Preface

Computational systems biology is the term that we use to describe computational methods to identify, infer, model, and store relationships between the molecules, pathways, and cells ("systems") involved in a living organism. Based on this definition, the field of computational systems biology has been in existence for some time. However, the recent confluence of high-throughput methodology for biological data gathering, genome-scale sequencing, and computational processing power has driven a reinvention and expansion of this field. The expansions include not only modeling of small metabolic *(1–3*) and signaling systems *(2*, *4)* but also modeling of the relationships between biological components in very large systems, including whole cells and organisms *(5–15)*. Generally, these models provide a general overview of one or more aspects of these systems and leave the determination of details to experimentalists focused on smaller subsystems. The promise of such approaches is that they will elucidate patterns, relationships, and general features, which are not evident from examining specific components or subsystems. These predictions are either interesting in and of themselves (e.g., the identification of an evolutionary pattern) or interesting and valuable to researchers working on a particular problem (e.g., highlight a previously unknown functional pathway).

Two events have occurred to bring the field of computational systems biology to the forefront. One is the advent of high-throughput methods that have generated large amounts of information about particular systems in the form of genetic studies, gene and protein expression analyses and metabolomics. With such tools, research to consider systems as a whole are being conceived, planned, and implemented experimentally on an ever more frequent and wider scale. The other event is the growth of computational processing power and tools. Methods to analyze large data sets of this kind are often computationally demanding and, as is the case in other areas, the field has benefited from continuing improvements in computational hardware and methods.

The field of computational biology is very much like a telescope with two sequential lenses: one lens represents the biological data and the other represents a computational and/or mathematical model of the data. Both lenses must be properly coordinated to yield an image that reflects biological reality. This means that the design parameters for both lenses must be designed in concert to create a system that yields a model of the organism, which provides both predictive and mechanistic information. The chapters in this book describe the construction of subcomponents of such a system. Computational systems biology is a rapidly evolving field and no single group of investigators has yet developed a complete system that integrates both data generation and data analysis in such a way so as to allow full and accurate modeling of any single biological organism. However, the field is rapidly moving in that direction. The chapters in this book represent a snapshot of the current methods being developed and used in the area of computational systems biology. Each method or database described within represents one or more steps on the path to a complete description of a biological system. How these tools will evolve and ultimately be integrated is an area of intense research and interest. We hope that readers of this book will be motivated by the chapters within and become involved in this exciting area of research.

Organization of the Book

This volume is organized into five major parts: Network Components, Network Inference, Network Dynamics, Function and Evolutionary Systems Biology, and Computational Infrastructure for Systems Biology. Each section is described briefly below.

Part I – Network Components

This section focuses on methods to identify subcomponents of the complete networks. Ultimately, such subcomponents will need to be integrated with each other or used to inform other methods to arrive at a complete description of a biological system. This section begins with two methods for the prediction of transcription factor binding sites. In the first, Chapter 1, Mariño-Ramirez et al. describe a method for the prediction of transcription factor binding sites using a Gibbs sampling approach. In Chapter 2, Liu and Bader show how DNA-binding sites and specificity can be predicted using sophisticated structural analysis. Chapters 3–5 discuss methods to predict protein–protein interaction (PPI) networks, and Chapter 6 builds on predicted PPIs to identify potential regulatory interactions. Finally, Chapter 7 discusses the inherent modularity that is observed in biological networks with a focus on networks of PPIs.

Part II – Network Inference

This section focuses on methodologies to infer transcriptional networks on a genome-wide scale. In general, the methods described within focus on using either mRNA expression data or mRNA expression data coupled with expression quantitative trait locus (eQTL) data. To a large extent, method development in this area is driven primarily by the ubiquitous mRNA expression data that are available in the public domain or that are relatively easily generated within a single laboratory. These methods have been tremendously enabled by the development of array technology and hence predominately model mRNA levels (as that is the most ubiquitous data type). Chapters 8 and 9 present two methods for identifying and modeling transcriptional regulatory networks, while Chapter 10 focuses on inferring mRNA expression networks from eQTL data. Chapter 11 is a review of different methods for inferring and modeling large scale networks from expression and eQTL data.

Part III – Network Dynamics

Systems are not static entities. They change over time and in response to a variety of perturbations. Ultimately, computational systems biology will have to develop methods and corresponding data sets that allow one to infer and model the kinetics and dynamics of reactions between all the chemical moieties in a cell. The chapters in this section focus on such methods. Chapter 12 discusses methods to infer both static co-expression networks and a finite-state Markov chain model for mimicking the dynamic behavior of a transcriptional network. Chapter 13 focuses on quantitative models of system behavior based on differential equations using biochemical control theory, whereas Chapter 14 focuses on the use of stochastic kinetic simulations. Both approaches have applications where one is superior to the other. At this point in time, it is not clear which methods will turn out to be most useful in dynamically modeling the largest number of biological systems. In general, this is likely the case for most of the technologies described in this book, so it is useful for readers to familiarize themselves with several concepts. Specifically, both Chapters 13 and 14 provide an excellent discussion of a variety of historical approaches to the dynamical modeling of biological systems and the relative merits and downsides to each. Chapter 15 provides an excellent introduction to considerations for the interplay between experimental design and dynamic modeling using lambda phage as an example system. The methods and considerations described within are generally applicable to other biological systems and highlight the importance of integrating the direction of wet bench work and computational modeling to more rapidly refine the models.

Part IV – Function and Evolutionary Systems Biology

The ultimate representation of the function of a given biological moiety is a complete description of all the reactions in which it participates and the relative rates of said reactions. At present, we are quite distant from this goal for most biological molecules or systems. However, we are able to use computational methods to predict the most likely functions of a given protein and even predict which portions and specific sequences of the protein contribute most to that function. This section is focused on methods used to infer protein function and on the relationships between function and evolution.

Ultimately, the reason to study and research “systems” biology is to understand biological function at a given hierarchical level (be it a single catalytic site or entire pathways). The interplay between the detailed atomic study of function and the large-scale study of systems will enable us to achieve this goal. This section contains chapters that address the interdependence of these two aspects: individual algorithms or techniques to understand the functional role of atoms or residues in single molecules (e.g., proteins), which in turn are extrapolated to understand their greater role in terms of biological or organismal function. Conversely and complementarily, the role of larger systems and their influence on single molecules is also explored. Together, all these chapters illustrate the strong dependence between single molecules and entire pathways or systems.

