Neural potentials of proteins extrapolate beyond training data

Geemi. P. Wellawatte
Department of Chemistry
University of Rochester
Rochester, NY, USA
gwellawa@ur.rochester.edu

Glen M. Hocky
Department of Chemistry
Simons Center for Computational Physical Chemistry
New York University
New York, NY, USA hockyg@nyu.edu

Andrew D. White[*]
Department of Chemical Engineering
University of Rochester
Rochester, NY, USA
andrew.white@rochester.edu

Abstract

We evaluate neural network (NN) coarse-grained (CG) force fields compared to traditional CG molecular mechanics force fields. We conclude NN force fields are able to extrapolate and sample from unseen regions of the free energy surface when trained with limited data. Our results come from 88 NN force fields trained on different combinations of clustered free energy surfaces from 4 protein mapped trajectories. We used a statistical measure named total variation similarity (TVS) to assess the agreement between free energy surfaces of mapped atomistic simulations and CG simulations from the trained NN force fields. Our conclusions support the hypothesis that constructing force fields on one region of the protein free energy surface can indeed extrapolate to unexplored regions. Additionally, the force matching error was found to only be weakly correlated with a force field's ability to reconstruct the correct free energy surface.

1 Introduction

Coarse-grained (CG) molecular dynamics (MD) is a tool to complement experiments.[1,2] A CG model can be considered a "reduced model" as not all degrees of freedom are considered explicitly. The objective is to eliminate irrelevant atomic details to gain computational advantages.[3] According to Noid[4], CG models provide a foundation to most scientific efforts by focusing over "essential" features of a system. CG MD enables sampling of thermodynamic systems at larger spacial and temporal scales, which are inaccessible at the all-atom resolution. As a result, CG MD is often used to study phenomena which occur at large timespans such as protein folding and multi-protein structure assemblies.[5,6] CG modeling is based implicitly on the separation of timescales between molecular motions. Therefore, this provide a practical alternative to uncover the underlying Hamiltonian of these reduced models at long timespans.[7] Works by Kidder et al.[8], ?], Noid[9],Saunders and Voth[10] and, Brini et al.[11] provide fundamental perspectives on CG modeling.

A CG model consist of two main components – 1) CG representation (mapping) and, 2) CG forcefield (FF). The first is a projection of the all-atom system into a reduced, "coarser" configurational space. A CG site which can be identified as a pseudo atom and should ideally encapsulate the physicochemical characteristics of a given group of atoms.[4,12] As we focus on CG FFs in this work, more discussions on selecting a suitable CG representation and their impact can be found in references 2, 3, 12, 13, 14, 15 and, 16. Next, a CG FF is a potential energy function which approximates the interactions between these pseudo CG atoms with

---

[*]Corresponding Author

the goal of capturing the eliminated atomistic level details.[4,17] A CG FF must be able to compute any equilibrium property that is expressed as an ensemble average of the CG coordinates.[18] Additionally, a CG FF can be thought of as a potential of mean force (PMF).[3,4,19] Foley et al.[3] highlight that a CG PMF is a configuration dependent free energy function, which should ideally preserve structural and thermodynamic properties at the lower resolution. Finding a fitting approximation of this PMF is one of the key challenges associated with CG modeling. Furthermore, since a CG model averages over the atomistic configurations, CG models are less transferable to different thermodynamic state points.[4,9,10,20] Thermodynamic inconsistency between atomistic and CG resolutions, parameterization of the CG FFs and, choice of CG representation are few factors which contribute to the limited transferability in CG modeling. According to Noid[9], these fundamental challenges arise due to the lack of understanding in the relationship between an atomistic and CG models.

Traditionally, two common approaches are used in developing CG FFs, 1) bottom-up approaches[2,21] and 2) top-down approaches.[10] A bottom-up approach rely on information from fine-grained (atomistic) models to approximate the PMFs, while a top-down approach aims to reproduce macroscopic properties.[3,4,22] However, the objective of both these approaches is that a CG model must reflect the "correct physics" of the atomistic system.[4] However, parameterizing a CG FF to approximate the behavior of an atomistic system is often a tedious and iterative task.

With the recent advances of deep learning, researchers have begun to focus on utilizing neural networks (NNs) as atomistic MD potentials[23–26] and CG potentials to study biomolecular systems.Wang et al.[19], Majewski et al.[27], Husic et al.[28], Doerr et al.[29], Zhang et al.[30] Additionally, it has been shown that NNs can be used for automating and optimizing CG mapping generation as well. Work in reference 12 is an example which demonstrate that CG mapping generation can be automated by training a graph neural network (GNN) based on human knowledge. The main assumption behind CG FF development is that, CG FFs can extrapolate to unseen regimes of the configurational space when parameterized/trained with limited data. However, little to no studies have investigated if this assumption is valid for NN based CG FFs when compared to physics informed traditional CG FFs. To address this research gap, we aim to evaluate the extrapolating capability of NNs as CG FFs. Similar to conventional CG FF, a NN based CG FF should approximate the underlying atomistic PMF as a function of the CG coordinates.

