

A comparative study of the reported performance of *ab initio* protein structure prediction algorithms

Glennie Helles

J. R. Soc. Interface 2008 **5**, 387-396
doi: 10.1098/rsif.2007.1278

References

This article cites 41 articles, 6 of which can be accessed free

<http://rsif.royalsocietypublishing.org/content/5/21/387.full.html#ref-list-1>

Email alerting service

Receive free email alerts when new articles cite this article - sign up in the box at the top right-hand corner of the article or click [here](#)

To subscribe to *J. R. Soc. Interface* go to: <http://rsif.royalsocietypublishing.org/subscriptions>

REVIEW

A comparative study of the reported performance of *ab initio* protein structure prediction algorithms

Glennie Helles*

University of Copenhagen, Universitetsparken 1, 2100 Copenhagen, Denmark

Protein structure prediction is one of the major challenges in bioinformatics today. Throughout the past five decades, many different algorithmic approaches have been attempted, and although progress has been made the problem remains unsolvable even for many small proteins. While the general objective is to predict the three-dimensional structure from primary sequence, our current knowledge and computational power are simply insufficient to solve a problem of such high complexity.

Some prediction algorithms do, however, appear to perform better than others, although it is not always obvious which ones they are and it is perhaps even less obvious why that is. In this review, the reported performance results from 18 different recently published prediction algorithms are compared. Furthermore, the general algorithmic settings most likely responsible for the difference in the reported performance are identified, and the specific settings of each of the 18 prediction algorithms are also compared.

The average normalized r.m.s.d. scores reported range from 11.17 to 3.48. With a performance measure including both r.m.s.d. scores and CPU time, the currently best-performing prediction algorithm is identified to be the I-TASSER algorithm. Two of the algorithmic settings—protein representation and fragment assembly—were found to have definite positive influence on the running time and the predicted structures, respectively. There thus appears to be a clear benefit from incorporating this knowledge in the design of new prediction algorithms.

Keywords: protein structure prediction; algorithms; performance; *ab initio*; *de novo*

1. INTRODUCTION

Ram Samudrala once wrote ‘Proteins don’t have a folding problem. It’s we humans that do’ (Samudrala 1990) and indeed that seems to be the case. For five decades, researchers all over the world have tried to break the code and predict the three-dimensional structure of proteins from their primary sequence. There are two very different approaches to protein structure prediction: comparative modelling and *ab initio* prediction.

In comparative modelling, predictions are based on knowledge of structures of already known proteins, such that the sequence of an unknown protein is aligned to known proteins and, if a homology of more than 35% exists, then the three-dimensional fold is assumed to be the same (Edwards & Cottage 2003). Significant progress has been made in comparative (also called homology) modelling, as the method has proven to be quite efficient and applicable for a majority of proteins (Zhang & Skolnick 2004).

*glennie@diku.dk

There are, however, three reasons why *ab initio* folding remains interesting. First of all, there still exists a large number of proteins which do not show any homology with proteins of known structure. Second, comparative modelling does not offer any insight as to why a protein adopts a certain structure; and third, although some proteins show high resemblance to other proteins they still adopt different structures, which in principle means that predictions made by comparative modelling are never fully reliable.

Many different definitions of *ab initio* algorithms exist. The same definition as in Hardin *et al.* (2002) is adopted here, such that the term is taken to mean to start without knowledge of globally similar folds, which allows for algorithms to use statistical information, secondary structure prediction and fragment assembly (referred to by some as *de novo* rather than *ab initio* prediction).

A vast number of *ab initio* algorithms have been proposed throughout the years, with two prominent focus areas, rapidity and quality. Some model only very general principles of protein folding (Li *et al.* 2005; Guo *et al.* 2007; Hockenmaier *et al.* 2007), which is fast but

typically not very accurate. On the other hand, some create an actual simulation of the folding process (Zagrovic *et al.* 2002), which yields excellent results but an unacceptable running time. Most structure prediction algorithms try to balance the two and lie somewhere in between.

This review compares a wide range of *ab initio* protein structure prediction algorithms in order both to identify current state-of-the-art algorithms and to make the effect of algorithm choice and algorithmic configuration stand out. In order to design new and better prediction algorithms, it is important to know the paths already travelled, and this review is also meant to facilitate this.

Some of the algorithms proposed have competed in the biannual Critical Assessment of Techniques for Protein Structure Prediction (CASP) competition (<http://www.predictioncenter.org/>), which provides an ultimate way for benchmarking prediction systems. However, the CASP competition is concerned only with quality of the predicted structures. Neither the algorithmic details nor the running time is considered. This review includes all elements that can affect performance. Both algorithms proposed in connection with the CASP free modelling category (for a review of the latest CASP competition the reader is referred to Jauch *et al.* (2007)), and algorithms proposed elsewhere are included if they have been published along with their results within the past 5 years. Several systems entered in the CASP competition are not published and, although the results are available on the Internet, the algorithmic details are unknown, and such systems are therefore excluded. Incidentally, the top-performing algorithms in the CASP competitions tend to be published, although the SBC system entered by Elofsson and Wallner along with the MQAP-Consensus system from Gattie and the Luethy system are all examples of systems that appeared to perform very well in the free modelling category of the latest CASP VII competition¹, but to the best of our knowledge are unpublished.

In §2, key parameters relevant for comparing prediction algorithms are identified, and §3 concerns performance comparison of the reported results from 18 prediction systems. The results of the comparison are discussed in §4 and our conclusions are summarized in §5.

2. ALGORITHMIC PERFORMANCE FACTORS

Many parameters influence the running time of a structure prediction algorithm and the quality of the result. In order to determine why some prediction systems are more successful than others, it is important to identify all of the elements that can influence performance. Depending on the problem, some search algorithms (e.g. genetic algorithm or simulated annealing) do for instance tend to perform better than others, and the choice of algorithm may thus influence performance. Some prediction systems use the same

underlying optimization algorithm but differ in the configuration of the chosen algorithm. In other words, they differ in the setting of algorithmic parameters such as protein representation, acceptable angle space and energy functions. Furthermore, the composition of the test set may also be responsible for the differences in reported performance.

