university of groningen

faculty of science and engineering

Coarse-Grained Force Field Optimisation: A Gaussian Process Regression Approach

Author:
Petra Navarčíková
(s4631447)

Supervisor:
prof. dr. ir. Patrick R. Onck
Second examiner :
Andrea Giuntoli, PhD

Bachelor's Thesis
To fulfill the requirements for the degree of
Bachelor of Science in Physics
at the University of Groningen

July 2, 2024

Contents

Abstract

Coarse-grained molecular dynamics simulations offer insight into fundamental biological processes which are challenging to investigate via all-atom simulations or complex experiments. Liquid-liquid phase separation drives formation of compartments, distinct chemical environments inside the cell, which directly affect cellular function and disease formation. In this work, a one-bead-per-amino-acid (1BPA) model is used to minimise the discrepancy between experimental and calculated radius of gyration (or Stokes radius) for a molecular data set comprising of 189 intrinsically disordered proteins (IDPs). The 1BPA force field is optimised via a supervised machine learning algorithm; Gaussian Process Regression. The GPR model predictions did not match molecular dynamics observations due to the small number of train data points and error metric definition in the target variable. Predictions were encompassed in the GPR confidence interval, which was relatively large due to under-fitting. Nevertheless, the newly determined 1BPA V3.0 shows considerable improvement compared to the previous 1BPA variants, with significant emphasis on aromatic amino acid interactions.

Acknowledgments

I would like to give my warmest thanks to Mark D. Driver, Patrick R. Onck and Andrea Giuntoli for their guidance and support.

Many thanks to Anton van Beek for sharing his expertise regarding Gaussian Processes.

We thank the Center for Information Technology of the University of Groningen for their support and for providing access to the Hábrók high performance computing cluster.

1 Introduction

Cells organise complex biochemical reactions via compartments, distinct chemical environments formed by membrane-bound or membrane-less organelles [1]–[4]. Compartmentalisation allows for efficient function without additional energy input, and has been shown to arise as a result of Liquid-Liquid Phase Separation (LLPS) since 1995 [5]. LLPS is a mechanism during which biomolecules phase separate and form (nano)droplets. Many research groups are currently probing this process to better understand its role in cellular function and disease formation [6], with LLPS being detected in indispensable biological processes including DNA damage repair, mRNA transcription, protein translation or membrane receptor assembly [6]. More importantly, LLPS is considered to be linked to the evolution of life due to its ubiquity in fundamental biological processes [7]. Biomolecular condensates such as Cajal bodies, the nucleolus or stress granules [8], act as membrane-less organelles in physiological function as opposed to isolation by lipid membranes [9]. LLPS is driven via intrinsically disordered regions (IDRs), which are characterised by lack of well-defined ordered structures [10]. Amino acids cause phase separation by weak interactions; π-stacking, electrostatic, cation-π and hydrophobic contacts [6]. As a result, the tendency to phase separate is directly linked to protein sequence and environmental conditions. Specific macromolecule concentration hallmarks LLPS occurrence. Notably, this concentration depends on several biophysical parameters, such as the salt concentration, temperature, and other ions [11].

Experimental investigation of IDRs and phase separation introduces challenges pertaining to sample quality, cost and highly complex experimental protocols. Frequently performed analyses include confocal fluorescence microscopy [12], Raman spectroscopy [13] or small-angle-X-ray-spectroscopy (SAXS) [14]. An efficient alternative to experiments are molecular dynamics simulations, whose main limitations consist of available computational resources. All-atom molecular dynamics simulations have shown considerable sucess, fully reproducing the experimental kinetics and energetics for IDR binding processes [15]. Nevertheless, all-atom simulations are disadvantageous when studying LLPS due to the large time/length scales. Simultaneous simulation of multiple proteins (necessary for LLPS investigation) is beyond the scope of all-atom resolution. This has led to independent development of numerous coarse-grained molecular dynamics models, underlined by the stickers-and-spacers framework proposed by Choi *et al.* [16]. Coarse-grained simulations have shown promising results while being validated against single molecule experimental data such as the radius of gyration [9]. Recently introduced sequence-specific LLPS prediction models include Mpipi (Joseph *et al.* [17]), CALVADOS (Tesei *et al.* [18]) or HPS and its variants (Kapcha and Rossky [19], Regy *et al.* [20], Tesei *et al.* [18]). Machine learning based coarse-grained models are also emerging, with Lotthamer *et al.* developing a deep-learning model ALBATROSS to estimate IDR conformation properties directly from the sequence [21].

The coarse-grained molecular dynamics model of interest in this work is the 1BPA (one-bead-per-amino-acid) model. The model's initial application was centered around the transport phenomena at the nuclear pore complex [22]–[24]. Subsequently, the model was adapted for LLPS prediction via adjusting the characteristic force field parameters by Driver and Onck [25], [26]. The 1BPA coarse-grained model was developed via a modified 8-6 Lennard-Jones potential coupled with bonded interactions (Section 2.1). Molecules of interest are simulated in implicit solvent, which accounts for macromolecule-solvent interactions without introducing solvent molecules to the system. While the latest iterations of the 1BPA model have shown 10% relative error, further improvements are vital for generalised LLPS investigation. Additional accuracy improvements require a broader exploration of the force field parameter space.

In recent years, machine learning (ML) models have shown considerable progress in tackling complex computational tasks with large datasets. The renowned neural network based model AlphaFold predicts protein structures with atomic accuracy even in cases in which no similar structure is known from experiments [27]. Its applicability demonstrated the potential impact of ML approaches to biophysics. Gaussian Progress Regression (GPR) is a ML algorithm widely applied in physical and chemical sciences [28]. Notable examples include force field parametrisation [29], Gaussian Approximation Potential (GAP) [30], crystal structure prediction and dielectric response properties [28]. GPR predictions offer an efficient alternative in exploring the coarse-grained force field parameter space as opposed to a brute-force approach. This choice is further supported by works of John *et al.* [30] and Giuntoli *et al.* [29].

1.1 Project aim

The aim of this project is to optimise the coarse-grained molecular dynamics 1BPA force field. GPR will help identify a new force field, which shows the least discrepancy between experimental and calculated single molecule data.

1.2 Thesis outline

First, the 1BPA framework is explained in section 2. The 1BPA model is followed by a function-space view introduction of the implemented ML algorithm; Gaussian Progress Regression. Next, the methods section 3 explains the molecular dynamics dataset and simulation settings together with the machine learning implementation details and performance. Obtained results are analysed in the results section 4. Then, the discussion section dissects limitations of the model design choices and acquired results. Finally, the main findings are summarised in the conclusion section 5. Supplementary information is provided in the appendix.

2 Theoretical background

2.1 Coarse-grained Molecular Dynamics: 1BPA model

This section contains the theoretical framework of the 1BPA model, developed by Ghavami *et al.* [22] and modified by Jafarinia *et al.* [31]. The 1BPA model is a coarse-grained implicit solvent model initially devised for probing the disordered domain of the nuclear pore complex. It includes both bonded ϕ_b and non-bonded interactions ϕ_{nb}:

$$\phi_{1BPA} = \phi_b + \phi_{nb} \tag{1}$$

Bending and torsion potentials classify as bonded interactions, with hydrophobic/hydrophillic, electrostatic and cation-π interactions belonging to non-bonded interactions. The implicit solvent is modeled by adjusting individual amino acid hydrophobicities. All amino acid beads have an average mass of 120 Da. Explicit hydrogen bonding interactions are not included, as the IDRs are highly flexible and do not form a secondary structure [32]. All numeric values discussed correspond to 1BPA V2.1 [33].

2.1.1 Bonded interactions

Pseudo-bond and pseudo-dihedral angular distribution between neighboring alpha-carbons in the polypeptide chain was modeled using bending and torsion potentials. Potentials were obtained from Ramachandran plots, two-dimensional graphs showing the relation between protein backbone dihedral angle pairs [34]. Assuming that the only degrees of freedom in a protein chain are two dihedral angles ψ and ϕ, the α carbons are connected via pseudo-bonds, with pseudo-bending angle θ and pseudo-dihedral agles α shown in Figure 1.

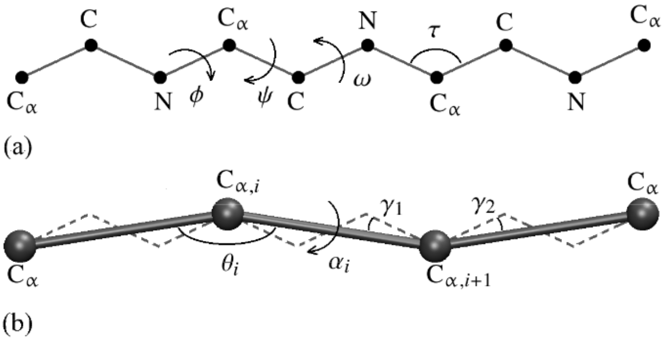

Figure 1: Mapping from all atom dihedral angle ψ and ϕ to coarse grained pseudo-bending angle θ and pseudo-dihedral angle α. (a) All atom model of the protein. (b) Coarse-grained version with pseudo-bonds between α carbons depicted as solid rods. Figure source: [22].

The relationship between (ψ,ϕ) and (θ, α) purely depends on geometry [22]. Unique bending and torsion angles are extracted from the Ramachandran pseudo-angles ψ and ϕ, defined in Appendix A.

Equation 2 summarises the 1BPA bonded interactions ϕ_b.

$$\phi_b = \phi_{bond} + \phi_{bend} + \phi_{torsion} \tag{2}$$

The bonding potential is characterised by a simple harmonic potential $\phi_{bond} = k(r-b)^2$, where the spring constant $k = 8030$kJ mol^{-1}nm^{-2} and pseudo-bond length $b = 0.38$ nm. The main advantage of the 1BPA bonded interactions lies in their residue and sequence specificity. The above defined bonded interactions model (Equation 2) successfully predicts the scaling law for denatured proteins [22].

2.1.2 Non-bonded interactions

1BPA non-bonded interactions account for hydrophobic/hydrophilic, electrostatic and cation-π interactions between amino acid pairs. Solvent polarity and screening of free ions are also included. Non-bonded interactions are defined as a sum of the above defined components:

$$\phi_{nb} = \phi_{hp} + \phi_{cp} + \phi_{el}, \tag{3}$$

Hydrophobic and hydrophilic interactions are modeled via a modified 8-6 Lennard-Jones potential,

$$\phi_{hp} = \begin{cases} \epsilon_{rep}\left(\frac{\sigma}{r}\right)^8 - \epsilon_{ij}\left[\frac{4}{3}\left(\frac{\sigma}{r}\right)^6 - \frac{1}{3}\right], & r \leq \sigma, \\ \left(\epsilon_{rep} - \epsilon_{ij}\right)\left(\frac{\sigma}{r}\right)^8, & \sigma \leq r, \end{cases} \tag{4}$$

where bead radius $\sigma = 0.6$ nm, repulsive energy $\epsilon_{rep} = 5$ kJ mol^{-1}, individual amino acid hydrophobicities ϵ_i and ϵ_j are grouped into $\epsilon_{ij} = \epsilon_{hp}\sqrt{(\epsilon_i\epsilon_j)^\alpha}$, with hydrophobic energy $\epsilon_{hp} = 6.5$ kJ mol^{-1} and scaling factor $\alpha = 0.15$.

Residue specific hydrophobicities were obtained from partition energy measurements [35], and scaled into the range [0,1]. ϵ_{ij} can be interpreted as interaction energy strength between a pair of amino acids. It is proportional to absolute hydrophobic strength between the most hydrophobic amino acids ϵ_{hp} with exponent α determined via calibration against FG-nup Stokes radii [32]. ϵ_{rep} defines the repulsive hydrophilic interaction intensity.

The difference between ϵ_{rep} and ϵ_{ij} determines the behaviour of the potential. The potential is attractive for $\epsilon_{ij} > \epsilon_{rep}$, repulsive for $\epsilon_{ij} < \epsilon_{rep}$ and neutral for $\epsilon_{ij} = \epsilon_{rep}$. Excluded volume effects are modeled in the $r \leq \sigma$ region.

The cation-π interactions are defined by

$$\phi_{cp}(r) = \epsilon_{cp,ij}[3\left(\frac{\sigma}{r}\right)^8 - 4\left(\frac{\sigma}{r}\right)^6], \tag{5}$$

where $\epsilon_{cp,ij}$ is the cation-π interaction energy for an amino acid pair. Cation-pi interactions were first implemented in the 1BPA model by Jafarinia *et al.* [31].

Electrostatic interaction between charged amino acids (R,K,D,E) are modeled via a modified Coulomb's law with Debye-Huckel screening,

$$\phi_{el} = \frac{q_i q_j}{4\pi\epsilon_0\epsilon_r(r)r}e^{(-\kappa r)}, \tag{6}$$

where q_i, q_j are the amino acid pair charges, κ is the Debye screening coefficient, with ϵ_r and ϵ_0 representing the permittivity of water and free space, respectively. Explicit expressions for κ and ϵ_r are described in Appendix A.

2.2 Gaussian Process Regression

This subsection contains the definition of Gaussian Process Regression via the function-space view coupled with application methods for 1BPA force field optimisation. Derivations were adapted from the following literature sources: Rasmussen and Williams [36], Deringer *et al.* [28], with supplementary information from Cheng and Wang [37] and Huang *et al.* [38].

Gaussian Process Regression (GPR) is a supervised machine learning method used for stochastic regression problems. Simple regression models such as Linear Regression require an explicit form of the fitted function, whereas Gaussian Progress Regression is non-parametric (not constrained to a specific function or number of parameters). This property is advantageous for processes without *apriori* known functional form.