Often, a NN based CG FF is trained by minimizing the force matching error (shown in equation 1). This refers to the squared error between mapped forces (atomistic forces mapped to CG atoms) and CG forces, which are computed from the predicted CG coordinates.[31,32]

$$L_{FM} = \sum_t \|\nabla_{\mathbf{m}}\widehat{F}(\mathbf{Mx_t}, \theta) + f^{\mathbf{m}}(\mathbf{x_t})\|^2 \tag{1}$$

Here, $\mathbf{M}$ is the mapping matrix which takes the N atomistic coordinates into n CG sites. $\nabla_{\mathbf{m}}\widehat{F}(\mathbf{Mx_t}, \theta)$ represents the gradient of the learned free energy function (effective CG forces) where $\mathbf{m}$ are the CG variables. Instantaneous CG forces mapped from the all-atom trajectory are represented by the last term in the equation 1. Note that, although NNs have shown to be promising as molecular FFs,[33–35] their training can be viewed as highly dependent on availability of useful data.[32,36] Generally, the applicability of NNs raise an open question "how well can these NN FFs extrapolate beyond training data?". Another important challenge with using NNs as CG potential is the lack of interpretability when compared to traditional FFs parameterized on empirical data. Therefore, Zeni, *et al.*[36] explain that it is not trivial whether NN potentials are able to exploit the extrapolation regime, specifically when the atomistic potential energy surface (PES) is smoothened by CG representations.

In this work, we aim to investigate the extrapolating capabilities of NN based CG potentials and the impact of the amount of data used in training. We aim to discuss whether NNs are indeed apt to be used in place of traditional, physics informed models. Finally, we question if force matching by itself is adequate to benchmark the performance of trained CG FFs. To study these research questions, we selected 4 (mini)proteins based on structural properties[37] – 1) a folded protein: P-Element somatic inhibitor miniprotein (PDB ID:2BN6)[38] 2) a half folded protein: Miniature Esterase (PDB ID: 1V1D)[39] 3) a small fast-folding protein near its melting point: Trp-cage (PDB ID: 2JOF)[40] and, 4) a disordered protein: β-amyloid peptide residues 10-35 (PDB ID: 1HZ3)[41]

First, we conducted atomistic simulations for these miniproteins with GROMACS software and mapped the trajectories to generate to a CG representation. These high dimensional mapped protein trajectories were

next projected to a low dimensional free energy space (FES) with Time-structure independent component analysis.[42–44] Next, the FES were clustered using a Markov State Model (MSM) based approach to identify 4 states (conformations). Various subsamples of the these states were selected systematically to train NN based CG FFs CGSchNet[19,28] and, TorchMD-Net.[35] We trained 88 NN FFs in total (11 CGSchNet FFs and 11 TorchMD-Net FFs for each of the 4 proteins). Finally, we proceeded to produce CG simulations from each FF and to evaluate the performances of the trained FFs. Note that, we used a metric named total variation similarity[45,46] (TVS) given in equation 2 to compare the similarity between the mapped and CG FES.

$$\text{TVS} = 1 - \max_{\phi \subseteq \Omega} |P_{mapped}(\phi) - P_{CG}(\phi)| + \zeta \tag{2}$$

$$\zeta = \frac{P_{CG}(\pi) - P_{CG}(\phi)}{P_{CG}(\pi)} \tag{3}$$

$$\pi = \{x \in \Omega | P_{CG}(x) \geq P_{mapped}(x)\} \tag{4}$$

The $|P_{mapped}(\phi) - P_{CG}(\phi|$ term in equation 2 measures the maximum possible distance between the mapped and CG FES over the measurable space Ω. ϕ is the maximizer in equation 2 which represents the mapped space. Here, ζ given in equation 3, is a penalty term which accounts for the probability of the CG trajectory explored beyond the regions of the mapped trajectory. As shown in equation 4, π defines the total explored space by the CG trajectory. We use TVS metric to evaluate the performance of CG FFs because it is a system agnostic metric. Therefore, TVS can be used to compare different CG models with dissimilar FFs and CG representations.

2 Methods

2.1 Simulation methods

All-atom simulations: For each protein, all-atom MD simulations were conducted with the AMBER99SB*-ILDN forcefield[47,48] and TIP3P water model[49] with neutralizing potassium ions added. All simulations were performed in GROMACS 2020.4.[50] Minimization and equilibration were performed according to a standard protocol[51] which involves up to 50000 steps of steepest descent minimization, followed by 100 ps of NVT equilibration with backbone atoms restrained. For each protein $15\mu s$ long NPT simulations were produced at $T = 300K$ for 2BN6, $T = 290K$ for 1V1D, $T = 350K$ for 2JOF and,$T = 310K$ for 1HZ3. These temperatures were selected empirically to ensure simulation temperature is below the melting point of each protein except for 2JOF.[52? ? ,53] For 2JOF we selected a slightly higher temperature to sample from both folded and non-folded state. Production simulations used a 2 fs timestep, a 1 nm cutoff for electrostatics, the v-rescale thermostat[54] with a 0.1 ps time constant, and Parrinello-Rahman barostat[55] using a 2 ps time constant. From these production runs, training data frames were generated by restarting from fixed points along these trajectories using the same MD parameters, but with velocities resampled for each run. Starting points for restart trajectories were checkpoint files separated every 50 ns starting after 2.5 microseconds. From these 250 checkpoint files, four 10 ns simulations were performed, where positions and forces were saved in double precision every 20 ps. Each frame was then treated for periodic boundary conditions to make the molecules whole GROMACS.

