Obtaining Soft Matter Models of Proteins and their Phase Behavior

Irem Altan[1,*] and Patrick Charbonneau[1,2]

[1] *Department of Chemistry, Duke University, Durham, North Carolina 27708, United States*

[2] *Department of Physics, Duke University, Durham, North Carolina 27708, United States*

[*] *Corresponding author: irem.altan@duke.edu*

January 17, 2019

Abstract

Globular proteins are roughly spherical biomolecules with attractive, and highly directional interactions. This microscopic observation motivates describing these proteins as patchy particles: hard spheres with attractive surface patches. Mapping a biomolecule to a patchy model requires simplifying effective protein-protein interactions, which in turn provides a microscopic understanding of the protein solution behavior. The patchy model can indeed be fully analyzed, including its phase diagram. In this chapter, we detail the methodology of mapping a given protein to a patchy model and of determining the phase diagram of the latter. We also briefly describe the theory upon which the methodology is based, provide practical information, and discuss potential pitfalls.

Keywords: soft matter, phase behavior, protein crystallization, coarse-grained simulation

1 Introduction

While all-atom simulations of a single solvated protein are now fairly run-of-the-mill, simulating protein crystallization in a similar way is far beyond computational reach. The operation would require simulating hundreds of copies of the macromolecule, over very long timescales. In addition, such simulations would contain so much information that teasing out the relevant physico-chemical features that drive crystal assembly would itself be challenging. To circumvent these obstacles, coarse-grained models are used to capture protein-protein interactions in an effective manner, and thus to hide from view (and from computations) most of the obfuscating details. The key operations for such coarse-graining are: (i) determining and characterizing the relevant features of protein-protein interactions, and (ii) solving the properties of the resulting effective model. The first is done using all-atom simulations, and the second with the coarse-grained model alone. From a conceptual viewpoint, the key difficulties consist of identifying the relevant features and of choosing an appropriate coarse-graining scheme. Here, we describe a procedure developed over the last few years that focuses on crystal contacts between proteins for the former, and uses patchy models for the latter.

An attentive reader will notice that this approach gives information about the crystal assembly of a protein of known structure, and thus that has presumably already been crystallized. This is the vicious cycle of protein crystallization (see Fig. 1). Despite this obvious drawback for the study of a specific protein, such a scheme can be employed to understand the generic features that control protein crystallization, and thus the different classes of macromolecular assembly.

In this chapter, we detail the methodology for obtaining effective representations of protein-protein interactions, patchy models, and the steps involved in determining their phase diagram. For each of these tasks, we also briefly summarize the underlying theory. We conclude by discussing common complications and workarounds.

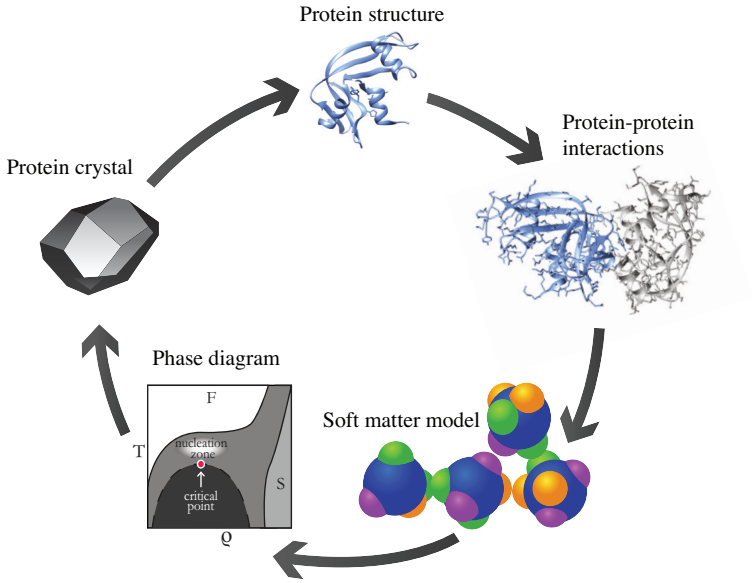

FIGURE 1: The vicious cycle of protein crystallization. The phase diagram of a given patchy model can be straightforwardly determined. The resulting phase information can in turn be used to optimize crystallization screens to reliably obtain protein crystals. However, constructing patchy models requires knowing protein-protein interactions, for which the protein structure itself is needed. Going through this process nevertheless results in a better understanding of how the microscopic properties of the protein control its phase behavior.

2 Materials

It may sound paradoxical but the most essential information needed to study protein crystal assembly is the protein crystal structure itself (Fig. 1). While high quality crystal structures are preferable, if the positions of some of the side chains cannot be resolved, it is still possible to add them on using tools such as KiNG [1], and then minimize the energy of the resulting configuration in order to avoid steric clashes. Once that has been resolved, the expensive computational work can begin.

In order to run all-atom simulations, molecular dynamics (MD) packages, such as Gromacs [2] or Amber [3], are essential. These packages include a variety of protein force fields and water models and are designed for sharing the computational load with graphical processing units

(GPUs). This is especially useful for simulating systems that contain a large number of non-bonded interactions, e.g., interactions that involve solvent molecules.

Once model parameters have been determined from all-atom MD simulations, the remainder of the work uses Monte Carlo (MC) simulations. Note that no generic MC code distribution is widely available, but the relevant methods for patchy models can be straightforwardly implemented based on Ref. [4]. Rovigatti et al. have also recently published a detailed review of the specific MC methods used for simulating patchy particles, along with an educational package for performing various such simulations [5, 6].

3 Methods

In this section we first describe how effective protein-protein interactions are obtained (Sec. 3.1), and how the nature of these interactions leads to a minimal patchy model. We then detail the process of obtaining the phase diagram of this model (Sec. 3.2). For the sake of concreteness, we use Gromacs for the first step and illustrate the overall process with a specific rubredoxin mutant [7] (Protein data bank [8] ID: 1YK4 [9]).

3.1 Effective Protein-Protein Interactions Through Umbrella Sampling

The change in free energy upon forming or destroying a protein-protein contact is determined from simulations that mimic experimental conditions and thus include solvent molecules and ions. In general, the free energy difference between two states along a reaction coordinate, ξ, is called the potential of mean force (PMF). For protein-protein interactions in particular, one needs to determine the PMF as a function of distance between two proteins, given a specific crystal contact. The natural choice for the reaction coordinate is then the protein-protein distance.