"A Gaussian process is a collection of random variables, any finite number of which have a joint Gaussian distribution." [36] Univariate Gaussian distributions are characterised by a single mean μ and variance σ^2 e.g., $Z \sim \mathcal{N}(\mu, \sigma^2)$. For a joint distribution of random variables, the different mean and variance values are expressed as vectors: $\mathbf{Z} \sim \mathcal{N}(\mu, \sigma^{\mathbf{2}})$ [39]. Similarly, a Gaussian Process is defined by its mean and covariance function, where the covariance is defined between two input vectors $\mathbf{x}$ and $\mathbf{x}'$:

$$\begin{aligned} \mathrm{m}(\mathbf{x}) &= \mathbb{E}[f(\mathbf{x})] \\ \mathrm{cov}(\mathbf{x}, \mathbf{x}') &= \mathbb{E}[(f(\mathbf{x}) - \mathrm{m}(\mathbf{x}))(f(\mathbf{x}') - \mathrm{m}(\mathbf{x}'))], \end{aligned} \tag{7}$$

with the expected value $\mathbb{E}$ and the function value $f(\mathbf{x})$.

Random variables of a Gaussian process are the function values $f(\mathbf{x})$ at a certain location $\mathbf{x}$:

$$f(\mathbf{x}) \sim \mathcal{GP}(\mathrm{m}(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')), \tag{8}$$

with kernel function $k(\mathbf{x}, \mathbf{x}')$, whose value for identical inputs $(\mathbf{x} = \mathbf{x}')$ corresponds to a variance. Although the functional form f is not known, the data consists of observations which are mapped via $f(\mathbf{x})$ from a multi-dimensional input to a scalar output [28]:

$$\mathcal{D} = \{\mathbf{x}_n; y_n\}_{n=1}^{N}, \tag{9}$$

with D representing the set of all GPR datapoints, y_n representing the target variable value for an input point $\mathbf{x}_n$ and total data set size N. The goal of GPR is to predict function values at an arbitrary point in space, having learnt the observed dataset. Figure 2 depicts a one-dimensional GPR example. The GPR predictor, which approximates the unknown underlying function f, is a sum of H (intentionally undefined) basis functions ϕ_h with corresponding weights w_h:

$$f(\mathbf{x}) = \sum_{h}^{H} w_h \phi_h(\mathbf{x}). \tag{10}$$

The weights are input point independent, identically distributed random variables drawn from Gaussian distributions with zero mean and variance σ_w^2:

$$P(w_h) \sim \mathcal{N}(0, \sigma_w^2), \tag{11}$$

leading not to a single estimate of $f(\mathbf{x})$ but to a distribution of estimators, which corresponds to a Gaussian prior probability distribution and is commonly called a Gaussian process (GP)

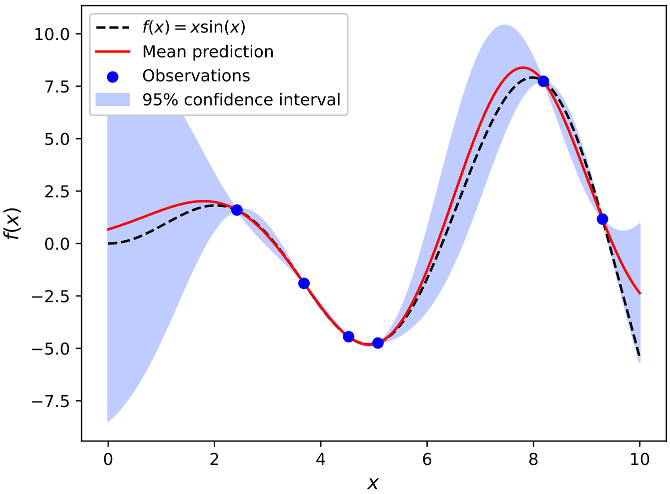

Figure 2: Example application of Gaussian Process Regression in 1D. True function $f(x)$ (black) is approximated by mean prediction (red), predicted on input x (dark blue) without noise. The 95% confidence interval is depicted as a shaded blue region. The mean prediction matches the true function at all data points due to noise free input. The confidence interval increases further away from available observations. Inspired by [40].

[28].
The covariance between two estimator values at input points $\mathbf{x}$ and $\mathbf{x}'$ is expressed as a sum of basis functions ϕ_h [1]:

$$\mathrm{cov}(f(\mathbf{x}), f(\mathbf{x}')) = \sigma_w^2 \sum_h^H \phi_h(\mathbf{x})\phi_h(\mathbf{x}') \tag{12}$$

The covariance is directly related to the kernel k, which is a similarity measure defining models of covariance. Specifying the kernel without defining the H basis functions is called the 'kernel trick' [28], which allows for computing k in implicit feature space.

$$k(\mathbf{x}, \mathbf{x}') \equiv \sigma_w^2 \sum_h^H \phi_h(\mathbf{x})\phi_h(\mathbf{x}') \tag{13}$$

2.2.1 Squared Exponential Kernel

"Kernel is a similarity measure between two data points, commonly denoted $k(\mathbf{x}, \mathbf{x}')$" [28]. $k(\mathbf{x}, \mathbf{x}')$ specifies the covariance between $f(\mathbf{x})$ and $f(\mathbf{x}')$, to describe the degree of statistical correlation between them [28]. In other words, kernel limits the function character to be either smooth, differentiable, periodic etc. Success of a GPR predictor without a large number of data points depends on kernel choice to a large extent. Ill-suited kernel type or parameters, denoted hyper-parameters, will considerably slow down the convergence as a function of data points. The most widely used kernel for Gaussian Progress Regression is the Squared Exponential, also

[1] The corresponding weights w_h have been integrated over, Deringer *et al.* [28] contains the full derivation.

known as the Radial Basis Function[41]:

$$k(\mathbf{x}, \mathbf{x}') = \sigma^2 \exp(-\frac{d\left(x_i, x_j\right)^2}{2l^2}). \tag{14}$$

The above equation defines the covariance between two input points $\mathbf{x}$ and $\mathbf{x}'$ as a function of their Euclidean distance $d(\mathbf{x}, \mathbf{x}')$ and the kernel hyper-parameters l^2 and σ^2, which correspond to the length scale and variance, respectively. The length scale is informally defined as the distance you have to move in the input space before the function value significantly changes [36]. σ^2 is a positive scalar, which sets the output variance [30]. Hyper-parameter choice and optimisation is discussed in Section 2.2.3.

2.2.2 From Observations to Predictions

Physical measurements always have an associated error. Observations can include noise ϵ, modeled as independent Gaussian noise with variance σ_n^2 [36].

$$y = f\left(\mathbf{x}\right) + \epsilon \tag{15}$$

The covariance between two observation values y and y' then becomes [28]

$$\operatorname{cov}(y, y') = k(\mathbf{x}, \mathbf{x}') + \sigma_n^2 \delta_{ij}, \tag{16}$$

where the noise variance σ_n^2 is added via the Kronecker delta δ_{ij}. As a result, the covariance between two distinct observations $(y \neq y')$ is simply the kernel function value between the corresponding input points $\mathbf{x}$ and $\mathbf{x}'$. All observations $\mathbf{y} = (y_1, ..., y_N)$ follow a multivariate Gaussian distribution with zero mean and covariance $K_{NN} + \sigma_n^2 I$. The covariance is defined by a $N \times N$ matrix K, which contains kernel results between all input point pairs together with the identity matrix I scaled by the noise variance i.e. generalised case of Equation 16:

$$\mathbf{y} \sim \mathcal{N}(0, K + \sigma_n^2 I). \tag{17}$$

The predictor is derived from the joint probability distribution of the observations $\mathbf{y}$ and the predicted function values $\mathbf{f}_*$ at new locations X_* [28].

$$\left[\begin{array}{c} \mathbf{y} \\ \mathbf{f}_* \end{array}\right] \sim \mathcal{N}\left(\left[\begin{array}{c} \mu(X) \\ \mu\left(X_*\right) \end{array}\right], \left[\begin{array}{cc} K(X, X) + \sigma_n^2 I & K\left(X_*, X\right)^{\mathrm{T}} \\ K\left(X_*, X\right) & K\left(X_*, X_*\right) \end{array}\right]\right) \tag{18}$$

For M new locations, the covariance matrix $K(X_*, X)$ has dimensions of $M \times N$. Conditioning the predicted function values $\mathbf{f}_*$ on the observed data $\mathbf{y}$ assuming zero mean leads to the GP predictive distribution [37]:

$$P\left(\mathbf{f}_* \mid X, \boldsymbol{y}, X_*\right) = \mathcal{N}(\hat{\mu}, \hat{\Sigma}) \tag{19}$$

Explicit definitions of $\hat{\mu}$ and $\hat{\Sigma}$ depend on Equation 18 and can be found in Appendix B. Appendix C contains a single prediction example for further explanation.

2.2.3 Hyper-parameter Optimisation

Predictor performance is largely influenced by hyper-parameter choice. Hyper-parameters of the above defined GPR predictor grouped in a vector $\boldsymbol{\theta}$ consist of kernel variance σ^2 and kernel lengths scales l^2 (one per feature)[38]:

$$\boldsymbol{\theta} = \{\sigma^2, l^2, ...\} \tag{20}$$

Optimal hyper-parameter set is determined via maximising the Log Marginal Likelihood (LML), also known as type II Maximum Likelihood (ML-II).

Marginal likelihood, denoted $P(\mathbf{y}|X)$, is the likelihood $P(\mathbf{y}|\mathbf{f}, X)$ times the prior $P(\mathbf{f}|X)$ integrated over the whole parameter space (all possible values of $\mathbf{f}$) [36].

$$P(\mathbf{y}|X, \boldsymbol{\theta}) = \int P(\mathbf{y}|\mathbf{f}, X)P(\mathbf{f}|X)d\mathbf{f} \tag{21}$$

Evaluating the integral after substituting $\mathcal{N}(\mathbf{0}, K)$ for the prior and $\mathcal{N}(\mathbf{f}, \sigma_n^2 I)$ for the likelihood yields,

$$\text{log}P(\mathbf{y}|X, \boldsymbol{\theta}) = -\frac{1}{2}\mathbf{y}^T(K + \sigma_n^2 I)^{-1}\mathbf{y} - \frac{1}{2}\text{log}|K + \sigma_n^2 I| - \frac{n}{2}\text{log}2\pi \tag{22}$$

which implicitly depends on $\boldsymbol{\theta}$ (included in the kernel values in matrix K). Hyper-parameters are updated according to the conjugate gradient method [38], which updates the individual hyper-parameter values such that the log marginal likelihood attains a maximum.

$$\frac{\partial}{\partial\theta_j}\text{log}P(\mathbf{y}|X, \boldsymbol{\theta}) = \frac{1}{2}\text{tr}\Big((\boldsymbol{\alpha}\boldsymbol{\alpha}^{-1} - (K + \sigma_n^2 I)^{-1})\frac{\partial(K + \sigma_n^2 I)}{\partial\theta_j}\Big) \tag{23}$$

Equation 23 defines the log marginal likelihood gradient for all hyper-parameters θ_i, with $\boldsymbol{\alpha}$ representing the covariance transformation of all observations $(K + \sigma_n^2 I)^{-1}\mathbf{y}$.

Log marginal likelihood can attain multiple stationary points in higher dimenions. Equation 23 simply converges to a local maximum/minimum closest to the initialisation. This poses a problem when trying to determine the optimal set of hyper-parameters. The widely adapted solution is a somewhat heuristic method; simply initialising at different points in the hyper-parameter space multiple times and choosing the best estimate. As such, maximising the log marginal likelihood can be likened to cross-validation; both approaches serve to evaluate multiple models (set of hyper-parameters) with the goal of finding the optimal solution.

3 Methods

3.1 Molecular Dataset

Dataset choice is crucial when validating any simulations against experimental work. The dataset comprises 189 intrinsically disordered proteins, selected according to available experimental data of interest: radius of gyration (R_g) or hydrodynamic radius (R_h). Radius of gyration of a particle is the root-mean-square distance of all particles from their center of gravity [42]. Another indicator of protein volume/compaction is the hydrodynamic radius defined as the effective radius of a molecule in a solution measured by assuming that it is a body moving through the solution and resisted by the solution's viscosity [43]. Gathered experimental data consists of 155 R_g instances and 34 R_h instances.

Not all literature sources quoted an experimental error. 97% of R_h molecules and 34% of R_g molecules reported an experimental error. A median value of the available experimental error $\sigma R_{g/h_{exp}}$ was used for analysing the R_h and R_g molecules: 0.03 nm and 0.09 nm, respectively. Initial 1BPA molecular data set focused on 20 FG-nup molecules [44], however the molecular data set has been considerably increased (189) resulting in lower bias towards previous applications. The molecular dataset can be considered fairly diverse, as supported by the sequence length analysis below. Literature sources for the molecular dataset can be found in Appendix D.

