Study of Protein Folding Kinetics using a Coarse-grained Model

Chester P Chu*

MIT Department of Physics

(Dated: May 17, 2007)

In this paper, the kinetics of protein folding is analyzed using the coarse-grained protein folding program OLIGOMER. It is found that in light of a Go-like model, the protein folds more effectively with a deeper native contact potential, as well as lower solvent viscosity. The temperature effect on protein folding kinetics is also explored, and the result resembles that of known results from previous research [10].

I. Introduction

Proteins are the most abundant biomolecules in nature. It exists in various different forms and sizes, and they all played an essential role in numerous biological functions, from catalyzing enzymatic reactions and aiding signal transduction pathways, to acting as molecular motors and immuno-molecules[1]. The shape and structure of the protein needs to be very specific pertaining to their specific functions in the organisms. But how freshly-made proteins fold into these specific structures remains a large interest within the biophysical and biochemical community for decades[2,3].

Various models for protein folding have been proposed. For instance, some insist on the involvement of external protein (chaperones) in the process of protein folding[4], while others believe that protein folding can be understood solely on the basis of statistics and physical forces[,5]. In this paper, the later view is adopted, and the mechanism (simulation) of protein folding is based solely on the interaction of the amino acids with their environment and themselves.

The program used for simulation of protein folding is OLIGOMER, developed by ACT-JST project[6]. The extensive nature of the protein is reduced, in which the amino acids are simply beads on the polypeptide backbone, such that the computational time for the simulation could be lessened. Contrary to the approach in [], an off-lattice Go model[7,8] is used in the OLIGOMER simulation[9], with the benefit of being able to analyze the folding of real-life protein.

In this paper, we will utilize the OLIGOMER program (and its PAV interface[]) to analyze the kinetics of protein folding.

* Electronic address: cpchu@mit.edu

II. Molecular Dynamics and Energetics[+]

The total potential energy of the protein (with N amino acids) is composed of four constituents:

- Bond angle potential: the steric strain manifested in the magnitude of the bond angle:

$$E_{angle} = \sum_{i=1}^{N-2} E_{angle}^{i} = \sum_{i=1}^{N-2} K_{angle}\left(\theta_i - \theta_i^0\right)$$

with the bond angle θ_i be the angle at the $(i+1)^{th}$ residue given by:

$$\cos\theta_i \equiv \frac{\left(\vec{\mathbf{r}}_{\mathbf{i}} - \vec{\mathbf{r}}_{\mathbf{i+1}}\right)\cdot\left(\vec{\mathbf{r}}_{\mathbf{i+2}} - \vec{\mathbf{r}}_{\mathbf{i+1}}\right)}{\left|\vec{\mathbf{r}}_{\mathbf{i}} - \vec{\mathbf{r}}_{\mathbf{i+1}}\right|\left|\vec{\mathbf{r}}_{\mathbf{i+2}} - \vec{\mathbf{r}}_{\mathbf{i+1}}\right|}$$

- Dihedral angle potential: the steric strain manifested in the magnitude of the angle made by the adjacent planes of N=C double bond:

$$E_{dihedral} = \sum_{i=1}^{N-3} E_{dihedral}^{i} = \sum_{i=1}^{N-3}\left[K_{dihedral}^{1}\left[1+\cos\left(\phi_i - \phi_i^0\right)\right] + K_{dihedral}^{3}\left[1+\cos 3\left(\phi_i - \phi_i^0\right)\right]\right]$$

with the dihedral angle ϕ_i given by:

$$\cos\phi_i \equiv \frac{\left\{\left(\vec{r}_i - \vec{r}_{i+1}\right)\times\left(\vec{r}_{i+2} - \vec{r}_{i+1}\right)\right\}\cdot\left\{\left(\vec{r}_{i+1} - \vec{r}_{i+2}\right)\times\left(\vec{r}_{i+3} - \vec{r}_{i+2}\right)\right\}}{\left|\left(\vec{r}_i - \vec{r}_{i+1}\right)\times\left(\vec{r}_{i+2} - \vec{r}_{i+1}\right)\right|\left|\left(\vec{r}_{i+1} - \vec{r}_{i+2}\right)\times\left(\vec{r}_{i+3} - \vec{r}_{i+2}\right)\right|}$$

- Native contact potential: the interaction energy between i^{th} and j^{th} residues[♠] that are within a cutoff distance of 7 Å from the native structure:

$$E_{native} = \sum_{native} K_{Go}\frac{1}{n-m}\left\{m\left(\frac{R_{ij}^0}{R_{ij}}\right)^n - n\left(\frac{R_{ij}^0}{R_{ij}}\right)^m\right\}$$

where R_{ij} is the distance between the i^{th} and j^{th} residues in the current structure, and R^0_{ij} is the corresponding distance for the native structure. The exponents m and n could be adjusted to match the shape of physical repulsion and attraction, such as m=6 and n=12 for Lennard-Jones potential. This potential is the essence of the Go-model—the existence of a deep potential well to favor the native structure.