The effect of one particular setting compared with another is difficult to document based on the results from the literature, because the configuration of individual prediction algorithms typically differ on several settings. Also, some configurations work well for one type of algorithm but not for other types of algorithms. This is perhaps particularly pronounced between algorithms that employ a multiple solution search strategy, i.e. algorithms that search the entire solution space at once (like genetic algorithms), and algorithms that employ a single solution search strategy, i.e. algorithms that search neighbourhoods (like Monte Carlo, MC). When one thus looks at only a single well-performing algorithm that uses a simplified protein representation, it is difficult to know if the good performance has anything to do with the protein representation or if it is really due to for instance the chosen restrictions on angle space or perhaps the chosen set of test proteins. However, if many systems achieve good performance using a specific protein representation, then that protein representation is most likely a good idea. Regardless of the algorithm used, the trick is to introduce restrictions to the search space in a way that yields the proper trade-off where the relative gain in speed does not exceed the relative loss in quality.

Protein structure prediction is highly complex, and restrictions that can decrease the solution space in order to make the problem more tractable are very attractive. The specific configuration of an algorithm reflects the restrictions that are introduced and as mentioned it is usually distinct for every prediction system. The purpose of this review is thus not simply to list the results obtained with the different algorithms, but also to compare the algorithms with respect to the settings in order to better understand why some prediction systems appear to perform better than others.

Subsections 2.1–2.5 deal with different factors that may affect the performance of the search algorithms used in prediction systems.

2.1. Representation

A protein can be represented in a number of ways ranging from an all-atom to a simple C_α -trace representation. The all-atom model is naturally the most accurate representation, but unfortunately it typically has a very direct negative effect on running time, as more atoms require more time per iteration of the algorithm. Excluding the small hydrogen atoms from the representation and compensating by making the binding atoms larger is a restriction that intuitively seems rather harmless, and it greatly reduces the number of atoms that need to be considered. Further reduction can be made by substituting explicit side-chain representation with a single point representing just the centre of mass. The CAlpha,

¹When looking at r.m.s.d. values published on <http://www.predictioncenter.org/casp7/Casp7.html>

CBeta, Side chain (CABS; Kolinski 2004) and the UNified RESidue (UNRES; Lee *et al.* 1999) models are popular examples of this type of reduction. Excluding side chains altogether and thereby including only backbone atoms is yet another simplification that can be made, and at the far end of the scale we have the C_α -trace representation, which is no doubt the most optimal representation with respect to running time. Of course, it is also a rather crude approximation of the protein.

It should be noted that, although these are the types of representations typically encountered in protein structure prediction systems, even cruder simplifications can be made. Experiments with designs of simplified residue alphabets have been made where the amino acids are no longer viewed as distinct, but grouped in categories according to their physical propensities. The best-known property-based sequence representation is probably the hydrophobic–hydrophilic alphabet, but many others exist (see for instance Camproux & Tuffery (2005)). However, such representations are primarily used in model systems rather than real structure prediction systems.

2.2. Dihedral angle space

In principle, an infinite number of angles, dihedral angles and bond lengths between atoms can be adopted, but, due to the physical propensities of atoms, certain bond lengths and angles are strongly favoured. By analysing known proteins, it is also clear that amino acids have a definite preference for specific dihedral angles (Ramachandran & Sasisekharan 1968). A very common way to reduce the solution space is thus to fix bond lengths and angles and put restriction on the dihedral angle space by, for example, incorporating rotamer libraries (Dunbrack 2002) or operating on lattices.

Many restrictions on the dihedral angle space mean that the algorithm will typically converge more quickly than if there are no or only very few restrictions. However, many restrictions on dihedral angle space also mean that a significant part of the solution space cannot be sampled, and that the native structure may be unattainable. The dihedral angle space sampled by prediction systems is nearly always restricted in one way or another.

2.3. Energy function

The energy function is probably the parameter that has received the most attention throughout the years, and for good reason, as the energy function has an unquestioned influence on the accuracy of the structures predicted. A rather diverse set of functions that range from very simple to highly complex exists, but a perfect energy function that will consistently identify the native structure among decoys independently of the protein has yet to be found. Simple energy functions are typically based on some very general principles of protein folding such as hydrophobic packing and hydrogen bonding whereas the more complex functions incorporate many other kinds of physical, chemical and statistical information, such as electrostatic potentials, secondary structure tendencies and so on.

Much research has been done in the field of energy functions (Skolnick 2006) and a number of major energy functions (also known as force fields) exist, such as CHARMM and AMBER (see for instance MacKerell (2004) for an overview), but most prediction algorithms define and use their own versions. A thorough description of the different energy functions is beyond the scope of this study, and the reader is referred to the individual papers for a detailed description of the particular energy function used.

Generally, the energy functions can, however, be divided into two groups: physics- and statistics-based energy functions. Physics-based energy functions rely on the calculation of energy in the protein, whereas statistics-based energy functions derive their potential from statistical observations. It is important to note that both types of energy functions are approximations, although statistics-based energy functions are perhaps generally considered the cruder approximation of the two.

2.4. Folding strategy

Predicting the structure of small proteins is naturally easier (although still hard) than predicting the structure of large proteins, since the solution space grows exponentially with the number of amino acids. This fact has motivated many to divide the proteins into a number of fragments whose structure is predicted separately and subsequently assembled—a strategy known as fragment assembly. The method is currently very popular and also very successful when either secondary structure prediction algorithms (like PSIPRED; Jones 1999) or fragment libraries are used to predict fragment structures.