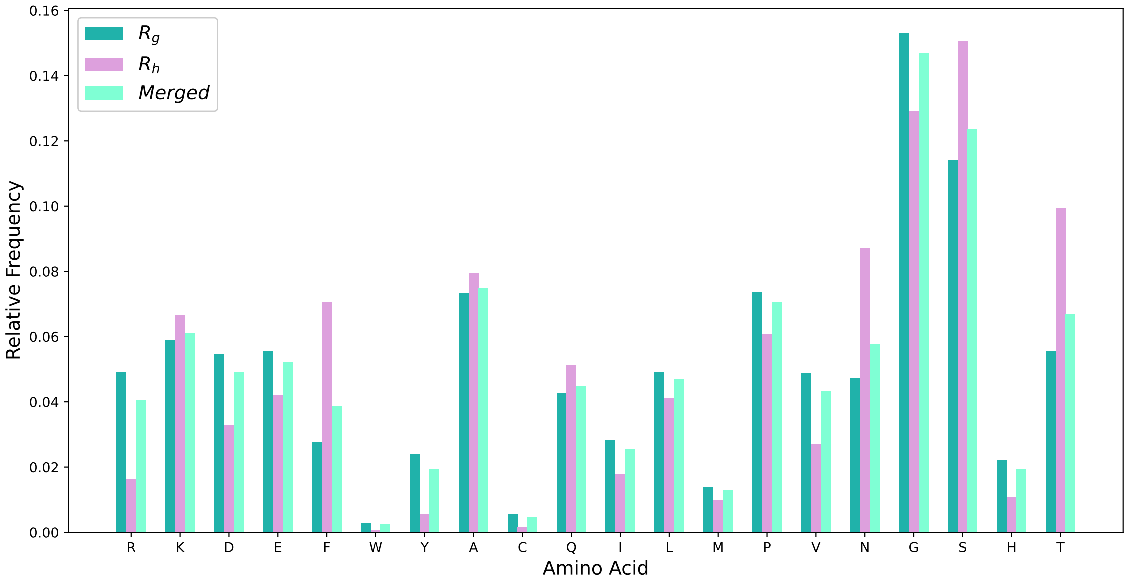

Figure 3: Amino acid sequence composition for all 189 molecules. Relative frequency is defined by summing individual amino acids from the full dataset and dividing by the total number of residues. Evaluated for three groups: R_g data, R_h data and Merged (full dataset).

Sequence composition can be found in Figure 3. All amino acids are represented, with a majority exhibiting relative frequency between 2.5-5%. Relative frequency refers to the fraction of amino acid in the full molecular dataset. Glycine (G) and Serine (S) are the most frequent, each representing around 12.5% of all amino acid residues. In contrast, Tryptophan (W) and Cysteine (C) are underrepresented. The optimised force field will ultimately be biased by this molecular sequence composition.

The longest chain comprises 625 residues, with the shortest chain of 15 residues. Median protein sequence length is 137 residues, mean protein sequence length is 185 residues, with a standard deviation from the mean of 85 residues. Approximately 60% of proteins falls into 100-270 residues range. Such broad sequence length distribution is desirable for increasing the applicability of the 1BPA force field.

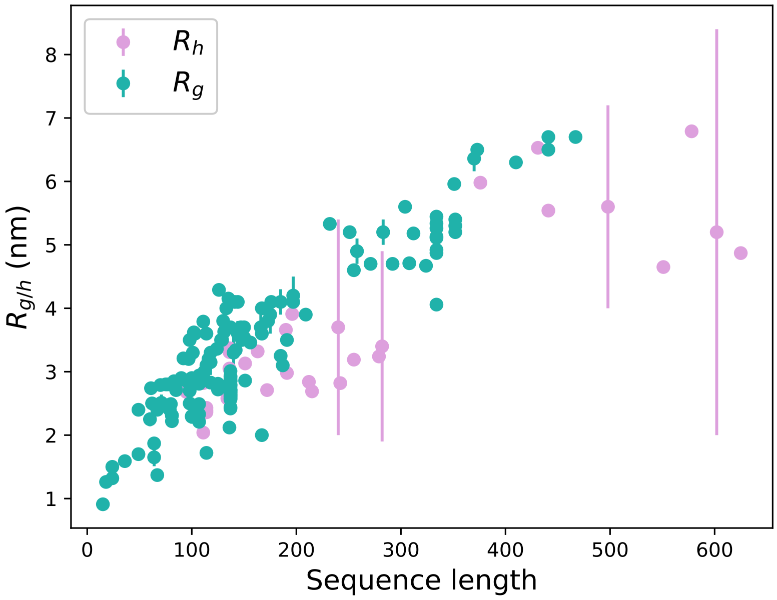

Figure 4: Experimental radius of gyration/hydrodynamic radius as a function of protein sequence length. Evaluated for two groups: R_g data and R_h data.

3.2 Molecular Dynamics Simulations and Analysis

Single molecule molecular dynamics (MD) simulations were executed in GROMACS 2019.6 [45]. Simulation settings were kept consistent for all force field parameter sets. Each MD simulation consisted of two main steps: Energy Minimisation (EM) and Production (MD). The molecule is initially relaxed to energetically favorable conformation (EM) and then simulated for an extended period of time to obtain statistically significant information (MD).

All simulations were performed at 300K and ion concentration of 150 mM. Langevin dynamics was used to propagate the system in time via steepest descent minimisation. The time step was set to 0.02 ps. Energy Minimisation consisted of 500000 steps or until the maximum force decreased below 1 kJ mol^{-1}nm^{-1}. The Production was carried out for 2.501 μs, with the initial 1 ns used for system equilibration. The simulation trajectory was recorded for 2.5 μs by assuming the canonical (NVT) ensemble, corresponding to constant number of particles, volume and temperature.

Trajectory analysis was implemented via MDAnalysis for R_g and R_h calculation [46]. Both properties were calculated per frame, with the corresponding mean $R_{g/h}$ and standard deviation $\sigma R_{g/h_{calc}}$ (for all MD frames) compared to the experimental value. The radius of gyration is defined as,

$$R_g = \sqrt{\frac{\sum_{i=1}^{n} m_i \left(\mathbf{r}_i - \mathbf{r}_{\mathrm{CM}}\right)^2}{\sum_{i=1}^{n} m_i}} \tag{24}$$

where $\mathbf{r}_i$ is the position vector of individual amino acid residue with mass m_i and $\mathbf{r}_{\mathrm{CM}}$ is the position vector of the protein center of mass. The 1BPA model approximates all amino acids to have identical mass, so the m_i factor cancels out from Equation 24. The number of amino acid beads is protein-specific and denoted by n.

The translational hydrodynamic radius R_h was calculated via the Hullrad method [47]. The protein volume is calculated with a convex hull, from which the hydrodynamic radius can be determined. A convex hull is a boundary of the smallest convex set enclosing all the points in a set [48]. The Hullrad method estimates the protein surface area from individual amino acid bead coordinates, which allows for computing the protein volume. Having obtained the volume V_{TH}, the translational hydrodynamic radius is defined as

$$R_h = F_T \times \sqrt[3]{\frac{3V_{TH}}{4\pi}}, \tag{25}$$

with the translational shape factor F_T. Further details regarding the Hullrad method can be found in the work of Fleming and Fleming [47].

3.3 Adapting Gaussian Progress Regression to Coarse-Grained Force Field Optimisation

This section contains details of adapting the Gaussian Process Regression model to 1BPA force field optimisation within the scikit-learn library [41]. The coarse-grained force field is characterised by a set of equations and constants explained in Section 2.1. A parameter set $\mathbf{x}$ refers to a unique vector of force field constants, which were varied in a physically feasible range.

$$\mathbf{x} = \Big[\alpha, \quad \epsilon_i, \quad \epsilon_{aromatic}, \quad \Delta_{catpi}\Big] \tag{26}$$

α is a scaling factor for individual amino acid hydrophobicities ϵ_i, $\epsilon_{aromatic}$ is the aromatic-aromatic amino acid interaction energy and Δ_{catpi} represents the shift in cation-π interaction energies wrt. 1BPA V1. The force field optimisation goal is to find a parameter set that results in the lowest discrepancy between experimental and calculated R_g (or R_h) for all 189 molecules in the dataset. Gaussian Process Regression was implemented to probe the relation between the parameter set values and error over all molecules.

Each parameter set was validated against R_g (or R_h), denoted by $R_{g/h}$ in the subsequent sections, with the following average relative error definition:

$$\delta R_{g/h} = \frac{1}{M} \sum_{i=1}^{M} \frac{\mid \left(R_{g/h_{calc}} - R_{g/h_{exp}}\right) \mid}{R_{g/h_{calc}}}, \tag{27}$$

with the total number of molecules M=189. The calculated and experimental $R_{g/h}$ are denoted $R_{g/h_{calc}}$ and $R_{g/h_{exp}}$, respectively. However, this metric does not account for experimental error, $\sigma R_{g/h_{exp}}$, or the calculated error $\sigma R_{g/h_{calc}}$. Mean absolute error (in the average relative error) was introduced to account for this dependency.

$$\Delta\delta R_{g/h} = \sqrt{\left(\frac{\sigma R_{g/h_{calc}}}{R_{g/h_{calc}}}\right)^2 + \left(\frac{\sigma R_{g/h_{exp}}}{R_{g/h_{exp}}}\right)^2} \times \frac{1}{M}\sum_{i=1}^{M} |\left(R_{g/h_{calc}} - R_{g/h_{exp}}\right)| \tag{28}$$

Due to the lack of data points prior to feature selection (essentially only 1BPA 2.1), the varied parameters were chosen according to physical intuition. Feature selection refers to the process of identifying force field parameters of interest, which were subsequently used to train the GPR model. Selected parameters define the input matrix X(Equation 29) which has dimensions of $\mathrm{n_{samples}} \times \mathrm{n_{features}}$: (N × 26). See Table 2 for a list of all 26 force field parameters.

$$X = \begin{bmatrix} \mathbf{x_1} \\ \mathbf{x_2} \\ \vdots \\ \mathbf{x_N} \end{bmatrix} \tag{29}$$

The number of parameter sets N increased throughout the implementation. Appendix E contains information regarding continuous data acquisition for the GPR model. Individual target variable $\mathbf{y}$ has 2 dimensions; including both average relative error and mean absolute error.

$$\mathbf{y} = \begin{bmatrix} \delta R_{g/h}, & \Delta\delta R_{g/h} \end{bmatrix} \tag{30}$$

All observations are grouped into a matrix Y with $(N \times 2)$ dimensions.

$$Y = \begin{bmatrix} \mathbf{y_1} \\ \mathbf{y_2} \\ \vdots \\ \mathbf{y_N} \end{bmatrix} \tag{31}$$

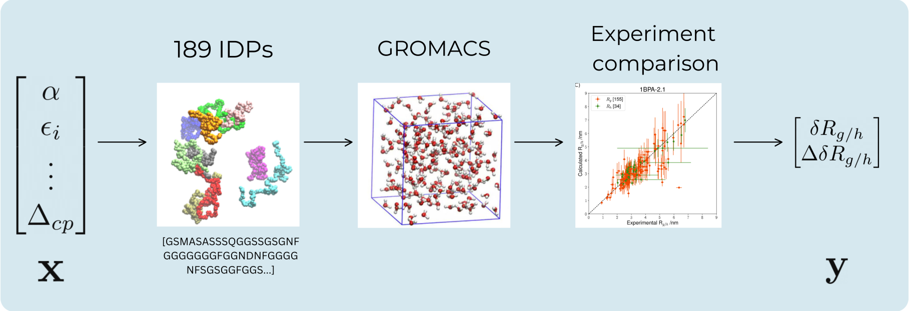
$\begin{bmatrix} \alpha \\ \epsilon_i \\ \vdots \\ \Delta_{cp} \end{bmatrix}$
$\mathbf{x}$
189 IDPs
[GSMASASSSQGGSSGSGNF GGGGGGGFGGNDNFGGGG NFSGSGGFGGS...]
GROMACS
Experiment comparison
1BPA-2.1
R_g [155]
R_h [34]
Calculated $R_{g/h}$ /nm
Experimental $R_{g/h}$ /nm
$\begin{bmatrix} \delta R_{g/h} \\ \Delta\delta R_{g/h} \end{bmatrix}$
$\mathbf{y}$

Figure 5: Acquisition of 1 GPR data point. A force field parameter set $\mathbf{x}$ is used to simulate 189 IDPs via GROMACS. Calculated $R_{g/h}$ are compared with the experimental values. The average relative error and mean absolute error are computed and stored in the target variable $\mathbf{y}$.

One GPR data point consists of a force field parameter set $\mathbf{x}$ and the corresponding average relative error with mean absolute error grouped into $\mathbf{y}$, (see Figure 5).

Both X and Y underwent pre-processing. Input features X were scaled to account for notable differences in magnitude, while the target variable Y was standardised in accordance with the assumption in Equation 17. The scaler of choice was Standard from the scikit-learn library [41], which scaled the input matrix X feature-wise to have zero mean and unit variance.

The kernel was defined by Squared Exponential, (scikit-learn.gaussian_process.kernels.RBF()), with an anisotropic length scale providing a distinct length scale l^2 per feature and a scaling factor σ^2. Noise in individual observations (mean absolute error) was passed as an addition to the diagonal elements in the kernel matrix. Hyper-parameters (see Equation 20) were optimised via maximising the Log Marginal Likelihood with 100 random initialisations.

The available observations, X and Y, were split into a train and test set in a 80:20 ratio. The GPR model was trained on the train set and evaluated on the test set. The following equation assessed model performance on the test set i.e. quantifying how much the GPR model predictions y_{predict} differ from the observation values y (corresponding to $\delta R_{g/h}$).

$$R^2 = 1 - \frac{\Sigma(y - y_{\text{predict}})^2}{\Sigma(y - \overline{y})^2} \tag{32}$$

The summation is over the test set. Having defined and trained the GPR model, the Latin HyperCube Sampling (LHS) method [49] was used to define a set of prediction input points X^* without a corresponding target value $\mathbf{y}$. The GPR model then predicted the target value for all elements of X^*. Predictions with the lowest target values were selected as GROMACS candidates and added to the total number of samples N. The candidates were simulated in GROMACS to assess whether the predictions matched the observed values (or not). The Root-Mean-Square-Error (RMSE) was computed to quantify the agreement between observations y_i and predictions $\hat{y}_i$ for m batch samples.

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{m}(y_i - \hat{y}_i)^2}{m}} \tag{33}$$

This procedure was repeated numerous times, in order to improve the GPR model and investigate as much of the force field parameter space as possible. The input matrix X was subsequently augmented with the new candidates and the GPR model was retrained. This iteration of identifying parameter sets of interest, simulating them in GROMACS and retraining the GPR model with increased input was repeated eight times.

