SCOPE: An Open-Source, C++ Implementation for Calculation of Protein Energetics from First Principles

Timothy Matthew Fawcett[1], **Stephanie Irausquin**[1], **Mikhail Simin**[1], **and Homayoun Valfar**[1]
[1]Computer Science and Engineering, University of South Carolina, Columbia, SC, USA

Abstract - *SCOPE (Semi Classical Open Source Protein Energy) is an open-source program that has been implemented in the Object-Oriented C++ language, capable of computing none-bonded energies for protein structures from first principles. SCOPE is also capable of manipulating protein structures within the Rotamer space instead of the typical Cartesian space. This approach simplifies calculation of the transitional force field through elimination of unnecessary terms such as bond lengths, bond angles, and other peptide geometrical constraints. Elimination of unnecessary force calculation is beneficial in improving computational performance while the OO approach results in better program maintenance and customization for other projects. Finally, the calculation of forces has been compared and confirmed with respect to other commonly used programs such as CHARMM and Xplor-NIH. Further development of SCOPE can be very beneficial in refinement of computationally modeled structures, or potentially Ab-Initio calculation of structures from first principles without any reliance on homology modeling.*

Keywords: protein structure generation, non-bonded energy, protein folding, protein structure refinement

1 Introduction

Proteins play a critical role in maintaining the homeostatic functions of a biological cell and are often referred to as the working molecules of a cell [1]. Although proteins are prevalently recognized for their enzymatic activities, they are also involved in structural or mechanical functions, as well as regulatory functions [1]. Given that a protein must be folded into its native structure in order to carry out its particular function, it is of no surprise that mis-folded proteins are linked with disease [2]. Certain cancers, cystic fibrosis and amyloid diseases such as Alzheimer's, Parkinson's, and Type II Diabetes are such examples [2-5]. Understanding the mechanisms involved in protein folding and protein structure prediction has never been so important. Collaborations between experimental and computational fields have the potential to aid in a number of different applications that will not only accelerate treatments and therapies for a number of diseases, but will also replace the use of costly and time consuming approaches with faster, cheaper computer simulations [6-9].

A protein's structure often dictates its function and therefore investigation of structure of biologically active proteins has intrigued scientists for several decades [7]. Within the last 25 years, the combination of both novel and powerful experimental and theoretical techniques, have contributed to a number of important advances in elucidating protein folding mechanisms; yet there are still challenges that need to be overcome in order to obtain a complete solution [7]. Currently, the "protein folding problem" is often described as 3 different problems: (1) the folding code – what thermodynamic balance of inter-atomic forces dictates protein structure; (2) protein structure prediction – how to predict a protein's native structure given its amino acid sequence; and (3) the folding process - the kinetics associated with how proteins fold quickly [6].

The concept of an energy landscape is fundamental to the mechanism of protein folding [10]. The thermodynamic hypothesis of protein folding states that a protein will fold to a certain form because it is the most favorable [11]. Here an open source software program, SCOPE (Semi Classical Open Source Protein Energy), is presented which allows the user to recreate structures and explore the calculated non-bonded energy potentials associated with those structures using only the initial structure and its dihedral angles as input. Furthermore, due to formulation of protein structure in the rotamer space, several of the traditional force-terms are no longer required. The simplified force field can result in a smoother and more manageable energy landscape.

2 Methods

2.1 Program Details

SCOPE utilizes an object oriented approach and is written in C++. The class structure starts from the fundamental *Atom* class and through compositional inheritance constructs the *AminoAcid*, and finally the *PolyPeptide* objects. The AminoAcid class contains an array of Atoms to represent an amino acid, while the PolyPeptide contains an array of amino acids which constitute a protein. The AminoAcid class is a factory class which constructs all 20 amino acids; it contains the attributes of the backbone atoms, as well as the ϕ, ψ, and ω angles, which are part of the REDCRAFT engine [12]. Because backbone atoms are the same for all amino acids except proline, there is only one array of atoms that contains the backbone atoms. The proline amino acid differs from all other amino acids in that it has no

amide hydrogen and its sidechain is linked back to the backbone atoms. Therefore, in the case that a proline amino acid is created, the array of atoms containing backbone atoms for all other amino acids is modified by converting each hydrogen backbone atom into a Cδ; the coordinates of the backbone atoms are then updated accordingly. When the protein is created by the use of ϕ and ψ angles, there is an assumption of a perfect geometry, this translates to perfect bond lengths and favorable bond angles for all atoms of each residue.