However, two things should be noted in this respect. First of all, using fragments one assumes that a fragment always folds into a number of predefined ways. Once a fragment is selected, it is considered rigid and it can be replaced only by another fragment. Long-range interactions in the protein under investigation are therefore not directly involved in shaping the fragments, which may not be prudent. Second, although algorithms that are based on secondary structure prediction and/or fragment libraries are currently better at generating native-like structures, they are, of course, forever bound to the limitation of systems relying on known structures, as secondary structure prediction algorithms are trained on known structures and fragment libraries built from known structures.

2.5. Test set

Unfortunately, a standard protein test set does not exist, and so yet another parameter that must be considered when evaluating the performance of a prediction algorithm is the set of test proteins. The number of proteins in the test set is of interest for statistical reasons. For a large test set, it is less likely that good results are obtained by mere ‘luck’ and the algorithm is more likely to be generally applicable than if the test set contained only a few test proteins. Since most prediction algorithms are quite time consuming, test sets are most often of limited size (up to 15),

Table 1. Performance comparison of 18 structure prediction algorithms.

| | rep. ^a | frag. ^b | move set ^c | energy func. ^d | running time ^e | no. of proteins ^f | length range ^g | class ^h | res. set size ⁱ | avg. r.m.s.d. ^j |
|---|-------------------|--------------------|-----------------------|---------------------------|---------------------------|------------------------------|----------------------------------|---|----------------------------|----------------------------|
| <i>MD simulation</i> | | | | | | | | | | |
| Pande group; Zagrovic <i>et al.</i> (2002) | all-atom | no | all | physics | years | 1 | [36] | α | 1 | 1.7 |
| Liwo group; Liwo <i>et al.</i> (2005) ^k | UNRES | no | all | physics | hours | 4 1 | [46–75] [48] | α α/β | 10 10 | 2.6 3.9 |
| <i>replica exchange Monte Carlo</i> | | | | | | | | | | |
| Shakhnovich group; Yang <i>et al.</i> (2006) | all-atom | no | all | statistics | weeks | 7 | [46–77] | α | 15 | 4.3 |
| | | | | | | | | | | |
| Touchstone II; Zhang <i>et al.</i> (2003) | CABS | yes | lattice | statistics | days | 43 | [36–157] | α | 2–22 | 6.7 |
| | | | | | | | | | | |
| Kolinski group; Latek <i>et al.</i> (2007) | CABS | yes | lattice | statistics | days | 4 | [66–86] | α | 1 | 4.8 |
| | | | | | | | | | | |
| ZAM; Ozkan <i>et al.</i> (2007) | all-atom | yes | restricted | physics | months | 3 2 4 | [39–153] [47–174] [35–57] | β α/β α/β | 4–62 2–56 1 | 8.2 8.3 3.0 |
| <i>Metropolis Monte Carlo</i> | | | | | | | | | | |
| Rosetta; Rohl <i>et al.</i> (2004); Bradley <i>et al.</i> (2005b) | all-atom | yes | restricted | statistics | months | 3 | [49–72] | α | 5 | 1.5 |
| | | | | | | | | | | |
| SimFold; Fujitsuka <i>et al.</i> (2006) | CABS-like | yes | restricted | mixed | months | 14 10 12 | [63–109] [28–72] [28–72] | α β α/β | 400 400 400 | 3.9 7.5 6.6 |
| PROTINFO; Hung <i>et al.</i> (2005) | all-atom | no | restricted | statistics | months | 1 1 | [53] [69] | α α/β | 5 5 | 5.0 4.3 |
| <i>hyperbolic Monte Carlo</i> | | | | | | | | | | |
| I-TASSER; Wu <i>et al.</i> (2007) | C_α | yes | part lattice | statistics | hours | 16 14 26 | [49–118] [47–117] [47–117] | α β α/β | 5 5 5 | 3.5 3.8 4.3 |
| <i>conformational space annealing (CSA)</i> | | | | | | | | | | |
| Profesy; Lee <i>et al.</i> (2004, 2005) | backbone | yes | restricted | physics | | 2 2 | [28–46] [20–57] | α β | 5 5 | 4.2 2.7 |
| <i>multi-objective evolutionary approach</i> | | | | | | | | | | |
| Nicosia; Cutello <i>et al.</i> (2006) | all-atom | no | restricted | physics | hours ^l | 3 | [34–70] | α | 1 | 3.3 |

Table 1. (Continued.)

| | rep. ^a | frag. ^b | move set ^c | energy func. ^d | running time ^e | no. of proteins ^f | length range ^g | class ^h | res. set size ⁱ | avg. r.m.s.d. ^j |
|---|-------------------|--------------------|-----------------------|---------------------------|---------------------------|------------------------------|---------------------------|--------------------|----------------------------|----------------------------|
| <i>evolutionary algorithms</i> | | | | | | | | | | |
| Schug; Schug & Wenzel (2006) | % H | no | restricted | physics | hours | 1 | [46] | α/β | 1 | 4.4 |
| Profet; Koskowski & Hartke (2004) | all-atom | no | all | physics | days | 1 | [60] | α | 10 | 3.9 |
| | | | | | | | | | | |
| <i>MOLS-genetic algorithms</i> | | | | | | | | | | |
| Gautham; Arunachalam <i>et al.</i> (2006) | all-atom | yes | restricted | physics | hours | 2 | [18–21] | β | 5–10 | 5.2 |
| | | | | | | | | | | |
| <i>branch and bound</i> | | | | | | | | | | |
| M. Paluszewski & P. Winter 2007, unpublished data | C_α | yes | restricted | statistics | days | 4 | [26–52] | α | 1 | 4.9 |
| | | | | | | | | | | |
| <i>αBB/CSA/MD</i> | | | | | | | | | | |
| ASTROFOLD; Klepeis & Floudas (2003) | all-atom | yes | restricted | physics | weeks | 1 | [43–65] | α | 10 000 | 4.6 |
| | | | | | | | | | | |
| <i>probabilistic distance geometry optimization</i> | | | | | | | | | | |
| PROPAINOR; Joshi & Jyothi (2003) | C_α | yes | all | statistics | hours | 21 | [69] | β | 1 | 6.2 |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |

although the largest test set seen in this study includes 125 test proteins (Zhang *et al.* 2003).