- Nonnative contact potential: the interaction energy between i^{th} and j^{th} residues that are beyond the native cutoff distance:

$$E_{nonnative} = \sum_{nonnative} \alpha K_{Go}\frac{1}{n-m}\left\{m\left(\frac{R_{nonnative}^0}{R_{ij}}\right)^n - n\left(\frac{R_{nonnative}^0}{R_{ij}}\right)^m\right\}$$

where $R^0_{nonnative}$ is the distance at which $E_{nonnative}$ is at minimum, and α is the

---

[+] This section follows the OLIGOMER manual closely.

[♠] with the constraint that $j \geq i + 4$

relative weakness of $E_{nonnative}$ compared with E_{native} (α needs to be small in order to favor the protein folding into the native structure instead of nonnative structure).

The total potential gives rise to the potential force **F**(t) that the protein experiences during the molecular simulation. On top of the potential force, a velocity-dependent frictional force (- ζ d**r**/dt) is also inserted to account for the effect of solvent viscosity as the protein is submerged inside some solvent. Furthermore, a random force **Γ**(t) is also included to account for the collision effect of solvent molecules on the protein. The random force has a temperature-dependent Gaussian probability profile:

$$P(\Gamma) = \sqrt{\frac{\Delta t}{4\pi\zeta k_B T}} \exp\left(-\frac{\Delta t}{4\zeta k_B T}\Gamma\right)$$

Therefore, the molecular dynamics of the protein in the simulation is governed by the Langevin equation that includes the three forces mentioned above:

$$m\frac{d^2\vec{\mathbf{r}}}{dt^2} = \vec{\mathbf{F}}(t) - \zeta\frac{d\vec{\mathbf{r}}}{dt} + \vec{\mathbf{\Gamma}}(t)$$

III. Proteins, Scales & Parameters used

In the simulation, protein 4HHB is folded into protein 2HCO. The structures of the two proteins are shown in Fig. 1. In order for the transformation to occur, "linear" portion of 4HHB must be folded.

(a) **(b)**

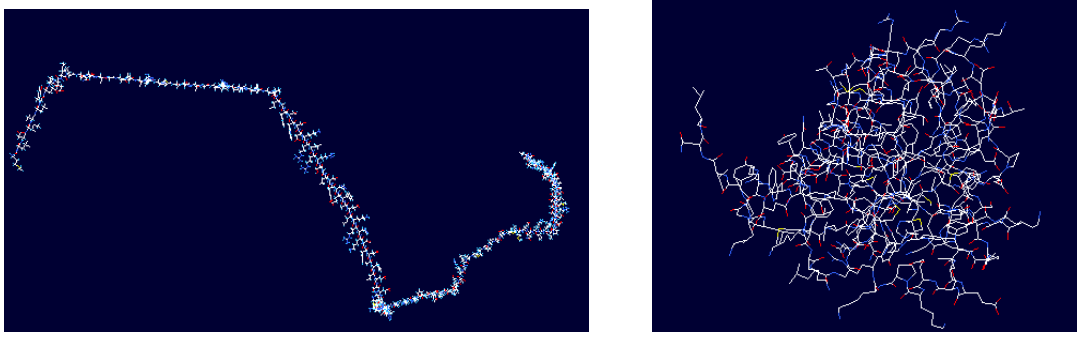

Fig 1. Structure of the two proteins involved in the simulation: (a) 4HHB, (b) 2HCO

Various physical quantities are rescaled in the program, and we shall continue using the rescaled values in the paper. In particular, 1 rescaled temperature unit is equivalent to 504 K, and 1 rescaled time unit is equivalent to 0.51 ps.

