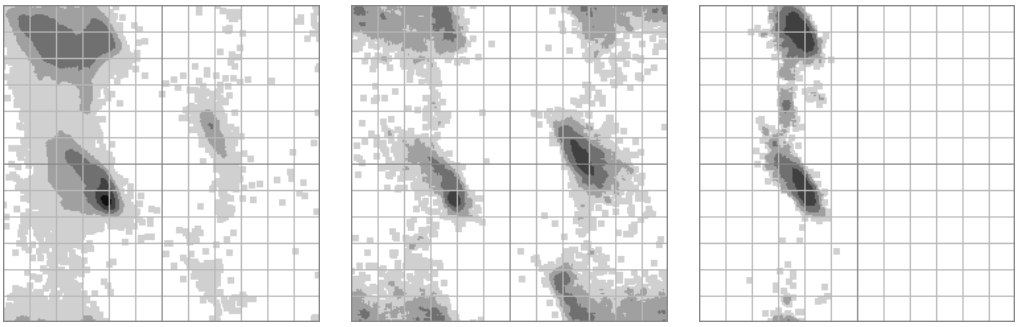
Table 1 shows the discrete models previously proposed by[19]. We used four of these discrete models: three four states models and one six states model. We have chosen the Rooman et al. six states model since it had better results than the one proposed by Park and Levitt. In the four states models, we chose model C because it was the best model and model A because it had the best results for the Alpha and Beta secondary structures. Model G was chosen randomly from the rest.

If we analyze the angle sets in terms of physical correction, we can notice that only the best model, C, is near the probable zones of the Ramachandran plot[20]. The Ramachandran plots depict the probability distribution for the $\phi$ and $\psi$ angles in known proteins. Although a model might be near the true structure even if the angles in that model fall outside the most probable zone, the torsion angles will probably be very different from the true angles. Figure 2 shows examples of Ramachandran plots taken from the web site of Deniz Yuret (http://www.denizyuret.com/bio/). Nevertheless, the torsion an-

| Name | Set of Pairs of Angles |
|---|---|
| ∗ A | {(-64,-40),(-123,134),(111,-46),(117,105)} |
| B | {(-66,-40),(-119,114),(-36,124),(132,-40)} |
| ∗ C | {(-63,-63),(-132,115),(-42,-41),(-44,127)} |
| D | {(-58,-31),(-127,126),(-97,-24),(109,108)} |
| E | {(-71,-57),(-131,122),(-42,-36),( 107,-25)} |
| F | {(-58,-51),(-133,135),(-33,174),(114,-40)} |
| ∗ G | {(-56,-48),(-129,128),(-108,35),(-31,-109)} |
| H | {(-74,-31),(-131,125),(-101,179),(105,-40)} |
| 6 states | {(-57,-47),(-139,135),(-119,113),(-49,-26),(-106,48),(-101,-127)} |
| ∗ Rooman et al.[21] | {(-65,-42),(-123,139),(-70,138),(-87,-47),(77,22),(107,-174)} |

Table 1
Discrete State models used in the work of Park and Levitt[19]. The ∗ signals the models used in this work.

gles (or dihedral angles) errors made by the models can be easily corrected in refinement steps. Therefore we will focus on the correction of the overall structure.



Non GLY/PRO GLY PRO

Fig. 2. Ramachandran plots for the proline (right), glycine (center) and other types of amino acids (left). Figures taken from the web site of Deniz Yuret (http://www.denizyuret.com/bio/).

## 4 Testing Method

We used an efficient clash detection algorithm, proposed by[4], and the side chain positioning algorithm presented earlier. The clash detection algorithm uses a three dimensional hash table and has constant clash detection time. To set the side chains we used a maximum number of possible alterations (see

section 2) equal to the number of amino acids divided by 10. Since the number of possible conflicts may increase with the size of the protein (but only until a core of well packed atoms is formed), we increase the number of possible alterations linearly with the size of the protein.

We have first tried the algorithm presented by[19] for testing discrete state models. The algorithm does a simple beam search using the RMSD distance as the scoring function. In a beam search, $n$ states are saved at any given time. Considering that there are $m$ choices in an $m$ state discrete model, the algorithm starts by testing the first $m$ choices. The algorithm then chooses the best $n$ states and tests all the choices for each of those states ($n \times m$ tests). The best $n$ states are chosen and the same steps are iterated until the final configurations are reached.