The lengths of the individual test proteins (e.g. number of amino acids) are also of interest, as the structures of small proteins are both easier and faster to predict than the structures of larger proteins. Skolnick and colleagues (Reva *et al.* 1998) have previously investigated the possibility of obtaining a native-like structure by mere chance, and concluded that generating a structure at random (although compact) that has an r.m.s.d. below 6 Å is highly unlikely for a chain greater than 60 amino acids, but naturally that chance increases for smaller proteins. An algorithm that is able to predict the structure of a protein of, say, 20 amino acids to an r.m.s.d. of 6 Å is much less impressive than an algorithm that can predict the structure of a protein of, say, 100 amino acids to an r.m.s.d. of 6 Å. Most test proteins contain fewer than 100 amino acids.

Finally, the structural classes of the test proteins are of interest. All proteins can generally be classified as α , β , α/β or coil (Orengo *et al.* 1997), where the first three categories are by far the most populated. A good prediction algorithm must be able to make equally good predictions regardless of structural class. Hence confidence in an algorithm relies also on the structural classes of the test proteins. It is much harder to conclude anything about the general applicability of an algorithm that has been tested on only a few proteins belonging to the same structural class than if the algorithm has been tested on a larger number of proteins from all structural classes. Most—but not all—prediction systems are tested on proteins from all structural classes (except coil structures which none of the systems included in this study are tested on).

3. PERFORMANCE COMPARISON

When configuring an algorithm for structure prediction, focus can be put on any or all of the parameters identified in §2, which is reflected in the numerous prediction algorithms proposed. The aim of this performance comparison is primarily to contrast different algorithmic approaches, but also to deduce any trends in the settings of the algorithms. One must, of course, be cautious when drawing conclusions about a given setting, as it is often tied to the algorithm and the test proteins, but, when comparing a relatively large number of algorithms, the results may nevertheless indicate some general trends that would be of interest in the design of new algorithms.

Performance is compared here between the reported results of 18 recently proposed prediction algorithms (published in the last 5 years). Several algorithms have been excluded (even relatively known *ab initio* systems such as Jones *et al.* (2005) and Zhou & Skolnick (2007)) because r.m.s.d. values between the native and predicted structures have not been published in their papers. There exist a number of alternative ways to compare structures (such as d r.m.s.d., GDT_TS, TM score, etc.), and while they may be better at expressing how well the algorithm performs in terms of, for example, substructure formation or core packing, the r.m.s.d. is the most commonly used descriptor. Furthermore, an overall low r.m.s.d. is the ultimate goal for a structure prediction algorithm if it is to

be used in practice. Algorithms designed to predict only specific types of proteins (such as membrane proteins) are excluded. Algorithms based on exact knowledge of the native structure are naturally also excluded, although the contribution from DeRonne & Karypis (2006) is very interesting from an algorithmic point of view. Finally, newer versions of the algorithms are assumed to be equally good or better than older versions, and therefore only the latest versions have been included.

Although two of the algorithms participated in the CASP VII competition (Bradley *et al.* 2005b; Wu *et al.* 2007), their results from the competition are not included. As mentioned earlier, r.m.s.d. values are available on the Internet, but have not yet been explicitly documented in the literature.

3.1. Results

The results of the performance comparison are presented in table 1. The three columns ‘Avg. r.m.s.d.’, ‘Res. set size’ and ‘Running time’ constitute the collected results for each algorithm.

The Avg. r.m.s.d. column specifies the average r.m.s.d. values for the best selected structures of all the test proteins. The Res. set size refers to the size of the result set, i.e. the number of predicted structures selected by the systems. For those systems that use clustering or refinement, it refers to the number of results selected after the initial results have been clustered or refined. Many are reluctant to pick one cluster over another and thus return a representative structure from each cluster. There may be significant differences between the representative structures chosen (see for instance Wu *et al.* 2007), but r.m.s.d. is usually not reported for all selected structures and thus only the lowest r.m.s.d. value among the selected structures is included in the r.m.s.d. average here.

The Running time column indicates how quickly the algorithm finds a solution for a protein. Running times generally depend on the length of the proteins, but as mentioned earlier—and as is evident from table 1—most test proteins included in the test sets are of comparable length. The computer power available differs greatly, but the time stated in the column gives a rough estimate of the time required for a single standard PC processor to reach a solution. Hence, if a group has used a cluster of 10 computers and 4 days to predict a structure, it will be marked as ‘months’ (approx. 40 CPU days). The algorithms (and results) included have all been published in the last 5 years; while that is a fairly short time range, it should be noted that computer power has increased significantly in that period (a standard PC is roughly three times faster today), and newer algorithms have thus not only the benefit of previous experience but also the benefit of significantly faster computers.

4. DISCUSSION

4.1. Algorithmic configuration

Detailed MD simulation of all-atom protein models such as Zagrovic *et al.* (2002) is typically performed in order to allow researchers to observe the folding

pathways—not merely to predict protein structure. MD simulation has, however, proved to be very accurate (for at least small proteins), and it has been included here because it is technically possible to use MD simulation for structure prediction, although it is computationally extremely heavy and renders the problem intractable for all but the smallest of proteins. In a sense, one might say that the goal of a prediction algorithm is to combine the accuracy of MD simulation with the speed of (most) search algorithms.

