Critical Review of the Rosetta Algorithm

Jake Wang
Biochemistry 218 Final Project

I. Introduction

Proteins are linear chains of amino acids that evolved the property of folding into unique three-dimensional structures in order to perform complex biological functions. Determining their structures has far-reaching implications in science and medicine. For example, once the structure of a protein is known, its function can be more readily elucidated. Additionally, appreciating how proteins fold can also facilitate drug design. While experimental determination of protein structure through X-ray crystallography and NMR spectroscopy remain the most accurate methods, these techniques are resource and time consuming. And with the advent of high-throughput sequencing, the ratio of known experimental structures to the number of all sequenced proteins is shrinking and now is less than 1 in 1000 (6). Thus, the need for computational methods for protein structure determination has never been greater.

Protein folding was first modeled by standard molecular dynamics (MD), which simulates physical movements of atoms by numerically solving the Newton's equations of motion for a system of interacting particles. Molecular mechanics force fields define the forces between particles and potential energy. MD is limited, however, by computational power. A nanosecond MD simulation of a 100-residue protein requires approximately 400 hours on a single processor. Additionally, the field has yet to settle on the most tractable and physically realistic model for water, nor is there a consensus on the values of parameters used in molecular mechanics potentials (1). The failings of this approach led to the development of lower complexity models.

Recent experimental data indicate that, when excised, local sequences of a peptide fold independent of the full protein (14). This suggests that perhaps in folding, local sequences have a propensity to form a limited number of structures, which influences the overall protein architecture. This "local bias" is one answer to the question of how to limit the conformational space searched in modeling algorithms. Rosetta, created in the Baker lab, implements this technique by coarsely sampling local structures for short segments of a polypeptide first. These fragments are then assembled randomly using a Monte Carlo simulated annealing search. Finally, the energy of the assembled models is minimized using a scoring function that accounts for nonlocal interactions such as compactness, hydrophobic burial, and specific pair interactions (disulfides and electrostatics). The figure below illustrates the conceptual basis for Rosetta; near native structures are labeled N.

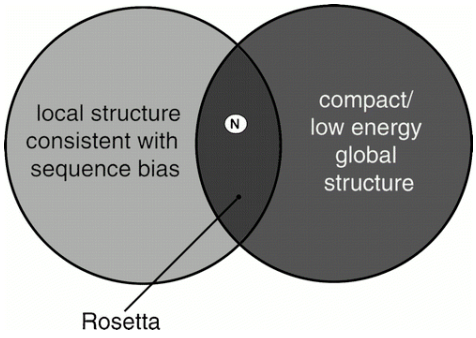

Figure 1. Conceptual Basis for Rosetta (1).

Since its birth, Rosetta has been identified by the community-wide critical assessment of structure prediction (CASP) experiments as one of the most successful current methods for *de novo* protein structure prediction (2). The purpose of this article is to review the Rosetta algorithm and its performance in CASP (particularly in CASP9) experiments. More importantly, we will address one major flaw in Rosetta that seems to have hindered progress in the field by examining how dynamic programming and a RNA structure determination method proposed by the Das lab offers a potential solution to this problem.

II. Methods

Fragment Library
Rosetta uses the Bayes statistical theorem to derive a structure from short fragments:

$$P(structure|sequence) = P(structure) \times \frac{P(sequence|structure)}{P(sequence)}$$

To calculate the full structure from the probabilities in the theorem, a fragment library must first be built by parsing input sequences into overlapping segments that are 3 and 9 residues long. These lengths were chosen because there is a greater correlation between local sequence and local structure for 3 and 9 residue fragments compared to other fragment lengths (<15 amino acids) (3). 200 of the most likely angles for these fragments are computed from X-ray resolved structures. Then the fragments are matched to ones in the protein data bank (PDB) via a PSIBLAST search. The last step of constructing the fragment library is to rank the matches by minimal steric overlap, favorable torsion angles and compatibility with secondary structure predictions made by Psipred, SAM-T99 and JUFO software.

Fragment Insertion and Assembly
Fragment assembly occurs by a Monte Carlo procedure, which begins with an arbitrary position in the protein in a fully extended conformation. A 3 or 9-residue fragment insertion window is randomly selected and a fragment from the top 25 matches in the ranked list for this position is inserted. For each insertion, the torsion angles in the protein segment are replaced with ones from the selected fragment. The resulting conformation's energy is calculated using scoring functions.