Mapping atomistic simulations: To map atomistic trajectories to the CG representations, MDAnalysis software[56,57] and "fast_forward" software tool[58] were used. The latter is a specific CG mapping tool designed for MARTINI modeling. MDAnalysis software was used for mapping the trajectories for other CG models found in this study. α-Carbon CG representation was used for NN based CG modeling. For MARTINI and OpenAWSEM, their respective mapping schemes were used.[59–62]

CG simulations: After training CGSchNet and TorchMD-Net models which were the chosen NN FFs, each FF was used to conduct NVT CG simulations with Langevin dynamics at the same temperatures as the all-atom simulations (300K, 290K, 350K and 310K). A time step of 2fs were used for all FFs. Each CG trajectory was initiated from a centroid configuration randomly selected from the testing states/clusters. For example, if a NN was trained from frames with clusters labeled as 1,2,3 then the starting configuration was from cluster 4. We used this approach to avoid the impact of the starting configuration during the CG production. With CGSchNet FFs we were only able to run 50 independent trajectories which were 0.02 ns - 0.2 ns long. Most simulation were not stable beyond 0.2 ns. With TorchMD-Net we produced 2 ns long CG trajectories with 10 replicas. To perform NVT CG simulations with MARTINI3 FF, we employed

Gō-like models in combination with the MARTINI3 model[63] following the standard protocol.[64] Each CG simulation was run for 10 ns with a timestep of 20 fs using GROMACS software at constant temperatures 300K, 290K, 350K and 310K for 2BN6, 1V1D, 2JOF, 1HZ3 proteins respectively. Explicit water was used to solvate the protein system; see SI further details. For the CG simulations with OpenAWSEM FF, we used Langevin dynamics at constant temperatures 300K, 290K, 350K and 310K for 2BN6, 1V1D, 2JOF, 1HZ3 miniproteins respectively. Each CG simulation was run for 1 ns with a timestep of 2 fs. Again the standard simulation protocol was followed.[65] The CG mappings used in MARTINI and OpenAWSEM analyses were the default mappings of MARTINI[59] and AWSEM[61] FFs. Simulations from MARTINI and OpenAWSEM FFs were stable and ran to completion. More details can be found in SI.

2.2 Clustering the configuration space

Firstly, the all-atom trajectories of 2BN6,[38] 1V1D,[39] 2JOF[40] and, 1HZ3[41] miniproteins were mapped into a CG representation, where each residue was represented with its α-Carbon atom. MDAnalysis[56,57] software was used for the mapping. We used the same configuration mapping operator for the force mapping as well. A reader may find more work on force mapping in references 66 and 67. Then each mapped trajectory was clustered into four states based on a Hidden-Markov State Model (HMSM)[68,69] using the PyEMMA python library as described in the following text.[70,71] These can be thought of as metastable states. Finally, configurations (snapshots from the trajectory) from various subsets of the clustered states were used for training separate FFs. Note that, the number of frames per cluster were not equal after the assignment. To avoid oversampling from one such state, we reduced the number of frames per state to match the minimum withing the four. The total number of frames used during training were 123456, 258676, 63364 and, 204332 for 2BN6, 1V1D, 2JOF and, 1HZ3 miniproteins respectively. The total number of frames per trajectory was less than the initial atomistic trajectory, because k-means clustering algorithm failed to assign many configurations to a cluster during Markov state modeling.
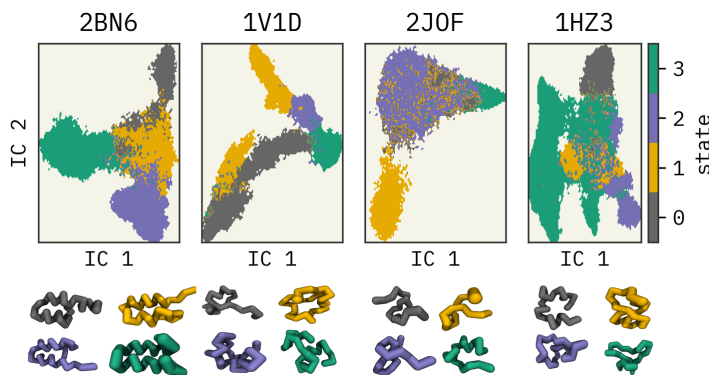

Figure 1: Four clusters of the miniproteins in the low dimensional space projected using the TICA method.[42] P-Element somatic inhibitor miniprotein (PDB ID:2BN6[38]), Miniature Esterase (PDB ID: 1V1D[39]), Trp-Cage miniprotein (PDB ID:2JOF[40]) and, β-amyloid peptide residues 10-35 (PDB ID: 1HZ3[41]) were used in this study. The centroid configuration of each cluster is illustrated in the bottom with their respective colors.

As mentioned previously, protein FES were clustered to 4 states based on a Hidden-MSM. In general, MSMs are used for analyzing dynamic data from MD simulations.[72,73] Four main steps are involved in building an MSM – 1) featurization 2) dimensionality reduction 3) clustering and 4) estimation of the transition matrix.[74] They are extensively discussed in literature on approximating observables from MD simulations.[75–79]

We selected the α-Carbon pairwise distances to featurize mapped trajectories for the dimensionality reduction. Time-lagged independent component analysis (TICA)[42–44,80] was used for this step. Next, these projected spaces were discretized using K-means[81] clustering to estimate an initial Markov State Model (MSM). 50, 75, 75 and 200 cluster centers were used for 2BN6, 1V1D, 2JOF, 1HZ3 miniproteins respectively. These cluster numbers were selected based on the VAMP2 scores.[82] These are the sums of singular values of the symmetrized MSM transition matrix; see SI. Respective lags of 100, 100, 100 and 10 were selected to build the MSMs. Lags were selected such that the implied timescales were constant with the statistical error (see SI). Furthermore, we validated the MSMs using Chapman–Kolmogorov tests (shown in SI).[83,84] Finally,

Table 1: Cluster combinations used for training

Cluster Percentage	100%	75%	50%	25%
FF label: clusters used in training	*FF0:* 1,2,3,4	*FF1:* 2,3,4 *FF2:* 1,3,4 *FF3:* 1,2,4 *FF4:* 1,2,3	*FF5:* 1,2 *FF6:* 3,4	*FF7:* 1 *FF8:* 2 *FF9:* 3 *FF10:* 4

HMSMs were estimated based on the reference MSMs where each trajectory was clustered into four coarser groups where each frame of the trajectories were assigned to a cluster. Christoforou et al.[85] describe a HMSM as a "kinetic" coarse-graining model which group the microstates identified by the k-means clustering algorithm. We followed a similar approach as Christoforou et al.[85] to cluster the projected configurational space. Figure 1 illustrates the four clusters of the reduced dimensional spaces along the first two independent components (IC1 and IC2) identified by TICA[42] and their centroid configurations. Further details for this procedure are given in the SI.