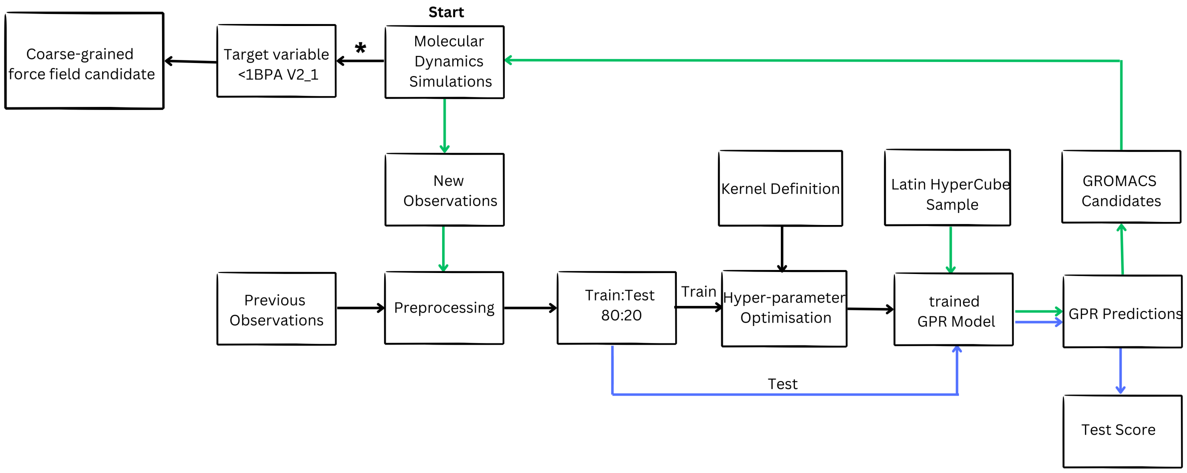

Figure 6: Force field optimisation process flow. New observations are obtained via GROMACS Molecular Dynamics simulations. They are combined with previously simulated force fields, scaled with a Standard scaler and split into train (80%) and test (20%) sets. The GPR model is defined with the Squared Exponential kernel and trained on the train set. Hyper-parameters are optimised via maximising the Log Marginal Likelihood. The GPR model is used for predicting expected values of the test set and Latin HyperCube Sample (LHS). Model performance on the test set is evaluated via the Score metric (Equation 32). A subset of LHS predictions is selected to be GROMACS simulation candidates. The observed target value is compared to 1BPA V2.1 and selected as a coarse-grained force field candidate.* (The iteration loop was stopped after acquiring 68 force fields with $\delta R_{g/h}$ lower than 1BPA V2.1 due to project timescale.)

The process flow above resulted in N=265 coarse-grained force field parameter sets. The force field corresponding to the minimum observed target value was selected as the best candidate. The newly determined hydrophobicity scale was compared to 87 experimental hydrophobcity scales collected by Simm *et al.* [50] combined with previous 1BPA versions. Agglomerative clustering analysis was adapted from Tesei *et al.* [18]. This method was conducted to validate the physical feasibility of the optimised force field.

Sensitivity analysis was implemented to investigate the relative 'importance' of individual force field parameters. Three approaches were evaluated; Linear Regression, GPR length scales and Sobol indices.

3.4 Gaussian Process Regression Performance

GPR data points were simulated in batches (Appendix E), with each new batch addition increasing the total number of data points presented to the model. The GPR model improvement is analysed via the Log Marginal Likelihood and measure of agreement between observations and model predictions.

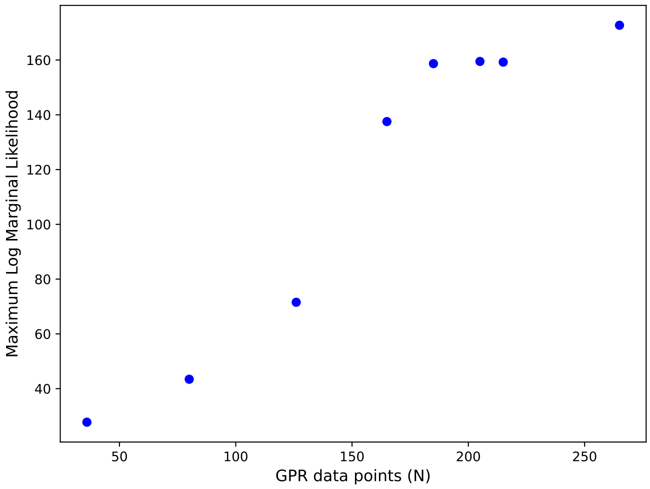

Figure 7: Maximum Log Marginal Likelihood (LML) as a function of GPR data points (N). Each dark blue mark corresponds to the maximised LML (with 100 random initialisations) for a total number of GPR data points (train and test).

The Log Marginal Likelihood increased together with the number of GPR data points as expected. More observations sample the force field parameter space with higher frequency, which allows for identifying better local minima. Each LML value is the result of 100 random initialisations. The largest obtained LML value does not seem to converge, although early indications are present around N=200. GPR hyper-parameters were optimised via maximising the Log Marginal Likelihood. Their values can be found in Appendix F. Most of the hyper-parameter varied drastically, with multiple reaching the upper bound of 100000. Consequently, the latest set of hyper-parameters might change with the addition of more GPR data points. Nevertheless, the Log Marginal Likelihood shows continuous improvement.

The GPR model performance was assessed throughout data acquisition. Initial batches were defined similarly to 1BPA V2.1, with later batches defined via Latin HyperCube Sampling (LHS). Consult Appendix E for detailed batch overview. Later batches, namely Batch 5(20 samples), Batch 6(10 samples) and Batch 7(50 samples) were selected based on GPR predictions. The figures below show a direct comparison between GPR predictions and GROMACS observations. Figure 8 shows many observations outside of the GPR confidence interval. The confidence interval increases as the value of the target variable $\delta R_{g/h}$ increases because of less dense sampling in that region. Many more data points were evaluated for $\delta R_{g/h}$ <0.2, since the ultimate goal is to find the lowest possible $\delta R_{g/h}$.

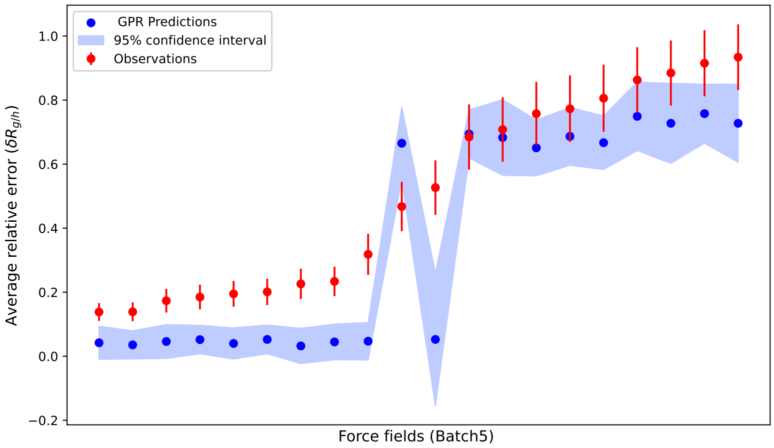

Figure 8: Batch 5: GPR predictions (dark blue) vs Observations (red). GPR confidence interval is shown in light blue. Predictions were trained on 167 data points.

Predictions of Batch 6 (Figure 9) are more accurate, with all observations included in the GPR confidence interval. Nevertheless, the GPR predictions are still slightly larger. It is important to note the difference between Batch 6 and other batches. A GPR model trained on solely 60 LHS defined data points was used for predicting Batch 6. Yet, Batch 6 predictions are more accurate (less discrepancy between predictions and observations) compared to Batch 5, which were trained on 167 GPR data points. This difference in performance and number of GPR data points shows that Latin HyperCube Sampling is a much more efficient method for investigating highly dimensional spaces.

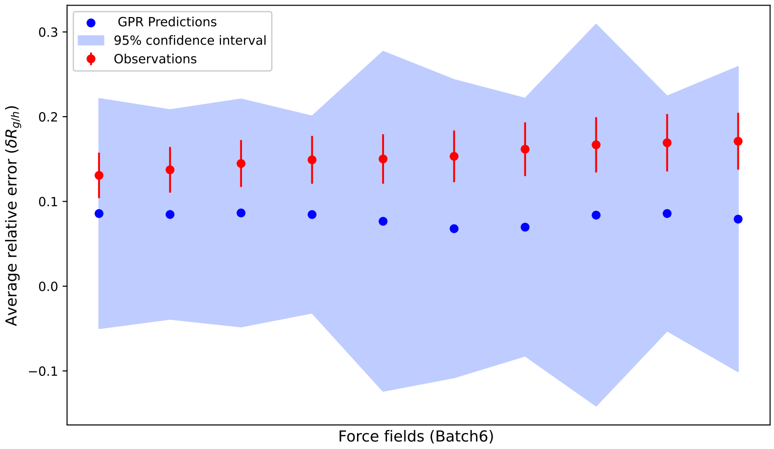

Figure 9: Batch 6: GPR predictions (dark blue) vs Observations (red). GPR confidence interval is shown in light blue. Predictions were trained on 60 LHS defined data points.

Similarly to Batch 6, Batch 7 observations (Figure 10) fall within the GPR confidence interval.

Yet, the GPR predictions are still not as accurate as desired. The confidence interval stays relatively large, even after introducing 215 observations. The GPR model underfitted due to small number of data points in a highly dimensional space and large error ($\Delta\delta R_{g/h}$) included in the kernel diagonal.

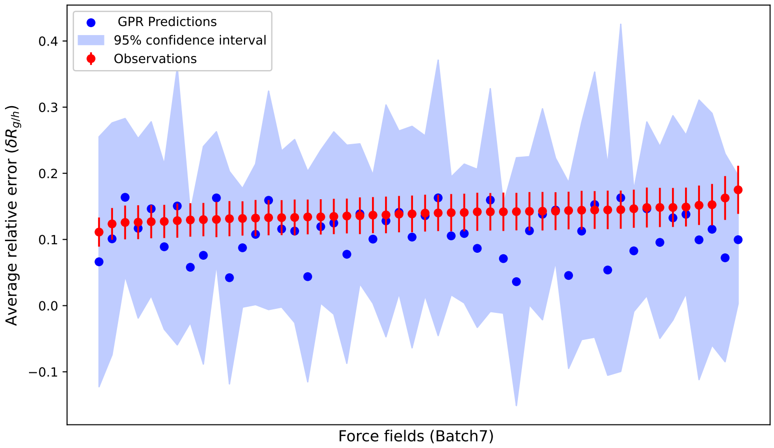

Figure 10: Batch 7: GPR predictions (dark blue) vs Observations (red). GPR confidence interval is shown in light blue. Predictions were trained on 215 data points.

Prediction accuracy between the later batches is quantified in Table 1. The Root-Mean-Square-Error (RMSE) decreased for later batches. Its rather small absolute value originates from $\delta R_{g/h}$ magnitude. It is important to note that RMSE is calculated without including the error in observations or predictions (Equation 33). This metric is batch size independent.

Predictions	**RMSE**
Batch5	0.181
Batch6	0.075
Batch7	0.047

Table 1: Root-Mean-Square-Error (RMSE) quantifying the discrepancy between GPR predictions and observations per batch.

4 Results

4.1 Optimised Coarse-Grained Force Field: 1BPA V3.0

The GPR model was used to investigate 265 coarse-grained force fields, out of which 68 had a lower average relative error $\delta R_{g/h}$ than V2.1. The lowest observed $\delta R_{g/h}$ was selected as the optimised coarse-grained force field. Figure 13 compares the chosen force field 1BPA V3.0 with the previous versions.

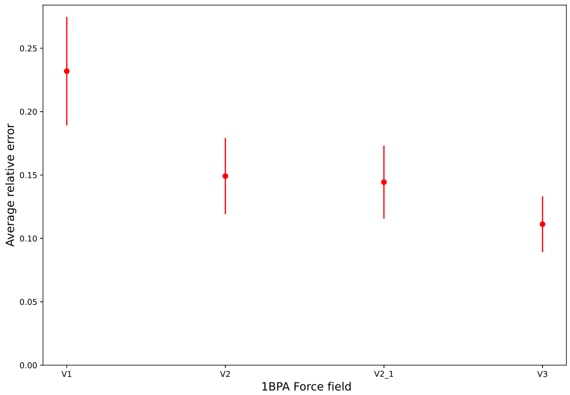

Figure 11: Comparison of the optimised force field; 1BPA V3.0 with previous versions. The plotted metric is the average relative error $\delta R_{g/h}$.

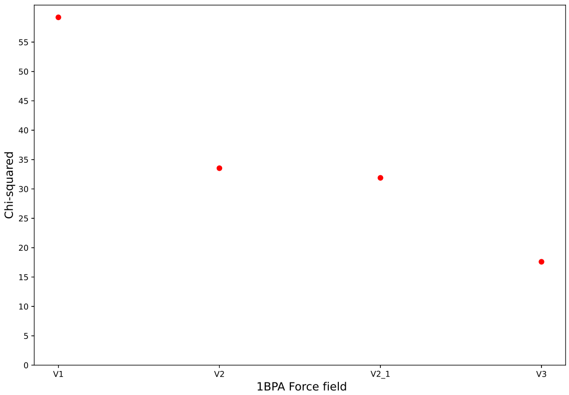

Figure 12: Comparison of the optimised force field; 1BPA V3.0 with previous versions. The plotted metric is the chi-squared value χ^2.

Figure 13: 1BPA force fields comparison

1BPA V3.0 has an average relative error $\delta R_{g/h} = 0.11 \pm 0.02$, as opposed to $\delta R_{g/h} = 0.14 \pm 0.03$ for 1BPA V2.1. The discrepancy between calculated and experimental $R_{g/h}$ has decreased by approximately 3%. The chi-squared metric shows even larger improvement: 17.60(V3.0) compared to 31.89(V2.1). This difference stems from $\delta R_{g/h}$ and χ^2 definitions. While $\delta R_{g/h}$ is defined by taking the average of single molecule errors, χ^2 is the sum of their squares. Consequently, χ^2 is prone to outliers and its magnitude will change based on the number of molecules in the data set. The average relative error is more robust and preferred for further investigation.