At equilibrium, the probability that the system is found at a given ξ is [10]

$$Q(\xi)\mathrm{d}\xi = \frac{\int \delta(\xi(\mathbf{r}^N) - \xi) e^{-\beta U(\mathbf{r}^N)} \mathrm{d}\mathbf{r}^N}{\int e^{-\beta U(\mathbf{r^N})} \mathrm{d}\mathbf{r}^N} \mathrm{d}\xi, \tag{1}$$

where δ is the Dirac delta function, $\mathbf{r}^N$ denotes the coordinates of the N particles of the system, and $U(\mathbf{r}^N)$ is the potential energy of a given configuration. This configuration is observed with a probability proportional to its Boltzmann weight, i.e., $e^{-\beta U(\mathbf{r}^N)}$ at inverse temperature $\beta \equiv 1/k_\mathrm{B}T$ where k_B is the Boltzmann constant. The constrained (Helmholtz) free energy, $A(\xi)$, as a function of the reaction coordinate, $-\beta A(\xi) = \ln Q(\xi)$, corresponds to the PMF. While it is theoretically possible to sample $Q(\xi)$ in a single molecular dynamics (MD) simulation, in practice the small Boltzmann weight of the dissociated configurations makes sampling exceedingly difficult. It is thus advantageous to introduce a series of biasing potential, $w_i(\xi)$, and to simulate the system with modified energy functions,

$$U_i'(\mathbf{r}^N) = U(\mathbf{r}^N) + w_i(\xi). \tag{2}$$

Sampling all ξ is then possible because the original energy barriers are lowered, or equivalently, the Boltzmann weight of the associated configurations is increased. A convenient choice of bias is a harmonic potential, i.e., a spring,

$$w_i(\xi) = k(\xi_i - \xi)^2, \tag{3}$$

where ξ_0 is the imposed protein-protein distance and k is the spring constant. Using this potential, we separate the protein-protein center of mass distance into M umbrella sampling windows, with different ξ_i (Fig. 2).

In what follows, we detail the computational steps involved in this procedure for a given protein. Note that a detailed tutorial for the process is available for Gromacs (see note 1). We here provide the details that are not mentioned in that tutorial or that differ for crystal contacts.

1. **Determine contacts.** Crystal contacts can be determined using PISA [11], an online tool that identifies protein-protein interfaces for a given `.pdb` (protein data bank) file. PISA takes

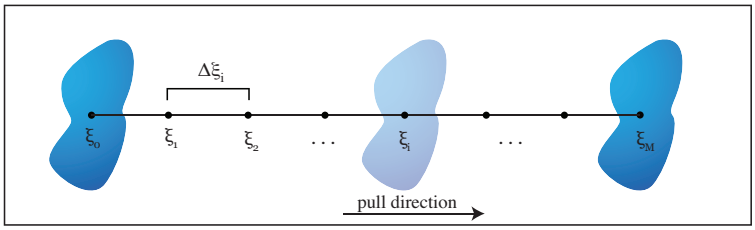

FIGURE 2: Two proteins are pulled away from each other to generate umbrella sampling windows centered at ξ_i. Each window is sampled by MD simulations using a biased interaction potential $U_i'(\mathbf{r}^N)$. The results for the different windows are then joined to reconstruct the overall PMF as a function of ξ.

into account protein symmetry and crystal periodicity, in addition to using a distance cutoff for determining contacts. For each contact, it also lists the residues involved in hydrogen bonding and salt bridges, and provides an estimate for the contact free energy. This estimate, however, is fairly rough, because many contributions, such as interactions with the crystallization cocktail or side-chain and backbone conformational changes, are not explicitly considered.

2. **Add any missing or incomplete residues.** When starting from a crystal structure, it is possible that entire residues or some of their side chains might be missing because they could not be crystallographically resolved. These should be added manually to the protein structure. Note that the accuracy of the orientation and the conformation of these residues is not critical at this stage (within chemical reasonableness), because the protein structure is minimized in subsequent steps, eliminating any steric clash that might arise. The interaction strength between patches also depends on the protonation states of the contained residues. A rough estimate for the protonation states can be obtained with propKa [12], keeping in mind that in most cases the solution conditions for crystallization experiments is such that the protein carries no net charge.

3. **Generate input files for each contact.** For each contact, separate `.pdb` files with two copies of the protein forming the contact should be generated. These `.pdb` files should then be converted to `.gro` files (the default structure file format for Gromacs) using the Gromacs

command `pdb2gmx`. This command also prompts the user to choose a force field and a water model. Once the `.gro` file is created, it is convenient to rotate the protein assembly so that the z-axis corresponds to the pull direction and to center them in the simulation box. The assembly should be at least 1 nm away from the box sides, in order to prevent one protein from interacting with its copy across the (periodic) box boundary, given that the cutoff for neighbor lists, electrostatic and van der Waals interactions is less than 2 nm. This centering and resizing can be achieved with the following command

```
gmx editconf -f box.gro -o newbox.gro -c -d 1.0
```

The next step is to elongate the box along the pull direction, making sure that the box is at least twice the pull direction plus the original box size. Suppose that the box size in `newbox .gro` is $10 \times 10 \times 10$ nm. In order to pull one of the protein copies 5 nm away one should run

```
gmx editconf -f newbox.gro -o newbox2.gro -center 5 5 5
    -box 10 10 20,
```

for the box to be resized to $10 \times 10 \times 20$ nm.

4. **Add solvent and ions.** Once the box dimensions are selected and the proteins are positioned, solvent and ions are inserted with the commands `gmx solvate` and `gmx genion`, respectively. Ideally, one should use the same ion type and concentration as in the crystallization cocktail. If the force field parameters for these specific ions are not readily available, however, one might consider replacing them with simple generic ions such as sodium and chloride. This replacement at least matches the ionic strength of the solvent and thus the extent of charge screening in the experimental setup.

5. **Minimize energy and equilibrate.** The energy of the resulting system should be minimized before running any simulation, because the positions of the waters and ions placed in the box, as well as the conformation and orientation of the inserted residues and side chains, need to be relaxed. To further relax the system, a short additional simulation should be performed in

which the number of particles, pressure, and temperature are kept constant (constant NPT), before generating input configurations for umbrella sampling. Note that keeping the center of mass of the protein-protein complex fixed for this step facilitates the remainder of the procedure.

6. **Generate configurations for umbrella sampling.** In order to generate initial configurations for umbrella sampling, a simulation is run in which one protein is pulled away from the other at a constant rate, using a harmonic potential with force constant k, while the other protein is restrained. (A convenient approach is to restrain three or four backbone atoms.) Typically, k ranging from 1,000 to 10,000 kJ $\text{mol}^{-1}\text{nm}^{-2}$ is appropriate for pulling the proteins apart. Since the resulting configurations are not necessarily equilibrated they need to be further relaxed after the pull, before being used as inputs to umbrella sampling simulations.