Although the beam search method obtained good results in the backbone only problem[19], for the problem presented here the results were much worse. In fact, when the protein size was greater than 80 amino acids, a solution with an RMSD near five angstrom was difficult to obtain with this method. The beam search approach is a non exact method. An explanation for this, already presented by[19], is that the neglected search states, although with worse RMSD values at the time they where removed from search, may in reality provide better fits in the long run. Since the number of states increases exponentially not even an increase in the beam size can prevent this. With side chains, this problem is greatly aggravated, since many states will prove to be dead ends, because of collisions, or will just be driven away from the best fit because of them. Because of these results we present a new search method that, although computationally more expensive, can avoid this problem.

Each discrete state model is tested by searching in the discrete state search space. Algorithm 1 presents the testing method proposed in this work. The search is performed by fitting the model to the real protein, using a best first approach with backtrack. For each amino acid, if the backbone conformation with lowest RMSD has no conflict with the previously set atoms, and if a non conflicting rotamer configuration is found, the algorithm sets the atoms and tries to set the next amino acid (running the *Next Amino Acid* procedure). During this step, previous side chains may be repositioned to accommodate the new amino acid atoms. If a clash is found that cannot be resolved or if the root mean square distance between the model and the protein exceeds a given threshold, the algorithm backtracks (running the *Backtrack* procedure). While backtracking, one of two actions occurs: if some of the possible configurations for the backbone were not tested the algorithm chooses the next conformation with lowest RMSD; if all configurations were tested the algorithm returns to the previously set amino acid and resumes its main procedure.

The algorithm starts with a minimum RMSD threshold of 5 angstroms and

**Algorithm 1** Testing method for the discrete models.

```
1: procedure FitModel(Pdb,Model)
2:     RmsdLimit = 5
3:     Start with the first amino acid of the sequence
4:     while RmsdLimit > 1 and there is a possible (φ, ψ) pair do
5:         Choose argmin_(φ,ψ) RMSD ({non tested (φ, ψ) pairs of Model })
6:         if RMSD < RmsdLimit and the backbone does not clash with any
               previous atom then
7:             for all rotamers indexed by (φ, ψ) do
8:                 if the side chain does not clash with any previous atom then
9:                     Set the Next Amino Acid
10:            if no side chain was found then
11:                Choose the rotamer that has less atomic conflicts and try to
                       correct the clash by changing a previously set side chain
12:                if no correction is possible then
13:                    Execute a Backtrack
14:                else
15:                    Set the Next Amino Acid
16:        else
17:            Execute a Backtrack

    procedure Backtrack          ▷ The chosen (φ, ψ) pair is not a possible
        configuration
        if ∃ non tested (φ, ψ) pairs of Model then
            Test the next (φ, ψ) pair with lowest RMSD (step 5)
        else
            Backtrack to the previous amino acid and test the next (φ, ψ) pair
                with lowest RMSD (step 4)

    procedure Next Amino Acid ▷ A valid conformation was found for this
        amino acid
        if this is the last amino acid of the sequence then
            RmsdLimit = RmsdLimit - 0.2
            Re-evaluate this amino acid with the new limit (step 5)
        else
            Set the next amino acid (step 4)
```

decrements 0.2 angstroms each time a model is found for the previous threshold. The algorithm stops if one of the following conditions is met: a 1 angstrom RMSD threshold limit is reached, no more configurations are possible or a time limit of one hour for each one hundred amino acids is reached. Although the number of possible configurations grows exponentially, an exponential increase of the time would be prohibitive. Since the linear increase may not be enough to produce equivalent results for all protein sizes, the test will also analyze the impact of the size of the protein on the model accuracy. We considered different thresholds for the side chain rotamer library (see section 2). In addition to

the 4% default value, we also tried 1%, 0.1% and without any cutoff (complete rotamer library). Since there is a time limit considering more rotamers or a more complex model may or may not produce better results.

## 5 Results

To test the accuracy of discrete models we compiled a set of protein structures of increasing size. The proteins also differ in terms of secondary structure composition (Alpha Beta, Mainly Alpha and Mainly Beta proteins). We have chosen the proteins from a list compiled in the What If database[16] (http://swift.cmbi.kun.nl/whatif/select/). In this database a technique similar to the PDB_SELECT program ([3,15]) was used to select a set of proteins from the PDB database[2] (http://www.rcsb.org/). Both techniques filter the proteins using a similarity threshold and a minimum resolution. Table 2 shows the set of chosen proteins and figure 3 shows the respective structures.