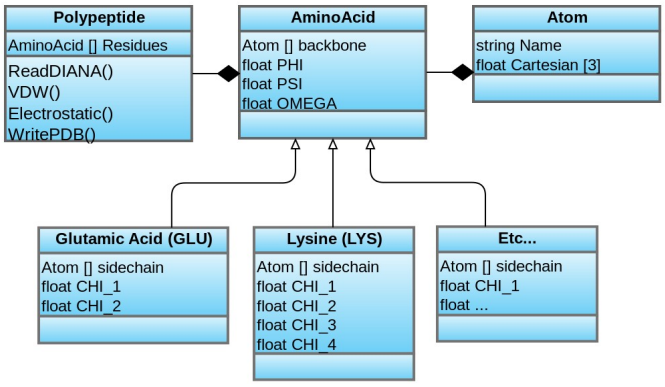

Figure 1: A UML diagram of the class structure for amino acids

The 20 different amino acids inherit the factory amino acid class (Fig. 1). Each class contains the appropriate side chain atoms as well as the χ angles for that particular amino acid. For example, Glycine has a side chain with a single hydrogen atom and no χ angle. Each amino acid class contains the same functions that will rotate their χ angles and update the positions of the side-chain atoms. The side-chain atoms of each amino acid records their own coordinates to a .pdb file.

2.2 Implementation

SCOPE expects two input files from the user. The first input file is a DIANA[13] file (.ang) whose format contains the dihedral angles of each residue; if available, the angles are listed in the following order: ϕ, ω, χ, and finally the ψ angle (Fig. 2). A protein is then generated one amino acid at a time by reading in each residue and rotating its angles so that they correspond to the values of the coinciding DIANA file.

```
# Structure of 3LAY001 from MOLMOL
 44 LEU   CHI1   -87.498 CHI2   155.773 PSI    -41.124
 45 THR   PHI    -58.676 CHI1    83.557 PSI    159.867
 46 THR   PHI    -60.703 CHI1   -61.376 PSI    -62.493
 47 GLU   PHI    -57.493 CHI1    47.340 CHI2   -84.354 CHI3   158.889
          PSI    -21.774
 48 GLN   PHI    -77.998 CHI1   -77.310 CHI2  -179.757 CHI3   -19.924
          PSI    -44.705
```

Figure 2: Example fo a DIANA formatted file

The second input file is a protein structure file (.psf) which contains information related to the topology of the molecule. This topology file provides a rich set of information such as which 3 atoms make a bond angle, and which 4 atoms make a dihedral angle. Both CHARMM [14] and Xplor-NIH [15] create a .psf file compatible with SCOPE's requirements. SCOPE utilizes the topology information to calculate Van der Waals energy and electrostatic energy of the protein. These energies are then output to the command line along with a .pdb file of the recreated protein.

Because of our previous assumption of perfect geometry during protein construction, SCOPE refrains from calculating the energies associated with bonded terms. As mentioned previously, SCOPE calculates the non-bonded Van der Waals and electrostatic energy terms seen in CHARMM and Xplor-NIH simultaneously. This is accomplished through a series of loops that compares each atom with every other atom. The algorithm begins by comparing the first atom to all other atoms, one at a time and computing a potential energy for each comparison. Similarly, the second atom is compared to all other atoms, except the first atom, one at a time and an energy term is computed for each comparison. These comparisons and energy calculations continue for all remaining atoms so that no duplicate calculations are made, thereby alleviating unnecessary calculations that would needlessly increase computational demands.