7. **Choose umbrella sampling windows and run simulations.** The M chosen configurations should cover the whole distance interval of interest, and their pair separation, $\Delta\xi_i$, (Fig. 2) should be such that the resulting umbrella sampling windows overlap sufficiently. In particular, a significant portion of the tails of distributions of ξ for each neighboring pair of windows should overlap (see Fig. 3a). The constant k for the umbrella simulations should be strong enough to keep the proteins at roughly the desired separation, but not so strong that the resulting distributions are overly narrow. Additional windows, as permitted by the box size, can nonetheless be added after the PMF is generated, if any pair of windows does not sufficiently overlap. Note that if the overlap is poor, one often observes discontinuities in the resulting PMF. This serves as a diagnostic.

8. **Generate PMF.** The PMF is constructed from the output of each umbrella sampling simulation using the command `gmx wham` (Fig. 3). The input files necessary for this step are the `.tpr` (The Gromacs executables for individual umbrella sampling simulations) and `pullf*.xvg`, which contain the force information for each window.

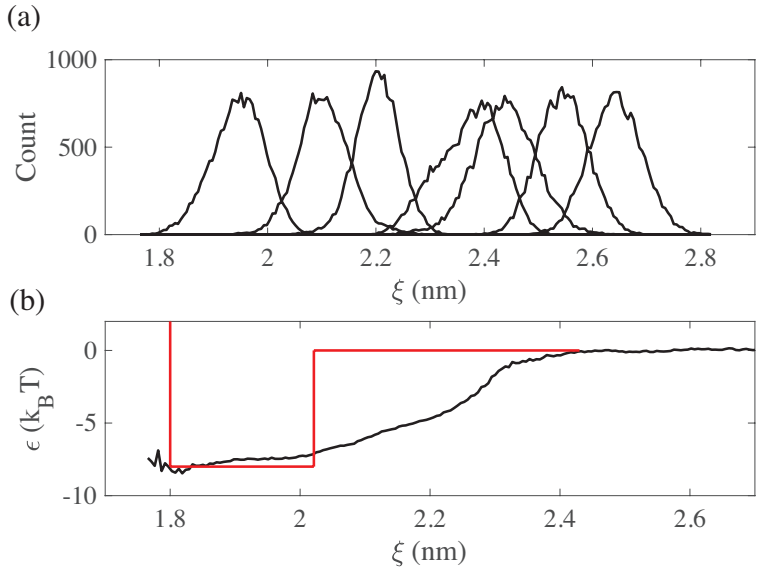

FIGURE 3: (a) Histograms generated by the weighted histogram analysis method (`gmx wham`) [13]. The distribution for each window overlaps well with those of their neighbors. (b) The PMF for a contact of rubredoxin as a function of pull distance, ξ, at $T = 300$K [7]. Infinite separation sets the zero of the energy. The square-well potential generated from this PMF by fitting the second virial coefficient is shown in red (see Section 3.2.1).

3.2 Phase Diagram

Our ultimate goal is to capture the solution and the assembly behavior of a protein from the simplest possible physical model. This is not only useful in making the simulations computationally tractable, but also serves as a consistency check for the microscopic features we previously identified as relevant. In that context, we consider globular proteins to be roughly spherical objects with anisotropic interactions that are dictated by their surface amino acids. These key features, together with the assumption that the relevant surface amino acids are involved in crystal contacts, suggest a minimal model comprising a hard sphere with attractive surface patches, i.e. a patchy model. Patchy models based on the Kern-Frenkel potential [14] and others have indeed been shown to recapitulate the phase behavior of various globular proteins [7, 15–18]. The location, interaction range and strength of the patches, as well as their angular width, can

be determined from all-atom simulations of the crystal structure. Note that this model is suitable for short-range interactions. That is, the protein should either be uncharged, or the ionic strength of the crystallization cocktail should be sufficiently high for charge-charge interactions to be screened. These conditions are precisely those used in crystallization experiments. Once this minimal model is parameterized, its phase diagram can be obtained using MC simulations. This section first describes a model that captures the properties of the effective protein-protein interactions computed in Sec. 3.1 (Sec. 3.2.1). The general idea of thermodynamic integration, which is used for calculating the free energy of a system from a reference state is then introduced (Sec. 3.2.2). The calculation of fluid (Sec. 3.2.3) and crystal (Sec. 3.2.4) free energies are subsequently described, as well as the procedure for obtaining a coexistence point between these two phases, and the Gibbs-Duhem integration scheme (Sec. 3.2.5) for obtaining the complete crystal solubility curve. Finally, we discuss how Gibbs Ensemble simulations can be used to obtain the (metastable) gas-liquid binodal (Sec. 3.2.6).

3.2.1 Patchy Models

In a patchy model, the interaction potential between two patchy particles i and j is

$$U(r_{ij}, \Omega_i, \Omega_j) = U_{\mathrm{HS}} + \sum_{\alpha,\beta=1}^{n} U_{\alpha\beta}(r_{ij}, \Omega_i, \Omega_j), \tag{4}$$

where U_{HS} is the hard sphere repulsion, which prohibits overlaps between particles of diameter σ, α and β label one of the n surface patches, r_{ij} is the distance between particles, and Ω_1 and Ω_2 describe the particle orientations (either in terms of Euler angles or quaternions). The attractive part of the potential, $U_{\alpha\beta}$, is then

$$U_{\alpha\beta} = v_{\alpha\beta}(r_{ij}) f_{\alpha\beta}(\Omega_i, \Omega_j), \tag{5}$$

with

$$v_{\alpha\beta}(r_{ij}) = \begin{cases} -\epsilon_{\alpha\beta}, & \sigma < r_{ij} \leq \lambda_{\alpha\beta}\sigma \\ 0, & \text{otherwise} \end{cases} \quad ; \quad f_{\alpha\beta}(\Omega_i, \Omega_j) = \begin{cases} 1, & \cos\theta_\alpha \geq \cos\delta_\alpha \text{ and } \cos\theta_\beta \geq \cos\delta_\beta \\ 0, & \text{otherwise} \end{cases} . \tag{6}$$

In other words, a square-well potential of range $\lambda_{\alpha\beta}\sigma$ controls the radial part of the attraction, and the widths of patches α and β, δ_α and δ_β, respectively (Fig. 4) set their angular range. That is, patches only attract when the vector joining their centers of mass passes through the patch of both particles.