| Name | Size | Type |
|---|---|---|
| 1r69 | 63 | Mainly Alpha |
| 1ctf | 69 | Alpha Beta |
| 1poh | 85 | Alpha Beta |
| 1o5u | 88 | All Beta (beta helix) |
| 1e9m | 106 | Alpha Beta |
| 1co6 | 107 | Mainly Alpha |
| 1vhh | 157 | Alpha Beta |
| 1kao | 167 | Alpha Beta |

Table 2
Protein data set

We have chosen different types of proteins and different sizes to study the impact of these features in the precision of the discrete models. We imposed a limit on the number of amino acids since larger proteins would require too much time to test. The largest proteins in the test set already allow a group of amino acids to form a core surrounded by other amino acids. Larger proteins would repeat this pattern. We also decided to focus our choice in globular proteins which are more packed and therefore more difficult.

Figure 4 shows the results using side chains cutoff of 4% and 1%. Figure 5 shows the results when using 0.1% and the complete rotamer library.

From the results on figure 4 it is possible to verify that, for the majority of the proteins an accurate model can be found. However, for some proteins, it is

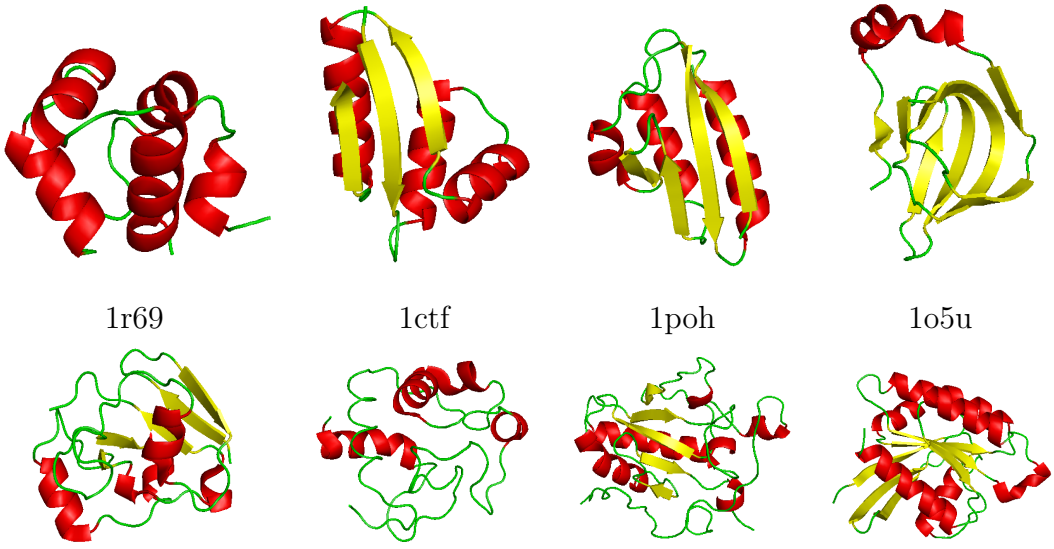
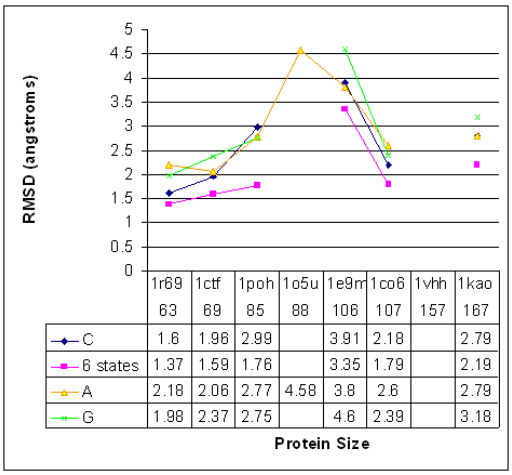
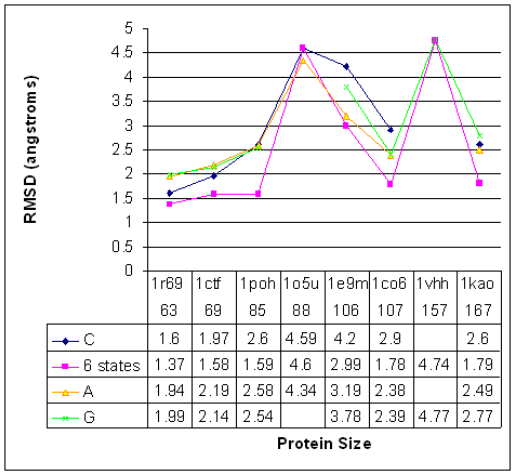



Fig. 3. Protein data set.