2.3 Training forcefields and running CG simulations

In this study, we selected two NN-based CG FFs; CGSchNet[28] and TorchMD-Net.[35] CGSchNet[28] is a modified version of the CGNet model[19] which learns the CG FES based on the force matching approach. In CGNet model, the inputs are hand-selected features such as bond distances, angles and dihedrals. However, in the CGSchNet model, the features are "learned" during training by leveraging the SchNet model.[25,86] Additionally, TorchMD-Net[35] provide state-of-the-art graph neural network (GNN) and equivariant transformer (ET) based NN potentials for molecular simulations. In this work, we used TorchMD-Net's GNN model for training CG FFs since the performance from both models were similar. The main difference between the two NN architectures arise from input featurization where CGSchNet model uses SchNet features and TorchMD-Net model embeds the atom types into a fixed embedding. Both architectures are based on the PyTorch[87] deep learning model builder. A technically oriented reader may further detail on the differences between the two models in their open source GitHub repositories; https://github.com/coarse-graining/cgnet and https://github.com/torchmd/torchmd. Note that we used TorchMD[29] which is the Python API for performing CG MD simulations using the TorchMD-Net trained FFs. CGSchNet tool is equipped with its own Python script for performing the simulations.

The key objective of this work is to investigate if NN CG FFs are able to extrapolate and sample from unseen regions of the reference FES. Therefore, during training, we subsampled different sets of identified clusters and trained multiple independent FFs per miniprotein (listed in Table 2.3). For example, to train "FF1" we used 75% of clusters, those labeled as 2,3 and 4 – data from cluster 1 were withheld. Note that the labels were generated randomly. The number of frames for each cluster were kept constant through downsampling to match the minimum number of frames among the 4 states with the intention of avoiding oversampling from one cluster. Hyperparameters used in training and train-validation error plots can be found in SI. Finally, the trained FFs were used to produce CG simulations (see Simulation methods section). Note that, due to the smoothness of the underlying CG FES, similar amount of sampling is obtained in these "ns" long simulations as compared to the original microseconds of training data, as well be shown shortly.

3 Results and discussion

First, we compared the performances of the *FF0* from CGSchNet and TorchMD-Net (trained with data from all four clusters) with state-of-the-art, physics informed FFs MARTINI[59,60] and OpenAWSEM.[62] MARTINI is possibly the most popularly used FF in CG simulations[88] of lipids,[89,90] proteins,[91,92] sugars[93] and other biomolecules.[94,95] OpenAWSEM is the implementation of AWSEM[61] CG FF for proteins within the GPU compatible OpenMM framework. AWSEM contains physics informed many-body effects and employ an implicit solvent environment.[61] This FF has been successfully applied to study protein structure prediction.[96–98]
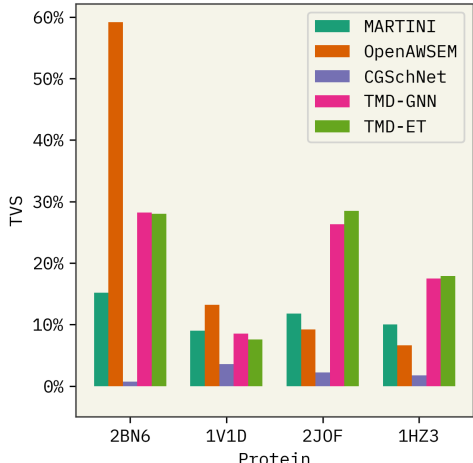
Figure 2: Comparison of CGSchNet and TorchMD-Net potentials with state-of-the-art methods MARTINI and OpenAWSEM. TMD-GNN and TMD-ET refer to the TorchMD-Net's graph neural network and equivariant transformer models. Higher TVS refers to high similarity between mapped and CG simulation distributions in the projected TICA spaces. 100% data were used to train the latter two FFs. PDB IDs of the 4 protein systems used are shown in the x-axis.

Based on the comparison illustrated in Figure 2 we observe the following; a) Performance of the OpenAWSEM FF decreases significantly with the increasing structural disorder of the miniproteins whereas MARTINI3 is not affected, b) CGSchNet has the lowest overall performance among all four FFs, c) TorchMD-Net's performance is comparable among all four proteins regardless of the structural disorder, d) TorchMD-Net's GNN and ET model demonstrate similar performances.