For most of the simulations, the contact potentials use the "default" setting of (m,n)=(10,12), instead of the Lennard-Jones value♥. Nevertheless, the default value

♥ This was done unintentionally. By the time I discovered the problem with this one particular parameter, it's already too late. However, I did try redoing the experiment with the Lennard-Jones (m,n)

(6Å) is still used for $R^0_{nonnative}$. For most of the simulation, we have used a small friction coefficient $\zeta = 0.005$, and a relative strength of nonnative-native contact potential $\alpha = 0.1$ to maintain a substantial advantage for the protein to stay in its native configuration. This has been tested as we run the simulation starting with the native conformation at 3 temperature units. Despite of the high temperature, the protein remain in its native structure for the entire time—giving a contact index (C, fraction of the protein being in the native position) of 1 during this simulation.

In order to get accurate assessments of the protein folding process, we should technically let each simulation run for a long time such that the protein is certain to reach a local/absolute minimum in the configuration space whose potential depth is much higher than the thermal fluctuations and work done by random forces. However, this would require a sufficiently long running time per simulation on my computer (on the order of several days), and therefore I have chosen a shorter simulation time of about 10^6 time units instead of 10^8 time units. A sample of one of the folding curves is shown in Fig 2.

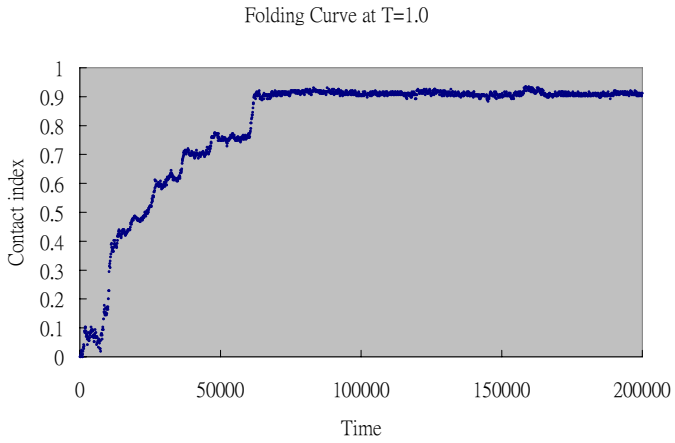

Fig 2. Contact index vs time at T=1.0.

IV. Kinetics & Temperature

The ultimate goal for our simulation is to determine how the kinetics of protein folding is dependent on temperature. The temperature dependence of protein folding kinetics has been explored with other simulation models, such as [10]. Generally, we expect the folding rate to be slower for very low temperature due to the temperature dependence of rate in Arrhenius equation. However, for high temperature, the folding rate is also low, because high thermal fluctuation could make the protein drift away from the native conformation on the reaction coordinate. Therefore, there exists a

---

with the limited time left on hand, but the results are strange, and I have also noticed there seems to be some problem with the randomization in the program when I run the Lennard-Jones (m,n).

minimum in the folding time as shown in [10]. And we hope to obtain similar results from our simulation.

To investigate the kinetics of folding, we need some means to measure the kinetics. One way to do so is to measure the time that protein takes to fold into a certain percentage of the native structure. This requires setting a cutoff value for the contact index. The folding time for different cutoff value of C is shown in Fig 3.