Two scoring functions are available. One is more coarse-grained but faster to compute whereas the second function is all-atomic and more accurate but also slower. The coarse-grained function takes into account the torsion angles of the backbone with the side chains described by a centroid located at the center of mass. In contrast, the all-atomic description considers all side-chain atoms, van der waals packing, hydrogen bonds and manifestations of water. The complete scoring function can be found in Rohl et. al., 2004 (2). If a conformational move decreases the energy of the overall structure (compared to before the fragment was inserted), the move is retained. Those that increase the energy are kept according to the Metropolis-Hasting acceptance probability $P=\exp(\Delta E/kT)$. This

requirement is necessary because some moves that increase energy allow the structure to escape from local minima in order to reach global minima. Each simulation attempts 28,000 fragment insertions, and of those, the final prediction is generally the conformation with the lowest energy.

Fragment insertion is a global move that generally affects the whole protein structure. Although a global move can change the overall structure faster, the acceptance probability is small because such rearrangements destroy the already formed local contacts, thereby increasing the energy dramatically. Thus, once the initial coarse tertiary structure is determined, the fragment insertion strategy must be replaced with more fine-grained potentials to allow for efficient model refinement. Finer sampling was accomplished with additional modification operators based on the following actions: random torsion angle perturbation, selection of globally nonperturbing fragments, rapid torsion angle optimization to offset global backbone perturbations, optimization of scoring function by gradient descent after a backbone modification and rapid optimization of side-chain rotamers (2). For example, the "crank" move is a combination of making an insertion at a selected window and of varying torsion angles of adjacent residues using a "wobble" operation. The two actions offset one another to reduce the overall perturbation. The figure below depicts the steps (A-D) involved in a "crank" fragment insertion. Initially in figure 1A, the nine-residue fragment insertion (red) causes a significant change in the original conformation (black to blue). However, after optimization of angles at two wobble sites (colored cyan and orange in 1B and 1C), the original and final structures are much more similar (black versus magenta in 1D).

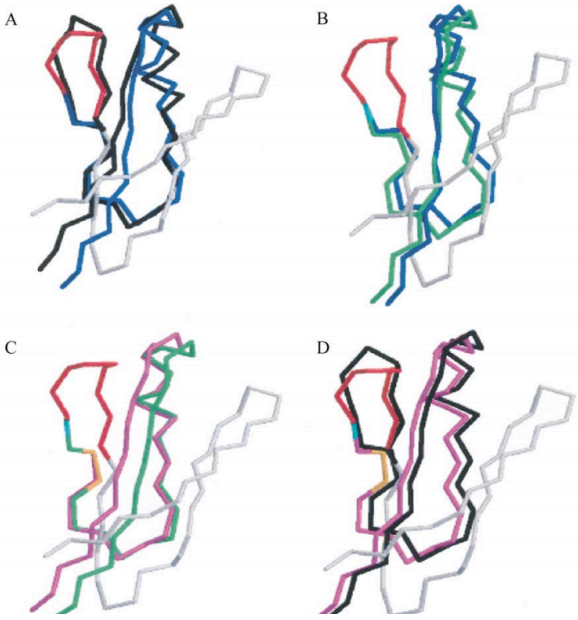

Figure 2. Example of a "crank" fragment insertion (2).

III. Advantages & Past Performance

By building a fragment library, Rosetta makes two assumptions that have made it one of the most successful structural determination programs. The first is that local amino acid sequence propensities bias each subsequence of a folding polypeptide chain toward a

limited number of alternative local structures before distant interactions narrow these down to one stable native arrangement (3). Secondly, Rosetta assumes that the distribution of configurations sampled by a peptide segment are well represented by the range of configurations already present in protein databases. Since local biases are influenced by subtle interactions including side-chain configurational entropy losses, current physical chemistry-based models do not capture all the nuances. Fortunately, Rosetta essentially circumvents the complex details of inter-atomic interactions through this knowledge-based step.

Rosetta has achieved remarkable success in advancing macromolecular modeling. Figure 3 shows successful predictions made by Rosetta at CASP 5. The target, T0135 in 3A has 106 residues and the target, T0171 in 3B has 69 residues. The structures on the left are the experimental structures. These models reached a Cα rmsd of approximately 4 Å.

