Stochastic Model of Protease-Ligand Reactions

Paul Anderson∗, Douglas Raiford, Deacon Sweeney, Travis Doom, and Michael Raymer
Wright State University
Dept. of Computer Science and Engineering
3640 Colonel Glenn Hwy
Dayton, OH 45435

Abstract

Prediction of protein tertiary structure based on amino acid sequence is one of the most challenging open questions in computational molecular biology. The two most common experimental methods for determining protein structure, X-ray crystallography and nuclear magnetic resonance (NMR) are both relatively expensive and time consuming processes. Further, some proteins (such as largely hydrophobic membrane integral proteins) are resistant to crystallization. Thus, computational approaches for determination of protein conformation are an attractive alternative to experimental procedures. While many such algorithms exist, none can yet achieve the same accuracy and reliability as experimental structure determination techniques. The objective of this research is to combine experimental evidence with computational modeling for high-confidence structure prediction. Limited proteolysis and chemical modification data is used to drive the modeling process towards physically realizable structures. Here we present a stochastic simulation of a limited proteolysis experiment that models the behavior of a protease-ligand reaction at an abstracted molecular level. The results of this simulation will be used to validate rate constant prediction methods that will then be used for the selection and refinement of candidate models for computational structure determination.

1 Introduction

Proteins are the molecular hardware that perform most of the metabolic work in both eukaryotic and prokaryotic cells. Understanding protein structure is an integral part of many real world applications, including drug design [4]. Since the structure of a protein is directly related to its biological function, the key to understanding the function of a protein starts with determining its structure [1].

Currently, there are two common experimental methods for determining the structure of a protein: X-ray crystallography and nuclear magnetic resonance (NMR) [3]. Using these two methods, over 30,000 protein structures have been determined. However, the total number of proteins is hypothesized to be at least 10 times this number [3]. A key difficulty with X-ray crystallography is that it can sometimes take several months or even years to resolve the structure of a single protein. Further, some proteins are resistant to crystallization and thus, not amenable to X-ray structure determination. An alternative to X-ray crystallography is NMR, which can also be used to determine the structure of a protein. The limited resolution of NMR structure determination limits its value for many important applications.

Computational techniques for determination of protein structure are attractive for many reasons. Effective structure prediction algorithms would result in significant reductions in the cost and human effort associated with structure determination. Computational methods might provide insight into the structure of membrane integral proteins and proteins that resist crystallographic methods. The relative speed of computational modeling versus experimental methods would allow much faster turnaround for applications such as drug target identification and lead development. For these and many other reasons, computational assignment of protein structure is an open and vibrant area of research. However, computational methods have so far been unable to provide the accuracy and resolution of experimental techniques. One approach for enhancing the accuracy of computational structure assignment is to incorporate experimental data from limited proteolysis and chemical modification experiments. These experiments are faster and less expensive than the crystallographic process, and can provide feedback to a structure assignment algorithm resulting in more reliable predictions of protein conformation.

The overall objective of our work is to develop a technique for the verification of computationally determined models of protein structure based on experimental data. The
method has two phases: the experimental phase and the computational phase. During the experimental phase, the protein is combined in solution with a chemical reagent, referred to as a probe. Each probe modifies the protein at specific locations, which are selected based on chemical, structural, and sequential features. Further, these features also determine the rate of the reaction. Then in the computational phase, experimental data is combined with computational data to determine the accuracy of the model in question. The simulation presented in this paper models the experiment using a protease as the probe.

In the experimental phase, the protease and protein are combined in solution. Then at specific time points, the concentrations of different protein fragments are measured by mass spectrometry [5], which separates the fragments based on their molecular weights. Finally, by plotting the concentration of a fragments over time, we can determine rate constants for the digestion reaction. These rate constants are functions of the features mentioned above and the concentration of protease in the experiment, but not of the amount of protein in the solution. The concentration of the protease is assumed to be constant during the reaction, and therefore, the concentration of the protease can be masked out of the reaction rates by dividing each reaction rate by the sum of all the reaction rates.

The rate of protein digestion is determined by the affinity of the protease to cut the protein at a specific location, which is a function of the chemical, structural, and sequential features. In particular, solvent exposure is expected to be a major determiner of reaction rate. The computational phase will verify a structural model by comparing its experimentally determined rates with its computational predicted rates. These two fingerprints will be compared to facilitate high-confidence structure prediction. A computational method is currently being developed to predict the reaction rates of a structural model.