Part V – Computational Infrastructure for Systems Biology

To represent and organize the large amounts of experimental data and software tools, database frameworks must be created and made available to the larger biological community. This chapter focuses on computational methods and databases as well as data representations necessary to both integrate and export systems biology information to an end user. The user may be the biologist searching for their gene of interest or they may be the bioinformatician looking for trends in protein function among higher eukaryotes. Several groups are working on this extremely difficult task of providing semantic meaning to the large amounts of underlying biological data collected from single and high-throughput experiments, as well as computational predictions. (As a parenthetical comment, this is a significantly much harder problem than one faced by Internet search engines such as a Google, which at this point do not provide any semantic meaning to a query.) We present only a few such examples in this section (and in this book). One primary focus is on the Bioverse framework, database, and web application, which was developed by the editors of this book. However, we also describe the Biozon as well as the SEBIN and CABIN frameworks. The abstract representations required to model biological systems are still in fruition, and a complement of many tools, technologies, databases, and algorithms will have to be integrated in the future as our knowledge expands.

Acknowledgments

It goes without saying that this volume would not have been possible without the efforts of countless biologists who provided the raw data that enable modeling of biological systems. We first and foremost thank all these researchers who provide the raw data for all the computational modeling that enables the field of "computational" systems biology. We also thank all the researchers who investigate these problems using sophisticated modeling techniques, particularly those described in this volume. We thank two particular editors of this volume, Reneé Ireton and Kristina Montgomery, who have dealt with the thankless job of undertaking the proofreading the chapters in this book and associated bureaucratic requirements. We finally thank the McDermott, Bumgarner, and Samudrala groups for their critical comments as this volume was being prepared, as well as our respective families for their patience. The administrative aspects of this work were in part supported by the NSF CAREER award to Dr. Ram Samudrala.

References

1. Ishii N, Robert, M, Nakayama, Y, Kanai, A and Tomita, M. Toward large-scale modeling of the microbial cell for computer simulation. J Biotechnol 2004;113(1-3):281–94.
2. Ekins S, Nikolsky, Y, Bugrim, A, Kirillov, E and Nikolskaya, T. Pathway mapping tools for analysis of high content data. Methods Mol Biol 2006;356:319–50.
3. Lafaye A, Junot, C, Pereira, Y, et al. Combined proteome and metabolite-profiling analyses reveal surprising insights into yeast sulfur metabolism. J Biol Chem 2005;280(26):24723–30.
4. Stevenson-Paulik J, Chiou, ST, Frederick, JP, et al. Inositol phosphate metabolomics: merging genetic perturbation with modernized radiolabeling methods. Methods 2006;39(2):112–21.
5. Ideker T, Thorsson, V, Ranish, JA, et al. Integrated genomic and proteomic analyses of a systematically perturbed metabolic network. Science 2001;292(5518):929–34.
6. Pe'er D, Regev, A, Elidan, G and Friedman, N. Inferring subnetworks from perturbed expression profiles. Bioinformatics 2001;17 Suppl 1:S215–24.

7. Pilpel Y, Sudarsanam, P and Church, GM. Identifying regulatory networks by combinatorial analysis of promoter elements. Nat Genet 2001;29(2):153–9.
8. Ideker T, Ozier, O, Schwikowski, B and Siegel, AF. Discovering regulatory and signalling circuits in molecular interaction networks. Bioinformatics 2002;18 Suppl 1:S233–40.
9. Kelley BP, Sharan, R, Karp, RM, et al. Conserved pathways within bacteria and yeast as revealed by global protein network alignment. Proc Natl Acad Sci U S A 2003;100(20):11394–9.
10. Shannon P, Markiel, A, Ozier, O, et al. Cytoscape: a software environment for integrated models of biomolecular interaction networks. Genome Res 2003;13(11):2498–504.
11. Ideker T. A systems approach to discovering signaling and regulatory pathways–or, how to digest large interaction networks into relevant pieces. Adv Exp Med Biol 2004;547:21–30.
12. Schadt EE, Monks, SA, Drake, TA, et al. Genetics of gene expression surveyed in maize, mouse and man. Nature 2003;422(6929):297–302.
13. Schadt EE and Lum, PY. Reverse engineering gene networks to identify key drivers of complex disease phenotypes. J Lipid Res 2006.
14. McDermott J and Samudrala R. Bioverse: Functional, structural and contextual annotation of proteins and proteomes. Nucleic Acids Res 2003;31(13):3736–7.
15. McDermott J, Bumgarner, R and Samudrala, R. Functional annotation from predicted protein interaction networks. Bioinformatics 2005;21(15):3217–26.

Chapter 2

Structure-Based *Ab Initio* Prediction of Transcription Factor–Binding Sites

L. Angela Liu and Joel S. Bader

Abstract

We present an all-atom molecular modeling method that can predict the binding specificity of a transcription factor based on its 3D structure, with no further information required. We use molecular dynamics and free energy calculations to compute the relative binding free energies for a transcription factor with multiple possible DNA sequences. These sequences are then used to construct a position weight matrix to represent the transcription factor–binding sites. Free energy differences are calculated by morphing one base pair into another using a multi-copy representation in which multiple base pairs are superimposed at a single DNA position. Water-mediated hydrogen bonds between transcription factor side chains and DNA bases are known to contribute to binding specificity for certain transcription factors. To account for this important effect, the simulation protocol includes an explicit molecular water solvent and counter-ions. For computational efficiency, we use a standard additive approximation for the contribution of each DNA base pair to the total binding free energy. The additive approximation is not strictly necessary, and more detailed computations could be used to investigate non-additive effects.

Key words: Transcription factor–binding sites, molecular dynamics, free energy, position weight matrix (PWM), multi-copy, thermodynamic integration, protein–DNA binding.

1. Introduction

Transcription factors are DNA-binding proteins that control gene expression *(1)*. They often recognize short DNA sequences (about six to eight base pairs long, roughly the number of base pairs exposed on the single face of a DNA major groove) that can be degenerate. Traditionally, binding sites have been obtained using

Jason McDermott et al. (eds.), *Computational Systems Biology, vol. 541*
© Humana Press, a part of Springer Science+Business Media, LLC 2009
DOI 10.1007/978-1-59745-243-4_2

experimental methods, including SELEX *(2)*, ChIP-chip *(3)*, protein-binding microarrays *(4)*, etc. These methods are often labor-intensive and expensive.

The binding sites of a transcription factor are intrinsically determined by the 3D structures of the protein and DNA and their structural complementarities. Binding sites of a transcription factor may also depend on its participation in a multi-protein complex. It is therefore desirable to predict these binding sites based on the 3D structures of transcription factors. This *ab initio* approach uses all-atom molecular simulation and remains a challenging problem. Several previous attempts *(5–7)* are limited to implicit solvent, enthalpic calculations of the free energy and frozen macromolecular backbones, all of which could lead to a bias in the binding site prediction.

In this chapter, we present an improved and, in principle, exact method (at least to the level of accuracy of molecular force fields) that can predict the transcription factor–binding sites using their structural information. There is no other required information, except for a well-chosen atomic force field for the representation of the protein–DNA complex.