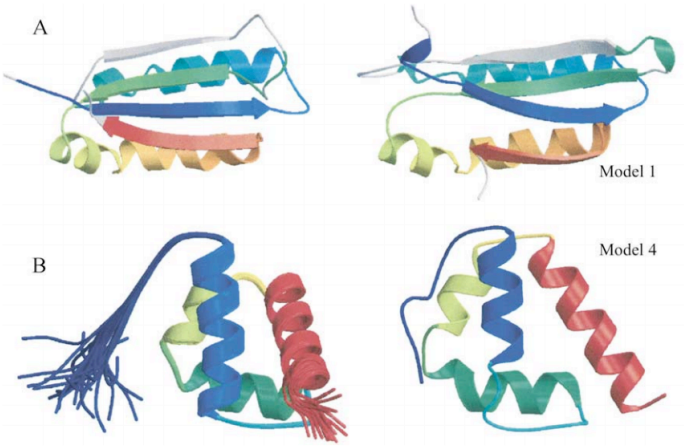

Figure 3. Rosetta-predicted protein structures at CASP 5 (2).

Moreover, in CASP 6, Rosetta successfully modeled all-beta and all-alpha proteins and became the first to refine a close-to-atomic level structure (1.59 Å Cα rmsd) from only the sequence (5). Most recently in CASP 9, Rosetta again distinguished itself by being noted as one of six groups that performed better than the rest (along with HHpredB, Zhang-Server, QUARK and Seok-server), especially at the local scale (15). The experiments included over 100 protein targets. Shown below are the rankings, which are based on three different scores (GDT-HA, GDC-All, and LDDT-All). The global distance test (GDT) counts the largest set of amino acid residues' Cα positions in the prediction structure that is within a certain distance from their position in the experimental reference structure. GDC-All is a similar measure but takes into account all non-hydrogen atoms. LDDT evaluates correct local interactions. As an example, the Rosetta model for a lactose-specific IIB component domain of the phosphoenolpyruvate carbohydrate phosphotransferase system (PTS) from *Streptococcus pneumonia*, obtained GDT-HA and LDDT-all scores of 71.6 and 89.8, respectively, both of which were 3 to 6 points higher than other groups as well as 15 points higher than a pseudo-model generated form the best template identified by PSI-BLAST.

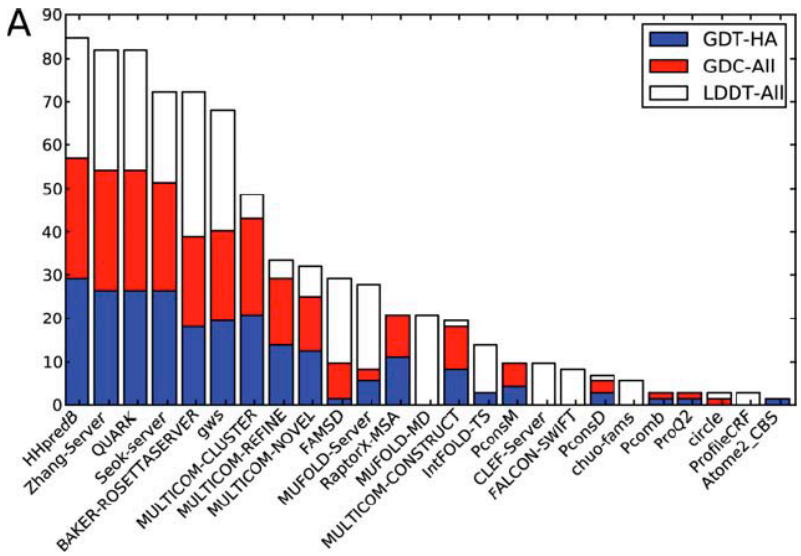

Figure 4. Ranking the top 25 groups on common targets in CASP9. The Rosetta algorithm is among the six best methods (15).

IV. Challenges and Bottlenecks

After significant advances in early CASP experiments, progress in the field in recent years has been more modest (7). And despite its effectiveness, inherent problems in Rosetta algorithm limit its ability to produce high-resolution models of even some small protein and RNA structures such as the 20-residue mini-protein Trp cage and a 10-residue protease-binding loop of the chymotripsin inhibitor from barley seeds (8). Inaccurate models may be attributed to flaws in the Rosetta energy function. For example, Rosetta's solvation model ignores nontrivial solvation structure around polar groups and "second shell" water effects (7). Additionally, Rosetta's hydrogen bond potential does not consider effects of charged atoms or cooperativity within H-bond networks. These issues may be rectified with improvements to the energy function by incorporating more physical information and parameters for solvation, H-bonds and electrostatic interactions.