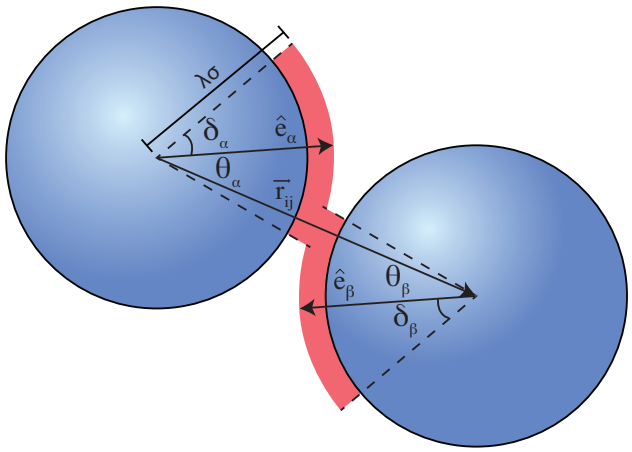

FIGURE 4: Two particles do not interact unless their patches overlap in distance and orientation. Here, $\hat{e}_\alpha$ and $\hat{e}_\beta$ point to the centers of patches α and β, respectively, hence they interact if both $\hat{e}_\alpha \cdot \hat{r}_{ij} = \cos\theta_\alpha \geq \cos\delta_\alpha$ and $\hat{e}_\beta \cdot \hat{r}_{ij} = -\cos\theta_\beta \leq \cos\delta_\beta$, with $r_{ij} \leq \lambda_{\alpha\beta}\sigma$.

The parameters for the radial part of the attraction are obtained from the PMF computed in Sec. 3.1 (Fig. 3a). The depth of the square-well potential, $\epsilon_{\alpha\beta}$ is that of the PMF for contact between patches α and β. The range of the square-well attraction is obtained by matching its contribution to B_2, the second virial coefficient, to that of the PMF, where

$$B_2 = -\frac{1}{2}\int (e^{-\beta U(\mathbf{r})} - 1)\mathrm{d}\mathbf{r}, \tag{7}$$

This integral is evaluated numerically for the PMF (its value denoted I) and analytically for

the α-β contact. For a given contact, the interaction range, $\lambda_{\alpha\beta}$, is found by equating the two results,

$$\lambda_{\alpha\beta} = \left(\frac{3I}{e^{\beta\epsilon_{\alpha\beta}} - 1} + 1 \right)^{1/3}. \tag{8}$$

Finally, the angular breadth of the interaction is set by running simulations in which the distance between the two proteins is fixed, but not their relative orientation. This is achieved by constraining the center of mass distance with a harmonic spring, at the equilibrium bonding distance. The deviation of the patch vectors from the center of mass axis is tracked in terms of the angle δ between them. The angular breadth, $\cos\delta_\alpha$, for patch α is taken to be the mean of the computed distribution for that angle, and the same for $\cos\delta_\beta$ of patch β.

3.2.2 Thermodynamic Integration

Once the patchy model is parameterized, various types of MC simulations are employed to trace out its phase diagram. For two or more phases to be in coexistence, their temperature, pressure, and chemical potential, μ, must all be equal. While P and T can be straightforwardly enforced, μ is more challenging. Simulations, like experiments, can only determine the *change* in free energy along a transformation, not its absolute value. One thus needs a reference state of known free energy and a transformation from that reference to the system of interest to calculate its free energy [4, 19]. Reference states that are of particular interest for us are the ideal gas and the Einstein crystal. Integrating from any of these states along an isotherm yields for the Helmholtz free energy

$$A(\rho, T) = A(\rho_0, T) + N \int_{\rho_0}^{\rho} \frac{P(\rho')}{\rho'^2} \mathrm{d}\rho', \tag{9}$$

where ρ_0 is the density of the reference system and $P(\rho)$ is the equilibrium pressure of the system at a density, ρ. Here, we consider the number density, such that $\rho \equiv N/V$, where V is the volume of the system. For numerical convenience, if the reference system is an ideal gas,

this expression is rewritten as [20]

$$A^{\text{fluid}}(T,\rho) = A^{\text{ideal}}(\rho) + N\int_0^{\rho}\left[\frac{P}{\rho'^2} - \frac{1}{\beta\rho'}\right]\mathrm{d}\rho', \tag{10}$$

where A^{ideal} is the free energy of the ideal gas (see note 2)

$$\frac{A^{\text{ideal}}(\rho)}{N} = \frac{1}{\beta}[\log(\rho\Lambda^3) - 1 + \frac{1}{N}\log(2\pi N)]. \tag{11}$$

where the de Broglie wavelength, Λ^3, is set to unity without loss of generality. Another option is to integrate at constant pressure, i.e., along an isobar, varying the temperature of the system

$$\beta\mu(T,P) = \beta\mu(T_0,P) - \int_{T_0}^{T}\frac{H(T')}{Nk_{\text{B}}T'^2}\mathrm{d}T', \tag{12}$$

where H is the enthalpy of the systems, or along an isochore,

$$\beta\frac{A(T,V)}{N} = \beta\frac{A(T_0,V)}{N} - \int_{T_0}^{T}\frac{U(T')}{Nk_{\text{B}}T'^2}\mathrm{d}T', \tag{13}$$

where U is the internal energy of the system.

3.2.3 Free Energy of the Fluid Phase

Using the principles of thermodynamic integration introduced above, the following steps summarize how we obtain the free energy of the fluid phase. To integrate along the isotherm from the ideal gas, we run NPT simulations at a set of pressures $\{P_1, \cdots, P_m\}$, where P_1 is a very low pressure and $\rho(P_m)$ is the density of interest. Figure 5(a) shows the numerical equation of state of the fluid phase for the patchy model of rubredoxin [7]. The integrand of Eq. (10) is calculated from these data points (Fig. 5b). The integrand gets noisy as pressure decreases, because both $1/\rho^2$ and $1/\rho$ diverge as $\rho \to 0$, hence the numerical error is then amplified. In that regime, one can use the fact that the integrand converges to B_2, Eq. (7), as $\rho \to 0$ to increase

numerical accuracy. There are three options for the rest of the thermodynamic integration: (i) continue integrating along the same isotherm to obtain the free energy as a function of pressure, (ii) integrate along an isobar using Eq. (12), or (iii) integrate along an isochore using Eq. (13).

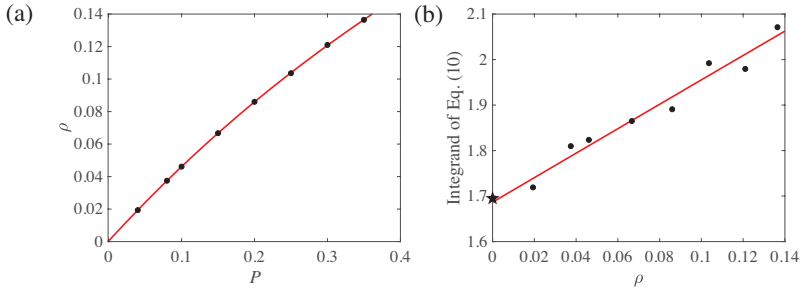

FIGURE 5: (a) Equation of state (density as a function of pressure), and (b) the integrand of Eq. (10) for the fluid phase of the patchy model of rubredoxin at $\beta = 0.2$ (in units of $1/k_{\mathrm{B}}T$) [7]. The star denotes B_2, the y-intercept of the integrand Eq. (7). Curves are polynomial fits to the data.