The theoretical basis for structure-based binding site predictions for transcription factors is the binding free energy of the protein–DNA complex, calculated as the difference in free energy between the solvated complex and the solvated individual protein and DNA components. A transcription factor could possibly bind to multiple different DNA sequences with comparable binding affinity. This is because both DNA and protein are highly flexible molecules. Once a DNA base pair is changed to a different base pair and its prior favorable contacts with the protein are disrupted, protein and DNA can relax and change their geometries to achieve alternative favorable binding conformation. Typically, a specific DNA sequence and a non-specific DNA sequence to the same transcription factor differ only in binding energy on the magnitude of 10 kcal/mol. This is roughly equivalent to the energy of breaking two to five hydrogen bonds, as hydrogen bonds formed between oxygen and nitrogen atoms are typically 2–5 kcal/mol.

The relative binding free energy of a transcription factor with two different DNA sequences can be obtained using the following thermodynamic cycle:

$$\begin{array}{ccc}
\mathrm{DNA(aq)}+\mathrm{protein(aq)} & \rightarrow & \mathrm{protein}-\mathrm{DNA\ complex\,(aq)}\ \Delta G \\
\downarrow \Delta G_{\mathrm{DNA}} & & \downarrow \Delta G_{\mathrm{comp}} \\
\mathrm{DNA}'\mathrm{(aq)}+\mathrm{protein(aq)} & \rightarrow & \mathrm{protein}-\mathrm{DNA}'\,\mathrm{complex\,(aq)}\ \Delta G'
\end{array}$$

$$\Delta\Delta G = \Delta G' - \Delta G = \Delta G_{\mathrm{comp}} - \Delta G_{\mathrm{DNA}}, \qquad [1]$$

where $\Delta\Delta G$ is the relative binding free energy of the protein with DNA and DNA'. The two horizontal reactions represent the association of the protein with two different DNA sequences. These binding free energies can be obtained from experimental measurement. The two vertical reactions represent mutations of changing the DNA sequence in the DNA duplex (ΔG_{DNA}) and the protein–DNA complex (ΔG_{comp}). In computation, it is the two vertical or "mutational" reactions that are calculated. There are two common methods for the calculation of such "mutational" free energies: free energy perturbation and thermodynamic integration. From our experience, we found that the latter method, thermodynamic integration, is easy to implement and provides more opportunities for extension such as free energy decomposition analyses. In this chapter, we use this method exclusively.

A prevalent representation of the transcription factor–binding site is a position weight matrix (PWM), which can be converted into a sequence logo for graphical representation. In order for the PWM representation to be valid, each base pair must contribute independently or additively to the total binding free energy, commonly called the "additive approximation". In transcription factor – DNA complexes that have relatively small deformations in the DNA structure, this assumption has been observed to be a fairly good simplification *(6)*. In this chapter, we will also use this additive approximation and point out ways to assess the non-additivity in **Note 1**.

At each base pair position along the DNA, there are four possible Watson-Crick base pairs. **Equation [1]** can be used to calculate the relative binding free energies among these four possible base pairs, which will result in a four-level energy diagram. The base pair with the lowest energy leads to the strongest binding, and is normally the base that appears in the experimental consensus binding sequence. These relative energies can be converted into probabilities using the Boltzmann factor, as in

$$\Pr(\mathrm{bp} = \alpha, \alpha \in \{\mathrm{A, C, G, T}\}) = \frac{\exp[-\beta(E_\alpha - E_0)]}{\sum\limits_{\gamma \in \{\mathrm{A,C,G,T}\}} \exp[-\beta(E_\gamma - E_0)]}, \quad [2]$$

where the four possible base pairs are labeled as A, C, G, or T; α and γ represent possible base pair identities; β is the inverse temperature (i.e., $1/k_{\mathrm{B}}T$, k_{B} is the Boltzmann constant and T is the temperature); E_α and E_0 represent the free energy of the base pair α and the free energy of a reference base pair. Then $(E_\alpha - E_0)$ corresponds to the $\Delta\Delta G$ of **Eq. [1]** for changing the reference base pair into base pair α. For convenience, we choose the base pair leading to the lowest free energy (thus the strongest binder) as the reference point.

These probabilities can then be converted into a sequence logo *(8)* using the following formula,

$$IC(l) = 2 + \sum_{\alpha \in \{\mathrm{A,C,G,T}\}} \Pr(\alpha, l) \log_2 \Pr(\alpha, l), \qquad [3]$$

where $IC(l)$ represents the information content (in bits) at base pair position l; $\Pr(\alpha,l)$ represents the probability (from **Eq. [2]**) of base pair α at position l. In the sequence logo, the letters A, C, G, and T (representing the corresponding base pair) are stacked on top of each other in the order of descending probability at each base pair position. The relative height of each base pair at a position is proportional to their corresponding probabilities. The maximum height of information content at each position is 2 bits, representing 100% conservation at the position; the minimum height is 0, representing equal probabilities for all four possible base pairs.

Taking the vertical reaction ΔG_{comp} as an example, the free energy simulation and analysis can be done as follows. We will use a single base pair change as an example, using the above-mentioned additive approximation. First, a protein–DNA complex structure is made. Then a base pair at a specific position is changed to another possible base pair. These two structures represent the reactant and the product of the reaction. Our job is to calculate the free energy change associated with the reaction. Because free energy is a state function, we can connect the reactant and the product using an arbitrary reaction path, and integrate the energy gradient along the path to obtain the total free energy change. This approach is called thermodynamic integration. The formal derivation of this method can be found in Leach's introductory modeling book *(9)*, as well as most of the theoretical background for this chapter. More advanced treatments are also available *(10)*. Here we list the equations that are pertinent to the discussion. The energy function of the system is

$$H = H_0 + (1 - \lambda) H_{\mathrm{reac}} + \lambda H_{\mathrm{prod}}, \qquad [4]$$

where H is the total Hamiltonian of the system that contains all the energetic terms; H_0 is the energy terms for the environmental atoms, comprising all those other than the reactant and product; H_{reac} and H_{prod} represent the energy terms associated with atoms in the reactant and the product, respectively; and λ represents the reaction coordinate (aka coupling parameter). Here the reactant refers to the original base pair; the product refers to the final base pair; and the environment refers to the atoms of the DNA backbone, other DNA base pairs, protein, and the solvent. From this equation we can see that the Hamiltonian becomes that of the reactant system when λ is 0, and becomes that of the product system when λ is 1. At intermediate λ values, the Hamiltonian corresponds to an artificial system that contains both the reactant atoms and the product atoms. The reactant and the product, however, do not have any interaction terms, allowing them to occupy the same space.