Different flavours of the MC search strategy are by far the most common types of algorithms used, but results for all algorithms are for the most part comparable with respect to accuracy. Studies that compare different MC search strategies are performed regularly (Gront *et al.* 2000). They typically show that one algorithm performs slightly better than those it is compared with, but this may be related to the test proteins rather than to the algorithm as such. From *table 1*, it would certainly seem like all types of MC searches show roughly the same performance. The I-TASSER algorithm (Wu *et al.* 2007) based on a Hyperbolic MC scheme stands out with results being superior to the others, particularly when it comes to running time.

Interestingly, excellent results are obtained quickly by the Liwo group (Liwo *et al.* 2005), who also use MD simulation but with a simplified residue representation. The force field used is designed to compensate for the missing atoms, and while some simulations—particularly of proteins containing β -sheets—do not converge to a final structure, the fact that the algorithm reached good results extremely fast indicates an enthralling potential. In fact, simplified residue representation appears to generally have a positive effect on the running time of the algorithms but a more or less undetectable effect on accuracy irrespective of the type of algorithm used. As mentioned previously, most prediction algorithms differ on multiple settings, and it is thus usually difficult to make any general conclusions about a particular setting. Nevertheless, in this case where many algorithms are aligned, it seems clear that the effect of a simplified protein representation on overall accuracy is minimal.

Algorithms based on fragment assembly are generally considered more successful than others (many of the best algorithms in the CASP competition use this folding strategy), and it certainly seems intuitively right that fragment assembly would be much faster. This is not evident from the results reviewed here where the strategy does not appear to have any major influence on running time. However, fragment assembly is most likely an important factor in the high accuracy achieved by algorithms such as I-TASSER and Rosetta, but one should also bear in mind that the use of fragments makes the system a borderline ‘comparative modelling’ system, which relies heavily on existing structures.

Aside from MD simulation, the dihedral angle space is restricted in nearly every algorithm. PROPAINOR stands out as it takes a completely different approach to the problem by sampling residue distance space rather than angle space—thereby making all dihedral angles possible. Most algorithms are off-lattice, but some make use of a lattice (at least for parts of the protein) and with an expected positive influence on at least the

running time (Zhang *et al.* 2003; Latek *et al.* 2007; Wu *et al.* 2007). It should also be noted that the restrictions on dihedral angle space used by most do not appear to have a detectable influence on the quality of the predicted structures.

With regards to the two types of energy functions, it would appear from this comparison that algorithms that use a physics-based energy function find solutions that are marginally better than the solutions found by algorithms with a statistics-based energy function. No influence on the running time of the algorithms can be observed. It should be noted that statistics-based energy functions vary greatly in the number of parameters they include and thus a general trend should not be extracted from this study.

4.2. Test set specific parameters

From *table 1*, it can be seen that only 11 out of the 18 algorithms have been tested on proteins from all structural classes (except coil). The Liwo group (Liwo *et al.* 2005) did in fact test proteins from the three main structural classes, but, as the algorithm did not converge for β -sheet structures, they have not reported any results for these proteins. The smallest test sets used included only one structure (Zagrovic *et al.* 2002; Schug & Wenzel 2006) while the largest set included 125 structures.

Surprisingly, there appears to be no correlation between running time and quality—in fact the fastest algorithms obtain some of the best results even when the lengths of the proteins are taken into consideration. Algorithms that have been tested on test sets, which include many and/or significantly larger proteins, would be expected to obtain a higher average r.m.s.d., but the results would also be more reliable as it is very difficult to tune parameters—intentionally or not—to produce good results on large test sets (as discussed in §2). TOUCHSTONE II (large test set) and to a certain extent PROPAINOR (large proteins) support this assumption, but the I-TASSER algorithm actually maintains excellent performance despite being tested on the second largest test set with proteins that are both structurally diverse and have an average length of 81 amino acids (*table 2*).

From the results presented, it is evident that predicting β -sheets is much more difficult, and most algorithms that are tested on proteins belonging to different structural classes perform worse on proteins that contain β -sheets, indicating that the energy function used is biased towards one kind of secondary structure (usually α -helices). The results reported for the ZAM (Ozkan *et al.* 2007) and Profesy (Lee *et al.* 2004, 2005) algorithms along with the results reported by Gautham (Arunachalam *et al.* 2006) actually showed better results for β -class structures, but that is most likely due to the short length of the selected β -class proteins. A few of the groups (Joshi & Jyothi 2003; Klepeis & Floudas 2003; Yang *et al.* 2006; Wu *et al.* 2007) stand out as they seem to obtain equally good results for all their proteins regardless of structural class.

Table 2. Summarized results. I-TASSER (italics) is found to be the overall best-performing algorithm.