The Van der Waals term is used to measure the attraction and repulsion of two atoms. The 12 – 6 Lennard-Jones Potential is used to calculate its value(1). In this equation, σ_{ij} represents the sum of the Van der Waals radii of the two atoms ($\sigma_{ij}=\sigma_i+\sigma_j$); ε_{ij} signifies the well depth of the graph calculated as $\varepsilon_{ij}=\sqrt{\varepsilon_i \varepsilon_j}$; and r_{ij} denotes the distance between the two atoms(1). The sigma and epsilon values for each atom are the same value in the CHARMM program. The Van der Waals potential can also be calculated using different number of bond exclusions. The default value is the 1-4 atom exclusion, which means 4 atoms with three bonds separating them are excluded from the calculation. A flag can be set when the program is executed to exclude nothing (every atom to atom calculation), 1-2 atom exclusions (2 atoms with a single bond), or 1-3 atom exclusions(3 atoms with 2 bonds are excluded). These exclusions are cumulative so a 1-4 atom exclusions includes the exclusion of 1-2 atoms and 1-3 atoms.

$$LJ=\varepsilon_{ij}\left(\frac{\sigma_{ij}}{r_{ij}}\right)^6-2\left(\frac{\sigma_{ij}}{r_{ij}}\right)^{12} \quad (1)$$

The electrostatic term is used to determine the electrical charge between two atoms. The electrostatic potential is found using Coulomb's Law (2). The charge of each atom is denoted by q_{ij}; ε_0 symbolizes the permittivity of vacuum; just as in the Lennard-Jones equation, the distance between the two atoms is represented by r_{ij}.

$$E = \frac{q_i q_j}{4\pi\varepsilon_0 r_{ij}} \quad (2)$$

Both the Van der Waals and electrostatic energy terms include a distance constraint (r_{ij}). This is to account for instances where the distance between two atoms may be extremely large; in such a case, non-bonded energies are not calculated but set to zero instead.

2.3 Testing Strategy

Initially, SCOPE's ability to generate structures was tested. This was accomplished by creating a peptide of 5 residues in MolMol [16], which is referred to as 5RES, with ϕ and ψ angles rotated to values different from that of MolMol's default ϕ and ψ angle values. The residues comprising 5RES were chosen randomly, with the exception of proline, which was specifically placed in the center of the peptide for its properties discussed previously (section 2.1). Next the two input files required by SCOPE (DIANA and psf file) were created. The DIANA input file was constructed in MolMol and a structure file was created using the CHARMM program. These files were then input into SCOPE. The resulting .pdb file generated by SCOPE was then compared to the original 5RES .pdb created in MolMol by calculating backbone root mean square deviation (RMSD) and also by comparing ϕ and ψ dihedral angles between the two.

Next, SCOPE's ability to construct energetically favorable structures was tested using 12 different proteins (1A1Z, 1DP3, 1TGR, 2J5Y, 1A1W, 3LAY, 1G10, 1J4V, 2EZM, 2EZN, 2MOB, 2PTV) from the Protein Data Bank (PDB) [17]. These particular proteins were selected so that it would be able to test a variety of secondary structures(i.e., alpha-helical, beta-strand, and alpha-beta mix). Both a DIANA file (using MolMol) and a structure file (using CHARMM) were created in order to generate a SCOPE .pdb file for each of the 12 PDB proteins. The resulting SCOPE generated .pdb file was then compared to its original .pdb file for each protein by calculating the backbone RMSD between the two. 1000 similar structures for each protein were created by perturbing or randomly altering the ϕ and ψ angles of the DIANA file; the resulting perturbed structures were all within 6Å of the SCOPE generated protein. Next, the Van der Waals potential was calculated for the SCOPE generated protein as well as the 1000 perturbed structures for each protein (therefore, 1001 structures for each protein) using both SCOPE and CHARMM. Similarly, the electrostatic potential was calculated in both SCOPE and CHARMM .

3 Results

3.1 Structure Generation

In order to test SCOPE's ability to generate structures comparable to other programs, 5RES peptide generated in MolMol was compared to the 5RES peptide generated in SCOPE. The resulting backbone RMSD between the 2 structures is 0.019Å. Comparison of ϕ and ψ angles between the two structures, as well as the peptide sequence, are listed in Table 1. The difference in the angles shown is due to numerical precision error between MolMol and Scope. MolMol will read in the coordinates of the atoms but when displayed within MolMol many of the coordinates have slight differences in the hundredths and thousandths place. Some examples of these numerical precision errors are listed table 1.