Based on the linear coupling scheme of **Eq. [4]**, the free energy change for changing the base pair is

$$\Delta G = \int_0^1 \mathrm{d}\lambda \left\langle \frac{\partial H}{\partial \lambda} \right\rangle_\lambda = \int_0^1 \mathrm{d}\lambda \left\langle H_{\mathrm{prod}} - H_{\mathrm{reac}} \right\rangle_\lambda, \qquad [5]$$

where the angular brackets "$\langle\,\rangle_\lambda$" represent an ensemble average at a particular value of λ. In practice, the free energy simulation is done using traditional molecular dynamics methods, except that the energy function is now evaluated using **Eq. [4]**. After every 1 ps or so, the simulation trajectory will be saved. When the simulation is done, the saved trajectory will be analyzed using **Eq. [5]** to obtain the ensemble average of the Hamiltonian gradient. Typically, a numerical integration scheme is used to compute the free energy change for the reaction, such as the trapezoidal rule.

2. Materials

We list here the required computational resources for carrying out the computations discussed in the next section. The computational cost is listed in **Note 2**.

The majority of the calculations are done using a molecular modeling package called CHARMM (http://www.charmm.org) that requires a license. The version for wide distribution as of Jan. 2009 is c35b1. We have carried out all calculations using version c32b1. CHARMM requires FORTRAN90 compiler. On a Linux computer with Intel processors, the GNU FORTRAN compiler suffices. On Apple PowerPC computers with IBM processors, the IBM FORTRAN compiler is required. The benchmarks for these two architectures lead to similar running time for identical molecular test systems in serial mode, where the Intel processor is 3.0 GHz and the IBM processor is 2.2 GHz. CHARMM requires a moderate amount of memory at about 250 MB on the above two architectures for a system with about 25,000 atoms. The CHARMM executable is also available at public supercomputer sites, such as BigBen at the Pittsburgh Supercomputing Center (PSC), which has a parallel version of CHARMM installed (proof of license is required for usage). Benchmarks for additional systems are available from CHARMM's website, which lists a wide range of supported architectures and compilers.

We chose CHARMM because we found it the easiest for implementing the calculations we desired (*see* **Note 3**). The CHARMM27 atomic force field has been well tested to be accurate for the description of proteins and nucleic acids. In tests on BigBen at PSC, we have found a drop-off in performance for running CHARMM on more than 16 compute nodes (32 CPUs) in parallel. This drop-off may be system-specific or even due to inexperience on our part. We did not investigate this issue because parallelization of the code is not particularly important for our calculations. Calculations may be trivially parallelized by simulating each nucleotide on a separate node (*see* **Note 4**).

3. Methods

3.1. Simulation Protocols for Native DNA Duplex or Native Protein–DNA Complex

The starting point of the simulation is the 3D structure of the protein–DNA complex of interest. The structure can be obtained from X-ray crystallography, NMR determination, or homology modeling. We outline the protocol in **Fig. 2.1** and explain the steps below. The same protocols are carried out for the protein–DNA complex as well as the DNA duplex in the complex. This is necessary according to the thermodynamic cycle in **Eq. [1]**.

3.1.1. Preparation of the Complete Structure File

CHARMM incorporates PDB *(11)* structural files to initiate the molecular modeling and simulation. The starting structure's PDB file must be edited to follow CHARMM's naming convention. This may be done manually. One can also write a computer program to do these modifications once they become familiar with the required changes for amino acids and nucleotides. If the starting structure is from crystallography, missing side chains will be added by CHARMM. If the structure file is obtained from NMR determination, the hydrogen atoms need to be removed first, and CHARMM's HBUILD module is used to add hydrogen atoms according to its own naming convention. Any water molecules that are resolved in the original structures are also removed.

A common practice for the assignment of charge state of titratable amino acid residues is to assign a + 1 charge for all basic residues including lysine and arginine, assign a – 1 charge to all acidic residues including glutamate and aspartate, and finally assign a +1 charge to histidine residues that are exposed at the protein surface. These assignments are appropriate for near-neutral pH values. If histidine is buried in the protein core, then more advanced studies are required to assign its proper protonation state. For the transcription factors we have simulated to date, all histidine residues are exposed at the surface.

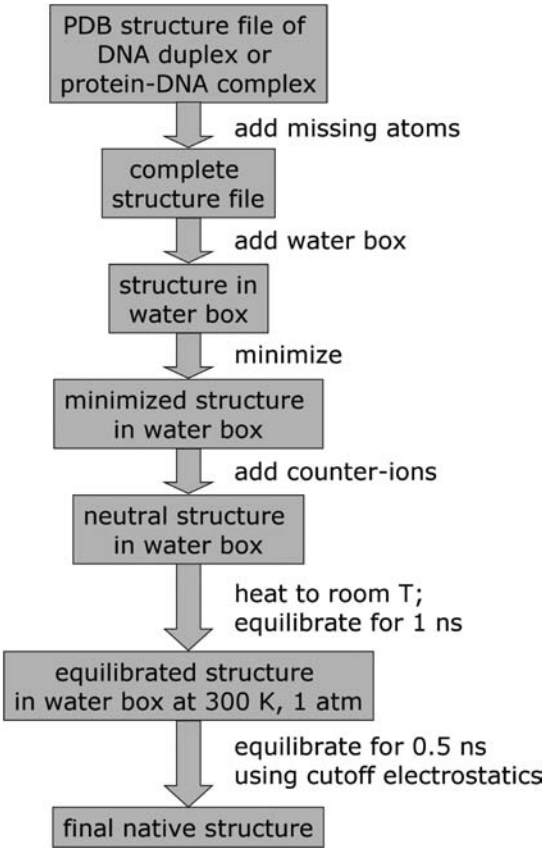
PDB structure file of DNA duplex or protein-DNA complex
add missing atoms
complete structure file
add water box
structure in water box
minimize
minimized structure in water box
add counter-ions
neutral structure in water box
heat to room T; equilibrate for 1 ns
equilibrated structure in water box at 300 K, 1 atm
equilibrate for 0.5 ns using cutoff electrostatics
final native structure

Fig. 2.1. Simulation protocol for generating a fully equilibrated native protein–DNA complex or DNA duplex structure in explicit solvent.

The ends of the protein contain a positively charged N-terminus and a negatively charged C-terminus. For the DNA section of the structure file, the 5′ end phosphate groups of both strands are removed. Other possible end-cappings are also supported in CHARMM.

Once the initial PDB file is edited to conform to CHARMM's convention and missing atoms are added, we will have a dry protein–DNA complex or DNA duplex with no solvent atoms.

3.1.2. Introduction of Explicit Water Molecules and Preparation of Minimized Structure in Water Box

Since we consider explicitly the role of water in the binding of protein and DNA, we now add water molecules to the dry complex structure (*see* **Note 5**) to form a solvated system in a periodic boundary condition. Because the water model TIP3P was used during the development of the current CHARMM force field CHARMM27, we recommend its usage over other water models.