| | no. of proteins ^a | avg. length ^b | avg. r.m.s.d. ^c | avg. r.m.s.d. ₁₀₀ ^d | run time ^e |
|---|------------------------------|--------------------------|----------------------------|---|-----------------------|
| Pande group; Zagrovic <i>et al.</i> (2002) | 1 | 36 | 1.7 | 3.48 | years |
| Liwo group; Liwo <i>et al.</i> (2005) | 6 | 57 | 4.0 | 5.56 | hours |
| Shakhnovich group; Yang <i>et al.</i> (2006) | 13 | 58 | 4.5 | 6.18 | weeks |
| Touchstone II; Zhang <i>et al.</i> (2003) | 125 | 87 | 7.7 | 8.28 | days |
| Kolinski group; Latek <i>et al.</i> (2007) | 9 | 78 | 6.2 | 7.08 | days |
| ZAM; Ozkan <i>et al.</i> (2007) | 9 | 46 | 2.8 | 4.57 | months |
| Rosetta; Rohl <i>et al.</i> (2004); Bradley <i>et al.</i> (2005a) | 16 | 69 | 3.8 | 4.66 | months |
| SimFold; Fujitsuka <i>et al.</i> (2006) | 38 | 76 | 5.9 | 6.84 | months |
| PROTINFO; Hung <i>et al.</i> (2005) | 2 | 61 | 4.7 | 6.24 | months |
| <i>I-TASSER</i> ; Wu <i>et al.</i> (2007) | 56 | 81 | 4.2 | 4.69 | hours |
| Profesy; Lee <i>et al.</i> (2004, 2005) | 4 | 38 | 3.5 | 6.78 | |
| Nicosia; Cutello <i>et al.</i> (2006) | 4 | 52 | 3.5 | 5.20 | hours |
| Schug; Schug & Wenzel (2006) | 1 | 60 | 3.9 | 5.24 | hours |
| Profet; Koskowski & Hartke (2004) | 3 | 28 | 4.0 | 11.17 | days |
| Gautham; Arunachalam <i>et al.</i> (2006) | 5 | 33 | 4.2 | 9.42 | hours |
| M. Paluszewski & P. Winter 2007, unpublished data | 6 | 60 | 5.2 | 6.96 | days |
| ASTROFOLD; Klepeis & Floudas (2003) | 8 | 76 | 5.3 | 6.15 | weeks |
| PROPAINOR; Joshi & Jyothi (2003) | 52 | 98 | 6.4 | 6.47 | hours |

^a Number of test proteins.^b Average length of the test proteins.^c Average r.m.s.d. values of the best selected structures found for all test proteins.^d Average r.m.s.d. values are normalized with respect to protein lengths.^e A rough estimate of how quickly a solution is found using a single processor.

4.3. Performance results

Generally, most results reported look impressive. It should of course be emphasized that the r.m.s.d. values reported here are for the selected structures with the lowest r.m.s.d. values found by the prediction systems. Note that most algorithms return numerous structures, some with high r.m.s.d. values and some with low r.m.s.d. values. A large result set size is generally less attractive—even if it includes a near native structure—because the algorithm as such is unable to separate that structure from the decoys. Returning many solutions does, however, not necessarily pose a problem, if there is some way to separate the ‘good’ structures from the ‘bad’ by, for example, using clustering (Yang *et al.* 2006) or other filtering techniques (Eyal *et al.* 2007). As shown by the Res. set size column in table 1, many systems return several solutions even after the results have been clustered and the best solution is then picked based on its r.m.s.d. score to the native protein. Of course, in order to function as a reliable prediction system, one must be able to pick the good structure without knowledge of the native structure.

Table 2 summarizes the performance result of the predictions. In the ‘Avg. r.m.s.d.₁₀₀’ column, the average r.m.s.d. values have been normalized with respect to the length of the test proteins (Carugo & Pongor 2001), which makes it easier to compare r.m.s.d. values for proteins of different lengths. Most of the included published results have an average r.m.s.d.₁₀₀ value of approximately 6 Å. Although the result reported by the Pande group using the MD simulation is the lowest, it has two major drawbacks: there are too few test proteins and the running time is very poor. Furthermore, the protein folded is very small (only 36 amino acids). From the r.m.s.d.₁₀₀ values, it is clear that the Rosetta algorithm (Bradley *et al.* 2005b) and the I-TASSER algorithm (Wu *et al.* 2007) are at a near tie,

which was also seen in the latest CASP VII competition (Jauch *et al.* 2007). The Rosetta algorithm (Bradley *et al.* 2005b) holds a long-standing record for achieving good results at the CASP competitions, and so the results of the 16 test proteins are considered quite reliable. The I-TASSER algorithm is, however, tested on a much larger test set of 56 proteins (including the same proteins as Rosetta was tested on), and, with an excellent running time that clearly outperforms Rosetta, it is here concluded to be the overall best-performing algorithm. As mentioned previously, the CASP VII results from I-TASSER and Rosetta are available on the Internet, but not included here. However, analysis of the r.m.s.d. values from CASP VII reveals a picture similar to what is seen here. Both I-TASSER and Rosetta use an MC sampling scheme (although different variants), fragment assembly and a statistics-based energy function, but they differ in protein representation and acceptable dihedral angle space.

Finally, the need for a standard protein test set of appropriate size must be emphasized. The trends observed concerning parameter settings in this study are based on (sparse) statistics, but could perhaps be made into actual conclusions if all research groups used the same test set (as is seen in other research areas). Furthermore, a standard test set would make it difficult to cheat and would allow for a more systematic and reliable evaluation of algorithms.

5. CONCLUSION

The parameters for proper comparison of protein structure prediction algorithms have been identified, and the performance of 18 different *ab initio* prediction algorithms has been compared with respect to these parameters. In lack of a standard protein test set, it is

usually difficult to evaluate the importance of one particular algorithmic setting over another, but, owing to the relatively large number of algorithms compared here, certain trends in the settings could be identified. Simplified protein representation was found to have seemingly undetectable influence on accuracy, but a definite positive influence on running time. The (very popular) fragment assembly folding strategy is most likely responsible for the high accuracy achieved by some groups (Bradley *et al.* 2005b; Wu *et al.* 2007), but it does not appear to have any general positive influence on running time. Half of the algorithms use a physics-based energy function and, although they appear to slightly outperform those using a statistics-based energy function, the complexity of energy functions makes it impossible to draw any reliable conclusions about the effect of physics-based versus statistics-based energy functions. Surprisingly, the overall best-performing algorithm—the I-TASSER algorithm (Wu *et al.* 2007)—is also one of the fastest algorithms included in this study.

As a final note, it should be mentioned that this type of performance comparison is made particularly difficult because research groups test their algorithms on their own selected proteins. A standard protein test set would greatly enhance any possible trends in algorithmic settings and could facilitate designs of new algorithms.