A rough approximation of the computational phase can be visualized by examining two possible models for a sample protein, shown in figure 1. The first row in the figure shows one possible structure of the protein (shown with two different renderings). The circles indicate areas of high reactivity. Examining the model on the first row, we see that the highly reactive areas are located in a pocket on the surface of the protein. This is a favorable environment for the protease and protein to interact. However, the areas of high reactivity are buried on the model in the second row. These regions are not favorable for the protease, and therefore, we would say that the first model is more likely than the second model.

The program described in this paper simulates the experimental phase of the structural verification method. The inputs to the simulation are protein sequence data, cut locations within the sequence, and cut-site rate constants. After the simulation is done, we are left with a set of concentration graphs that are synonymous with the output from the experimental phase. We then perform nonlinear curve fitting to determine the original cut-rate constants that were included in the input of the program. This verifies that the simulation and equations used in the nonlinear curve-fitting are consistent. Then, assuming that our simulation correctly models the protease-protein reactions at a high level, our fragment concentration equations are correct.

2 Methods

2.1 Simulation

The simulation is the repeated procedure of finding and processing the next reaction. This procedure can be broken down as follows: first, find the next reaction stochastically; second, stochastically calculate the time of this reaction; third, execute the reaction. The next reaction is found by calculating the relative frequency of each reaction and then randomly selecting a reaction with a probability proportional to its relative frequency. For example, if a simple protein with three cut sites has the following reaction rates: 0.25, 0.5, and 0.9; then, the relative frequency of the first reaction rate is $0.25/(0.9+0.5+0.25)=1/7$. This indicates that the first cut site will be selected as the next reaction approximately 14% of the time. Furthermore, the time of the next reaction is found using the following equations [2]:

\[
\tau = \frac{1}{a_0} \ln\left(\frac{1}{\text{rand}(0, 1)}\right) \quad (1)
\]

\[
a_0 = \sum_{i=1}^{N} \text{avail}_i \times \text{rate}_i \quad (2)
\]

\[
N = \text{number of proteins in the simulation} \quad (3)
\]
Execution of the reaction modifies internal data structures. These data structures keep track of the number of proteins that have been cut at specific locations throughout the simulation. Furthermore, to model the behavior of the reaction, data structures are needed that can retain the state of each individual molecule. In our model, we instantiate an object for each protein. Each object contains a series of bit-vectors that represent the cut locations within the sequence. When a cut is made on a specific protein at a specific location, a "1" is placed in the appropriate bit-vector location. The main loop of our code predicts which reaction will occur next and the time at which that reaction will occur. The code then randomly selects a protein that has not been cut in the selected location and applies the cut to the object. At predetermined time intervals (i.e., mimicking experimental procedures), our code visits each of the proteins and calculates which fragments have been liberated. This data is stored in a fashion that is conducive to future graphic display.

2.2 Rate Equations

The process of protein digestion can be modelled by a set of differential (rate) equations. Our first step in deriving these equations is to visualize the process of cutting up a protein. The notation used to describe the state of a protein with \( N \) possible cut sites is

\[
P_{i_1 i_2 i_3 \ldots i_N} = \# \text{ of proteins in a given state}
\]

\[
\begin{cases}
  i_j = 0 & \text{not cut at } i_j \\
  i_j = 1 & \text{cut at } i_j
\end{cases}
\]

Using this notation, a protein with three cut-sites and rates \( k_1, k_2, \) and \( k_3 \) can take one of the following paths during digestion:

\[
\begin{align*}
P_{000} & \xrightarrow{k_1} P_{010} & \xrightarrow{k_2} P_{011} \\
 & \xrightarrow{k_2} P_{110} & \xrightarrow{k_2} P_{111} \\
 & \xrightarrow{k_2} P_{001} & \xrightarrow{k_1} P_{011} \\
 & \xrightarrow{k_2} P_{001} & \xrightarrow{k_1} P_{011} \\
\end{align*}
\]