Once the water molecules are added, a series of minimizations are required to allow the water molecules to relax around the macromolecules. The recommended minimization algorithms are Steepest Descent at the initial stage of the minimizations, and then Adopted Basis Newton-Raphson method for more refined minimizations. We use 1,000 steps of the former and 3,000 steps of the latter. The energy of the system should be decreasing steadily and reach stability. However, we do not advise running long minimizations to achieve convergence (to reach absolute zero K in temperature), as all our molecular dynamic simulation and free energy calculations need to be carried out at room temperature.

3.1.3. Introduction of Counter-Ions

After the system in the water box is minimized, we use the Charmm script file written by Rick Venable (available from Charmm Discussion Forum Script Archive at http://www.charmm.org/ubbthreads/ubbthreads.php?Cat=0) to replace an appropriate number of water molecules with counter-ions. For the protein–DNA complexes we have studied, typically about ten sodium ions are required to neutralize the system. For a 10-base pair DNA duplex, 18 sodium ions are required. In Venable's script, the same number of water molecules as the desired counter-ions are selected at random and replaced by sodium ions. Then the system is minimized for 50 steps by Steepest Descent and by Adopted Basis Newton-Raphson. One hundred different sets of water selections are done. The lowest energy configuration among them is chosen to proceed to the next step.

3.1.4. Heating and Equilibration of the Structure

Since minimization freezes many degrees of freedom of the system, the solvent box is roughly about 50 K in temperature. Now we heat the system to room temperature and equilibrate it for 1.5 ns. We ramp up the temperature linearly from 50 to 300 K over 50 ps at a heating speed of 5 K per ps. During equilibration, constant temperature (300 K) and constant pressure (1 atm) are maintained using Charmm's CPT keyword. This corresponds to the NPT ensemble. A time step of 1 fs is used. SHAKE is used to constrain all the bonds with hydrogen atoms to be at the equilibrium values. All other degrees of freedom are allowed.

The Block module for free energy analysis has a limitation in that it requires the electrostatic interactions to be evaluated using non-Ewald methods, i.e., spherical cutoffs. Long, computationally expensive cutoffs are required to obtain an adequate representation of long-range electrostatic interactions. To reach a compromise between accuracy and computational saving, we carry out initial equilibration of the system for 1 ns using Ewald summation method Particle Mesh Ewald. Then we switch to spherical cutoff scheme using a cutoff value of 14 Å. Further equilibration of 0.5 ns is run at this condition.

After the 1.5 ns equilibration, the native protein–DNA complex structure is now considered well equilibrated. We need to note here that this equilibration time is still far too short for the equilibration of the counter-ions, which typically requires much longer equilibrations on the scale of tens to hundreds of ns. Please *see* **Note 6** for strategies that avoid long equilibrations for ions.

3.2. Simulation Protocols for Free Energy Calculations

3.2.1. Multi-Copy Base Pairs

CHARMM supports dual-topology, which means that in the "mutational" reactions, the reactant and the product chemical groups co-exist in the structure. This is also known as "multi-copy" representation, where multiple functional groups occupy possibly the same space; their interactions with the rest of the system are scaled by a coupling parameter, but there are no interactions among the multiple copies. As we have discussed in the Introduction, thermodynamic integration is an established method for calculating the free energy change associated with changing one functional group in the multi-copy into another. In the simulations that are discussed in this chapter, we consider only the co-existence of two possible base pairs at any base pair position. **Figure 2.2** illustrates the construction of such structures. We call these 2-base multi-copy base pairs, or in short multi-copy base pairs. Details on how to create structures with multi-copy bases and how to enable CHARMM to evaluate their force and energy functions are in **Notes 7** and **8**.

3.2.2. Using BLOCK for Simulation and Free Energy Analysis

The BLOCK module in CHARMM allows straightforward force and energy evaluation of multi-copies. Here we use a simple example to illustrate its usage. Imagine a protein–DNA complex in which one base pair is a multi-copy base. Using **Eq. [4]**, the total Hamiltonian that contains the contributions from the environment, the reactant, and the product will be further separated into six contributions, as in **Eq. [6]** in BLOCK.

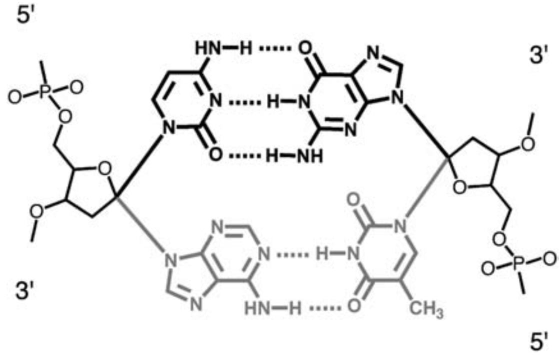

Fig. 2.2. Schematic diagram of multi-copy base pair. Single base pair change from A (*gray, bottom base pair*) to C (*black, top base pair*) is used as an example. The multi-copy base pair is referred to as A/C. The left strand is treated as the leading strand. The bases within each physical base pair interact normally, as evidenced by the hydrogen bonds (*dotted lines*) between complementary bases. The gray (reactant) atoms do not interact with the black (product) atoms.

$$\begin{array}{cccc} & \text{Env} & \text{Reac} & \text{Prod} \\ \text{Env} & 1 & 1-\lambda & \lambda \\ \text{Reac} & & 1-\lambda & 0 \\ \text{Prod} & & & \lambda \end{array} \qquad [6]$$

This matrix is used for force and energy evaluation. The labels "Env", "Reac", and "Prod" represent the environmental atoms, the atoms in the reactant, and the atoms in the product, respectively. The matrix is symmetric so the lower half is not shown. Each element in the matrix represents the interaction term between the atoms in the corresponding row and the atoms in the corresponding column. For example, the term H_0 in **Eq. [4]** is 1 in the matrix and represents the interactions between atoms in the environment. Note that the interactions that involve reactant atoms are scaled by the (1–λ) coupling parameter, whereas those involving product atoms are scaled by λ, just like in **Eq. [4]**. Finally, there are no interactions between the atoms of the reactant and the atoms of the product, hence the zero in the matrix.

BLOCK uses a different matrix to calculate the Hamiltonian gradient in **Eq. [5]** for the free energy analysis. The matrix is listed in **Eq. [7]**.

$$\begin{array}{cccc} & \text{Env} & \text{Reac} & \text{Prod} \\ \text{Env} & 0 & -1 & 1 \\ \text{Reac} & & -1 & 0 \\ \text{Prod} & & & 1 \end{array} \qquad [7]$$

Note that at all values of λ, the analysis matrix is of the same form.

Because of the flexibility of BLOCK, multiple multi-copy bases can be studied at the same time, and the dynamics and analysis matrices need to be adjusted correspondingly. The environmental atoms can also be further partitioned so that their contributions to the free energy can be calculated separately.