REFERENCES

Arunachalam, J., Kanagasabai, V. & Gautham, N. 2006 Protein structure prediction using mutually orthogonal latin squares and a genetic algorithm. *Biochem. Biophys. Res. Commun.* **342**, 424–433. ([doi:10.1016/j.bbrc.2006.01.162](https://doi.org/10.1016/j.bbrc.2006.01.162))

Bradley, P., Malmström, L., Qian, B., Schonbrun, J., Chivian, D., Kim, D. E., Meiler, J., Misura, K. M. S. & Baker, D. 2005a Free modeling with Rosetta in CASP6. *Proteins: Struct. Funct. Bioinform.* **61**, 128–134. ([doi:10.1002/prot.20729](https://doi.org/10.1002/prot.20729))

Bradley, P., Misura, K. M. S. & Baker, D. 2005b Towards high-resolution de novo structure prediction for small proteins. *Science* **309**, 1868–1871. ([doi:10.1126/science.1113801](https://doi.org/10.1126/science.1113801))

Camproux, A. C. & Tuffery, P. 2005 Hidden Markov model-derived structural alphabet for proteins: the learning of protein local shapes captures sequence specificity. *Biochim. Biophys. Acta* **1724**, 394–403.

Carugo, O. & Pongor, S. 2001 A normalized root-mean-square distance for comparing protein three-dimensional structures. *Protein Sci.* **10**, 1470–1473. ([doi:10.1110/ps.690101](https://doi.org/10.1110/ps.690101))

Cutello, V., Narzisi, G. & Nicosia, G. 2006 A multi-objective evolutionary approach to the protein structure prediction problem. *J. R. Soc. Interface* **3**, 139–151. ([doi:10.1098/rsif.2005.0083](https://doi.org/10.1098/rsif.2005.0083))

DeRonne, K. W. & Karypis, G. 2006 Effective optimization algorithms for fragment-assembly based protein structure prediction. In *Computational Systems Bioinformatics Conference*, Stanford, CA, USA, pp. 19–29. London, UK: Imperial College Press.

Dunbrack, R. L. 2002 Rotamer libraries in the 21st century. *Curr. Opin. Struct. Biol.* **12**, 431–440. ([doi:10.1016/S0959-440X\(02\)00344-5](https://doi.org/10.1016/S0959-440X(02)00344-5))

Edwards, Y. J. K. & Cottage, A. 2003 Bioinformatics methods to predict protein structure and function. *Mol. Biotechnol.* **23**, 139–166. ([doi:10.1385/MB:23:2:139](https://doi.org/10.1385/MB:23:2:139))

Eyal, E., Frenkel-Morgenstern, M., Sobolev, V. & Pietrokovski, S. 2007 A pair-to-pair amino acids substitution matrix and its applications for protein structure prediction. *Proteins* **67**, 142–153. ([doi:10.1002/prot.21223](https://doi.org/10.1002/prot.21223))

Fujitsuka, Y., Chikenji, G. & Takada, S. 2006 SimFold energy function for de novo protein structure prediction: consensus with Rosetta. *Proteins* **62**, 381–398. ([doi:10.1002/prot.20748](https://doi.org/10.1002/prot.20748))

Gront, D., Kolinski, A. & Skolnick, J. 2000 Comparison of three Monte Carlo conformational search strategies for a protein like homopolymer model: folding thermodynamics and identification of low-energy structures. *J. Chem. Phys.* **113**, 5065–5071. ([doi:10.1063/1.1289533](https://doi.org/10.1063/1.1289533))

Guo, Y. Z., Feng, E. M. & Wang, Y. 2007 Optimal HP configurations of proteins by combining local search with elastic net algorithm. *J. Biochem. Biophys. Methods* **70**, 335–340. ([doi:10.1016/j.jbbm.2006.08.001](https://doi.org/10.1016/j.jbbm.2006.08.001))

Hardin, C., Pogorelov, T. V. & Luthey-Schulten, Z. 2002 *Ab initio* protein structure prediction. *Curr. Opin. Struct. Biol.* **12**, 176–181. ([doi:10.1016/S0959-440X\(02\)00306-8](https://doi.org/10.1016/S0959-440X(02)00306-8))

Hockenmaier, J., Joshi, A. K. & Dill, K. A. 2007 Routes are trees: the parsing perspective on protein folding. *Proteins* **66**, 1–15. ([doi:10.1002/prot.21195](https://doi.org/10.1002/prot.21195))

Hung, L., Ngan, S., Liu, T. & Samudrala, R. 2005 PROTINFO: new algorithms for enhanced protein structure predictions. *Nucleic Acids Res.* **33**, W77–W80. ([doi:10.1093/nar/gki403](https://doi.org/10.1093/nar/gki403))

Jauch, R., Yeo, H. C., Kolatkar, P. R. & Clarke, N. D. 2007 Assessment of CASP7 structure predictions for template free targets. *Proteins: Struct. Funct. Bioinform.* **66**, 57–67. ([doi:10.1002/prot.21771](https://doi.org/10.1002/prot.21771))

Jones, D. T. 1999 Protein secondary structure prediction based on position-specific scoring matrices. *J. Mol. Biol.* **292**, 195–202. ([doi:10.1006/jmbi.1999.3091](https://doi.org/10.1006/jmbi.1999.3091))

Jones, D. T., Bryson, K., Coleman, A., McGuffin, L. J., Sadowski, M. I., Sodhi, J. S. & Ward, J. J. 2005 Prediction of novel and analogous folds using fragment assembly and fold recognition. *Proteins: Struct. Funct. Bioinform.* **61**, 143–151. ([doi:10.1002/prot.20731](https://doi.org/10.1002/prot.20731))

Joshi, R. R. & Jyothi, S. 2003 *Ab-initio* prediction and reliability of protein structural genomics by PROPAINOR algorithm. *Comput. Biol. Chem.* **27**, 241–252. ([doi:10.1016/S0097-8485\(02\)00074-8](https://doi.org/10.1016/S0097-8485(02)00074-8))

Klepeis, J. L. & Floudas, C. A. 2003 ASTRO-FOLD: a combinatorial and global optimization framework for *ab initio* prediction of three-dimensional structures of proteins from the amino acid sequence. *Biophys. J.* **85**, 2119–2146.