But, the major bottleneck in many of these problems is simply that the native structure is not sampled. This issue precedes the energy function problem because in this case, the native structure's energy is never even evaluated by scoring functions. Incomplete conformational sampling arises from the absence of native torsions in the fragment library. In other words, if a 3-nt or 9-nt fragment in the protein is unique and cannot be found in the protein database, the native structure containing such a fragment will obviously not be sampled by Rosetta. This is a fundamental challenge that faces all template-based modeling. In 2011, the Das lab conducted an enumerative stepwise ansatz to enable RNA loop modeling to atomic-accuracy (9). This approach can be implemented in the Rosetta framework and could be the solution to the conformational sampling bottleneck. It employs a recursive step-by-step enumeration of millions of conformations for each monomer to cover all build-up paths. But, we will first introduce the dynamic programming algorithms that underlie this method.

V. Dynamic Programming

Dynamic programming algorithms are utilized in many prominent computational biology programs including BLAST, FASTA (sequence search databases), CLUSTALW (multiple sequence alignments) and HMMER (profile hidden Markov models) (10). Such algorithms consist of *three* major components. We will illustrate these parts in the context of pairwise alignment as a simple example. We will be aligning two sequences x and y with lengths M and N, respectively.

1. First, we need to *recursively define the optimal score* by breaking up the problem into smaller independently optimizable pieces. Starting from the end of the sequences, the residues x_M and y_N could be aligned to each other. However, x_M could also be aligned to a gap (while y_N is paired with another base upstream of x_M), and vice versa for y_N. Thus we must be able to calculate the optimal score for all three cases because the optimal alignment will correspond to the highest scoring case. In the first scenario, the score of the entire alignment, S_T, equals $S(x_M,y_N) + S(M-1, N-1)$, which is the sum of the score for aligning x_M to y_N and the score for the optimal alignment of all bases before them. If x_M is aligned to a gap (second case), $S_T = S(M-1, N) + g$, where g is the gap penalty. Similarly, if y_N is aligned to a gap, $S_T = S(M, N-1) + g$. To calculate these other scores, we need the scores of even simpler problems: S(M-2, N-2), S(M-2, N-1), S(M-1, N-2), S(M-2, N), S(M-1, N-1), S(M, N-2). The advantage of a recursive definition of the optimal score is that the optimal alignment of $x_1 \ldots x_{M-1}$ to $y_1 \ldots y_{N-1}$ is independent of the score $S(x_M, y_N)$. By solving tiny alignment subproblems, the overall optimal alignment can be determined.
2. To keep track of each subproblem, a *dynamic programming matrix* is employed to memorize the solutions of optimal subproblems in an organized tabular form. For example, in pairwise sequence alignment, the optimal scores S(i, j) for the ith and jth residue is recorded in the (i, j) cell of the matrix shown below:

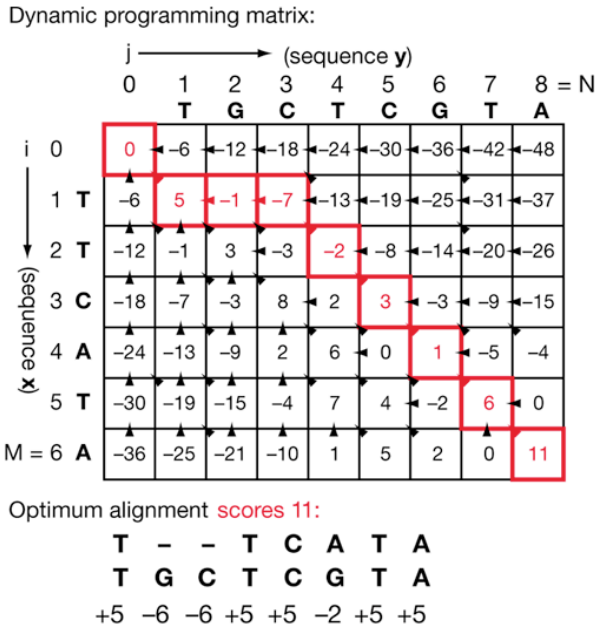

Figure 5. Dynamic programming matrix. The cells in the optimum path are highlighted in red and the arrows depict "traceback pointers" to mark which of the three cases were optimal for reaching each cell (10).