3.2.3. Simulation of Multi-copy Structures

For each multi-copy structure we create in **Section 3.2.1**, the following simulation protocol is used.

A short minimization is needed in order to resolve the potential bad contacts caused by the introduction of the multi-copy base pair. We use 100 steps of Steepest Descent and 100 steps of Adopted Basis Newton-Raphson for this purpose. Then the system is heated from 50 to 350 K over a linear ramp for 15 ps at a speed of 20 K per ps. Then the system is equilibrated at 350 K for 15 ps. After that, a linear ramp is used to cool the system down to 300 K at a speed of –10 K per ps for 5 ps. The system is then equilibrated at 300 K for 65 ps. This heat-cool-cycle is similar to the annealing process, except that we only heat the system up to 50 K above room temperature. The current force field is still expected to be reasonable in describing the system. The purpose of this heat-cool-cycle is to help the new multi-copy structure overcome the energy barriers that could trap the structure in the conformation favorable only to the native structure (*see* **Note 9**). Finally, a 100 ps production run is done (*see* **Note 10**), during which the system configuration is saved at every 0.5 ps. A time step of 1 fs is used. SHAKE is again applied to constrain bonds involving hydrogen atoms.

During the simulation, we use IMAGE to describe the cubic-shaped periodic boundary condition. The BLOCK matrix of **Eq. [6]** is used for force and energy evaluation. We assume that the density of the system is well-equilibrated over 1.5 ns simulation (**Section 3.1.4**). So the box size is fixed here using the final box size from the 1.5 ns equilibration, and the NVT ensemble is run. After the production run is finished, we examine all the saved configurations to calculate the free energy change using the BLOCK analysis matrix of **Eq. [7]**.

The saved configurations in the trajectory might be correlated among adjacent frames. To correct for this effect, we use **Eq. [8]** in estimating the sampling error.

$$E = \sigma\left[\left(\frac{1}{N}\right)\frac{1+c}{1-c}\right]^{1/2} = \left[\sum_{f=1}^{N}\frac{(x^2-\bar{x}^2)}{N(N-1)}\frac{1+c}{1-c}\right]^{1/2}, \qquad [8]$$

where E is the estimated error of the free energy change ΔG, x represents ΔG value at each frame, c is the correlation between adjacent frames, f is the frame number from 1 to N (total number of frames), and σ is the standard deviation of ΔG for all frames. Systematic and statistical errors that could exist in the simulation and free energy calculations are summarized in **Notes 9** and **10**.

3.2.4. Tournament Approach

According to **Eq. [1]**, two free energy calculations (one for the complex and one for the DNA duplex) are required to obtain the relative binding free energy for a single base pair change as in $\Delta\Delta G = \Delta G_{\mathrm{comp}} - \Delta G_{\mathrm{DNA}}$. At each base pair position, we evaluate the free energy changes for three multi-copy structures for both the DNA duplex and the protein–DNA complex. We carry out three $\Delta\Delta G$ calculations as a tournament, which contains three ΔG_{DNA} and three ΔG_{comp} calculations. Two competitions for multi-copy A/T and C/G are carried out first to obtain $\Delta\Delta G_{\mathrm{A/T}}$ and $\Delta\Delta G_{\mathrm{C/G}}$. The two winners then compete in the second round, e.g., $\Delta\Delta G_{\mathrm{A/C}}$ when A and C are the two winners. These three relative free energies are sufficient to describe the energy diagram of all four possible base pairs. These energies are then converted into probability and sequence logo representation using **Eqs. [2]** and **[3]**.

4. Notes

1. *Correlation between adjacent base pairs.* The "additive approximation" in Section 1 might not always be valid depending on the transcription factor. We can estimate the correlation between adjacent base pairs by the following test. It is analogous to comparing the energy change caused by two separate single mutations of the DNA and the energy change caused by a double mutation of the DNA. For example, one might be interested in the correlation between positions five and six. The user will first do two separate free energy evaluations for position five and position six. Only one base pair is changed at any time. Then, the user calculates the free energy change caused by changing positions five and six simultaneously. Taking multi-copy base pair A/C as an example, we use A5C and A6C to describe the base pair change at these two positions. The non-additivity can be estimated by $\Delta\Delta G_{\mathrm{A5C,A6C}} - (\Delta\Delta G_{\mathrm{A5C}} + \Delta\Delta G_{\mathrm{A6C}})$. These calculations can help quantify the non-additivity as well as the correlation between adjacent base pairs.
2. *Total computational cost and monetary equivalent.* The computational cost for obtaining the binding sites as a PWM for a transcription factor is about 400 CPU-days on a single Intel 3.0 GHz processor. The calculations in **Sections 3.1** and **3.2** are both included. We also list in **Table 2.1** the computational cost on the supercomputer BigBen at PSC. The total computational cost for the prediction of one transcription factor is about 1.2 CPU-years, or $1,200 if we assume one CPU-year is about $1,000.
3. *Force field and multi-copy implementations.* We compare four popular molecular modeling packages here, CHARMM, AMBER, NAMD, and GROMACS, and explain the reasons based on which we choose CHARMM in our simulations (**Section 2**).

   CHARMM was the first package to be developed and has the most capabilities and functions. CHARMM and AMBER are written in FORTRAN, and the GROMACS is written in C. NAMD is developed using similar philosophy of CHARMM, but is written in C++/C. All four packages can carry out traditional molecular dynamics simulations, and lead to similar results when the same force field is used.

   Many packages allow the user to choose a specific force field. The CHARMM27 force field is currently recommended for use in CHARMM. It can accurately characterize proteins and nucleic acids, and has overcome problems associated with the older versions. AMBER parm99 and parm03 are force fields

Table 2.1
Computational cost for the prediction of transcription factor–binding sites on supercomputer BigBen at Pittsburgh Supercomputing Center

	Counter	CPU hour	CPU days
Native structures			
Protein–DNA complex, 1.5 ns equilibration		1,200	50
DNA duplex, 1.5 ns equilibration		1,200	50
Multi-copy structures			
Number of free energy evaluations per ΔG	1	160	
Number of ΔG′s per ΔΔG	2		
Number of ΔΔG′s per base pair in tournament	3		
Number of base pair positions	8		
Number of free energy evaluations per protein	48	7,680	320
Total cost per protein		10,080	420

recommended for use in Amber. However, the A-DNA form tends to be over-stabilized in these force fields *(13, 14)*. Gromacs uses OPLS force field for all-atom simulations that leads to good results for proteins but is less characterized for nucleic acids. Namd allows the user to choose whether they want to use Charmm, Amber, or Gromacs force fields.

All these force fields use pairwise additive energy functions, typically including the bond length, bond angle, dihedral angle, van der Waals, and electrostatic interaction terms. Two library files are used for the implementation of the force field. The topology library file contains the list of these terms, whereas the parameter library file contains the force constants and other relevant constants.