First, CG trajectories from forcefields *FF0* from CGSchNet, TorchMD-Net (GNN and ET models), MARTINI and OpenAWSEM were projected onto the first two independent components (ICs) identified for the mapped trajectory with TICA. The similarity between the two FES were calculated with TVS,[45,46] see equation 2. TVS metric is used to evaluate the performance of the CG models with respect to the atomistic references. At the bottom of figure 3, the cartoon representations of the 4 miniproteins and 20 frames from the mapped trajectories are shown. The structural disorder of the proteins increase from left to right in the figure (1HZ3 has the highest disorder).[37] Note that the projected mapped FES of MARTINI and OpenAWSEM FFs are dissimilar to the FES from the two NN-FFs due to the used CG representations. The projected space is a function of the CG coordinates. In CGSchNet and TorchMD-Net CG models, each amino acid was represented with their α-Carbons, while in MARTINI and OpenAWSEM their specific mappings (4 heavy atom to 1 CG atom) were used.[59,62]

We observed that the CG trajectories of the CGSchNet FFs explore a broader region in their 2D FES beyond the reference trajectory, resulting in lower TVS scores. This observation can be interpreted that CGSchNet-FF0 tend to explore from physically non-meaningful regions. This could explain why the CG simulations from CGSchNet FFs did not run to completion for all 44 simulations. Total trajectory times were between 0.02 ns - 0.2 ns and none of the CG simulations were stable beyond 0.2 ns. Although we attempted to improve the performance of CGSchNet FFs with hyperparameter tuning, we were unsuccessful in our attempts. While we were able to minimize the "over extrapolation" by increasing the friction term in Langevin Dynamics, the overall performance was not significantly improved. Wang et al.[19] state that a prior energy term was added to their GNN architecture to avoid sampling from physically non-meaningful regions. Therefore, we expect that by optimizing the prior term, CGSchNet maybe improved. However, we did not attempt to alter the initial architecture, as this was beyond the scope of our work.

In comparison, we were able to produce longer, stable simulations for 2 ns each with TorchMD-Net. Note that 2 ns in CG coordinates are comparable to the microsecond lengthscale of the atomistic simulations. We selected this cut off specifically to evaluate the robustness of CG potentials to low availability of training data. Furthermore, we observe in figure 3 that TorchMD-Net is able to explore the mapped trajectory within the bounds while avoiding physically non-meaningful regions. This demonstrates the capability of
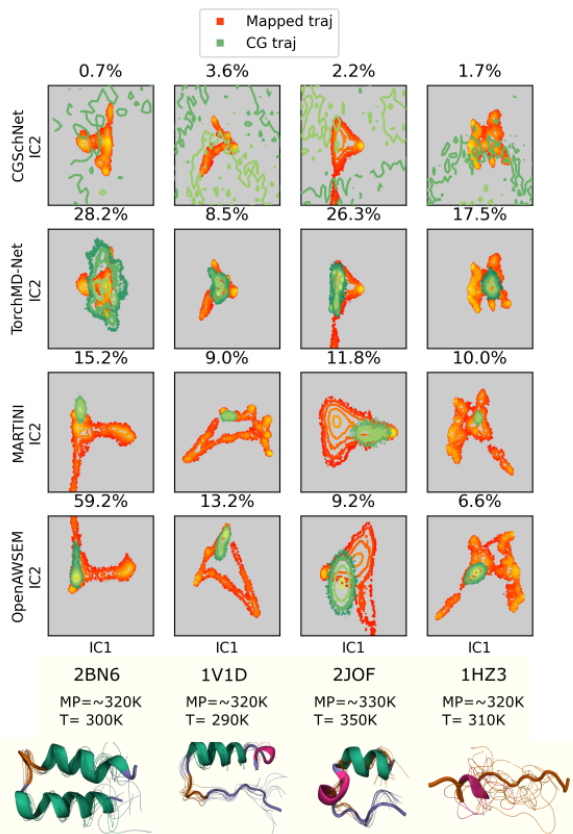
Figure 3: Mapped and CG FES from *FF0* – FFs trained with all 4 states. Top: projected miniprotein mapped and CG trajectories from CGSchNet, TorchMD-Net, MARTINI and OpenAWSEM FFs. Total variation similarity (TVS) between the mapped and CG FES are annotated as a percentage. Higher TVS indicates higher similarity. Bottom: cartoon representations of miniprotein trajectories annotated with approximate melting and simulation temperatures. P-Element somatic inhibitor miniprotein (PDB ID:2BN6), Miniature Esterase (PDB ID: 1V1D), Trp-Cage miniprotein (PDB ID:2JOF) and, β-amyloid peptide residues 10-35 (PDB ID: 1HZ3) were used in this study. Conformational ensembles are 20 random frames after a weighted iterative alignment following the procedure of Ref. 99.

NNs to be used as CG FFs with optimized model architectures. With these observations, we conclude that TorchMD-Net outperforms CGSchNet and the performance is consistent across all 4 proteins with varying degrees of disorder. These observations raise another question, "does that the model architecture significantly impact the performance of an FF?". To answer this non-trivial question, we compared the performances of TorchMD-Net's GNN and ET models. Based on the results in figure 2, we note that both GNN and ET models have almost identical TVS values. This suggests that the role of the model architecture does not play a significant role in the overall behavior. However, further investigations are needed to establish a profound conclusion. We did not pursue to answer this question, as the main aim of this work is to investigate if CG NN-FFs are able to extrapolate. Also note that, with hyperparameter tuning and architectural changes, the performance of the FFs can be improved.