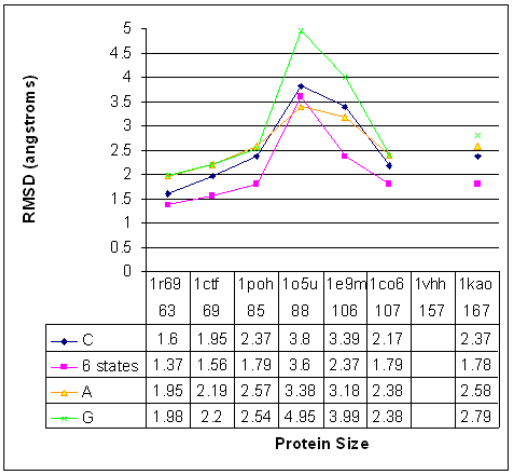

Side Chain Cutoff 4%



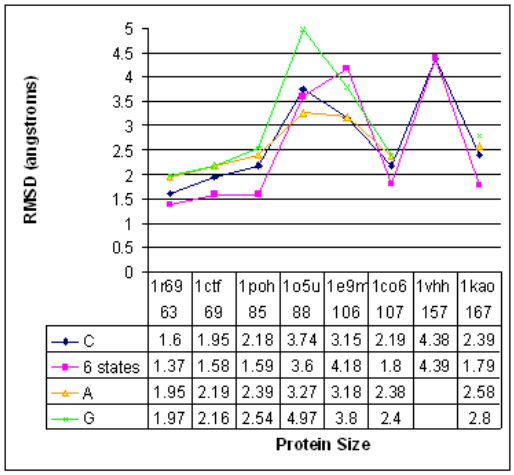

Side Chain Cutoff 1%

Fig. 4. Results of the Discrete State Models Using Side Chain cutoff of 4% and 1%.

hard to find a model and, in some cases, no model can be found. For the cases were an accurate model was found the results are consistent with the ones presented by[19] (2.22 to 2.43 angstroms for the four states models and 1.74 for the six states model). These results were expected since the side chain will only affect the fit in places where the protein is more compact. In those places, the minor errors in the discrete state models may not provide enough space for the side chains to be positioned. For the small proteins, the fitting problem is much easier since the error accumulation in the fitting will be small, and also because the side chains can almost always be positioned by choosing a




Side Chain Cutoff 0.1%




No Side Chain Cutoff

Fig. 5. Results of the Discrete State Models Using Side Chain cutoff of 0.1 percent and using all side chains in the rotamer library.

conformation that points to the exterior of the protein.

The cases where the results are worse in the side chain discrete models happen mainly with proteins with a large number of beta structures or that have a dense core. In the first case, because the flexibility of the beta structures is harder to model and in the second case, because it is harder to pack the atoms. One of the most difficult proteins to model, despite its small size, was the 1o5u mainly beta protein. The modeling difficulty of the beta strands is aggravated by the rolling conformation formed by the sheet. We also verify that, for the four states models, the C model has the best results. Moreover, we confirm the increased performance of model A on beta structures shown in the work of[19].

The six states model performs better than the four states models. However, the increase in the performance would probably not compensate for the increase in search space in a search procedure. Notice that, in this problem we are using the root mean square distance (RMSD) instead of the scoring functions used in ab-initio folding. The information given by a scoring function is less precise and the space that needs to be searched is much greater, even in a four state model. In a six state model that space will be even greater. We can probably achieve the same result from a four states model after a refinement process and we would benefit greatly from the smaller size of the search space during the search process.

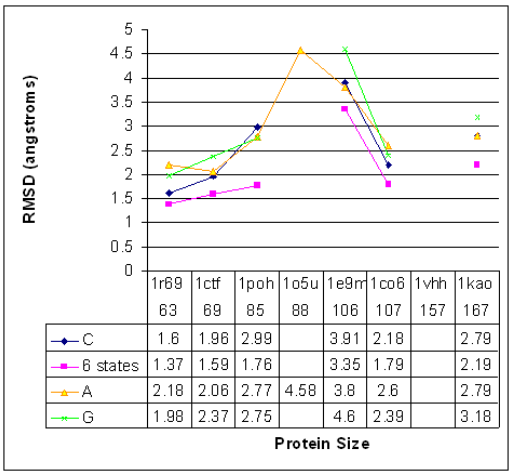
From tables 4 and 5 we can see the impact of the size of the rotamer library in the search. The performance of the models initially increases with the size of the protein. However, for the 0.1% threshold, we can already see that, for the bigger proteins, the performance sometimes decreases. When we avoid the