Kolinski, A. 2004 Protein modeling and structure prediction with a reduced representation. *Acta Biochim. Polon.* **51**, 349–371.

Koskowski, F. & Hartke, B. 2004 Towards protein folding with evolutionary techniques. *J. Comput. Chem.* **26**, 1169–1179. ([doi:10.1002/jcc.20254](https://doi.org/10.1002/jcc.20254))

Latek, D., Ekonomiuk, D. & Kolinski, A. 2007 *Protein structure prediction: combining de novo modeling with sparse experimental data*. New York, NY: Wiley InterScience.

Lee, J., Liwo, A. & Scheraga, H. A. 1999 Energy-based de novo protein folding by conformational space annealing and an off-lattice united-residue force field: application to the 10–55 fragment of staphylococcal protein A and to apo calbindin D9K. *Proc. Natl Acad. Sci. USA* **96**, 2025–2030. ([doi:10.1073/pnas.96.5.2025](https://doi.org/10.1073/pnas.96.5.2025))

Lee, J., Kim, S.-Y., Joo, K., Kim, I. & Lee, J. 2004 Prediction of protein tertiary structure using PROFESY, a novel method based on fragment assembly and conformational space annealing. *Proteins: Struct. Funct. Bioinform.* **56**, 704–714. ([doi:10.1002/prot.20150](https://doi.org/10.1002/prot.20150))

Lee, J., Kim, S.-Y. & Lee, J. 2005 Protein structure prediction based on fragment assembly and parameter optimization. *Biophys. Chem.* **115**, 209–214. (doi:10.1016/j.bpc.2004.12.046)

Liwo, A., Khalili, M. & Scheraga, H. A. 2005 *Ab initio* simulations of protein-folding pathways by molecular dynamics with the united-residue model of polypeptide chains. *Proc. Natl Acad. Sci. USA* **102**, 2362–2367. (doi:10.1073/pnas.0408885102)

Li, Z., Zhang, X. & Chen, L. 2005 Unique optimal foldings of proteins on a triangular lattice. *Appl. Bioinform.* **4**, 105–116. (doi:10.2165/00822942-200504020-00004)

MacKerell Jr, A. D. 2004 Empirical force fields for biological macromolecules: overview and issues. *J. Comput. Chem.* **25**, 1584–1604. (doi:10.1002/jcc.20082)

Orrego, C. A., Michie, A. D., Jones, S., Jones, D. T., Swindells, M. B. & Thornton, J. M. 1997 CATH-a hierachic classification of protein domain structures. *Structure* **5**, 1093–1108. (doi:10.1016/S0969-2126(97)00260-8)

Ozkan, S. B., Wu, G. A., Chodera, J. D. & Dill, K. A. 2007 Protein folding by zipping and assembly. *Proc. Natl Acad. Sci. USA* **104**, 11 987–11 992. (doi:10.1073/pnas.0703700104)

Ramachandran, G. N. & Sasisekharan, V. 1968 Conformation of polypeptides and proteins. *Adv. Protein Chem.* **23**, 283–438.

Reva, B. A., Finkelstein, A. V. & Skolnick, J. 1998 What is the probability of a chance prediction of a protein structure with an RMSD of 6 Å? *Fold des* **3**, 141–147. (doi:10.1016/S1359-0278(98)00019-4)

Rohl, C. A., Strauss, C. E. M., Misura, K. M. S. & Baker, D. 2004 Protein structure prediction using rosetta. *Methods Enzymol.* **383**, 66–93. (doi:10.1016/S1359-0278(98)00019-4)

Samudrala, R. 1990 *Genes, macromolecules & computers*. See <http://www.ram.org/ramblings/dream/>.

Schug, A. & Wenzel, W. 2006 An evolutionary strategy for all-atom folding of the 60-amino-acid bacterial ribosomal protein L20. *Biophys. J.* **90**, 4273–4280. (doi:10.1529/biophysj.105.070409)

Skolnick, J. 2006 In quest of an empirical potential for protein structure prediction. *Curr. Opin. Struct. Biol.* **16**, 166–171. (doi:10.1016/j.sbi.2006.02.004)

Wu, S., Skolnick, J. & Zhang, Y. 2007 *Ab initio* modeling of small proteins by iterative TASSER simulations. *BMC Biol.* **5**, 17. (doi:10.1186/1741-7007-5-17)

Yang, J. S., Chen, W. W., Skolnick, J. & Shakhnovich, E. I. 2006 All-atom *ab initio* folding of a diverse set of proteins. *Structure* **15**, 53–63. (doi:10.1016/j.str.2006.11.010)

Zagrovic, B., Snow, C. D., Shirts, M. R. & Pande, V. S. 2002 Simulation of folding of a small alpha-helical protein in atomistic detail using worldwide-distributed computing. *J. Mol. Biol.* **323**, 927–937. (doi:10.1016/S0022-2836(02)00997-X)

Zhang, Y. & Skolnick, J. 2004 The protein structure prediction problem could be solved using the current PDB library. *Proc. Natl Acad. Sci. USA* **102**, 1029–1034. (doi:10.1073/pnas.0407152101)

Zhang, Y., Kolinski, A. & Skolnick, J. 2003 TOUCHSTONE II: a new approach to *ab initio* protein structure prediction. *Biophys. J.* **85**, 1145–1164.

Zhou, H. & Skolnick, J. 2007 *Ab initio* protein structure prediction using chunk-TASSER. *Biophys. J.* **93**, 1510–1518. (doi:10.1529/biophysj.107.109959)