3.2.4 Free Energy of the Crystal Phase

The Frenkel-Ladd method [21] is a thermodynamic integration scheme to obtain the free energy of a crystal using an Einstein crystal with a fixed center of mass as a reference. Particles are then restrained to their equilibrium lattice positions and orientations, and do not otherwise interact. The interaction energy of this system is

$$U_E = \sum_{i=1}^{N} \kappa(\mathbf{r}_i - \mathbf{r}_{i,0})^2 + \sum_{i=1}^{N} \kappa\eta g(\Omega_i). \tag{14}$$

The first term restrains the positions $\mathbf{r}_i$ of the N particles to their positions $\mathbf{r_{i,0}}$ by a harmonic potential with spring constant $\kappa \in [0, \infty)$. The second term penalizes particles that deviate from their equilibrium orientations, by the potential $g(\Omega) = 1 - \cos(\psi_{i\alpha}) + 1 - \cos(\psi_{i\beta})$ [7], where $\psi_{i\alpha}$ ($\psi_{i\beta}$) is the angle between the vector that defines patch α (β) in its equilibrium and instantaneous orientations, and patches α and β are chosen arbitrarily among the surface

patches (see note 3). The free energy of the ideal Einstein crystal with fixed center of mass, A_E, has both translational and orientational contributions $A_E = A_{E,t} + A_{E,o}$ where [19, 20]

$$\beta\frac{A_{E,t}}{N} = -\frac{3}{2}\frac{N-1}{N}\log\Big(\frac{\pi}{\beta\kappa}\Big) - \frac{3}{2N}\log(N) \tag{15}$$

$$\beta\frac{A_{E,o}}{N} = -\log\Big(\frac{1}{8\pi^2}\int \mathrm{d}\Omega\, e^{-\kappa\eta g(\Omega)/k_\mathrm{B}T}\Big). \tag{16}$$

This reference system is converted to the interacting protein crystal in three steps.

1. **Switch on interactions.** The free energy change in this step is

$$\Delta A_1 = -\log\Big\langle e^{-\beta(\widetilde{U}-U_0)}\Big\rangle_E + U_0, \tag{17}$$

where $\langle\cdot\rangle_E$ denotes an averaging over ideal Einstein crystal configurations, $\widetilde{U}$ is the energy of the interacting Einstein crystal without the harmonic spring contributions, and U_0 is the interacting crystal ground state energy. For large enough κ (denoted κ_max) the contribution from the thermal average vanishes, i.e. $\Delta A_1 \approx U_0$. This condition sets κ_max.

2. **Turn off position and orientation restraints.** In this step, the springs are turned off, i.e. $\kappa \to 0$. The change in free energy of this process is

$$\Delta A_2 = -\int_0^{\kappa_\mathrm{max}} \mathrm{d}\kappa'\Big\langle\frac{\partial U_E}{\partial\kappa'}\Big\rangle_{NVT\kappa'} = -\int_0^{\kappa_\mathrm{max}} \mathrm{d}\kappa'\Big\langle\sum_{i=0}^{N}(\mathbf{r}_i-\mathbf{r}_{i,0})^2 + \eta\sum_{i=0}^{N} g(\Omega_i)\Big\rangle_{NVT\kappa'}. \tag{18}$$

Note that because κ_max can be very large, it is often more convenient to use $\log\kappa$ as the integration variable

$$\Delta A_2 = -\int_{-\infty}^{\log\kappa_\mathrm{max}} \mathrm{d}(\log\kappa')\,\kappa'\Big\langle\sum_{i=0}^{N}(\mathbf{r}_i-\mathbf{r}_{i,0})^2 + \eta\sum_{i=0}^{N} g(\Omega_i)\Big\rangle_{NVT\kappa'}, \tag{19}$$

where the integrand is evaluated by running NVT simulations at different κ. The first term in the integrand is the mean square displacement and the second term is a measure of the

orientational displacement, which are both easily measured in simulations. The integral can then be evaluated using Gaussian Quadrature [22] with 20 to 40 logarithmically-spaced points. Because the integrand vanishes when $\log\kappa \to -\infty$ the integration can start from a small (non-zero) κ. Evaluating the translational contribution to the Einstein crystal free energy is straightforward Eq. (15), and although the orientational part Eq. (16) cannot be calculated analytically, for large $\kappa_{\max}\eta$, one can use the saddle point approximation

$$\int \mathrm{d}\Omega e^{-\beta\kappa_{\max}\eta g(\Omega)} \approx e^{-\beta\kappa_{\max}\eta g(\Omega_0)}\sqrt{\frac{2\pi}{\beta\kappa_{\max}\eta g''(\Omega_0)}}. \tag{20}$$

Here we have approximated $g(\Omega)$ with its second order Taylor expansion, and Ω_0 is the orientation that minimizes $g(\Omega)$. The orientational contribution to the Einstein crystal free energy then becomes

$$\beta\frac{A_{\mathrm{E,o}}}{N} \approx \frac{3}{2}\log(\beta\kappa_{\max}\eta) + \frac{1}{2}\Big\{8\pi \det\left(H[g(\Omega_0)]\right)\Big\}, \tag{21}$$

where $\det(H(g(\Omega_0))$ is the determinant of the Hessian computed at the minimum of $g(\Omega)$ [7]. Note that one should check whether $\kappa_{\max}$ is indeed large enough by verifying that higher order terms in the Taylor expansion are negligible.

An estimate for the plateau value for the integrand of Eq. (19) is can be analytically estimated for sufficiently large κ, and thus serves as a consistency check. The orientational contribution to this quantity is calculated using the saddle point approximation to $A_{E,o}$, Eq. (21),

$$\kappa\eta\Big\langle\sum_{i=0}^{N} g(\Omega_i)\Big\rangle_{\kappa} = \kappa\eta\frac{\partial A_{E,o}}{\partial(\kappa\eta)} = \kappa\eta\frac{3}{2\beta}\frac{\beta}{\beta\kappa\eta} = \frac{3N}{2\beta}, \tag{22}$$

and the translational contribution can be estimated using the expression derived for the hard sphere mean square displacement [21]. In the limit of very large κ, this result should be exact because translational and orientational displacements are then too small to break any bond.