Additionally, TorchMD-Net's performance is comparable to MARTINI and OpenAWSEM FFs when trained with all available data. It can also be seen in figure 3, while TVS of MARTINI and OpenAWSEM FFs are comparatively higher, they tend to be localized around the starting configurations unlike TorchMD simulations. We expect that this restricted sampling of MARTINI and OpenAWSEM FFs can be improved by increasing the CG simulation length. However, note that TorchMD-Net simulations can explore the reference FES more within the same time scale. Furthermore, OpenAWSEM is seemingly affected by the structural changes of the miniproteins while the NN CG FFs are indifferent. This is a noteworthy observation as one of the main challenges in the field of CG modeling is lack of transferrable FFs. Based on these observations, we conclude that NN based CG FFs can satisfactorily learn the physicochemical behaviors of the underlying
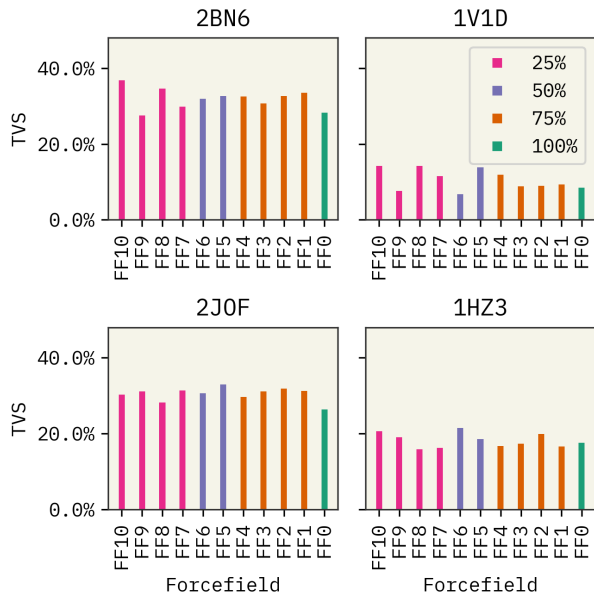
Figure 4: Impact of the amount of data in training of TorchMD-Net FFs. Labels of the FFs indicate the percentage of states used in training. See Table 2.3.

systems and are not yet utilized to the full capacity. While NN provide the advantages such as ease of training and transferability, conventional FFs are better at capturing the physical knowledge. Therefore, we can be hopeful that NN can be integrated with empirical CG models to expand the boundaries of research.

Next, we proceeded to focus on the impact of training data on the performance of TorchMD-Net FFs. Note that we did not pursue this investigation with CGSchNet model due to the poor performance observed previously. Figure 4 illustrates the performances of TorchMD-Net FFs trained with various combinations of clusters. Surprisingly, we observe that the percentage of clusters used in training is not strongly correlated with the performance of the FFs. For example, we see that the TVS of *FF0* trained with data from all 4 states is comparable to *FF7-10* trained with data from only one cluster. This validates the hypothesis that CG FFs indeed can extrapolate beyond the available knowledge and having large amounts of training data does not necessarily improve the performance. Fu et al.[31], arrived at a similar conclusion, where they observed that the performance of learned NN FFs cannot be improved by increasing the amount of training data. Stocker et al.[100] state that one way of improving the robustness of NN FFs is by including distorted and off-equilibrium conformations during training.

During training of the NN based FFs, we observed that the both CGSchNet and TorchMD-Net had similar validation errors. However, their performances varied drastically in the simulation phase. In other words, both NN FFs showed similar trends during training and lacked overfitting whilst the stability of CG simulations were significantly different. As we mentioned earlier, we faced difficulty in conducting stable, long simulations with CGSchNet. Baffled by this, we questioned, "isn't a low test error an indication of the overall performance?". Therefore, we assessed the suitability of benchmarking a NN based FF against the force matching error. We compared the force matching error (validation error) on the test data of 88 CGSchNet and TorchMD-Net FFs with their respective TVS values. This aligns with the findings by Fu et al.[31], who showed that machine learning FFs with a lower force matching error is not an indication of the performance. They showed that learned FFs can fail to reproduce simulation based observable such as radial distribution functions and to produce stable simulations. Our results shown in figure 5 also indicate that, although the test errors from both CGschNet TorchMD-Net models only differ by $\pm$ 0.2 kcal/(mol.Å) their TVS differ by $\sim 20\%$. We can conclude there is only a weak correlation between force matching error and the FF's expected behavior as indicated by TVS. Therefore, we highlight the force matching error should not be the only benchmark when developing FFs.[31,101]
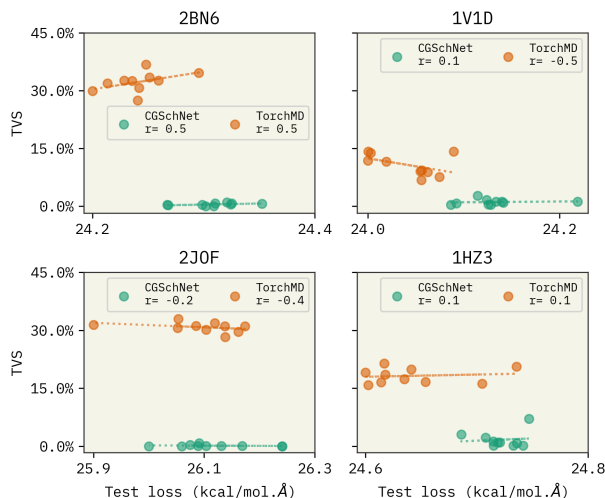
Figure 5: Variation of TVS with force matching error of all CGSchNet and TorchMD-Net FFs. The x-axis denotes average validation error of the last 3 epochs. TVS indicates the similarity between mapped and CG trajectories from the trained FFs.