Table 1: Peptide of 5 residues(5RES) created MolMol to test the ϕ and ψ angles assigned in MolMol to the ϕ and ψ angles created with SCOPE.

Residue	Original φ	Original ψ	SCOPE φ	SCOPE ψ
TYR	180	150	180	149.956
GLN	90	60	90.041	59.985
PRO		30		29.926
LYS	-30	0	-29.927	-0.001
ALA	-60	180	-59.954	180

3.2 Protein Model Generation & Non-bonded Energy Evaluations

The previously mentioned 12 proteins were used to test SCOPE's accuracy in representation of protein structures and calculation of potential energies. For each protein, comparisons between the protein obtained from the PDB (original) and the same protein generated by SCOPE (SCOPE) were made by calculating the backbone RMSD between the two. The resulting structural similarity results are shown in Table 2.

Table 2: The different calculations of backbone RMSD between 12 test proteins obtained from the PDB and the same proteins recreated using SCOPE.

Protein	**Secondary Structure**	**Size (Amino Acids)**	**BB RMSD to DIANA file**
1A1Z	α	83	0.633
1DP3	α	55	0.226
1TGR	α	52	0.468
2J5Y	α	61	0.318
1A1W	α	83	0.654
3LAY	α	79	0.631
1G10	α/β	102	0.548
1J4V	β	101	0.441
2EZM	β	101	0.667
2EZN	β	101	0.676
2MOB	α/β	94	0.536
2PTV	β	96	0.354

Each SCOPE generated protein was then perturbed into 1000 structures. The phi and psi angles were rotated by +/- 2 degrees to create 1000 different structures within 6 angstroms. The Van der Waals and electrostatic potential energies were calculated separately in both SCOPE and CHARMM for each of the 1000 derivative structures. Figure

3 displays the correlation for the Van der Waals potential calculated by CHARMM and SCOPE for protein 3LAY and figure 4 reveals the correlation between the electrostatic potential calculated by CHARMM and SCOPE for the same protein (3LAY). Figures 5 – 8 contain the correlation between the Van der Waals potential and the electrostatic potential calculated by CHARMM and SCOPE for proteins 1G10 and 1J4V.

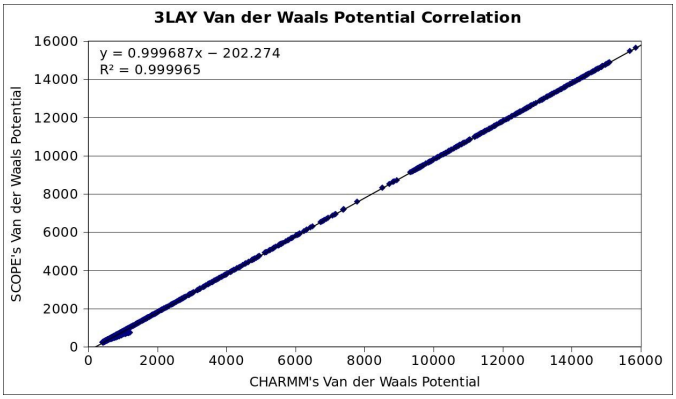

Figure 3: The Van der Waals Correlation between the CHARMM program and SCOPE for protein 3LAY.

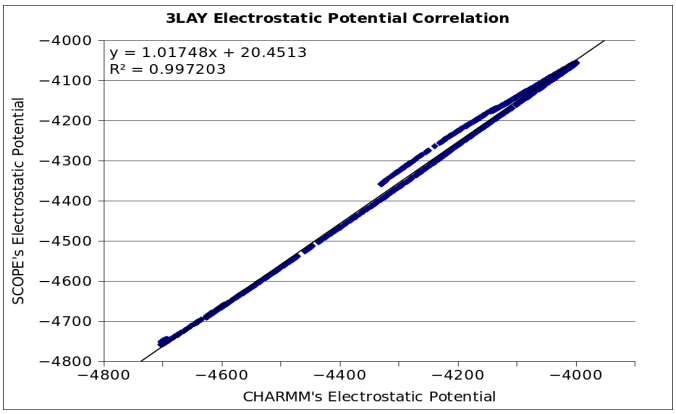

Figure 4: The electrostatic Potential between the CHARMM program and SCOPE for protein 3LAY.