Although 1BPA V3.0 results in the lowest $\delta R_{g/h}$, it must be physically feasible before its application to more complex molecular dynamics investigation. Table 2 serves as direct comparison between V3.0 and V2.1.

Parameter	1BPA 2.1	1BPA 3.0	Δ
α	0.15	0.17	0.02
A	0.7	0.67	-0.03
C	0.68	0.77	0.09
D	0.005	0.03	0.025
E	0.005	0.01	0.005
F	0.8	0.58	-0.22
G	0.45	0.5	0.05
H	0.56	0.68	0.12
I	0.98	0.67	-0.31
K	0.01	0.13	0.12
L	1	0.78	-0.22
M	0.78	0.91	0.13
N	0.28	0.58	0.3
P	0.67	0.62	-0.05
Q	0.4	0.44	0.04
R	0.01	0.06	0.05
S	0.42	0.46	0.04
T	0.43	0.35	-0.08
V	0.94	0.8	-0.14
W	0.8	0.65	-0.15
Y	0.55	0.85	0.3
F-F	7.5	7.46	-0.04
Y-Y	8	6.38	-1.62
W-W	8.5	7.44	-1.06
H-H	6.5	7.1	0.6
Δ_{catpi}	0	-1.9	-1.9

Table 2: Force field parameter comparison between 1BPA V2.1 and V3.0. Δ quantifies absolute change in all force field parameters between V2.1 and V3.0.

The scaling factor α has increased to 0.17. This effectively shifts the boundary between hydrophillic and hydrophobic amino acids. The hydrophobicity scale includes numerous changes. Charged amino acids are still on the lower end, but the most hydrophillic amino acid is now Glutamic acid (E). All charged AA hydrophobicities have increased and their values differ more compared to V2.1. Other notable change occurred in the aromatic group. The relative ordering of aromatic-aromatic interactions has now become (Y-Y,H-H,W-W,F-F). Phenylalanine (F) has shifted down drastically, it now falls within the polar hydrophobicity range. On the other hand, Tyrosene (Y) has increased. As a result, Y-Y interaction is now comparable with Y-X interaction, where X represents any aliphatic amino acid. The aromatic interaction energies differ based on the amino acid pairing. While F,H,W and Y follow the ϵ_{ij} relation in Section

2.1.2 with aliphatic amino acids, aromatic-aromatic interactions such as F-F,H-H,W-W and Y-Y have a predetermined interaction energy defined in Table 2. Polar residues are also modified, with Asparagine (N) being the most hydrophobic (within the polar group). The higher end of the hydrophobicity scale now ends with Methionine (M). Finally, the cation-π shift is almost back to the value in 1BPA V1 (corresponding to $\Delta_{catpi} = -2$), reducing the strength of cation-π interactions.

The optimised force field parameters hint at dynamics mostly driven by the aromatic amino acid group. The V3.0 hydrophobicity scale was compared to 90 other HP scales, both experimental (87) and previous 1BPA iterations (3). Results of the Agglomerative Clustering can be found in Appendix H. The 1BPA variants are quite similar, with V3.0 being the most distinct.

Lastly, sensitivity analysis consisting of three different methods; Linear Regression, GPR Length scales and Sobol indices was implemented. Sensitivity indices can be found in Appendix I. The outcome was inconclusive as discussed in the next section.

4.2 Discussion

The process of identifying an optimal force field parameter set had several limitations including the 1BPA model, available molecular data, Gaussian Process Regression performance and project timescale. This section dissects individual contributions of the above mentioned factors. Impact on result accuracy is emphasised, along with suggestions for concrete future improvements.

The 1BPA model is a coarse-grained molecular dynamics model. Its computational efficiency comes at a cost of numerous approximations. Molecules are simulated in an implicit solvent, which does not account for individual solvent interactions and dynamics. The residue-residue interactions are hence modified to include solvent effects. Interpretation of the determined amino acid hydrophobicity scale should be adjusted accordingly. Furthermore, the 1BPA model assigns an identical mass and bead radius to all amino acids, which differs from experimental masses and radii.

Additionally, the 1BPA model does not account for partial charges. Histidine, (which is protonated 50% of the time under simulation conditions), has no net charge in the current 1BPA model. This discrepancy was briefly investigated with the previous force field version, with little apparent effect(data in Appendix G). This is to be expected given the abundance of histidine in the molecular data set. The amino acid composition of the molecular data set influences optimised force field parameters. Small variations in hydrophobicity of an overly represented amino acid (e.g. S) can significantly affect the average relative error ($\delta R_{g/h}$). This dependence will further propagate into the sensitivity analysis, which is discussed further below.

Finally, the 1BPA model and molecular dataset are biased by the initial exploration of the nuclear pore complex. The model was developed in order to investigate nuclear transport phenomena. Although it has since been adapted for LLPS investigation, previous versions were validated against single molecule experimental data, largely consisting of FGNups. The 1BPA hydrophobicity scale was compared with another coarse-grained computational HP scale, namely HPS [19] and HPS-Urry [20]. These scales were validated on a more general set of intrinsically disordered proteins. Even though the HP scales differ due to the potential form and validation dataset, an agglomerative analysis showed that the 1BPA V3.0 HP scale is not too dissimilar from the HPS-Urry scale (Appendix H).

Molecular dynamics simulations were executed in GROMACS. Each force field parameter set comprised of 189 individual IDP simulations. As a consequence of the sampling during the MD stage, replicas are assumed to have the same target variable value. That is to say, simulating the same force field parameter set multiple times would result in the same target value. This assumption was validated with two randomly selected force fields, whose target variable replicas differed at the third decimal; 0.709 ± 0.1 vs 0.708 ± 0.1. As a result, random measurement error was not obtained by repeat measurements and the mean absolute error (in the average relative error) was used as an alternative. Still, its definition propagates 189 single molecule errors (experimental and calculated) into a single force field error. Translating errors from the molecule level to the force field level is not straightforward and further validation is necessary. To summarise, the assigned force field error is two orders of magnitude larger than the GROMACS replication error, with its effect on the GPR model described further below.

Gaussian Process Regression was implemented to probe the relation between coarse-grained molecular dynamics force field and the discrepancy between experimental and calculated single molecule data. Model features (varied force field parameters) were selected according to the desired model selectivity, rather than conventional dimensionality reduction methods such as Principal Component Analysis (PCA). The number of investigated force field parameter sets at the model development stage equaled one (1BPA V2.1). It follows that dimensionality reduction was not feasible. Data generation was the focus during the model development stage.

Initial force fields were defined by small deviations from 1BPA V2.1, rather than sophisticated sampling methods. Although this region is known to show promising behavior, exploring broader region of the force field parameter space is desirable when identifying an optimal force field. Latin HyperCube Sampling (LHS) was chosen as an efficient high-dimensional sampling method for further explorations. While applying LHS sampling from the start would result in equally dense observation regions (in the force field parameter space), this approach does not inherently bias the GPR model. The only consequence is the fact that the GPR model showed more constrained confidence regions near the 1BPA V2.1 force field, due to higher sampling frequency. Having said that, a comparison of Batch 6 with Batch 5 shows that using a LHS defined train set improves GPR prediction. The Score metric (Equation 32) was not included in the GPR performance analysis. The test score was largely influenced by the different sampling methods, resulting in unreliable model evaluation.

The Squared Exponential Kernel with anisotropic length scales (one per feature) was implemented due to its smooth nature. Individual length scales allow for further understanding of individual force field parameters and their effect on the target variable. Kernel-hyper parameters varied drastically as more data points were used for training the model. Many length scales reached the default upper bound of 100000 within the scikit-learn library. The latest iteration did not show a clear convergence. As a result, the latest length scale values cannot be a reliable indicator of parameter sensitivity.

Three methods were investigated in order to quantify model sensitivity to individual force field parameters; Linear Regression, GPR length scales and Sobol indices. All three methods are inconclusive and serve as a qualitative indication, rather than a reliable metric. Linear coefficients from Linear Regression are not reliable because there is no indication that the relationship between force field parameters and $R_{g/h}$ is linear. GPR length scales did not converge, hence they cannot truthfully encompass the sensitivity behaviour. Finally, the Sobol method involves more than one million GPR predictions, which do not to match the observations, see Figures 8, 9 and 10.

Most GPR models estimate observation noise when presented with input replicas with differing target values. This was not possible with the above described GROMACS simulated data points because of the project timescale. As an alternative, an observation specific error (mean absolute error in the average relative error) was added to the diagonal of the kernel matrix. It is probable that this design choice resulted in underfitting due to the error magnitude. The GPR confidence interval remains quite broad, which is an indicator of low sampling or underfitting. Varying the noise metric is recommended for investigating this behaviour further.

GPR hyper-parameters were optimised via maximising the Log Marginal Likelihood. The number of random initialisations was set to 100, as 1000 increased run time but did not significantly improve Log Marginal Likelihood. Log Marginal Likelihood is a point estimate, rather than a fully Bayesian estimator approach. The hyper-parameters themselves follow a probability distribution, which is not encompassed by point estimate methods such as maximising the Log Marginal Likelihood (ML-II). The result of ML-II is a single hyper-parameter value without a confidence interval. Fully Bayesian estimators have better performance (lower MSE) on parameter estimation compared to Maximum Likelihood Estimation as demonstrated in [51].

Due to the number of input dimensions (26), a substantial number of GROMACS simulated force field parameter sets was required for meaningful GPR predictions. The latest iteration of the GPR model was trained on 212 unique force fields. Although this number of observations allowed the model to differentiate between inputs, at least 1000 or 2000 would be desirable for accurate GPR predictions [52]. This limitation is further supported by Figure 10, which shows the discrepancy between observed and GPR predicted target variable values. Further acquisition of GPR data points (force fields) would substantially increase GPR model accuracy, which is directly linked to force field optimisation efficiency. Current coverage of the force field parameter space (265 force fields) does not exclude the possibility of another region of interest.

The GPR model was implemented via the scikit-learn package [41]. While straightforward to use, it led to specific design choice constraints. Numerous Python open source packages targeted to Gaussian Processes are freely available. GP-Plus [53], Gpy [54], GPflow [55] and GPytorch [56] were developed solely for Gaussian Process applications. Other Bayesian modelling packages that offer Gaussian Process implementation include PyStan [57], PyMC [58] or pyGPgo [59]. Exploring the functionalities of the above mentioned packages will allow for more informed design choices and further customising of the GPR predictor for coarse-grained force field optimisation.

The optimal coarse-grained force field resulted in a significant improvement between experimental and calculated $R_{g/h}$: $\delta R_{g/h} = 0.11 \pm 0.02$, as opposed to $\delta R_{g/h} = 0.14 \pm 0.03$ for V2.1. The GPR model identified 68 candidates, V3.0 corresponding to the lowest observed $\delta R_{g/h}$. Although the individual force field parameter changes seem reasonable, further validation is crucial. The 1BPA V3.0 force field was optimised solely via a machine learning approach based on R_g and R_h. Application to collective biological processes such as liquid-liquid phase separation is required to establish its status as an improved coarse-grained force field.

5 Conclusion

Gaussian Process Regression (GPR) is a powerful machine learning algorithm that can predict a target variable value from multi-dimensional input without the need for an explicit functional form. 265 force field parameter sets were investigated, with Latin HyperCube Sampling (LHS) allowing for efficient investigation of a 26-dimensional parameter space. Molecular data set comprising of 189 intrinsically disordered proteins was used for force field optimisation by minimising the discrepancy between experimental and calculated radius of gyration or Stokes' radius. The number of investigated force fields seemed not enough to determine a global optimum, although the hyper-parameters were on their way to converge even as the Log Marginal Likelihood increased with the number of training data points. We found that the sensitivity analysis was inconclusive. The GPR model performance showed signs of underfitting, which could be mitigated by another error metric or use of a different GPR targeted software package. Despite the above mentioned limitations, 68 force fields performed better than the previous 1BPA V2.1 version. The selected 1BPA V3.0 had an average relative error $\delta R_{g/h} = 0.11 \pm 0.02$, which is a 21% reduction in error compared to V2.1 error ($\delta R_{g/h} = 0.14 \pm 0.03$). The newly determined force field is physically feasible according to agglomerative clustering with 87 experimental hydrophobicity scales. Liquid-liquid phase separation simulations are the essential next step in establishing 1BPA V3.0 as an optimised coarse-grained force field.

References

[1] A. A. Hyman, C. A. Weber, and F. Jülicher, "Liquid-Liquid Phase Separation in Biology," *Annual Review of Cell and Developmental Biology*, vol. 30, no. Volume 30, 2014, pp. 39–58, Oct. 2014. DOI: 10.1146/annurev-cellbio-100913-013325. [Online]. Available: https://www.annualreviews.org/content/journals/10.1146/annurev-cellbio-100913-013325 (visited on 05/27/2024).

[2] A. Aguzzi and M. Altmeyer, "Phase separation: Linking cellular compartmentalization to disease," *Trends in cell biology*, vol. 26, no. 7, pp. 547–558, 2016.

[3] S. Alberti, A. Gladfelter, and T. Mittag, "Considerations and challenges in studying liquid-liquid phase separation and biomolecular condensates," *Cell*, vol. 176, no. 3, pp. 419–434, 2019, doi : 10.1016/j.cell.2018.12.035.

[4] S. F. Banani *et al.*, "Biomolecular condensates: Organizers of cellular biochemistry," *Nature Reviews Molecular Cell Biology*, vol. 18, no. 5, pp. 285–298, May 2017, ISSN: 1471-0080. DOI: 10.1038/nrm.2017.7. [Online]. Available: https://doi.org/10.1038/nrm.2017.7.