4 Conclusions

Based on our results, we observe that TorchMD-Net trained with limited data is comparable to the two physics informed FFs - (MARTINI[59] and OpenAWSEM[62]). Unlike CGSchNet, TorchMD-Net FFs strictly explore around the same FES as the mapped trajectories indicating, TorchMD-Net FFs tend to avoid physically improbable configurations. We also observe that the amount used in training does not impact the overall performance of FFs trained with TorchMD-Net, for proteins with varying degrees of protein disorder. This lessens the need to generate long scale atomistic trajectories with millions of data frames, which are already time and resource expensive. Mainly, we observe that while NN FFs are unaffected by the availability of data, they are able to extrapolate to unseen regions of the FES. When compared with empirical FFs MARTINI and OpenAWSEM, TorchMD-Net has significant advantages – easy to train, explore a large region of the reference FES within a short timespan, higher transferability. Additionally, we find that force matching error NN of CG FF is not strongly correlated with a model's accuracy. This highlights the need to explore a better benchmark for FF training. However, we are hopeful that research can be significantly accelerated with the integrated utility of deep learning in conventional CG models.

5 Supplementary Materials

We have included methodologies, data and results used during training, conducting CG simulations in the Supplementary Materials.

- Figure S1: Discretized reduced 2D configurational spaces with k-means centers and Implied timescale plots.
- Figures S2-S5: Chapman–Kolmogorov tests for proteins 2BN6, 1V1D, 2JOF, 1HZ3
- Figure S6: Train and validation error plots of TorchMD-Net trained FFs.
- Figure S7: Train and validation error plots of CGSchNet trained FFs.
- Figure S8: Mapped and CG FES of the trained TorchMD-Net's GNN FFs.
- Figure S9: Mapped and CG FES of the trained TorchMD-Net's ET FF0.
- Figure S10: Total energy during CG simulations from TochMD-Net FF0.
- Figure S11: Performance of all CGSchNet FFs.
- Figure S12: Mapped and CG FES of the trained CGSchNet FFs.
- Table S1: Parameters used in cluster assignment of mapped trajectories.

- Table S2: Hyperparameters used for training TorchMD-Net forcefields.
- Table S3: Hyperparameters used for training CGSchNet forcefields.
- Table S4: CG simulations parameters: Trained TorchMD-Net FFs.
- Table S5: CG simulations parameters: Trained CGSchNet FFs.

6 Acknowledgement

References

[1] Gary S. Ayton and Gregory A. Voth. Simulation of biomolecular systems at multiple length and time scales. *International Journal for Multiscale Computational Engineering*, 2(2), 2004. ISSN 1543-1649.

[2] Sergei Izvekov and Gregory A. Voth. A multiscale coarse-graining method for biomolecular systems. *The Journal of Physical Chemistry B*, 109(7):2469–2473, 2005. doi:10.1021/jp044629q. URL `https://doi.org/10.1021/jp044629q`. PMID: 16851243.

[3] Thomas T. Foley, M. Scott Shell, and W. G. Noid. The impact of resolution upon entropy and information in coarse-grained models. *The Journal of Chemical Physics*, 143(24):243104, 2015. doi:10.1063/1.4929836. URL `https://doi.org/10.1063/1.4929836`.

[4] W. G. Noid. Perspective: Coarse-grained models for biomolecular systems. *The Journal of Chemical Physics*, 139(9):090901, 2013. doi:10.1063/1.4818908. URL `https://doi.org/10.1063/1.4818908`.

[5] Sebastian Kmiecik, Dominik Gront, Michal Kolinski, Lukasz Wieteska, Aleksandra Elzbieta Dawid, and Andrzej Kolinski. Coarse-grained protein models and their applications. *Chemical Reviews*, 116(14): 7898–7936, 2016. doi:10.1021/acs.chemrev.6b00163. URL `https://doi.org/10.1021/acs.chemrev.6b00163`. PMID: 27333362.

[6] Vladimir Voynov and Justin A Caravella. *Therapeutic proteins: methods and protocols*. Springer, 2012. doi:10.1007/978-1-61779-921-1_27.

[7] Cecilia Clementi. Coarse-grained models of protein folding: toy models or predictive tools? *Current Opinion in Structural Biology*, 18(1):10–15, 2008. ISSN 0959-440X. doi:https://doi.org/10.1016/j.sbi.2007.10.005. URL `https://www.sciencedirect.com/science/article/pii/S0959440X07001753`. Folding and Binding / Protein-nucleic acid interactions.

[8] Katherine M. Kidder, Ryan J. Szukalo, and W. G. Noid. Energetic and entropic considerations for coarse-graining. *The European Physical Journal B*, 94(7):153, Jul 2021. ISSN 1434-6036. doi:10.1140/epjb/s10051-021-00153-4. URL `https://doi.org/10.1140/epjb/s10051-021-00153-4`.

[9] W. G. Noid. *Systematic Methods for Structurally Consistent Coarse-Grained Models*, pages 487–531. Humana Press, Totowa, NJ, 2013. ISBN 978-1-62703-017-5. doi:10.1007/978-1-62703-017-5_19. URL `https://doi.org/10.1007/978-1-62703-017-5_19`.

[10] Marissa G. Saunders and Gregory A. Voth. Coarse-graining methods for computational biology. *Annual Review of Biophysics*, 42(1):73–93, 2013. doi:10.1146/annurev-biophys-083012-130348. URL `https://doi.org/10.1146/annurev-biophys-083012-130348`. PMID: 23451897.

[11] Emiliano Brini, Elena A. Algaer, Pritam Ganguly, Chunli Li, Francisco Rodríguez-Ropero, and Nico F. A. van der Vegt. Systematic coarse-graining methods for soft matter simulations – a review. *Soft Matter*, 9:2108–2119, 2013. doi:10.1039/C2SM27201F. URL `http://dx.doi.org/10.1039/C2SM27201F`.