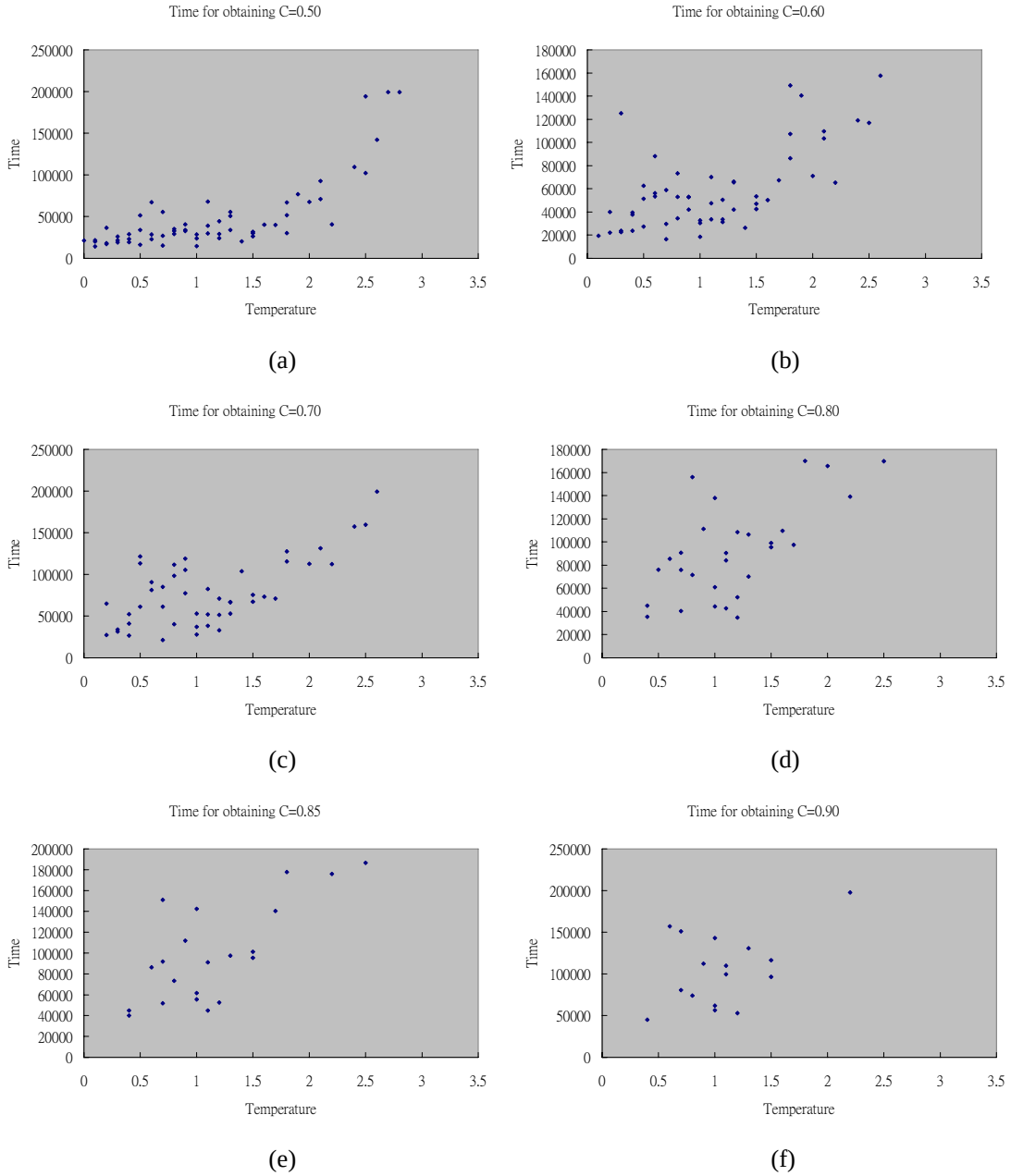

Fig 3. The folding time for obtaining a contact index of (a) 0.50, (b) 0.60, (c) 0.70, (d) 0.80, (e) 0.85, and (f) 0.90

Unfortunately, the folding time result above only show some slight increase of folding time at the higher temperature, but fails to show increase of folding time for lower temperature. However, the discrepancy is not completely surprising. First of all, the number of simulation per temperature is too low. Since the folding process could potentially be stuck indefinitely at some local energy minima, and thereby causing great variation in folding time. To avoid these type of data outliners from affecting our result, we need a great ensemble of folding simulation data; and 1-3 simulation per temperature may not be sufficient. Secondly, the determination of folding time is somewhat heuristic. In the course of simulation, the protein could explore in the configuration space and resulting in numerous crossing of the cutoff C value. In that case, we have to decide on one of the times to be the folding time[♦]. This could potentially be a source of dramatic error, especially in the case of high temperature where fluctuation is prominent.

To avoid the problem of folding time measurement, we could treat the problem differently by plotting the contact index at the end of simulation. This is shown in Fig 4. Although not perfect, the plot does resemble some aspect of the result in [10]: proteins are folding slower at lower and higher temperatures, and there is some middle region with optimal folding rate.