The matrix is filled starting with the easiest and smallest problems such as the scores S(0, 0), S(i,0) and S(0, j). Each cell is then filled with the optimal score calculated recursively from the three adjacent cells to the upper left, above and to the left. These solutions lend themselves to solving progressively bigger problems until the matrix is filled, at which point the last score computed S(M, N) is the score of the optimal alignment.

3. Even though the optimal score of the complete sequence alignment is known, we still have to recover the actual alignment corresponding to this score through a recursive "*traceback*" of the matrix. Starting in cell (M, N), we move backwards to one of the three previous cells whose score was used to calculate the current cell's score. Then, we continue to retrace the steps until we reach cell (0, 0). The sum of movements through the matrix yields the optimal alignment.

It is worth noting that dynamic programming remains extremely computationally demanding, but it will always guarantee a mathematically optimal solution. Another caveat is that only scoring systems than allow the optimal solution to be broken into independent parts can be used in dynamic programming.

VI. RNA Loop Modeling Using Enumerative Stepwise Ansatz

The same principles in dynamic programming of pairwise alignment can be applied to folding RNA structures. Like before, we will demonstrate the technique in folding RNA through a simplification of the problem by assuming that the native RNA structure can be attained simply by maximizing the number of base pairs. Thus, our scoring system is +1 per base pair and 0 for everything else. The optimal score for the best structure for a subsequence from position i to j in a sequence of length N is defined as S(i, j). Next, we recognize that S(i, j) can be solved recursively in terms of the optimal scores of smaller subsequences just like in pairwise alignment. Eddy et. al. 2004 outlines the different possible structures of nested base pairs on i..j (11):

1. i and j are base-paired together; they are added onto a structure for i+1..j-1 (Fig. 6.1).
2. i is unpaired; it is added onto a structure for i+1..j (Fig. 6.2)
3. j is unpaired; it is added onto a structure for i..j-1 (Fig. 6.3)
4. i, j are paired, but not to each other; the structure for i..j combines substructures for two subsequences: i..k and k+1..j (Fig. 6.4)

The four possibilities are shown in figure 6. Red dots indicate the new bases that are added onto previously calculated optimal substructures.

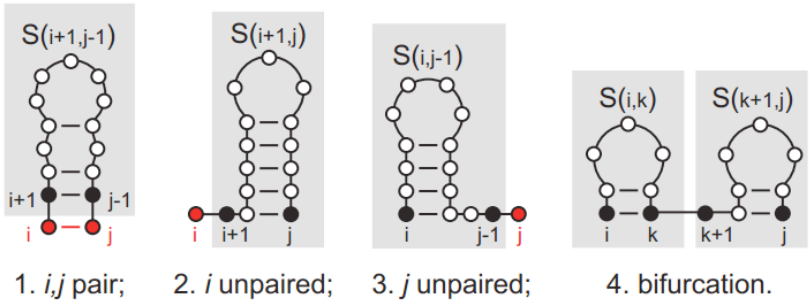

Figure 6. Four possible structures for a subsequence i, j (11).

For case 1, the score that we add for the base pair i, j does not affect the optimal structure for i+1..j-1. In fact, the optimal structure on i+1..j-1 and its score s(i+1, j-1) are independent of anything else that is built on top of it. Thus for the first scenario, S(i, j) = S(i+1, j-1) +1 since i and j can base pair. The same recursive methods apply for the other cases. In case 2, S(i,j) = S(i+1, j) + 0 since i is unpaired. Likewise, in case 3, S(i,j) = S(i, j-1) + 0 because j is unpaired. Finally, in case 4, S(i, j) = S(i, k) + S(k+1, j) since S(i, k) and S(k+1, j) are separate sub-structures and i and j are paired but not to each other. After determining the scores for all four possibilities, the greatest of the four corresponds to the optimal score S(i, j). As we did before, these scores can be tabulated in a triangular matrix starting from subsequences of length 0 or 1, which have no base pairs [S(i, i) = S(i, i-1) =0], and proceeding step by step from there. To recover the optimal structure, we conduct the traceback method again.