3. **Release the crystal center of mass.** Removing the constraint over the center of mass finally gives

$$\Delta A_3 = \frac{1}{\beta}\log(\rho). \tag{23}$$

Cumulating these results gives the absolute free energy of the crystal of patchy particles

$$A = A_E + \Delta A_1 + \Delta A_2 + \Delta A_3 \tag{24}$$

at a given density and temperature. Integration along an isobar, isotherm, or an isochore within the crystal phase can then be performed to explore a broad range of conditions within that phase.

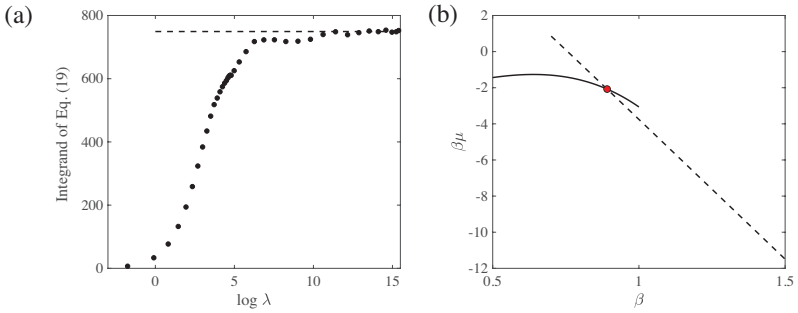

FIGURE 6: (a) The integrand in Eq. (19). The dashed line is the plateau value estimated from the translational and orientational displacements of particles at large κ (see Eq. (22) and Ref. [21]). (b) The chemical potential, $\mu = \frac{A}{N} - \frac{P}{\rho}$ of the fluid phase obtained by integrating Eq. (10) with the data of Fig. (5)(solid line) and the crystal phase (dashed line) for a fixed $P = 0.35$. Their intersection gives a coexistence point between the two phases (red point).

3.2.5 Gibbs-Duhem Integration

Given a coexistence point between the crystal and the liquid, the Gibbs-Duhem relation,

$$t(P(\beta),\beta) \equiv \left(\frac{\mathrm{d}P}{\mathrm{d}\beta}\right)_{\mathrm{coex}} = -\frac{H_{\mathrm{crys}}/N - H_{\mathrm{liq}}/N}{\beta(1/\rho_{\mathrm{crys}} - 1/\rho_{\mathrm{liq}})} = -\frac{\Delta H/N}{\Delta(1/\rho)}, \tag{25}$$

can be integrated to obtain coexistence points at different temperatures[23]. This can be done using a numerical method, such as predictor-corrector algorithms, and evaluating the thermodynamic quantities via MC simulations. The general idea is as follows.

1. Start from a known coexistence point (P_0,β_0), consider a system at $\beta_1 = \beta_0 + \Delta\beta$, where $\Delta\beta$ is small (see below).
2. Guess the coexistence pressure at this temperature according to the appropriate predictor formula.
3. Run NPT simulations of the two phases simultaneously to equilibrate $\Delta H/N$ and $\Delta(1/\rho)$. Correct the pressure prediction using these quantities according to the appropriate corrector formula.
4. Repeat 2 and 3 until convergence.

The chosen integration scheme depends on how many coexistence points are known (see Table 1). At the start of the process, only one such point, (P_0, β_0), is known. A short simulation is run for both phases to obtain $t(P_0, \beta_0)$. The guess for the pressure, P_1, for the next coexistence temperature, $\beta_1 = \beta_0 + \Delta\beta$, and its correction are then given by the trapezoid rule, after calculating $t(P_1, \beta_1)$ with the initial guess. The third and fourth coexistence points can be calculated using the midpoint rule. Once four points are obtained, additional ones can be found iteratively using Adams rule (see note 4). While $\Delta\beta$ should be large enough to allow for an efficient tracing using a too large a value causes numerical instability [24]. One way to validate the resulting coexistence line is to repeat this procedure starting from different, well separated coexistence points.

3.2.6 Obtaining the Gas-Liquid Binodal

Coexistence points on the gas-liquid binodal can certainly be obtained by slightly modifying the above procedure, but a more straightforward approach is to use Gibbs Ensemble simulations [4, 25], which is specifically designed for identifying coexistence between homogeneous phases of

Method	Predictor	Corrector
Trapezoid	$P_{i+1} = P_i + \Delta\beta t_i$	$P_{i+1} = P_i + \frac{\Delta\beta}{2}(t_i + t_{i+1})$
Midpoint	$P_{i+2} = P_i + 2\Delta\beta t_{i+1}$	$P_{i+2} = P_i + \frac{\Delta\beta}{3}(t_{i+2} + 4t_{i+1} + t_i)$
Adams	$P_{i+4} = P_{i+3} + \frac{\Delta\beta}{24}(55t_{i+3} - 59t_{i+2} + 37t_{i+1} - 9t_i)$	$P_{i+4} = P_{i+3} + \frac{\Delta\beta}{24}(9t_{i+4} + 19t_{i+3} - 5t_{i+2} + t_{i+1})$

TABLE 1: Predictor-corrector algorithms, where $t_i \equiv t(P_i, \beta_i)$ for $\beta_i = \beta_0 + i\Delta\beta$, is defined in Eq. (25)).

intermediate density. In this method, two boxes of fluid are simulated simultaneously. Their total volume and number of particles are kept constant but boxes can exchange volume as well as particles between each other. The density of each box then converges to the gas or the liquid density, ρ_g and ρ_l respectively (see note 5). The binodal is then obtained as follows.

1. **Obtain a few coexistence points from Gibbs Ensemble simulations.** Gibbs Ensemble simulations are run for a set of temperatures below the estimated critical temperature, T_c, from generalized law of corresponding states [26], starting from an intermediate fluid density, $\rho \approx 0.3$.