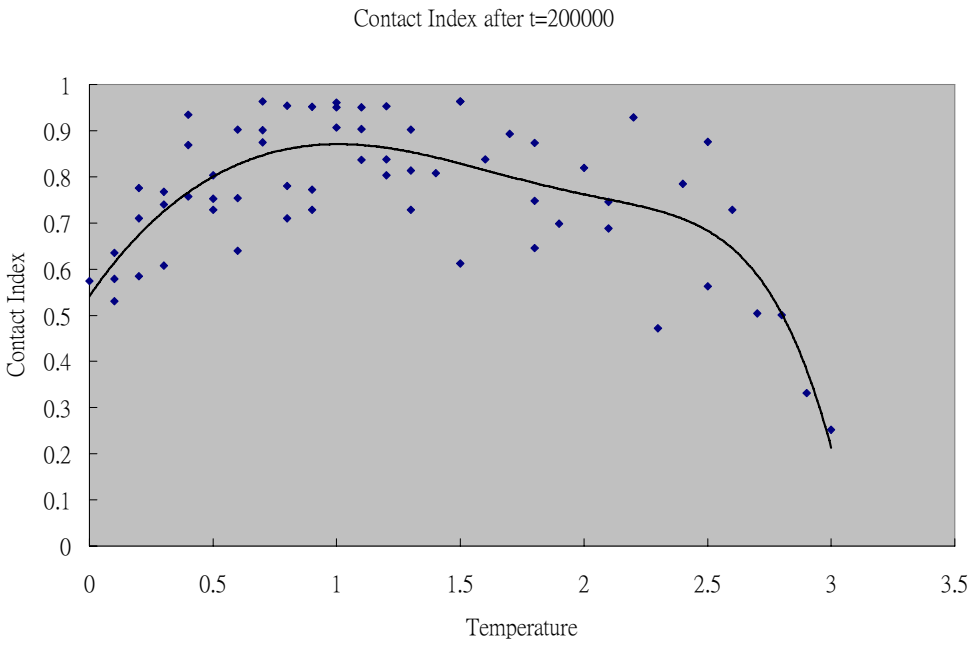

Fig 4. Contact index at the end of each simulation (t = 200000). The black line is a best-fit polynomial (order 6) that illustrates the overall shape of the distribution.

[♦]In this analysis, I have picked the last time the curve crosses the cutoff value. This could potentially cause bias against the higher temperature data, as their fluctuation would "delay" the recorded folding time.

V. Relative Strength of Contact Potential

As mentioned earlier, the essence of the Go model is to single out the native structure. This requires a huge difference between the contact potential for native structure and nonnative structure. This is controlled by the parameter α: if $\alpha<<1$, then the Go model greatly favors protein folding into the native structure. This is illustrated in a series of protein simulation running at the same temperature (as well as other parameters), but with different values for α. The result is shown in Fig 5.