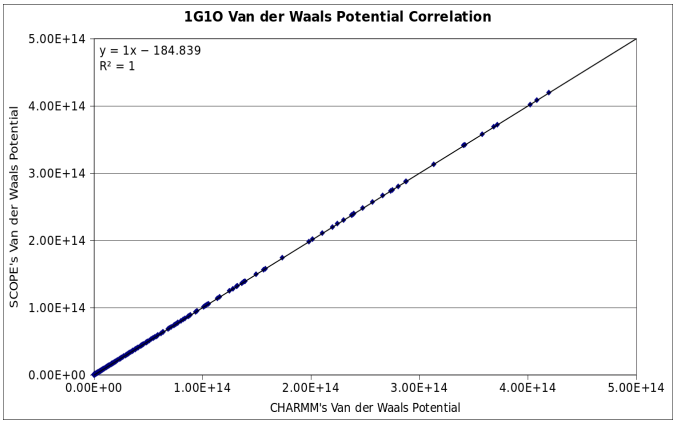

Figure 5: The Van der Waals Potential between the CHARMM program and SCOPE for protein 1G10.

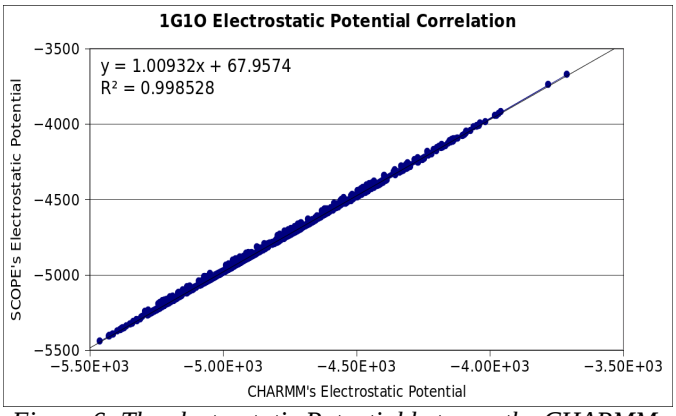

Figure 6: The electrostatic Potential between the CHARMM program and SCOPE for protein 1G10.

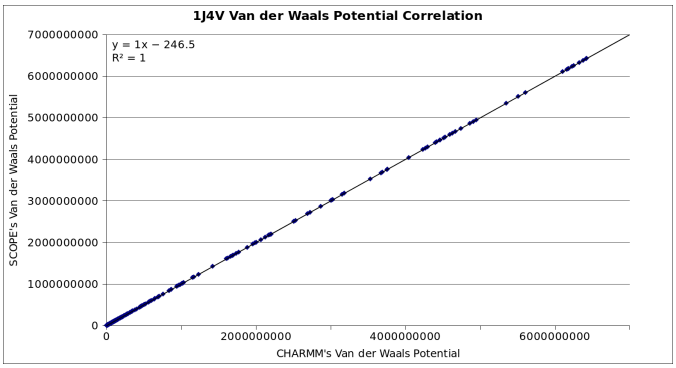

Figure 7: The Van der Waals potential correlation between the CHARMM program and SCOPE for protein 1J4V