2. **Fit coexistence data to obtain the full binodal.** The physical universality of the gas-liquid transition allows for tracing the binodal using only a few coexistence points. The full binodal, including the critical point, can then be calculated by fitting two universal equations: a scaling law and the law of rectilinear diameters [4]. The former gives an estimate of T_c,

$$\log(\rho_l - \rho_g) = \log B + \beta \log(T - T_c), \tag{26}$$

where $\beta = 0.32$ (not to be confused with the inverse temperature) is the magnetization exponent for the Ising universality class, and B is a proportionality constant [27]. Once

T_c is found, the critical density, ρ_c, can be obtained from the law of rectilinear diameters, which describes the asymmetry of the gas-liquid binodal away from T_c,

$$\frac{\rho_l + \rho_g}{2} = \rho_c + A(T - T_c), \tag{27}$$

where A and ρ_c are determined by fitting. The coexistence binodal (see Fig. 8) is then given by

$$\rho = \rho_c + A(T - T_c) \pm \frac{B(T - Tc)^\beta}{2}. \tag{28}$$

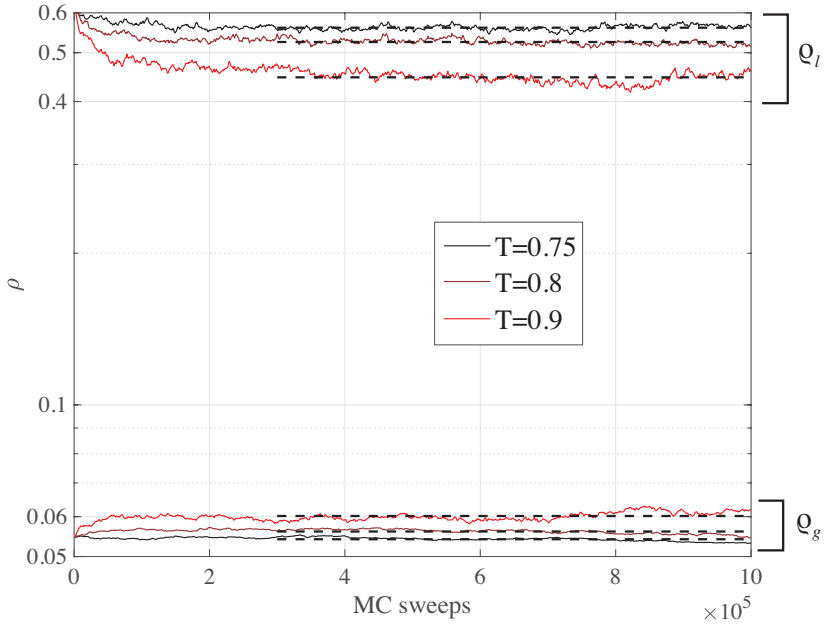

FIGURE 7: Evolution of ρ_l and ρ_g throughout the Gibbs Ensemble simulations for various temperatures. Note that the densities converge to their coexistence values after roughly 2×10^5 MC sweeps. Average densities are calculated (dashed lines) after the equilibration period. These three pairs of data points are those that appear in the final phase diagram (Fig. 8).

4 Notes

The above procedure results in the phase diagram for a simple globular protein as seen in Fig. 8. In what follows we discuss a number of geometric issues and how they can be avoided, as well as briefly mention possible ways of increasing the model complexity.

1. The tutorial prepared by Justin Lemkul can be accessed at `http://www.bevanlab.biochem.vt.edu/Pages/Personal/justin/gmx-tutorials/umbrella/index.html`
2. While the simplest method for obtaining the free energy of a fluid is Widom insertion [4], at high density it is more accurate to use Eq. (10).
3. The parameter η is a proportionality constant chosen such that the strengths of both restraints can be tuned by κ alone.
4. It is often not advantageous to use higher order predictor-corrector formulas. These not only require a higher number of coexistence points but also exhibit stability issues.
5. If the temperature is close to the critical temperature the small density difference can cause the boxes to switch identity (liquid↔gas) (Fig. 7), which limits the efficacy of the approach in this regime.
6. **Geometric Constraints:** Because globular proteins are not actually spherical, the onset of harsh repulsion for each contact PMF can occur at slightly different distances. In the scheme above, the chosen particle diameter should be the same for all contacts. The smallest center of mass distance should then be taken as the hard sphere diameter and the other PMFs should be translated such that the onset of attraction coincides with that diameter.

   Another problem that can arise due to deviations from sphericity is that a simple projection of the patch position on the sphere may not result in all patches interacting within the relevant crystal symmetry. In this case, patch vectors and unit cell parameters can be perturbed slightly to ensure that all bonds are satisfied in the crystal phase. This modification is known to have but a very limited effect on the phase diagram [28].
7. **Increasing Model Complexity** The simple patchy model described here does not capture

the phase behavior of certain proteins. Numerous modifications patchy models have been proposed to capture these effects. The impact of shape anisotropy[29–31], patch mobility [32], and the interaction potential form [33, 34] have been investigated in the context of general self-assembly. Such features can be considered if the microscopic properties of the protein of interest suggest that more complex models are required. The application of these features to specific protein systems is still an open area of research.

8. Data and scripts relevant to this work have been archived and can be accessed at `http://dx.doi.org/doi:XX.XXXX/xxxx`.

References

[1] Chen V, Davis I, Richardson D (2009) KiNG (Kinemage, next generation): A versatile interactive molecular and scientific visualization program. Protein Sci 18(11):2403–2409

[2] Berendsen H, van der Spoel D, van Drunen R (1995) Gromacs: A message-passing parallel molecular dynamics implementation. Comput Phys Commun 91(1):43–56

[3] Case D, Cerutti D, Cheatham T III, Darden T, Duke R, Giese T, Gohlke H, Goetz A, Greene D, Homeyer N, Izadi S, Kovalenko A, Lee T, LeGrand S, Li P, Lin C, Liu J, Luchko T, Luo R, Mermelstein D, Merz K, Monard G, Nguyen H, Omelyan I, Onufriev A, Pan F, Qi R, Roe D, Roitberg A, Sagui C, Simmerling C, Botello-Smith W, Swails J, Walker R, Wang J, Wolf R, Wu X, Xiao L, York D, PA K (2017) Amber 2017. Tech. rep., University of California, San Francisco

[4] Frenkel D, Smit B (2001) Understanding Molecular Simulation: From Algorithms to Applications. Academic Press

[5] Rovigatti L, Russo J, Romano F (2018) How to simulate patchy particles. Eur Phys J E 41(5):59

[6] Rovigatti L, Romano F, Russo J (2018) lorenzo-rovigatti/patchyparticles v1.0.1. DOI 10.5281/zenodo.1171695, URL `https://doi.org/10.5281/zenodo.1171695`

[7] Fusco D, Headd J, De Simone A, Wang J, Charbonneau P (2014) Characterizing protein crystal contacts and their role in crystallization: Rubredoxin as a case study. Soft Matter 10(2):290–302

[8] Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H, Shindyalov IN, Bourne PE (2000) The protein data bank. Nucleic Acids Res 28(1):235–242

[9] Bönisch H, Schmidt C, Bianco P, Ladenstein R (2005) Ultrahigh-resolution study on pyrococcus abyssi rubredoxin. I. 0.69 Å X-ray structure of mutant W4L/R5S. Acta Crystallogr D 61(7):990–1004