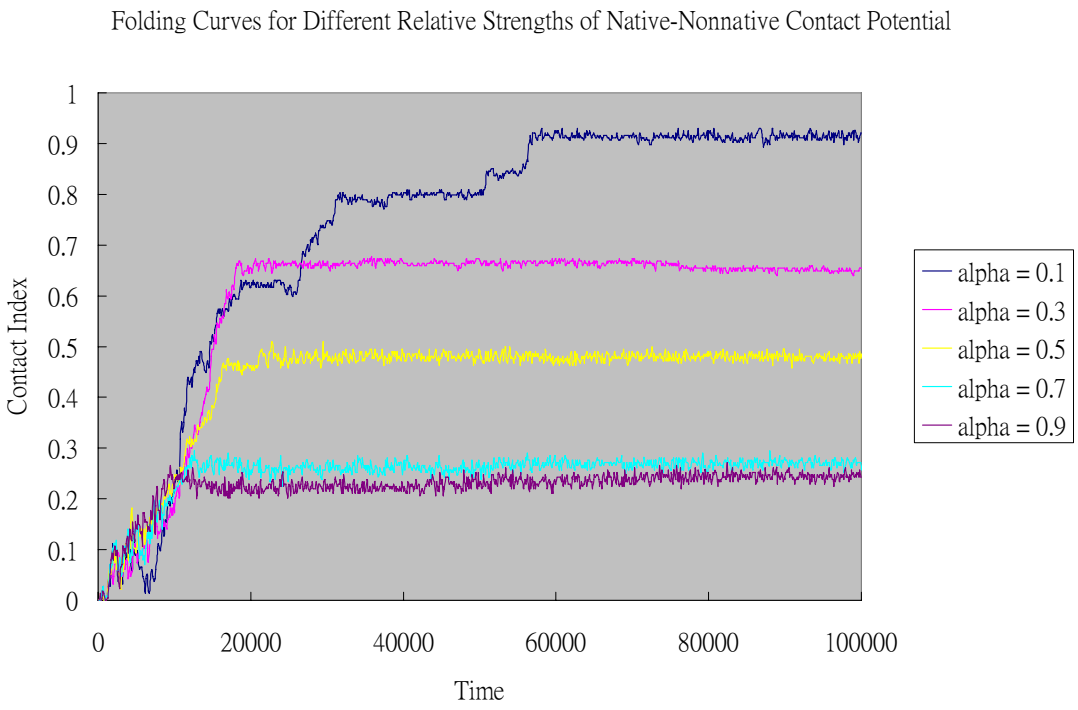

Fig 5. Contact index vs time for different relative strengths of native-nonnative contact potential, at T=1.0.

As expected, when the difference between native and nonnative contact potential is big, the protein tends to fold more successfully—this could be seen in the higher contact index for smaller α value. One interesting feature of the folding curves is that other than α=0.1 case, all the proteins have folded to the "steady state" at roughly the same rate. This shows that the lowering of native state energy has minimal effect on the kinetics of folding. This can be understood from consider Arrhenius equation: the rate of reaction is only dependent on the activation energy, and not the free energy difference between initial and final states. By lowering the energy of the native state, this only changes the free energy difference (and thus changing the equilibrium constant), but it does not really change the activation energy, and therefore has minimal effect on the kinetics of protein folding.

VI. Effects of Viscosity/Friction

Throughout most of the earlier simulations, a relatively small friction coefficient ζ is used. One can examine the effect of friction by varying the friction coefficient while holding other parameter fixed. The result is shown in Fig 6. This result is also anticipated—with higher friction coefficient, the protein will be harder to fold to native state. Thus, the curves with higher ζ fail to achieve higher contact index.

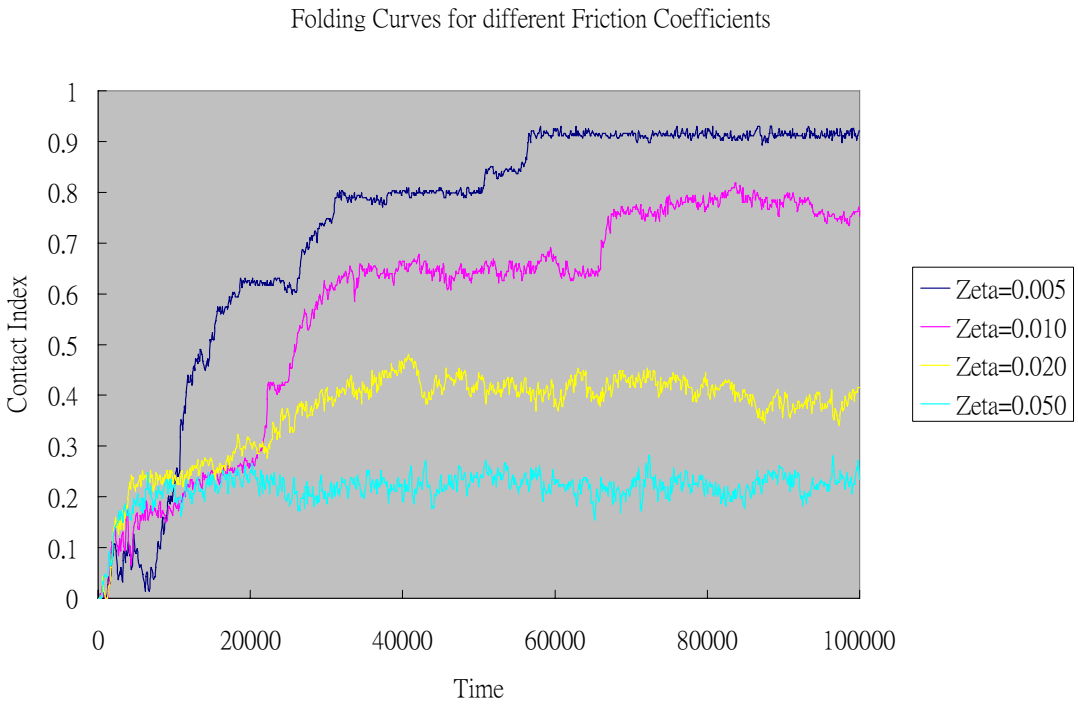

Fig 6. Contact index vs time for proteins in different solutions (different values for friction coefficient ζ), at T=1.0.