The most important factor that leads us to choose Charmm is its “dual-topology” implementation. Amber and Gromacs support only “single-topology”, which means that if a “mutational” free energy perturbation is to be carried out, the two end points (reactant and product) must be similar in structure and number of atoms. In practice, they typically differ in only a small functional group *(15)*. This poses serious challenges for the perturbations of two groups of varying number of atoms. For instance, one might be interested in finding the free energy change associated with morphing an A = T base pair into a T = A base pair along a linear coupling path. For this mutation, the total numbers of atoms in the two end states are the same. However, because the atom types and parameters are very different for bases adenine and thymine, this morphing and free energy calculation was difficult for us to implement in Amber and, presumably, Gromacs. One possible solution is to introduce a common intermediate topology, calculate two free energy changes of the two end states morphing into the intermediate, and then calculate the sum of the two to obtain the total free energy change. As we have already mentioned in **Note 9**, free energy calculations have a large innate systematic error, and we decided against using two "single-topology" simulations to mimic a single "dual-topology" calculation.

In contrast, both Charmm and Namd support "dual-topology". However, Namd only supports free energy perturbation for "mutational" reactions, which is generally less accurate than thermodynamic integration. This is because in the free energy perturbation formula *(9, 10)*, the free energy change is obtained as the ensemble average of the exponential of the energy function of the system. If we assume these energy function evaluations are Gaussian-distributed, which is often true, then only one of the tails of the Gaussian curve will contribute to the final free energy change, since all the other energy values nearly contribute nothing to the ensemble average of the exponentials. However, given the same trajectory, if we use thermodynamic integration, then all these configurations will contribute to the final free energy change. A second factor is that, to our knowledge, Namd only permits single mutations. In Charmm, the Block module allows us to carry out simulations of multiple mutations at the same time, which could lead to significant computational saving.

4. *Parallelization*. Charmm, as pointed out in **Section 2**, does not scale very well in parallel. However, inefficient parallelization is at best a minor concern for our study, because the calculations we have described in **Section 3.2** are trivially parallelizable by running a free energy calculation at each base pair on a different node. For a binding site of length eight, 48 free energy evaluations are required to obtain the relative binding free energies at all eight base pair positions (*see* **Table 2.1** second column). The only exception to the trivial parallelization is the initial long equilibration (for 1.5 ns, **Section 3.1.4**) for generating configurations of the native protein–DNA complex and DNA duplex. If parallel runs are to be planned, we advise a short benchmark to be done first to establish the optimal number of processors for each system of different size. For the protein–DNA complexes we have studied (about 25,000 atoms in total), we

found that the optimal number of processors was eight on the supercomputer BigBen at PSC, and the 1.5 ns equilibrations typically take about 6 days.

5. *Addition of water box as solvent.* In **Section 3.1.2**, water box is added to the dry protein–DNA complex or DNA duplex. A CHARMM script file written by Lennart Nilsson can be used to add a small box of water with a maximal number of water molecules of 9,999. A modified script file written by Davit Hakobyan can be used to add larger boxes of water exceeding 10,000 water molecules. Both script files assume periodic boundary condition. These script files can be downloaded from the "Script Archive" on the "CHARMM Discussion Forum" (http://www.charmm.org/ubbthreads/ubbthreads.php?Cat=0). The TIP3P water model is used in these scripts.

   For periodic boundary conditions, there are a variety of available box shapes to choose when adding water molecules using the above-mentioned scripts. Since we rely on the BLOCK module, which in turn requires the IMAGE module of CHARMM, to carry out free energy simulation and analysis, we use the cubic box shape, which is supported by IMAGE. It is also possible to use other more spherical-like box shapes, such as truncated octahedron, but it requires the creation of the corresponding IMAGE file by the user.

6. *Other treatments of counter-ions.* Two simple strategies are listed here, which avoid running long equilibrations for the ions in the system (**Section 3.1.4**). First, the system can be studied without counter-ions as a non-neutral system. This means that **Section 3.1.3** can be bypassed. The Ewald summation and spherical cutoff methods for electrostatic interactions are still valid in non-neutral systems. However, for certain molecular systems, salt concentration is an important factor for structural stability. In this case, both positive and negative ions should be added in order to obtain the desired salt concentration. Second, one can use a simple uniform neutralizing background to achieve neutral system. This is typically achieved by setting the $k = 0$ term in the Ewald sum to zero (this term is automatically zero for a charge-neutral system). Simulations with a uniform neutralizing background may require modifications to be made to the standard CHARMM source code.

7. *Generation of structures with multi-copy bases.* There are two types of files that must be created for the study of multi-copy structures in **Section 3.2**: PDB and an extended topology library file. We explain the method for creating PDB files with multi-copy bases in this section. The extended topology file is explained in **Note 8**.

First of all, a library of all 2-base multi-copy PDB files is made. There are several ways of doing this. We use the standard base geometry in Ref. *(12)* to create PDB files for each base. These base geometries do not contain the backbone geometry or hydrogen atoms. One can then use CHARMM to read this PDB file and use "IC BUILD" and HBUILD routines to create a complete PDB file for each DNA nucleotide. Note that CHARMM's default nucleotides are for RNA, so patches need to be applied to convert them into DNA nucleotides. After the set of PDB files are prepared for the four DNA nucleotides, the atomic entries for the base atoms are concatenated to form the 2-base multi-copy PDB files. We use the following shorthand for multi-copy bases, e.g., A/C represents the multi-copy base of changing adenine to cytosine in the leading (1st) strand of the DNA (C/A is not needed as it is simply the reverse reaction of A/C). There are six files needed for describing all possible 2-base multi-copies that constitute the library: A/C, A/G, A/T, C/G, C/T, and G/T.

Second, a fully equilibrated native DNA duplex or protein–DNA complex structure is modified to create all possible multi-copy structures for each base pair position. For a 10-base pair DNA, there are 60 multi-copy structures. We developed a C++ program to replace the original base pair by one multi-copy base pair from the above-mentioned library. Three rotations are required to align the *N*-glycosidic bond, then align the base atoms to preserve Watson-Crick base-pairing arrangement, and finally align the original plane of the base with the new multi-copy plane. For the complementary strand, the complementary multi-copy base is used so that proper base pairing is achieved.

8. *Topology files for multi-copy bases.* The multi-copy bases of the previous section are not yet integrated in the CHARMM27 topology files ("top_all27_prot_na.rtf"). The user needs to create topology entries for the six multi-copy bases (**Note 7**) and append them to the original library file. The interested users can consult CHARMM27's topology library file, "top_all27_prot_na.rtf", which is distributed with the package, to learn the proper naming conventions CHARMM uses for protein and nucleic acids.