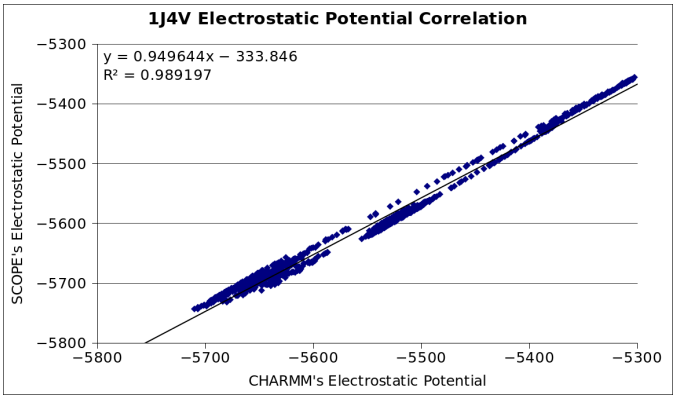

Figure 8: The Electrostatic potential correlation between the CHARMM program and SCOPE for protein 1J4V

4 Discussion

SCOPE's ability to generate structures comparable to those constructed in MolMol is demonstrated using the constructed 5RES peptide and 12 proteins representing different structural categories and sizes. In all of these exercises, the constructed structures by SCOPE are nearly identical to their original counterparts generated by MolMol. The subtle differences that are observed are due to more precise representation of structures by SCOPE. Inherently,

PDB file format imposes a limited numerical precision in representing the atomic coordinates in the Cartesian space. The backbone RMSD between the 5RES peptide generated in MolMol and the 5RES peptide generated in SCOPE is very low (0.019Å).

It is important to note that due to peculiarities of MolMol, the ϕ angles of prolines are not computed and therefore not reported in the DIANA format. Manual editing of the DIANA file is to capture the ϕ angle of prolines. In some instances other violations of standard peptide geometry causes a significant distortion of structures. For example, our preliminary calculations of backbone RMSD between original proteins and that same protein generated with SCOPE (data not shown) revealed problematic values (in excess of 15Å), which is explained using the 3LAY protein as an example. 3LAY contains two prolines at residues 20 and 43 of the protein. After carefully examining these specific residues some interesting observations were made as to why there appeared to be such huge diversions in backbone RMSD. In the case of residue 20, the original structure has an ω angle of –165 degrees, yet SCOPE was not able to rotate the ω angle accordingly. With regard to residue 43, the original structure contains a ϕ angle that is rotated to -53 degrees, yet the corresponding ϕ angle generated in SCOPE defaults to -72.3 degrees. Because MolMol does not calculate the ϕ angles of proline, the Diana file created from MolMol does not write out a ϕ angle for the proline and the angle is not rotated properly. To circumvent the issue, our solution was to add the ϕ angle to the proline in the DIANA file and rotate the ω angles in the original structure to be 180 degrees. These changes allowed for a reduction in the backbone RMSD from 0.855Å to 0.631Å (Fig. 9).

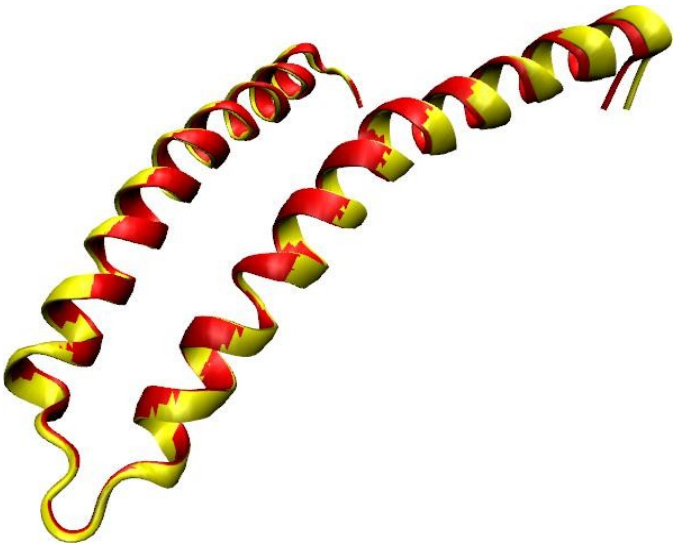

Figure 9: Comparisons between the 3LAY protein from MolMol created by the DIANA file (seen here in red) and the same protein generated by SCOPE using the DIANA file (seen here in yellow) after modifications produced a reduced backbone RMSD (0.631Å).