[5] H. Walter and D. E. Brooks, "Phase separation in cytoplasm, due to macromolecular crowding, is the basis for microcompartmentation," *FEBS letters*, vol. 361, no. 2-3, pp. 135–139, Mar. 1995, ISSN: 0014-5793. DOI: 10.1016/0014-5793(95)00159-7.

[6] J. Li *et al.*, "Post-translational modifications in liquid-liquid phase separation: A comprehensive review," *Molecular Biomedicine*, vol. 3, p. 13, Dec. 2022. DOI: 10.1186/s43556-022-00075-2. [Online]. Available: https://link.springer.com/10.1186/s43556-022-00075-2.

[7] V. Rangachari, "Biomolecular condensates – extant relics or evolving microcompartments?" *Communications Biology*, vol. 6, pp. 1–8, Jun. 2023. DOI: 10.1038/s42003-023-04963-3. [Online]. Available: https://www.nature.com/articles/s42003-023-04963-3.

[8] E. A. Abbondanzieri and A. S. Meyer, "More than just a phase: The search for membraneless organelles in the bacterial cytoplasm," en, *Current Genetics*, vol. 65, no. 3, pp. 691–694, Jun. 2019, ISSN: 1432-0983. DOI: 10.1007/s00294-018-00927-x. [Online]. Available: https://doi.org/10.1007/s00294-018-00927-x (visited on 05/27/2024).

[9] G. L. Dignon *et al.*, "Sequence determinants of protein phase behavior from a coarse-grained model," *PLOS Computational Biology*, vol. 14, pp. 1–23, Jan. 2018, doi: 10.1371/journal.pcbi.1005941. [Online]. Available: https://journals.plos.org/ploscompbiol/article?id=10.1371%2Fjournal.pcbi.1005941.

[10] A. K. Dunker *et al.*, "What's in a name? Why these proteins are intrinsically disordered: Why these proteins are intrinsically disordered," *Intrinsically Disordered Proteins*, vol. 1, no. 1, e24157, Jan. 2013, ISSN: 2169-0707. DOI: 10.4161/idp.24157. [Online]. Available: http://www.tandfonline.com/doi/abs/10.4161/idp.24157 (visited on 06/02/2024).

[11] X. Tong *et al.*, "Liquid–liquid phase separation in tumor biology," *Signal Transduction and Targeted Therapy*, vol. 7, p. 221, Jul. 2022. DOI: 10.1038/s41392-022-01076-x. [Online]. Available: https://www.nature.com/articles/s41392-022-01076-x.

[12] S. Ray, N. Singh, *et al.*, "α-Synuclein aggregation nucleates through liquid–liquid phase separation," *Nature Chemistry*, vol. 12, no. 8, pp. 705–716, Aug. 2020, ISSN: 1755-4349. DOI: 10.1038/s41557-020-0465-9. [Online]. Available: https://doi.org/10.1038/s41557-020-0465-9.

[13] P. Dogra *et al.*, "Intermolecular charge-transfer modulates liquid–liquid phase separation and liquid-to-solid maturation of an intrinsically disordered ph-responsive domain," *Journal of the American Chemical Society*, vol. 141, no. 51, pp. 20 380–20 389, 2019. DOI: 10.1021/jacs.9b10892. eprint: https://doi.org/10.1021/jacs.9b10892. [Online]. Available: https://doi.org/10.1021/jacs.9b10892.

[14] M. Wolf *et al.*, "Effective interactions in protein–salt solutions approaching liquid–liquid phase separation," *Journal of Molecular Liquids*, vol. 200, pp. 20–27, 2014, ISSN: 0167-7322. DOI: https://doi.org/10.1016/j.molliq.2014.08.006. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S016773221400364X.

[15] N. Stanley, S. Esteban-Martín, and G. De Fabritiis, "Progress in studying intrinsically disordered proteins with atomistic simulations," *Progress in Biophysics and Molecular Biology*, vol. 119, no. 1, pp. 47–52, Oct. 2015, ISSN: 00796107. DOI: 10.1016/j.pbiomolbio.2015.03.003. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0079610715000395 (visited on 06/02/2024).

[16] J.-M. Choi, A. S. Holehouse, and R. V. Pappu, "Physical Principles Underlying the Complex Biology of Intracellular Phase Transitions," *Annual Review of Biophysics*, vol. 49, no. Volume 49, 2020, pp. 107–133, May 2020. DOI: 10.1146/annurev-biophys-121219-081629. [Online]. Available: https://www.annualreviews.org/content/journals/10.1146/annurev-biophys-121219-081629 (visited on 05/27/2024).

[17] J. A. Joseph *et al.*, "Physics-driven coarse-grained model for biomolecular phase separation with near-quantitative accuracy," *Nature Computational Science*, vol. 1, pp. 732–743, Nov. 2021, doi : 10.1038/s43588-021-00155-3. [Online]. Available: https://www.nature.com/articles/s43588-021-00155-3.

[18] G. Tesei *et al.*, "Accurate model of liquid–liquid phase behavior of intrinsically disordered proteins from optimization of single-chain properties," *Proceedings of the National Academy of Sciences*, vol. 118, e2111696118, Nov. 2021. DOI: 10.1073/pnas.2111696118. [Online]. Available: https://www.pnas.org/doi/full/10.1073/pnas.2111696118.

[19] L. H. Kapcha and P. J. Rossky, "A simple atomic-level hydrophobicity scale reveals protein interfacial structure," *Journal of molecular biology*, vol. 426, no. 2, pp. 484–498, 2014, doi: 10.1016/j.jmb.2013.09.039.

[20] R. M. Regy *et al.*, "Improved coarse-grained model for studying sequence dependent phase separation of disordered proteins," en, *Protein Science*, vol. 30, no. 7, pp. 1371–1379, Jul. 2021, ISSN: 0961-8368, 1469-896X. DOI: 10.1002/pro.4094. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1002/pro.4094 (visited on 06/03/2024).

[21] J. M. Lotthammer *et al.*, "Direct prediction of intrinsically disordered protein conformational properties from sequence," *Nature Methods*, vol. 21, pp. 465–476, Jan. 2024, doi: 10.1038/s41592-023-02159-5. [Online]. Available: https://www.nature.com/articles/s41592-023-02159-5.

[22] A. Ghavami, E. van der Giessen, and P. R. Onck, "Coarse-Grained Potentials for Local Interactions in Unfolded Proteins," *Journal of Chemical Theory and Computation*, vol. 9, pp. 432–440, Jan. 2013. DOI: 10.1021/ct300684j. [Online]. Available: https://doi.org/10.1021/ct300684j.

[23] M. Dekker, E. Van der Giessen, and P. R. Onck, "Phase separation of intrinsically disordered fg-nups is driven by highly dynamic fg motifs," *Proceedings of the National Academy of Sciences*, vol. 120, no. 25, e2221804120, 2023.

[24] A. Fragasso *et al.*, "A designer FG-Nup that reconstitutes the selective transport barrier of the nuclear pore complex," *Nature Communications*, vol. 12, no. 1, p. 2010, Mar. 2021, ISSN: 2041-1723. DOI: 10.1038/s41467-021-22293-y. [Online]. Available: https://doi.org/10.1038/s41467-021-22293-y.

[25] M. Driver and P. Onck, *Selective phase separation of transcription factors is driven by orthogonal molecular grammar*. Apr. 2024. DOI: 10.1101/2024.04.12.589262.

[26] M. D. Driver, J. Postema, and P. R. Onck, "The effect of dipeptide repeat proteins on fus/tdp43-rna condensation in c9orf72 als/ftd," *bioRxiv*, 2024. DOI: 10.1101/2024.05.21.595197. eprint: https://www.biorxiv.org/content/early/2024/05/23/2024.05.21.595197.full.pdf. [Online]. Available: https://www.biorxiv.org/content/early/2024/05/23/2024.05.21.595197.

[27] J. Jumper *et al.*, "Highly accurate protein structure prediction with AlphaFold," *Nature*, vol. 596, no. 7873, pp. 583–589, Aug. 2021, ISSN: 1476-4687. DOI: 10.1038/s41586-021-03819-2. [Online]. Available: https://doi.org/10.1038/s41586-021-03819-2.

[28] V. L. Deringer *et al.*, "Gaussian Process Regression for Materials and Molecules," *Chemical Reviews*, vol. 121, pp. 10 073–10 141, Aug. 2021. DOI: 10.1021/acs.chemrev.1c00022. [Online]. Available: https://doi.org/10.1021/acs.chemrev.1c00022.

[29] A. Giuntoli *et al.*, "Systematic coarse-graining of epoxy resins with machine learning-informed energy renormalization," *npj Computational Materials*, vol. 7, no. 1, p. 168, Oct. 2021, ISSN: 2057-3960. DOI: 10.1038/s41524-021-00634-1. [Online]. Available: https://www.nature.com/articles/s41524-021-00634-1 (visited on 04/13/2024).

[30] S. John, *Many-Body Coarse-Grained Interactions using Gaussian Approximation Potentials*. Sep. 2017. DOI: 10.1101/2024.04.12.589262.

[31] H. Jafarinia, E. v. d. Giessen, and P. R. Onck, "Phase Separation of Toxic Dipeptide Repeat Proteins Related to C9orf72 ALS/FTD," *Biophysical Journal*, vol. 119, no. 4, pp. 843–851, Aug. 2020, ISSN: 0006-3495. DOI: 10.1016/j.bpj.2020.07.005. [Online]. Available: https://www.cell.com/biophysj/abstract/S0006-3495(20)30532-4 (visited on 05/30/2024).

[32] A. Ghavami *et al.*, "Probing the Disordered Domain of the Nuclear Pore Complex through Coarse-Grained Molecular Dynamics Simulations," *Biophysical Journal*, vol. 107, no. 6, pp. 1393–1402, Sep. 2014, ISSN: 0006-3495. DOI: 10.1016/j.bpj.2014.07.060. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4167297/ (visited on 05/31/2024).

[33] M. Driver and P. Onck, *Selective phase separation of transcription factors is driven by orthogonal molecular grammar: Supplementary Information*. Apr. 2024. DOI: 10.1101/2024.04.12.589262.

[34] S. W. Park *et al.*, "Revisiting the Ramachandran plot based on statistical analysis of static and dynamic characteristics of protein structures," *Journal of Structural Biology*, vol. 215, p. 107 939, Mar. 2023, ISSN: 1047-8477. DOI: 10.1016/j.jsb.2023.107939. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1047847723000023.

[35] D. Eisenberg *et al.*, "Analysis of membrane and surface protein sequences with the hydrophobic moment plot," eng, *Journal of Molecular Biology*, vol. 179, no. 1, pp. 125–142, Oct. 1984, ISSN: 0022-2836. DOI: 10.1016/0022-2836(84)90309-7.

[36] C. E. Rasmussen and C. K. I. Williams, *Gaussian processes for machine learning* (Adaptive computation and machine learning), en. Cambridge, Mass: MIT Press, 2006, ISBN: 978-0-262-18253-9.

[37] Z. Chen and B. Wang, "How priors of initial hyperparameters affect Gaussian process regression models," *Neurocomputing*, vol. 275, Jan. 2018, ISSN: 0925-2312. DOI: 10.1016/j.neucom.2017.10.028. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S092523121731679X.

[38] H. Huang *et al.*, "Gaussian Process Regression With Maximizing the Composite Conditional Likelihood," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–11, 2021. DOI: 10.1109/TIM.2021.3104376. [Online]. Available: https://ieeexplore.ieee.org/document/9512548.

[39] D. Forsyth, "Useful probability distributions," in *Probability and Statistics for Computer Science*, D. Forsyth, Ed., Springer International Publishing, 2018, p. 124, ISBN: 978-3-319-64410-3. DOI: 10.1007/978-3-319-64410-3_5. [Online]. Available: https://doi.org/10.1007/978-3-319-64410-3_5.

[40] V. Dubourg *et al.*, "Scikit-learn: Machine learning in Python:gaussian processes regression-basic introductory example,"

[41] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[42] O. Kratky and P. Laggner, "X-ray small-angle scattering," in *Encyclopedia of Physical Science and Technology (Third Edition)*, Third Edition, Academic Press, 2003, pp. 939–988, ISBN: 978-0-12-227410-7. DOI: https://doi.org/10.1016/B0-12-227410-5/00832-2. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B0122274105008322.

[43] J. Daintith, *Dictionary of Chemistry 6 ed.* Oxford University Press, Sep. 2008. DOI: 10.1093/acref/9780199204632.001.0001.

[44] J. Yamada *et al.*, "A Bimodal Distribution of Two Distinct Categories of Intrinsically Disordered Structures with Separate Functions in FG Nucleoporins *," *Molecular & Cellular Proteomics*, vol. 9, pp. 2205–2224, Oct. 2010, doi : 10.1074/mcp.M000035-MCP201. [Online]. Available: https://www.mcponline.org/article/S1535-9476(20)34524-2/abstract.

[45] M. J. Abraham *et al.*, "Gromacs: High performance molecular simulations through multi-level parallelism from laptops to supercomputers," *SoftwareX*, vol. 1-2, pp. 19–25, 2015, ISSN: 2352-7110. DOI: https://doi.org/10.1016/j.softx.2015.06.001. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2352711015000059.

[46] R. J. Gowers *et al.*, “MDAnalysis: A Python Package for the Rapid Analysis of Molecular Dynamics Simulations,” in *Proceedings of the 15th Python in Science Conference*, S. Benthall and S. Rostrup, Eds., 2016, pp. 98–105. DOI: 10.25080/Majora-629e541a-00e.