cutoff we can see this result again. This happens because, when we increase the size of the library, a greater number of side chain conformations may be tested. Consequently, although some conflicts may now be avoidable, the cost of the extra tests might not allow for the same number of conformations to be tested. Moreover, even the conflicts that were avoided may prove to be a dead end and cost the algorithm even more time. This increase of time for each side chain positioning will have a even greater impact in an ab-initio search algorithm, since no exact measure like the RMSD exists. Without an exact measure, the number of wrong conformations searched will be much greater and the increased computational cost of the enlarged library will further decrease the performance.

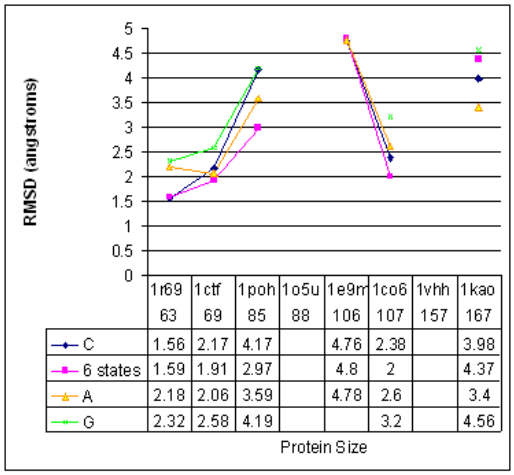
The type of model used and the threshold limit will have to be chosen according to the type of proteins (specially their secondary structure and size) and the search algorithm. If no information is known about the protein, a generic model like the C model or the six states model will probably be the best choice. For a search algorithm, with no specific information, a four states model with a side chain threshold of 1% will probably have best results because of the increase in the size of the searched space. If the protein is small, there are fewer possible conformations, and more detailed models can be found using more rotamers or even using the 6 states model. For specific cases, where there is some information about the protein, a model that increases the performance can probably be built.

Notice that if we use a threshold limit of, for instance, 5 angstroms, and we cannot find a model, we cannot be sure that no model can reach a 5 angstrom RMSD. When we stop the model construction, it is above the threshold, but subsequent choices could lower the RMSD. We chose not to pursue an exact solution since we needed to significantly increase the search space. However, we tried a limited solution that allows the algorithm to try a fixed number of steps (ten) even if the threshold is surpassed. If the algorithm can find a configuration ahead that is lower then the threshold, the algorithm continues, if not, it backtracks. The idea is to see if it is possible for the algorithm to correct a previous error. Table 6 shows results for the algorithm with the non rigid RMSD limit.

From the results on table 6 we can see that the cost of trying to correct a conformation error are too high even with a small limit. It is probable that the conflicts that produce a higher RMSD value are not easily resolved and that the increased flexibility on the limit would only increase the time spent on those conflicts.




Rigid RMSD cutoff




Non Rigid RMSD Cutoff 0.1%

Fig. 6. Results for the 4% side chain cutoff using the RMSD rigid cutoff and the non rigid cutoff.

## 6 Discussion

The results show that, although the discrete models presented in the work of[19] are well optimized for backbone fitting, when side chains are used and clashes are disallowed the accuracy decreases greatly for some proteins.

For small proteins the problem does not exist, since the side chains may be set to the outside of the structure. However, for larger proteins, especially for proteins with beta sheets, the accuracy is significantly lower than the one obtained for the backbone only tests. However, it is possible to obtain structures with root mean square error close or lower than 4 angstroms, which is still a very good, low detail, representation for the protein and a good starting point for the refinement algorithms. Moreover, for many proteins the results are near the 2 angstroms RMSD value and have, therefore, an accuracy equivalent to the backbone discrete models.

Since even low detail structures are very useful to determine protein function, these results show that the presented discrete models may be used in ab-initio protein folding. Moreover, since the models used in this work have been optimized for backbone models, it is possible that these results may improve by optimizing the models for structures with side chain representations.

There are still questions to be answered in respect to the discrete models evaluated in this work. In the protein folding process, one important aspect is that the scoring function is well adapted to the model that is being used. The model must be able to represent the positive and negative aspects of the structure, in a way that the scoring function can detect. However, no model will be useful

if it cannot represent a structure near the native protein structure. In this work, we have shown that it is possible to achieve a structure that is near the native protein structure. Moreover, by considering side chains and disallowing clashes, we have already obtained physically correct structures. This reduces the noise for the scoring function, since positive aspects of a structure, like the proximity of two amino acids, could be present in a physically impossible structure.