Protein representation in rotamer space has some distinct advantages. One such advantage is related to the reduced set of information that is needed to reconstruct a protein structure. The backbone only will have dihedral angles ϕ, ψ, and ω. If an all atom version is used then the x, y, z coordinates of 7 atoms need to be known for a total of 35 different parameters. So in the backbone alone, the rotamer representation reduces the number of parameters from 35 to 3.

The use of rotamer space to construct a protein also has some disadvantages, which primarily relate to the loss of information. Bond angles created under the rotamer geometry in both MolMol and SCOPE may differ from the bond angles that are present in the original file obtained from the Protein Data Bank. This was observed when bond angles were calculated between all bond angles for 3LAY, demonstrating bond angles that differed by as much as 65 degrees. Another problem may arise with bond lengths. When bond lengths were compared between all bond lengths for 3LAY, although small, the maximum difference was 0.07Å. The major differences in bond angles and possibly bond lengths led to major differences in atom coordinates, further contributing to high values in backbone RMSD (data not shown).

To alleviate this issue, structures made manually in MolMol (not input as a .pdb file) use the perfect geometry assumption. This allows for structures to be created in MolMol and then compared to structures generated by SCOPE. Using the amino acids of residues 18 – 66 of the 3LAY structure, since a proline is located centrally to the structure, a protein in MolMol was manually created and then the structure was re-created in SCOPE. Comparisons between the two structures revealed a backbone RMSD of 0.290 Å (Fig. 10). Therefore by assuming perfect geometry for bond lengths, bond angles, and the ω angles SCOPE can accurately depict the structure.

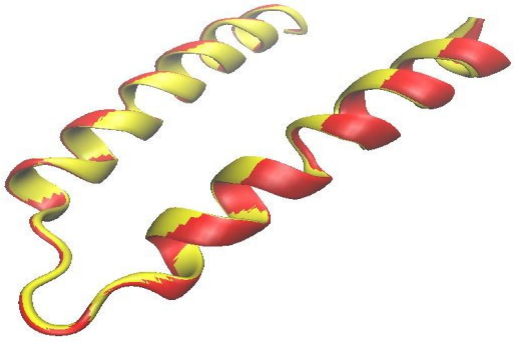

Figure 10: Residues 18 – 66 created by MolMol and SCOPE with an RMSD of 0.290.

Our initial non-bonded energy calculation comparisons between the CHARMM and SCOPE programs revealed large differences between the two. After further investigation it was realized that the coordinates of the atoms in CHARMM and the coordinates of the atoms in SCOPE contained different levels of accuracy. Though the difference in accuracies was only in a few decimal places, the energy spike was magnified since atoms at close distances cause the Van der Waals term to become basically exponential(1). Once the accuracy was fixed to the same number of decimal places, however, the non-bonded energies became extremely correlated.

In fact for all proteins, both non-bonded energies were very strongly correlated between CHARMM and SCOPE with R^2 values ranging from 0.99 to 1.0 and 0.96 to 0.99 for Van der Waals and electrostatic energies respectively (data not shown). Because correlations in non-bonded energies between the CHARMM and SCOPE programs were highly similar, these findings were demonstrated using only proteins 3LAY, 1G10, and 1J4V as examples (Figs. 3 - 8). .

Not all of the protein's non-bonded energy terms have a perfect linear correlation. The reason for the discrepancy is that SCOPE does not use an N-terminus residue or a C-terminus residue of the protein while CHARMM, creates the protein with both terminal residues. As a result, the energies from the HT2, HT3, OT1, and OT2 atoms are ignored but the HT1 atom is calculated in CHARMM leaving only the H atom on the first residue to be calculated in SCOPE. Some proteins have the same coordinates for the H atom and the HT1 atom resulting in a higher correlation while the proteins that differed in the coordinates resulted in the lower correlations.

Future work on the SCOPE program will start with adding on to the forcefield. The next term to be added will be a hydrogen-bond term that can be used to help with refinement of protein structures. Also, the addition of a Levenberg-Marquardt minimization algorithm will facilitate refinement of protein structures.