   The lines starting with "ATOM" in the PDB file are used by CHARMM to define the 3D coordinate of each atom. However, PDB files do not specify which atom is bonded with which one. The topology library file contains the information of the bonding arrangement and connectivity of each monomer unit (amino acids for proteins and nucleotides for DNAs), so that all the bonds, angles, and dihedral angles can be included in the evaluations of the force and energy. Therefore, it is of paramount importance that the topology of the molecular system is properly built.

For each nucleotide in the topology library file, “top_all27_prot_na.rtf”, there are the following sections of information: the atom types, the atomic charges, the bonds that connect the atoms, the hydrogen bond donor and acceptor atoms, and the internal coordinates required for adding missing hydrogen atoms and side chains for “IC BUILD” and HBUILD. Since all the entries of the nucleic acid nucleotides share identical backbone section (phosphate and sugar group), only the entries corresponding to the base atoms need to be combined to form the multi-copy base section. All the sections that correspond to base atoms need to be combined. The hydrogen bond sections are necessary if the HBOND module of CHARMM is to be used for hydrogen bond analysis.

An important addition to the multi-copy topology library file is the non-bonded exclusion section between atoms of the two bases in the multi-copy. For example, if A/C multi-copy is made, the atom section of the topology file must specify that the base atoms of the cytosine do not have any non-bonded (including electrostatic and van der Waals) interactions with the adenine base atoms.

As bond angles and dihedral angles are not explicitly listed in the topology files, the keyword “SETUP” is needed for generating them in CHARMM. This step will add one unwanted bond angle and four unwanted dihedral angles between the two bases in the multi-copy. So the keyword “DISCONNECT” should be used for these two bases, which will remove the unwanted angles from future force and energy evaluation. Using this method, the user will also need to append a few fictitious force field parameters to the standard parameter file (“par_all27_prot_na.prm”) for the unwanted angles. The force constant values do not matter, as they are removed from the force and energy evaluation by the “DISCONNECT” step.

9. *Systematic error*. As we can see from the Introduction, the relative binding free energy of a protein with two different DNA sequences is usually small. This creates a problem if the systematic and statistical errors of the calculation are larger than the relative energy difference we want to calculate. Statistical errors can be overcome by running longer simulations to collect independent data values for analysis. Systematic error is still a hard problem and there is no sound solution for its complete removal. Systematic error in molecular dynamics

simulation and free energy calculation is typically a result of poor sampling of the entire conformational space. It may also be due to biases in the molecular force field. Sufficient sampling of alternative favorable conformations of the protein and DNA is necessary. However, because these macromolecular systems contain tens of thousands of atoms and huge number of degrees of freedom, the entire conformational space is combinatorially large. This rugged energy surface often presents energy barriers between adjacent local minima, possibly limiting the sampling space. The heat-cool-cycle step we use in **Section 3.2.3** is an attempt to overcome local energy barriers.

For protein–DNA complexes, a problem that could cause insufficient sampling is the long-lived hydrogen bonds between protein and DNA bases. The hydrogen bonds formed with the DNA backbone generally do not contribute to the binding specificity, unless the backbone geometry is highly dependent on the base identity. If there is a particular hydrogen bond that exists between a protein residue and a DNA base pair throughout the simulation of the native complex, one must closely examine what is the fate of this hydrogen bond in the multi-copy complex structures. Since the multi-copy base pair is larger and needs more space, a prior stable hydrogen bond might become unstable due to strong van der Waals repulsion, and that part of the configurational space will no longer be sampled, leading to a bias in the calculations. This can also be true is there is a persistent and stable water-mediated hydrogen bond between the protein and the DNA. For such cases, other force field choices might need to be explored, such as the "soft core potential" that tones down van der Waals repulsion to allow bulky groups in a crowded space.

10. *Statistical error*. The duration of the production run during which trajectory frames are saved for future free energy analysis is important. Good statistics can in generally be achieved by running a sufficiently long production. However, the ensemble average we want to calculate **Eq. [5]** converges at about 100 ps (**Section 3.2.3**), indicating that longer productions than that will lead to the same free energy results. This production duration might be different for different systems. Therefore, it is important that the users examine the convergence of the ensemble average to reach a good compromise of convergence and statistical significance.

Acknowledgments

LAL acknowledges funding from the Department of Energy (DE-FG0204ER25626). JSB acknowledges funding from NSF CAREER 0546446, NIH/NCRR U54RR020839, and the Whitaker foundation. We acknowledge a starter grant and an MRAC grant of computer time from the Pittsburgh Supercomputer Center, MCB060010P, MCB060033P, and MCB060056N.

References

1. Pabo CO, Sauer RT. Transcription factors: structural families and principles of DNA recognition. Annu Rev Biochem 1992, 61:1053–1095.
2. Tuerk C, Gold L. Systematic evolution of ligands by exponential enrichment: RNA ligands to bacteriophage T4 DNA polymerase. Science 1990, 249(4968):505–510.
3. Ren B, Robert F, Wyrick JJ, et al. Genome-wide location and function of DNA binding proteins. Science 2000, 290(5500):2306–2309.
4. Mukherjee S, Berger MF, Jona G, et al. Rapid analysis of the DNA-binding specificities of transcription factors with DNA microarrays. Nat Genet 2004, 36(12):1331–1339.
5. Morozov AV, Havranek JJ, Baker D, Siggia ED. Protein-DNA binding specificity predictions with structural models. Nucleic Acids Res 2005, 33(18):5781–5798.
6. Paillard G, Lavery R. Analyzing protein-DNA recognition mechanisms. Structure (Camb) 2004, 12(1):113–122.
7. Endres RG, Schulthess TC, Wingreen NS. Toward an atomistic model for predicting transcription-factor binding sites. Proteins 2004, 57(2):262–268.
8. Schneider TD, Stephens RM. Sequence logos: a new way to display consensus sequences. Nucleic Acids Res 1990, 18(20):6097–6100.
9. Leach A. Molecular Modelling: Principles and Applications, 2nd ed. Prentice Hall, Harlow, England; New York, 2001.
10. Frenkel D, Smit B. Understanding Molecular Simulations: From Algorithms to Applications, 2nd ed. San Diego: Academic Press, 2002.
11. Berman HM, Westbrook J, Feng Z, et al. The protein data bank. Nucleic Acids Res 2000, 28(1):235–242.
12. Olson WK, Bansal M, Burley SK, et al. A standard reference frame for the description of nucleic acid base-pair geometry. J Mol Biol 2001, 313(1):229–237.
13. Cheatham TE, III, Young MA. Molecular dynamics simulation of nucleic acids: successes, limitations, and promise. Biopolymers 2000, 56(4):232–256.
14. Mackerell AD, Jr. Empirical force fields for biological macromolecules: overview and issues. J Comput Chem 2004, 25(13):1584–1604.
15. Kollman P. Free energy calculations: applications to chemical and biochemical phenomena. Chem Rev 1993, 93:2395–2417.