[47] P. J. Fleming and K. G. Fleming, “HullRad: Fast Calculations of Folded and Disordered Protein and Nucleic Acid Hydrodynamic Properties,” English, *Biophysical Journal*, vol. 114, no. 4, pp. 856–869, Feb. 2018, ISSN: 0006-3495. DOI: 10.1016/j.bpj.2018.01.002. [Online]. Available: https://www.cell.com/biophysj/abstract/S0006-3495(18)30065-1 (visited on 05/31/2024).

[48] A. N. Gamby and J. Katajainen, “Convex-hull algorithms: Implementation, testing, and experimentation,” *Algorithms*, vol. 11, no. 12, 2018, ISSN: 1999-4893. DOI: 10.3390/a11120195. [Online]. Available: https://www.mdpi.com/1999-4893/11/12/195.

[49] C. Song and R. Kawai, “Monte carlo and variance reduction methods for structural reliability analysis: A comprehensive review,” *Probabilistic Engineering Mechanics*, vol. 73, p. 103 479, 2023, ISSN: 0266-8920. DOI: https://doi.org/10.1016/j.probengmech.2023.103479. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0266892023000681.

[50] S. Simm *et al.*, “50 years of amino acid hydrophobicity scales: Revisiting the capacity for peptide classification,” *Biological research*, vol. 49, pp. 1–19, 2016, doi : 10.1186/s40659-016-0092-5.

[51] F. Alduais, “Comparison of classical and bayesian estimators to estimate the parameters in weibull distribution under weighted general entropy loss function,” *International Journal of ADVANCED AND APPLIED SCIENCES*, vol. 8, pp. 57–62, Mar. 2021. DOI: 10.21833/ijaas.2021.03.008.

[52] A. van Beek, personal communication, Jun. 14, 2024.

[53] A. Yousefpour *et al.*, “Gp+: A python library for kernel-based learning via gaussian processes,” *Advances in Engineering Software*, vol. 195, p. 103 686, 2024, ISSN: 0965-9978. DOI: https://doi.org/10.1016/j.advengsoft.2024.103686. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0965997824000930.

[54] GPy, *GPy: A gaussian process framework in python*, http://github.com/SheffieldML/GPy, since 2012.

[55] A. G. d. G. Matthews *et al.*, “GPflow: A Gaussian process library using TensorFlow,” *Journal of Machine Learning Research*, vol. 18, no. 40, pp. 1–6, Apr. 2017. [Online]. Available: http://jmlr.org/papers/v18/16-537.html.

[56] J. R. Gardner *et al.*, “Gpytorch: Blackbox matrix-matrix gaussian process inference with gpu acceleration,” in *Advances in Neural Information Processing Systems*, 2018.

[57] A. Riddell, A. Hartikainen, and M. Carter, *Pystan (3.0.0)*, PyPI, Mar. 2021.

[58] O. Abril-Pla *et al.*, “Pymc: A modern, and comprehensive probabilistic programming framework in python,” *PeerJ Computer Science*, vol. 9, e1516, 2023.

[59] J. Jiménez and J. Ginebra, “Pygpgo: Bayesian optimization for python,” *Journal of Open Source Software*, vol. 2, no. 19, p. 431, 2017. DOI: 10.21105/joss.00431. [Online]. Available: https://doi.org/10.21105/joss.00431.

[60] L. E. Kapinos *et al.*, “Karyopherin-Centric Control of Nuclear Pores Based on Molecular Occupancy and Kinetic Analysis of Multivalent Binding with FG Nucleoporins,” *Biophysical Journal*, vol. 106, pp. 1751–1762, Apr. 2014, doi : 10.1016/j.bpj.2014.02.021. [Online]. Available: https://www.cell.com/biophysj/abstract/S0006-3495(14)00227-6.

[61] V. H. Ryan *et al.*, “Mechanistic View of hnRNPA2 Low-Complexity Domain Structure, Interactions, and Phase Separation Altered by Mutation and Arginine Methylation,” *Molecular Cell*, vol. 69, no. 3, 465–479.e7, Feb. 2018, ISSN: 1097-4164. DOI: 10.1016/j.molcel.2017.12.022.

[62] G. Bianchi *et al.*, “Distribution of Charged Residues Affects the Average Size and Shape of Intrinsically Disordered Proteins,” *Biomolecules*, vol. 12, p. 561, Apr. 2022, doi : 10.3390/biom12040561. [Online]. Available: https://www.mdpi.com/2218-273X/12/4/561.

[63] G. Tesei *et al.*, “Conformational ensembles of the human intrinsically disordered proteome,” *Nature*, vol. 626, pp. 897–904, Jan. 2024, doi: 10.1038/s41586-023-07004-5. [Online]. Available: https://www.nature.com/articles/s41586-023-07004-5.

[64] T. A. Jowitt *et al.*, “Order within disorder: Aggrecan chondroitin sulphate-attachment region provides new structural insights into protein sequences classified as disordered,” *Proteins: Structure, Function, and Bioinformatics*, vol. 78, pp. 3317–3327, 2010. DOI: 10.1002/prot.22839. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/prot.22839.

[65] A. Bremer *et al.*, “Deciphering how naturally occurring sequence features impact the phase behaviours of disordered prion-like domains,” *Nature Chemistry*, vol. 14, pp. 196–207, Feb. 2022. DOI: 10.1038/s41557-021-00840-w.

Appendices

A 1BPA model details

Bending type ϕ_{bend}	*Torsion type* $\phi_{torsion}$
ZGX	ZGGX
ZPX	ZGPX
ZXX	ZGXX
ZGP	ZPGX
ZPP	ZPPX
ZXP	ZPXX
	ZXGX
	ZXPX
	ZXXX

Table 3: Unique bending types and torsion types derived from pseudo-angles. Naming convention is as follows: Glycine (G), Proline (P), 18 amino acids (excluding G and P)(X) and any amino acid (Z). Order is defined from the N-terminus (left) to the C-terminus (right) [22].

The Debye screening coefficient is defined as

$$\kappa = \left(\frac{\epsilon_0 \epsilon_r k_b T}{2 N_A e^2 I} \right)^{-0.5}$$

where ϵ_0 and ϵ_r represent the permittivity of free space and water respectively, k_b is the Boltzmann's constant and T is temperature set to 300K. N_A is Avogadro's number, e is the elementary charge and experimental ion concentration I equals 150 mM (unless specified otherwise). Permittivity of water follows the sigmoid function,

$$\epsilon_r(r) = S_s \left[1 - \frac{r^2}{z^2} \frac{e^{r/z}}{\left(e^{r/z} - 1\right)^2} \right]$$

where $Ss = 80$, and $z = 0.25nm$.

B GPR: Estimators

$$\hat{\mu} = K\left(X_*, X\right)^{\mathrm{T}} \left(K(X,X) + \sigma_n^2 I\right)^{-1} \boldsymbol{y}$$
$$\hat{\Sigma} = K\left(X_*, X_*\right) - K\left(X_*, X\right)^{\mathrm{T}} \left(K(X,X) + \sigma_n^2 I\right)^{-1} K\left(X_*, X\right)$$

C GPR: Single Prediction Example

The posterior predictor is assembled from the prior distribution conditioned on N observations via Bayes rule. In other words, we are interested in the probability of a new observation y_{N+1} given previous observations $\mathbf{y}$ [28].

$$P(y_{N+1}|\mathbf{y}) = \frac{P(y_1, y_2, ...y_N, y_{N+1})}{P(\mathbf{y})} \tag{34}$$

Given that both $\mathbf{y}$ and the joined distribution of $\mathbf{y}$ together with the new observation y_{N+1} follow a (distinct) Gaussian distribution, the probability distribution $P(y_{N+1})$ is also Gaussian.

$$P(y_{N+1}) \sim \mathcal{N}(\bar{y}_{N+1}, \text{var}(y_{N+1})) \tag{35}$$

Definitions of the mean and variance for the probability distribution of the new observation y_{N+1} are given below. Note that the variance does not depend on previous observations $\mathbf{y}$, only on the input points $\mathbf{x}$.

$$\begin{aligned} \bar{y}_{N+1} &= \mathbf{k}^T(K + \sigma_n^2 I)^{-1}\mathbf{y} \\ \text{var}(y_{N+1}) &= k(\mathbf{x}_{N+1}, \mathbf{x}_{N+1}) + \sigma_n^2 - \mathbf{k}^T(K + \sigma_n^2 I)^{-1}\mathbf{k} \end{aligned} \tag{36}$$

Vector $\mathbf{k}$ defines the kernel values evaluated between the new input $\mathbf{x}_{N+1}$ and the previous input points X, which is a $N \times N$ matrix [28].

D Molecular Dataset Literature Sources

R_h		R_g	
Source	*Molecules*	*Source*	*Molecules*
Yamada *et al.* [44]	20	Lotthamer *et al.* [21]	137
Kapinos *et al.* [60]	4	Dignon *et al.* [9]	9
Ryan *et al.* [61]	1	Tesei *et al.* (2021) [18]	5
Bianchi *et al.* [62]	9	Tesei *et al.* (2023) [63]	3
		Jowitt *et al.* [64]	1
Subtotal	34	*Subtotal*	155
Total	**189**		

Table 4: Literature sources overview for molecule data set. Dataset by Lotthamer *et al.* includes molecules from Bremer *et al.* [65], Joseph *et al.* [17] and Tesei *et al.* [18] .

E Batches

GPR data points were continuously acquired in batches. The section below defines each batch and its dimensions: ($\text{n}_{\text{samples}} \times \text{n}_{\text{features}}$).

- Batch 1 (36,26): Force field parameter sets defined by combining the parameter values below. All other parameters, including the hydrophobicity scale are based on 1BPA V2.1.

$$\alpha \in [0.12, 0.15, 0.20, 0.25]$$

$$[\text{F-F,Y-Y,W-W,H-H}] \in [[7.5, 8.0, 8.5, 6.5], [7.5, 8.0, 8.5, 6.0], [7.0, 7.5, 8.0, 6.0]]$$

$$\Delta_{catpi} \in [0, -1, -2]$$

- Batch 2A (44,26) and Batch 2B (48,26): Force field parameter sets defined by small perturbation from the 1BPA V2.1 hydrophobicity scale. All amino acids are divided into four groups with both intra(2A) and inter(2B) group effects accounted for.

1. Charged (Hydrophillic) : [R,K,D,E]
2. Polar : [S,T,N,Q,G]
3. Aliphatic (Hydrophobic): [A,V,I,L,P,C,M]
4. Aromatic: [F,Y,W,H]

- Batch 3 (39,26): Force field parameter sets sampled via Latin HyperCube Sampling. Samples were scaled in the following bounds.

  $$[\alpha, \epsilon_i, \epsilon_{aromatic}, \Delta_{catpi}] :$$ [2]

  $$[0.10, 0.6, 0.6, 0, 0, 0.5, 0.25, 0.5, 0.6, 0, 0.6, 0.6, 0.25, 0.6, 0.25, 0, 0.25, 0.25, 0.6, 0.5, 0.5, 6, 6, 6, 6, -2],$$

  $$[0.25, 1, 1, 0.25, 0.25, 1, 0.6, 1, 1, 0.25, 1, 1, 0.6, 1, 0.7, 0.25, 0.6, 0.6, 1, 1, 1, 8, 8, 8, 8, 0],$$

- Batch 4 (20,26): Force field parameter sets defined with GPR predictor trained on batches 1, 2A, and 2B. Latin HyperCube Sample of 1000 force field parameter sets (bounded by limits from Batch 3 definition) are predicted via trained GPR. 10 largest and 10 lowest target variable predictions were selected, defining Batch 4.

- Batch 5 (20,26): Force field parameter sets defined with GPR predictor trained on batches 1, 2A,2B and 3. Latin HyperCube Sample of 1000 force field parameter sets (bounded by limits from Batch 3 definition) are predicted via trained GPR. 10 largest and 10 lowest target variable predictions were selected, defining Batch 5.

- Batch 6 (10,26): Force field parameter sets defined with GPR predictor trained on batches 3 and 4. Latin HyperCube Sample of 1000 force field parameter sets (bounded by limits from Batch 3 definition) are predicted via trained GPR. 10 lowest target variable predictions were selected, defining Batch 6.

- Batch 7 (50,26): Batch 7 was defined by zooming into the previously identified regions of interest. Optimised force field candidates ($<$ 1BPA V2.1) were grouped into 2 clusters via Agglomerative Clustering:

  - Cluster1: Force fields similar to 1BPA V2.1
  - Cluster2: Other

  Two sets of lower and upper LHS limits were defined accordingly, effectively zooming in the force field parameter space. Each above defined cluster provided a distinct LHS sample with 100 samples. All 200 samples were then presented to the GPR model, (trained on batches 1,2A,2B,3,4, and 5). Lowest predicted target values were selected for Batch7 definition. Selection was purposefully uneven, with 40 parameter sets drawn from Cluster 2 LHS sample and 10 from Cluster 1 LHS sample. As a result, Batch 7 contained unevenly distributed parameter sets, with more focus on previously unexplored local minima.

[2] ϵ_i refer to individual AA hydrophobicites (sorted in alphabetical order) and $\epsilon_{aromatics}$ refers to $\epsilon_{F-F}, \epsilon_{Y-Y}, \epsilon_{W-W}, \epsilon_{H-H}$.