Of course, the scoring function used here is not sufficient for RNA structure prediction. Instead, in RNA structure prediction programs, secondary structures are evaluated based on thermodynamics to make sure the prediction has a globally minimum energy structure. These functions approximate the overall free energy by incorporating parameters for different types of loops and base pairing interactions (11). The Das lab shows how a recursive stepwise ansatz can take advantage of such energy functions and dynamic programming to systematically sample RNA loop conformations at atomic resolution (9).

Recursive stepwise ansatz is similar to ab initio methods that were explored early in the field by the Scheraga lab in the 1980s and Levinthal in 1968. These were largely abandoned in favor of Monte Carlo and knowledge-based methods in order to overcome limited computational power (12). However, as mentioned above, it has been shown that biomolecules with noncanonical properties are challenging to model de novo using Rosetta and knowledge-based methods because of incomplete conformational sampling. For example, a fragment assembly of RNA with full-atom refinement method (analogous to Rosetta for RNA modeling), FARFAR, was tested on a benchmark of 32 RNA motifs. It could only obtain near atomic accuracy models in half of the cases and failed to construct models within 1.5 Å rmsd of the crystallographic conformation (9). The J2/4 loop of the TPP riboswitch could not be solved by FARFAR even though it is only five nucleotides in length because of its irregular properties such as noncanonical loop torsions. It soon became clear that the lack of these native torsions in the fragment library prevented accurate modeling. In fact, the native structure could be recovered simply by inserting native torsions into the fragment library. Enumerative single-nucleotide

building, which permits fine-grained exploration of torsional conformations that form well-packed structures with multiple hydrogen bonds, circumvents this conformational sampling bottleneck. Many of the conformations produced by this approach include rare torsional combinations observed in native loops but are absent from consensus rotamers in the PDB.

Recently in 2011, Sripakdeevong et. al. employed this technique as the stepwise assembly (SWA) method in Rosetta to solve RNA loop-modeling problems that have confounded current knowledge-based methods. The general recursive scheme consists of a single-nucleotide building step, a bulge-skip building step (to allow for extra-helical bulges), a chain closure step and a clustering step. Because exhaustive enumeration for even a five base-pair loop like the J2/4 loop requires too much computational power, they calculated a low-energy ensemble of models for subregions. These subregions are then combined by chain closure, and the 1,000 lowest energy cluster centers are selected to be analyzed. For the J2/4 loop, they obtained a 0.85 Å rmsd model, which was far closer to the native structure than the prediction made by FARFAR as shown in figure 7.

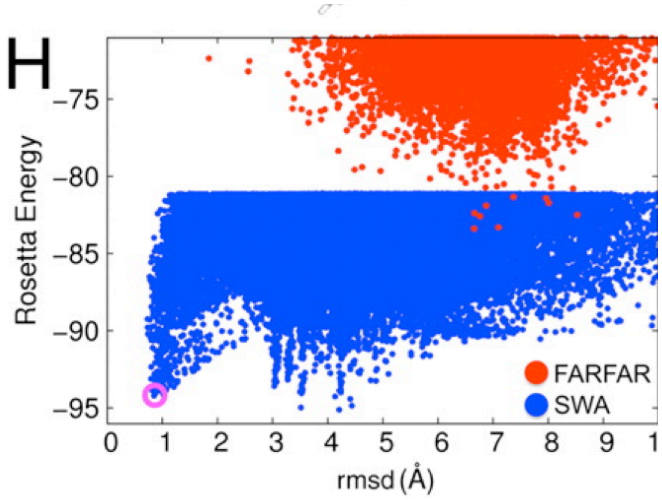

Figure 7. Rosetta all-atom energy vs. all-heavy-atom rmsd to the crystallographic conformation. De novo models derived by SWA are shown in blue while predictions made by FARFAR are represented in red (9).

In total, SWA obtained near native models (<1.5 Å rmsd) for ten of fifteen test cases (targets included RNA loops such as a 10-nt loop of the large ribosomal subunit from *Haloarcula marismortui*) compared to the four cases recovered by FARFAR. Furthermore, of the five remaining cases, SWA predictions actually had lower energies than the optimized experimental structures. SWA did sample models within 1.5 Å rmsd of the experimental conformation for four of those five cases, but they were not chosen as one of the lowest energy cluster centers. Sripakdeevong et. al. suggest that problems with the Rosetta all-atom energy function may account for why some models have lower energy than native structures. Nevertheless, it seems that SWA resolves the conformational sampling bottleneck. What's more, SWA successfully predicted the structure of a tetraloop/receptor motif (C7.2 tetraloop-docked receptor) in a blind trial (no known experimental structure). The model possessed noncanonical features (same-

stranded G-A base pair and an extrahelical bulge) that were consistent with the chemical behavior of the actual receptor in chemical modification experiments. Although only results for the implementation of SWA in single-stranded RNA loop modeling have been published, there is great optimism that the strategy should be applicable to a diverse class of macromolecular modeling problems from modeling multiple RNA strands to predicting protein structure at high resolution.