VII. Further Directions

In this work, we have utilized the coarse-grained protein simulation program OLIGOMER to study some kinetic aspects of protein folding. In particular, we have showed that the native structure in a Go model is quite robust toward thermal fluctuation, and the relative strength of Go model contact potentials does indeed give rise to effective folding into native structure. Furthermore, the retarding effect of viscosity on protein folding is also illustrated.

There are a lot more aspect of protein folding kinetics to be explored for this OLIGOMER program. For instance, we should go back to Lennard-Jones potential's exponents, or any other exponents, to see the relative effect of attraction and repulsion on the kinetics of folding.

The native-nonnative distinction was drawn somewhat arbitrarily, and it is interesting to see how changing the cutoff distance for native structure would change the dynamics. Alternatively, this can be also explored by altering $R^0_{nonnative}$, which can

be easily implemented with the help of the PAV interface.

However, before doing the above, one might want to get a more efficient and quantitative way of analyzing the kinetics of protein folding. Our inefficiency in determining an appropriate folding time has led to the somewhat chaotic result of folding time vs temperature as described previously. Even though we bypass the problem of folding time evaluation by looking at the contact index after long simulation time, and get a result resembling what is anticipated to be the temperature dependence of protein folding kinetics; the lack of correlation in the data is simply unsatisfactory and the methodology used in evaluating kinetics might need to be modified. As mentioned, the problem could be relieved with longer simulation time and more simulations. But are there alternative methods to tackle this problem?

Here's a slightly different method for accounting the kinetics of protein folding. As noticed from manually searching for the folding time, certain numerical sequences of contact index are repeating again and again across different simulation runs. This is probably the manifestation of the limited number of possible reaction route that a protein could take as it tries to move toward the native conformation in the configuration space. This recurrence of numerical sequence could be proven very useful in comparing the kinetics of protein folding. Suppose one knows a few numerical sequences of contact index that the protein must go through to reach the native structure. By using methods of sequence alignment, these numerical sequences could be identified for different simulation runs. Finally, to compare the folding time, one could focus on the time these different simulations take when going through these numerical sequences[♣]. Although this might not be very effective in terms of computation power, this could potentially be useful in situation when the number of simulation is limited.

VIII. Acknowledgment

The author would like to thank Professor Mirny and Peter Virnau for their guidance and assistance. Also, I would like to thank Grigory Kolesov for solving various computer/programming problems that come up in the course of this project.

---

[1] Nelson, D.L., Cox, M.M. *Lehninger Principles of Biochemistry*. *4th Ed.* 75-237.
[2] Ueda, Taketomi & Go, "Studies on protein folding, unfolding and fluctuations by computer simulation. I. The effects of specific amino acid sequence represented by specific inter-unit interactions", *Int. J. Pept. Prot. Res.*, **7,** 445-459 (1975)
[3] Pace C., Shirley B, McNutt M, Gajiwala K. "Forces contributing to the conformational stability of proteins", *FASEB J*, **10** (1): 75-83 (1996).

[♣] overlooking the possible of repetition of the same number in that numerical sequence as if they are gaps in sequence alignment problems.

[4] Lee S., Tsai F., “Molecular chaperones in protein quality control”. *J Biochem. Mol. Biol.* **38** (3): 259-85 (2005).
[5] Mirny, L. A., Abkevich, V. & Shakhnovich, E. I. “Universality and diversity of the protein folding scenarios: a comprehensive analysis with the aid of a lattice model”. *Fold Des* **1**, 103-116 (1996).
[6] The ACT-JST OLIGOMER and Protein Analyzer & Viewer (PAV) homepage: http://act.jst.go.jp/content/h13/life_sci/L02/PageMain.html
[7] Nymeyer, Garcia & Onuchuic, “Folding funnels and frustration in off-lattice minimalist protein landscapes”, *Proc. Natl. Acad. Sci. USA,* **95,** 5921-28 (1998)
[8] Koga, Takada, “Roles of native topology and chain-length scaling in protein folding: a simulation study with a Go-like model”, *J. Mol. Biol.,* **313**, 171-80 (2001)
[9] OLIGOMER manual.
[10] A. Gutin, A. Sali, V. Abkevich, M. Karplus, and E. I. Shakhnovich, **“**Temperature dependence of the folding rate in a simple protein model: Search for a ‘glass’ transition”, *J. Chem. Phys.* **108**, 646 (1998).