F GPR Hyper-parameters

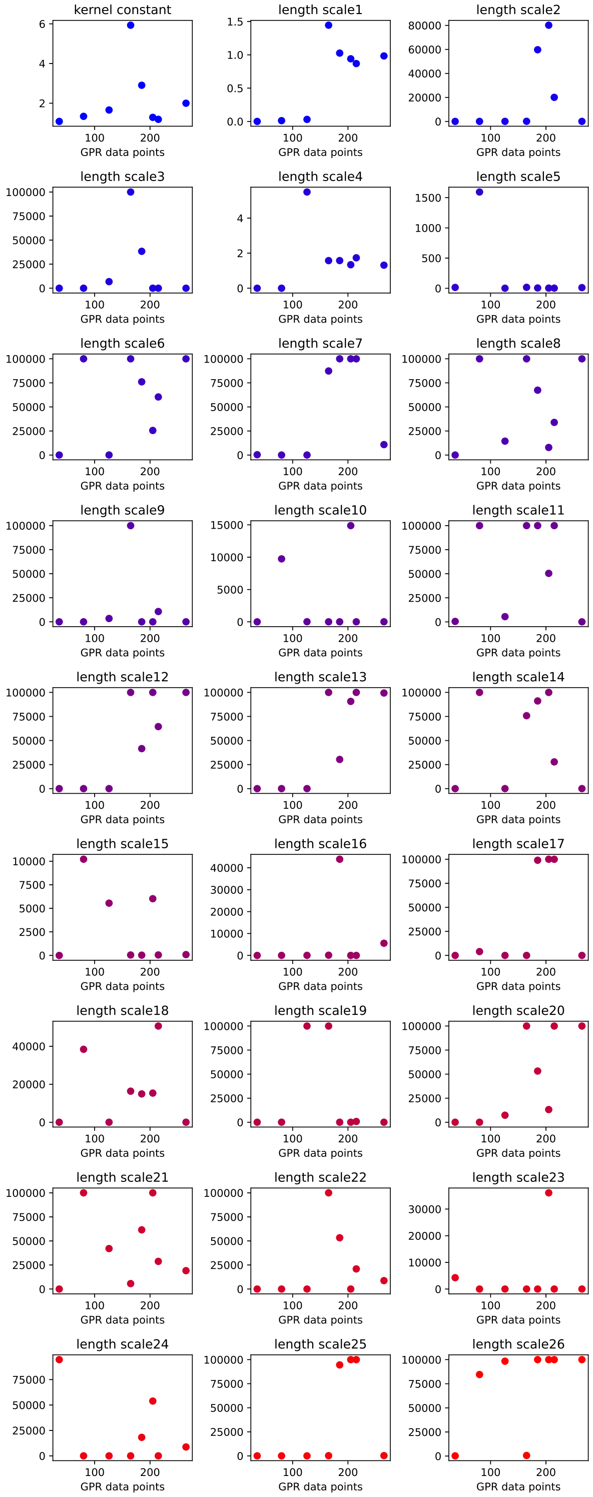

Figure 14: Hyper-parameters as a function of GPR data points.

G Histidine charge dependence

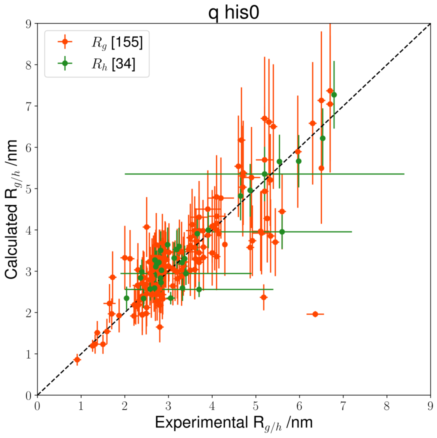

Figure 15: Comparison of all 189 IDR $R_{g/h}$ calculations wiht experiment for a coarse grained force field with no histidine charge.

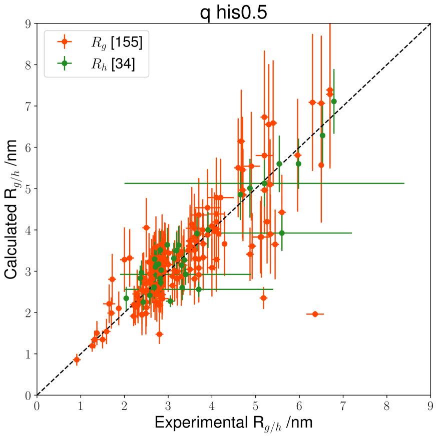

Figure 16: Comparison of all 189 IDR $R_{g/h}$ calculations with experiment for a coarse grained force field with 0.5 histidine charge.

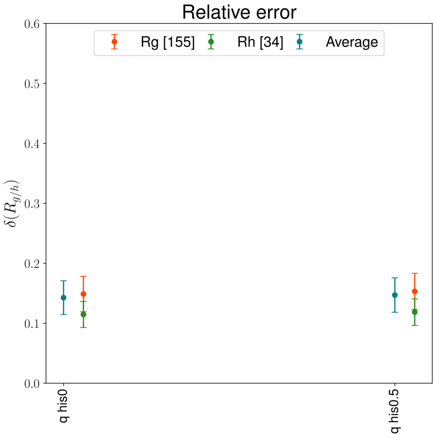

Figure 17: Average relative error comparison for two cases; no net histidine charge (q his0) and 0.5 histidine charge (q his0.5)

Figures 15 and 16 show minor differences in single molecule $R_{g/h}$ calculations. Averaging over all 189 IDRs (Figure 17) shows no visible distinction between a coarse-grained force field with no net histidine charge and 0.5 histidine charge.

H Hydrophobicity scales analysis

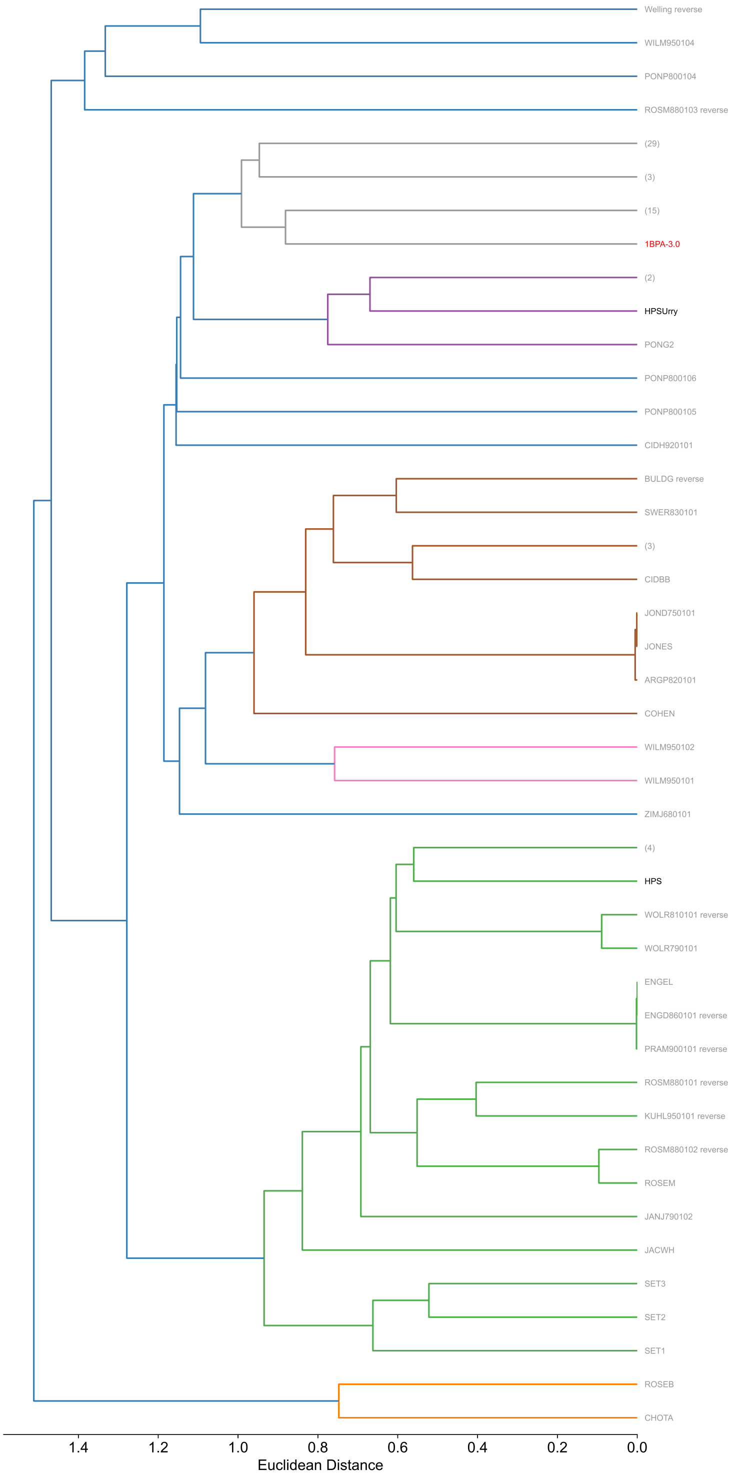

Figure 18: Agglomerative clustering of 91 hydrophobicity scales. 1BPA V3.0 is highlited in red, with HPS and HPS-Urry highlighted in black. Numbers in brackets indicate the number of HP scales belonging to that node. Previous 1BPA versions are not shown.

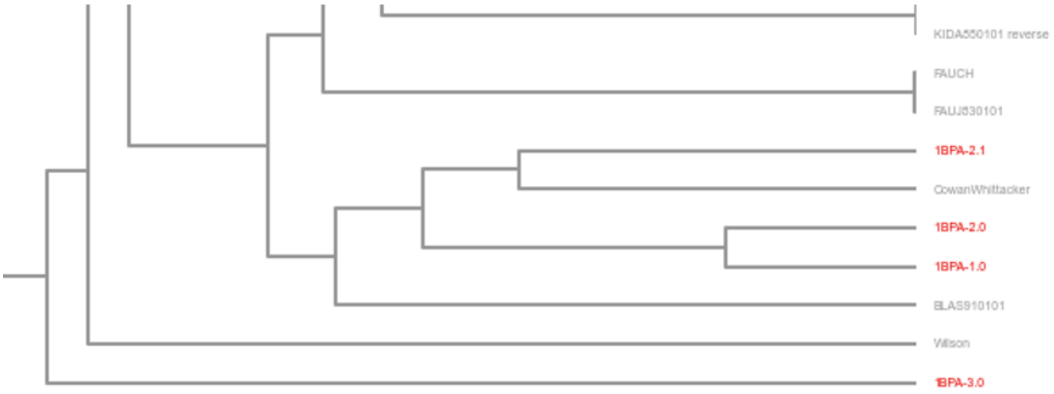

Figure 19: Agglomerative clustering analysis, zoomed in to show previous 1BPA hydrophobicity scales (in red). All fall into the same grey cluster, as expected.

I Sensitivity analysis

Sensitivity analysis was implemented to identify force field parameters with largest impact on $\delta R_{g/h}$. Three methods were evaluated: Linear Regression, GPR length scales and Sobol indeces. All methods provide qualitative rather than quantitative results. Further exploration is necessary.

The magnitude of Linear Regression coefficient represents the change in $\delta R_{g/h}$ upon a unit change in the corresponding parameter. However, Linear Regression has numerous prior assumptions (such as homoscedasticity), which have not been validated. Additionally, Linear Regression assumes a linear relation between the force field parameters and the target value $\delta R_{g/h}$. The exact functional form is not yet known, further undermining this sensitivity analysis method.

GPR length scales quantify the change in $\delta R_{g/h}$ upon a small change in a given parameter. Accordingly, the inverse can be interpreted as a measure of sensitivity. Optimised length scales varied rapidly as new GPR data points were introduced to the model. The Log Marginal Likelihood plot (Figure 7) shows a continuous increase, which indicates that the latest hyper-parameter values might not be fully optimised. Due to the lack of hyper-parameter convergence, corresponding sensitivity analysis is not indicative of actual model behaviour.

Total order Sobol indices quantify the effect of a single parameter (including its interaction with other parameter). The Sobol method consists of at least 10^5 model evaluations. In our case, the GPR model predicted more than 1 million $\delta R_{g/h}$ values. The latest iteration of the GPR model predictions do not fully match GROMACS observations, which is why the Sobol indices are also not a valid sensitivity metric.

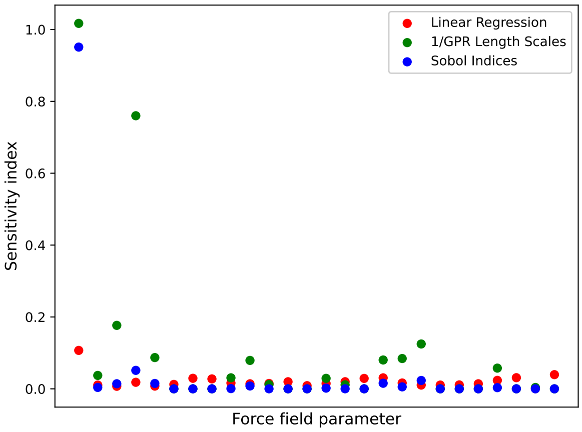

Figure 20: Comparison of 3 sensitivity analyses; Linear Regression(red), GPR length scales (green) and Sobol indices (blue). Higher sensitivity index has more effect on output variable.

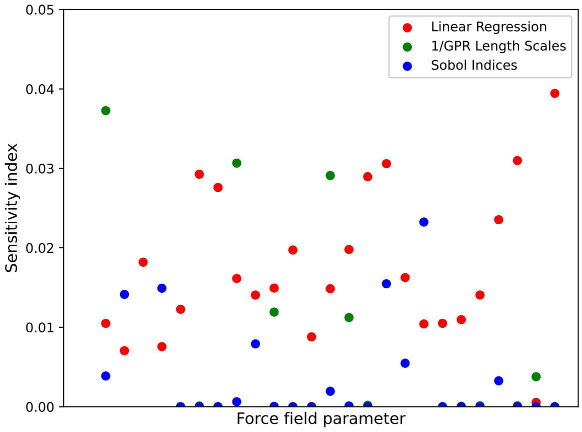

Figure 21: Comparison of 3 sensitivity analyses: Zoomed in below 5%

Investigated methods do not show a reliable trend and should only be taken as a qualitative measure.