[10] Kästner J (2011) Umbrella sampling. Wiley Interdiscip Rev Comput Mol Sci 1(6):932–942

[11] Krissinel E, Henrick K (2007) Inference of macromolecular assemblies from crystalline state. J Mol Biol 372(3):774–797

[12] Rostkowski M, Olsson M, Søndergaard C, Jensen J (2011) Graphical analysis of ph-dependent properties of proteins predicted using propka. BMC Struct Biol 11(1):1

[13] Hub J, De Groot B, Van Der Spoel D (2010) g_wham – a free weighted histogram analysis implementation including robust error and autocorrelation estimates. J Chem Theory Comput 6(12):3713–3720

[14] Kern N, Frenkel D (2003) Fluid–fluid coexistence in colloidal systems with short-ranged strongly directional attraction. J Chem Phys 118(21):9882–9889

[15] Fusco D, Charbonneau P (2016) Soft matter perspective on protein crystal assembly. Colloids Surf B 137:22–31

[16] Sear R (1999) Phase behavior of a simple model of globular proteins. J Chem Phys 111(10):4800–4806

[17] Wentzel N, Gunton J (2008) Effect of solvent on the phase diagram of a simple anisotropic model of globular proteins. J Phys Chem B 112(26):7803–7809

[18] Dixit N, Zukoski C (2002) Crystal nucleation rates for particles experiencing anisotropic interactions. J Chem Phys 117(18):8540–8550

[19] Vega C, Sanz E, Abascal J, Noya E (2008) Determination of phase diagrams via computer simulation: Methodology and applications to water, electrolytes and proteins. J Phys Condens Matter 20(15):153101

[20] Romano F, Sanz E, Sciortino F (2010) Phase diagram of a tetrahedral patchy particle model for different interaction ranges. J Chem Phys 132(18):184501

[21] Frenkel D, Ladd A (1984) New Monte Carlo method to compute the free energy of arbitrary solids. Application to the fcc and hcp phases of hard spheres. J Chem Phys 81(7):3188–3193

[22] Riley KF, Hobson MP, Bence SJ (2006) Mathematical methods for physics and engineering: a comprehensive guide. Cambridge University Press

[23] Kofke D (1993) Gibbs-Duhem integration: A new method for direct evaluation of phase coexistence by molecular simulation. Mol Phys 78(6):1331–1336

[24] Kofke D (1993) Direct evaluation of phase coexistence by molecular simulation via integration along the saturation line. J Chem Phys 98(5):4149–4162

[25] Panagiotopoulos A (1987) Direct determination of phase coexistence properties of fluids by Monte Carlo simulation in a new ensemble. Mol Phys 61(4):813–826

[26] Noro MG, Frenkel D (2000) Extended corresponding-states behavior for particles with variable range attractions. J Chem Phys 113(8):2941–2944

[27] Blote H, Luijten E, Heringa J (1995) Ising universality in three dimensions: A Monte Carlo study. J Phys A 28(22):6289

[28] Fusco D, Charbonneau P (2013) Crystallization of asymmetric patchy models for globular proteins in solution. Phys Rev E 88(1):012721

[29] Tang Z, Zhang Z, Wang Y, Glotzer S, Kotov N (2006) Self-assembly of CdTe nanocrystals into free-floating sheets. Science 314(5797):274–278

[30] Ye X, Chen J, Engel M, Millan J, Li W, Qi L, Xing G, Collins J, Kagan C, Li J, et al. (2013) Competition of shape and interaction patchiness for self-assembling nanoplates. Nat Chem 5(6):466

[31] Glotzer S, Solomon M (2007) Anisotropy of building blocks and their assembly into complex structures. Nat Mater 6(8):557

[32] Bianchi E, Capone B, Kahl G, Likos C (2015) Soft-patchy nanoparticles: Modeling and self-organization. Faraday Discuss 181:123–138

[33] de las Heras D, da Gama M (2016) Temperature (de)activated patchy colloidal particles. J Phys Condens Matter 28(24):244008

[34] Wilber AW, Doye JP, Louis AA, Noya EG, Miller MA, Wong P (2007) Reversible self-assembly of patchy particles into monodisperse icosahedral clusters. J Chem Phys 127(8):08B618

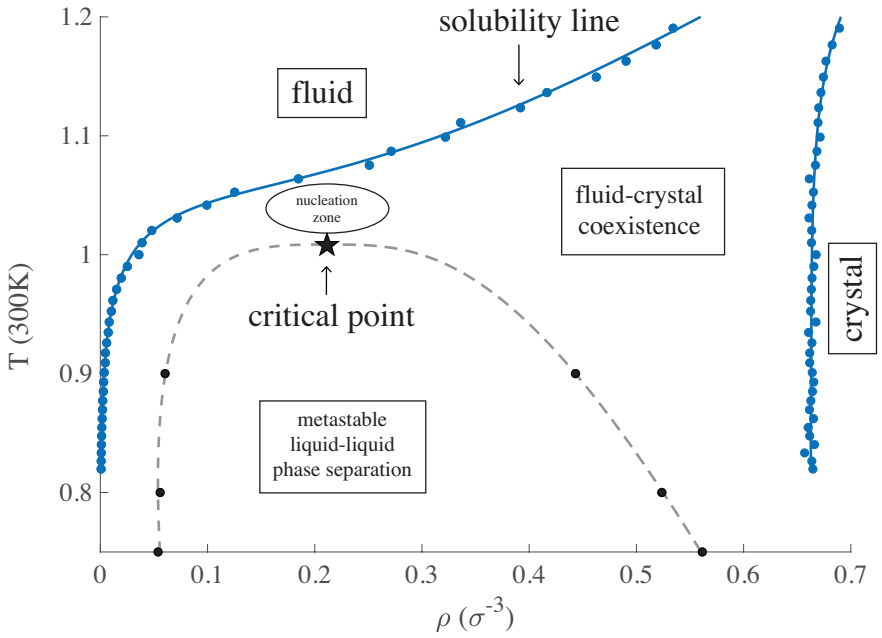

FIGURE 8: The $T - \rho$ phase diagram of a patchy model of rubredoxin [7]. Blue points denote the solubility line and are obtained from Gibbs-Duhem integration. Black points are gas-liquid coexistence points obtained from the Gibbs Ensemble simulations. The fit to the gas-liquid binodal (gray dashed line) terminates at the resulting critical point (black star). Below this line, the system exhibits a metastable liquid-liquid phase coexistence regime in which protein solutions often gel in experiments. This long-lived state often precludes crystallization. The region between the solubility line and T_c is called the nucleation zone because supersaturated solutions in this region are more likely to produce crystals by avoiding gelation.