SCOPE is a simple open source program that uses only structure and angle files to reconstruct proteins and output an energy analysis of the newly created structure. Because the program is written in C++, users are given the flexibility to make modifications, such as adding extra energy terms, that are relevant to the task at hand. SCOPE's utility can also be expanded by using it in combination with other protein folding programs, such as REDCRAFT, in order to determine energetically favorable structures.

5 References

[1] Lodish, H., Berk, A., Zipursky, S. L., Matsudaira, P., Baltimore, D. & Darnell, J.. Molecular Cell Biology, 4th edition. . W.H. Freeman, 2000.

[2] Dobson, C. M.. Protein misfolding, evolution and disease. *Trends in Biochemical Sciences* (1999) **24**: 329-332.

[3] Kim J & Holtzman DM. Prion-Like behavior of amyloid-b. *Science* (2010) **330**: pp. 918-919.

[4] Wu B, Chien EYT, Mol CD, Fenalti G, Liu W, Katritch V, Abagyan R, Brooun A, Wells P, Bi FC, Hamel DJ, Kuhn P, Handel TM, Cherezov V & Stevens RC. Structures of the CXCR4 Chemokine GPCR with Small-Molecule and Cyclic Peptide Antagonists. *Science* (2010) **330**: pp. 1066-1071.

[5] Powers ET & Balch WE. Protection from the outside. *Nature* (2011) **471**: pp. 42-43.

[6] Dill, K. A., Ozkan, S. B., Shell, M. S. & Weikl, T.R.. The protein folding problem. *Annu Rev Biophys* (2008) **37**: 289-316.

[7] Radford, S. E.. Protein folding: progress made and promises ahead.. *Trends in Biochemical Sciences* (2000) **25**: 611-618.

[8] Andreeva A & Murzin AG. Structural classification of proteins and structural genomics: new insights into protein folding and evolution. *Acta Crysallogr Sect F Struct Biol Cryst Commun* (2010) **66**: pp. 1190-1197.

[9] Cellmer T, Buscaglia M, Henry ER, Hofrichter J & Eaton WA. Making connections between ultrafast protein folding kinetics and molecular dynamics simulations. *PNAS* (2011) **108**: pp. 6103-6108.

[10] Dobson, C. M.. Protein folding and misfolding. *Nature* (2003) **426**: 884-890.

[11] Anfinsen CB. Principles that govern the folding of protein chains. *Science* (1973) **181**: pp. 223-230.

[12] Bryson M, Tian F, Prestegard JH & Valafar H. REDCRAFT: a tool for simultaneous characterization of protein backbone structure and motion from RDC data. *J Magn Reson* (2008) **191**: pp. 322-334.

[13] Guntert P, Mumenthaler C & Wuthrich K. Torsion angle dynamics for NMR structure calculation with the new program DYANA.. *J Mol Biol* (1997) **273**: pp. 283-298.

[14] Brooks, B. R., Brooks III, C. L., Mackerell Jr, A. D., Nilsson, L., Petrella, R. J., Roux, B., Won, Y., Archonits, G., Bartels, C., Boresch, S., Caflisch, A., Caves, L., Cui, Q., Dinner, A. R., Feig, M., Fischer, S., Gao, J., Hodoscek, M., Im, W., Kuczera, K., Lazaridis, T., Ma, J., Ovchinnikov, V., Paci, E., Pastor, R. W., Post, C. B., PU, J. Z., Schaefer, M.,

Tidor, B., Venable, R. M., Woodcock, H. L., Wu, X., Yang, W., York, D. M. & Karplus, M.. CHARMM: The Biomolecular Simulation Program. *Journal Computational Chemistry* (2009) **30**: 1545-1614.

[15] Schwieters CD, Kuszewski JJ, Tjandra N & Clore G. The Xplor-NIH NMR molecular structure determination package. *J Magn Reson* (2003) **160**: pp. 65-73.

[16] Koradi R, Billeter M & Wuthrich K. MOLMOL: A program for display and analysis of macromolecular structures. *J Mol Graphics* (1996) **14**: pp. 51-55.

[17] Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H, Shindyalov IN & Bourne PE. The Protein Data Bank. *Nucleic Acids Res* (2000) **28**: pp. 235-242.