After examining the possible paths, we derived the following set of differential equations:

\[
\frac{d[P_{000}]}{dt} = -k_1 \cdot [P_{000}] - k_2 \cdot [P_{000}] - k_3 \cdot [P_{000}] \\
\frac{d[P_{010}]}{dt} = -k_1 \cdot [P_{010}] - k_2 \cdot [P_{010}] - k_3 \cdot [P_{010}] \\
\frac{d[P_{100}]}{dt} = -k_1 \cdot [P_{100}] - k_2 \cdot [P_{100}] - k_3 \cdot [P_{100}] \\
\frac{d[P_{110}]}{dt} = -k_1 \cdot [P_{110}] - k_2 \cdot [P_{110}] - k_3 \cdot [P_{110}] \\
\frac{d[P_{011}]}{dt} = -k_1 \cdot [P_{011}] - k_2 \cdot [P_{011}] - k_3 \cdot [P_{011}] \\
\frac{d[P_{111}]}{dt} = -k_1 \cdot [P_{111}] - k_2 \cdot [P_{111}] - k_3 \cdot [P_{111}]
\]

Where \( [P_{i_1 i_2 i_3}] \) is the amount or concentration of a particular protein with a given configuration or state. So, \( \frac{d[P_{i_1 i_2 i_3}]}{dt} \) is the rate of change in the amount protein in a specific state. These differential equations can be solved and yield the following concentration equations:

\[
\begin{align*}
[P_{000}] &= N(e^{-k_1 t})(e^{-k_2 t})(e^{-k_3 t}) \\
[P_{010}] &= N(e^{-k_1 t})(e^{-k_2 t})(1 - e^{-k_3 t}) \\
[P_{100}] &= N(e^{-k_1 t})(1 - e^{-k_2 t})(e^{-k_3 t}) \\
[P_{110}] &= N(1 - e^{-k_1 t})(e^{-k_2 t})(e^{-k_3 t}) \\
[P_{011}] &= N(1 - e^{-k_1 t})(e^{-k_2 t})(1 - e^{-k_3 t}) \\
[P_{111}] &= N(1 - e^{-k_1 t})(1 - e^{-k_2 t})(1 - e^{-k_3 t})
\end{align*}
\]

Where \( N \) is the amount or concentration of the protein at the start of the reaction. Furthermore, we propose the concentration for a specific state with an arbitrary number of cut-sites follows a specific pattern:

\[
\psi(i_j) = \begin{cases} 
  (e^{-k_1}) & \text{not cut at } i_j \\
  (1 - e^{-k_2}) & \text{cut at } i_j
\end{cases}
\]

We have verified and derived this pattern for up to three cut-sites, but we believe it holds for a protein with any number of cut-sites.

3 Results

3.1 Two Cut Verification

We began our experiments using a small protein with two cut-sites. The simulation was run with 750 proteins for 60 seconds (chosen arbitrarily). The cut-rate constants for the two cut-sites, \( k_1 \) and \( k_2 \), were 0.9 and 0.12 s\(^{-1}\), respectively. This configuration can generate six different fragments (Figure 2): the original protein, the leftmost fragment, the rightmost fragment, the middle fragment (both
sites cut), the left-middle fragment (cut at the second site), and the right-middle fragment (cut at the first site).

Since the simulation is a stochastic process, multiple runs were employed to smooth the concentration graphs. Then, using the concentration graph of the middle fragment (Figure 3), we verified the rate constants of the two cut-sites within a 95% confidence interval. The predicted values of $k_1$ and $k_2$ were 0.9051, 0.1212, and 0.5038 s$^{-1}$, and had 95% confidence intervals of [0.880.93], [0.120.12], and [0.50, 0.51] s$^{-1}$, respectively.

![Figure 2. Possible fragments from two-cut protein](image)

**Figure 2. Possible fragments from two-cut protein**

![Figure 3. Double Cut, Middle Fragment Concentration vs. Time](image)

**Figure 3. Double Cut, Middle Fragment Concentration vs. Time**

### 3.2 Three Cut Verification

As further support for our model’s ability to generate fragments appropriately and our ability to derive the correct rate constants from said fragment plots, we extended our analysis to three cuts. Again, we ran the simulation for 60 seconds with 750 proteins. The cut-rate constants were 0.9, 0.12, and 0.50 s$^{-1}$ for $k_1$, $k_2$, and $k_3$, respectively.

After running the simulation, we curve-fit the fragment concentration $[P_{101}]$ graph to find the rate constants $k_1$, $k_2$, and $k_3$ (figure 4). From this we determined the values of $k_1$, $k_2$, and $k_3$ were 0.9051, 0.1212, and 0.5038 s$^{-1}$, and had 95% confidence intervals of [0.880.93], [0.120.12], and [0.50, 0.51] s$^{-1}$, respectively.