VII. Conclusion

When Rosetta was first introduced to the structural modeling community, it was attractive because of its use of fragment library to conduct a coarse-grained conformational search before its finer, all-atom refinement stage. Indeed, incorporation of this "knowledge-based" algorithm has allowed Rosetta to excel as evidenced by its performance in the CASP competitions. However, progress has slowed since CASP8 and CASP9, and demand for even higher resolution models has increased. Ironically, it has become clear that relying on already experimentally determined structures in protein databases (once an advantageous tool) prevents sampling of the lowest energy conformations of a large class of proteins, especially those with noncanonical properties. Here, we introduce one of the most promising methods that attempts to overcome this barrier: a recursive stepwise ansatz founded on the principles of dynamic programming. The success with which it has predicted simple RNA-loop structures suggest that step-by-step build-up approaches may be the answer to structural modeling especially with the advent of massive parallelization of high-performance computer clusters. As CASP 10 approaches, it will be exciting to see if they are in fact the breakthrough the field needs to model protein structure more accurately and efficiently.

Works Cited

1. Bonneau, R. and Baker, D. Ab initio protein structure prediction: progress and prospects. *Annu Rev Biophys Biomol Struct*. 2001;30:173-189.
2. C. Rohl, C. Strauss, K. Misura, and D. Baker. Protein structure prediction using Rosetta. *Methods Enzymol*. 2004; 383:66–93.
3. Simons KT, Kooperberg C, Huang E, Baker D. Assembly of protein tertiary structures from fragments with similar local sequences using simulated annealing and Bayesian scoring functions. *J Mol Biol.* 1997;268:209–225.
4. R. Jauch, H. C. Yeo, P. R. Kolatkar, and N. D. Clarke. Assessment of casp7 structure predictions for template free targets. Proteins: Structure, Function, and Bioinformatics. *Proteins*. 2007; 69(Suppl 8):57-67.
5. P. Bradley, L. Malmstrom, B. Qian, J. Schonbrun, D. Chivian, D. Kim, J. Meiler, K. Misura, and D. Baker. Free modeling with Rosetta in CASP6. *Proteins*. 2005; 61(Suppl 7):128–134.
6. Moult, J., Fidelis, K., Kryshtafovych, A. and Tramontano, A. Critical assessment of methods of protein structure prediction (CASP)—round IX . *Proteins*. 2011;79: 1–5.
7. A. Kryshtafovych, K. Fidelis, J. Moult. CASP9 results compared to those of previous CASP experiments. *Proteins*. 2011;79(Suppl 10):196-20
8. Das R. Four small puzzles that Rosetta doesn't solve. *PLoS ONE.* 2011;**6**:e20044.
9. Sripakdeevong P, Kladwang W, Das R. An enumerative stepwise ansatz enables atomic-accuracy RNA loop modeling. *Proc Natl Acad Sci.* 2011;108(51):20573-8.
10. Eddy S.R. What is dynamic programming? *Nat. Biotechnol.* 2004;22:909-910.
11. Eddy S. R. How do RNA folding algorithms work? *Nat. Biotechnol.* 2004; 22:1457-1458.
12. Moult J, Fidelis K, Kryshtafovych A, Rost B, Tramontano A. Critical assessment of methods of protein structure prediction—round VIII. *Proteins*. 2009; 77(Suppl 9):1–4.
13. Das R, Baker D. Macromolecular modeling with rosetta. *Annu. Rev. Biochem.* 2008;**77**:363–382.
14. Baker D. A surprising simplicity to protein folding. *Nature.* 2000;405:39–42.
15. Mariani, V., Kiefer, F., Schmidt, T., Haas, J. and Schwede, T. Assessment of template based protein structure predictions in CASP9. *Proteins*. 2011;79: 37–58.