In future work we propose to study efficient algorithms to search near native conformations in the discrete state model space. This study may also be further improved by constructing a better adapted discrete model with side chains.

## References


[1] B. Berger and T. Leighton. Protein folding in the hydrophobic-hydrophilic (HP) is NP-complete. *Proceedings of the Second Annual International Conference on Computational Molecular Biology*, pages 30–39, 1998.

[2] H. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. Bhat, H. Weissig, I. Shindyalov, and P. Bourne. The Protein Data Bank. *Nucleic Acids Research*, 28(1):235–242, 2000.

[3] J. Boberg, T. Salakoski, and M. Vihinen. Selection of a representative set of structures from Brookhaven Protein Data Bank. *Proteins*, 14(2): 265–76, 1992.

[4] M. Bugalho and A. L. Oliveira. An efficient clash detection method for molecular structures. Technical Report 21, INESC-ID, August 2007.

[5] A. Canutescu, A. Shelenkov, and R. Dunbrack. A graph-theory algorithm for rapid protein side-chain prediction. *Protein Science*, 12(9):2001–2014, 2003.

[6] P. Crescenzi, D. Goldman, C. Papadimitriou, A. Piccolboni, and M. Yannakakis. On the complexity of protein folding. *Journal of Computational Biology*, 5(3):423–466, 1998.

[7] M. DePristo, P. de Bakker, S. Lovell, and T. Blundell. Ab initio construction of polypeptide fragments: Efficient generation of accurate, representative ensembles. *Proteins Structure Function and Genetics*, 51(1): 41–55, 2003.

[8] R. Dunbrack and M. Karplus. Conformational analysis of the backbone-dependent rotamer preferences of protein sidechains. *Nature Structural Biology*, 1(5):334–340, 1994.

[9] R. Dunbrack Jr. Rotamer libraries in the 21st century. *Current Opinion in Structural Biology*, 12(4):431–40, 2002.

[10] R. Dunbrack Jr and F. Cohen. Bayesian statistical analysis of protein side-chain rotamer preferences. *Protein Science*, 6(8):1661–1681, 1997.

[11] R. Dunbrack Jr and M. Karplus. Backbone-dependent rotamer library for proteins. Application to side-chain prediction. *Journal of Molecular Biology*, 230(2):543–74, 1993.

[12] A. Fraenkel. Complexity of protein folding. *Bulletin of Mathematical Biology*, 55(6):1199–1210, 1993.

[13] N. Gibbs, A. Clarke, and R. Sessions. Ab initio protein structure prediction using physicochemical potentials and a simplified off-lattice model. *Proteins Structure Function and Genetics*, 43(2):186–202, 2001.

[14] W. Hart and S. Istrail. Robust proofs of NP-hardness for protein folding: general lattices and energy potentials. *Journal of Computational Biology*, 4(1):1–22, 1997.

[15] U. Hobohm and C. Sander. Enlarged representative set of protein structures. *Protein Science*, 3(3):522, 1994.

[16] R. Hooft, C. Sander, and G. Vriend. Verification of protein structures: side-chain planarity. *Journal of Applied Crystallography*, 29(6):714–716, 1996.

[17] E. Huang, P. Koehl, M. Levitt, R. Pappu, and J. Ponder. Accuracy of side-chain prediction upon near-native protein backbones generated by ab initio folding methods. *Proteins Structure Function and Genetics*, 33 (2):204–217, 1998.

[18] B. Ma and R. Nussinov. The Stability of Monomeric Intermediates Controls Amyloid Formation: A $\beta$ 25-35 and its N27Q Mutant. *Biophysical Journal*, 90(10):3365–3374, 2006.

[19] B. Park and M. Levitt. The complexity and accuracy of discrete state models of protein structure. *Journal of Molecular Biology*, 249:493–507, 1995.

[20] C. Ramakrishnan and G. Ramachandran. Stereochemical criteria for polypeptide and protein chain conformations: II. Allowed conformations for a pair of peptide units. *Biophysical Journal*, 5(6):909, 1965.

[21] M. Rooman, J. Kocher, and S. Wodak. Prediction of protein backbone conformation based on seven structure assignments. Influence of local interactions. *Journal of Molecular Biology*, 221(3):961–79, 1991.