![Figure 4. Triple Cut, Internal Fragment $([P_{101}]$) Concentration vs. Time](image)

**Figure 4. Triple Cut, Internal Fragment $([P_{101}]$) Concentration vs. Time**

### 4 Discussion

#### 4.1 Validating Rate Equations

The primary goal of this project was to develop a simulation that can validate our proposed concentration equations. Incorrectly, the original concentration equations assumed that an internal fragment was liberated by a specific sequential reaction. That is, two consecutive cut-sites were always cut in the same order. We used our simulation to prove that these initial equations were incorrect, since the cuts could occur in any order. This led to the derivation of the new concentration equations that did not assume any order in the reaction. As was shown in the results sections, our simulation verified that these equations were correct given the assumptions stated in the introduction.

#### 4.2 Determining the Rate Constants

We also used the simulation to validate the experimental methods used to determine the rate constants. It was originally assumed that the rate constant of a specific cut-site could only be determined by examining all of the possible fragments liberated by a specific cut. For example, our two-cut simulation can create the fragments shown in Figure 2. Again, it was assumed that the middle fragment and the left-middle fragment were both needed to determine the
rate constant of cut-site 2. Specifically, it was thought that these two rates should be summed to find the actual value. We used our simulation to test this theory and found that if the fragment contained the cut-site in question, you could determine the rate constant from that fragment alone. This leads to the result that multiple fragments can be used individually to find the same rate constants. Considering the two-cut protein again, we can determine the rate constant of the second cut-site, \( k_2 \), by curve-fitting either the concentration graph of the left-middle, middle, right-middle, or rightmost fragments. Figure 5 shows the curve-fitting results for the left-middle fragments. Curve-fitting the middle fragment produces the value of 0.12 s\(^{-1}\) for \( k_2 \), and curve-fitting the left-middle fragment finds a value of 0.1136 s\(^{-1}\) for \( k_2 \). Both of these nonlinear curve-fits found the correct value of \( k_2 \) within their 95% confidence intervals.

![Left-middle Fragment Concentration vs. Time](image)

**Figure 5. Concentration Graph of \([P_{01}]\)**

### 4.3 Exploring the Biology behind the Reactions

During the simulation, one of our key assumptions is that proteins do not radically unfold during the experiment. We can support this assumption in the lab by strictly controlling the experimental parameters: concentration of protease and protein. While this assumption is valid, it is only an approximation. In reality, as the protein is digested, it begins to lose its structure (i.e. it starts to unfold). This creates a problem because the rate constants are functions of the protein’s shape. We will refer to this problem as over-digestion.

To solve this problem, we propose using a smaller time window of the data. In other words, instead of using 60 seconds of an experiment, we could use 30 seconds or some other fraction of the experiment. However, when we tried this with our simulation it became apparent that this degraded our ability to find the correct rate constants. Using only 30 seconds of the three-cut simulation, we approximately predicted the rate constants \( k_2 \) and \( k_3 \) from an internal fragment (i.e. \([P_{011}]\)). The value of \( k_3 \) was predicted as 0.5243 s\(^{-1}\), which is still within a reasonable margin of the original value of 0.50 s\(^{-1}\). However, since we are curve-fitting simulation data and not experimental data, we expect very accurate curve-fitting results. When using experimental data, this loss of accuracy could become very detrimental. This will be assessed once we have collected enough experimental data.

### 4.4 Future Work

The next step for this research will be to integrate it with experimental data that will be provided by Dr. Alter’s lab in the Biology Department (Biomedical Sciences Program) at Wright State University. Once this data is available, the simulation can be used to investigate the experimental and computational phases that were mentioned in the Introduction. One area of investigation will be to use the simulation to help develop a computational phase that can compare the predicted rates with the experimental rates. The stochastic simulation can be used to generate a time course for predicted rates, and then, that time course can be compared to the experimental time course. An example benefit of this approach would be that early time points are given a higher weight than later time points; thus, attempting to avoid some of the problems of over-digestion.

### 5 Acknowledgments

Thanks are extended to Brent Foy, Ph.D. for help setting up our stochastic simulations and for help deriving the rate equations, Michael Raymer, Ph.D., Gerald Alter, Ph.D., and Travis Doom, Ph.D. for interesting discussions and continuous support.

### References