[12] Zhiheng Li, Geemi P. Wellawatte, Maghesree Chakraborty, Heta A. Gandhi, Chenliang Xu, and Andrew D. White. Graph neural network based coarse-grained mapping prediction. *Chem. Sci.*, 11: 9524–9531, 2020. doi:10.1039/D0SC02458A. URL `http://dx.doi.org/10.1039/D0SC02458A`.

[13] Maghesree Chakraborty, Chenliang Xu, and Andrew D. White. Encoding and selecting coarse-grain mapping operators with hierarchical graphs. *The Journal of Chemical Physics*, 149(13):134106, 2018. doi:10.1063/1.5040114. URL `https://doi.org/10.1063/1.5040114`.

[14] Maghesree Chakraborty, Jinyu Xu, and Andrew D White. Is preservation of symmetry necessary for coarse-graining? *Physical Chemistry Chemical Physics*, 22(26):14998–15005, 2020.

[15] Charly Empereur-Mot, Luca Pesce, Giovanni Doni, Davide Bochicchio, Riccardo Capelli, Claudio Perego, and Giovanni M. Pavan. Swarm-cg: Automatic parametrization of bonded terms in martini-based coarse-grained models of simple to complex molecules via fuzzy self-tuning particle swarm optimization. *ACS Omega*, 5(50):32823–32843, 2020. doi:10.1021/acsomega.0c05469. URL `https://doi.org/10.1021/acsomega.0c05469`. PMID: 33376921.

[16] Zhiyong Zhang, Lanyuan Lu, Will G Noid, Vinod Krishna, Jim Pfaendtner, and Gregory A Voth. A systematic methodology for defining coarse-grained sites in large biomolecules. *Biophysical journal*, 95 (11):5073–5083, 2008.

[17] Jay W. Ponder and David A. Case. Force fields for protein simulations. In *Protein Simulations*, volume 66 of *Advances in Protein Chemistry*, pages 27–85. Academic Press, 2003. doi:https://doi.org/10.1016/S0065-3233(03)66002-X. URL `https://www.sciencedirect.com/science/article/pii/S006532330366002X`.

[18] Jonas Köhler, Yaoyi Chen, Andreas Krämer, Cecilia Clementi, and Frank Noé. Force-matching coarse-graining without forces. *arXiv preprint arXiv:2203.11167*, 2022.

[19] Jiang Wang, Simon Olsson, Christoph Wehmeyer, Adrià Pérez, Nicholas E. Charron, Gianni de Fabritiis, Frank Noé, and Cecilia Clementi. Machine learning of coarse-grained molecular dynamics force fields. *ACS Central Science*, 5(5):755–767, 2019. doi:10.1021/acscentsci.8b00913. URL `https://doi.org/10.1021/acscentsci.8b00913`. PMID: 31139712.

[20] Aleksander E.P. Durumeric, Nicholas E. Charron, Clark Templeton, Félix Musil, Klara Bonneau, Aldo S. Pasos-Trejo, Yaoyi Chen, Atharva Kelkar, Frank Noé, and Cecilia Clementi. Machine learned coarse-grained protein force-fields: Are we there yet? *Current Opinion in Structural Biology*, 79: 102533, 2023. ISSN 0959-440X. doi:https://doi.org/10.1016/j.sbi.2023.102533. URL `https://www.sciencedirect.com/science/article/pii/S0959440X23000076`.

[21] Dirk Reith, Mathias Pütz, and Florian Müller-Plathe. Deriving effective mesoscale potentials from atomistic simulations. *Journal of Computational Chemistry*, 24(13):1624–1636, 2003. doi:https://doi.org/10.1002/jcc.10307. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/jcc.10307`.

[22] Zack Jarin, James Newhouse, and Gregory A. Voth. Coarse-grained force fields from the perspective of statistical mechanics: Better understanding of the origins of a martini hangover. *Journal of Chemical Theory and Computation*, 17(2):1170–1180, 2021. doi:10.1021/acs.jctc.0c00638. URL `https://doi.org/10.1021/acs.jctc.0c00638`. PMID: 33475352.

[23] Jörg Behler and Michele Parrinello. Generalized neural-network representation of high-dimensional potential-energy surfaces. *Phys. Rev. Lett.*, 98:146401, Apr 2007. doi:10.1103/PhysRevLett.98.146401. URL `https://link.aps.org/doi/10.1103/PhysRevLett.98.146401`.

[24] Stephan Thaler and Julija Zavadlav. Learning neural network potentials from experimental data via differentiable trajectory reweighting. *Nature communications*, 12(1):6884, 2021.

[25] Kristof Schütt, Pieter-Jan Kindermans, Huziel Enoc Sauceda Felix, Stefan Chmiela, Alexandre Tkatchenko, and Klaus-Robert Müller. Schnet: A continuous-filter convolutional neural network for modeling quantum interactions. *Advances in neural information processing systems*, 30, 2017.

[26] J. S. Smith, O. Isayev, and A. E. Roitberg. Ani-1: an extensible neural network potential with dft accuracy at force field computational cost. *Chem. Sci.*, 8:3192–3203, 2017. doi:10.1039/C6SC05720A. URL `http://dx.doi.org/10.1039/C6SC05720A`.

[27] Maciej Majewski, Adrià Pérez, Philipp Thölke, Stefan Doerr, Nicholas E Charron, Toni Giorgino, Brooke E Husic, Cecilia Clementi, Frank Noé, and Gianni De Fabritiis. Machine learning coarse-grained potentials of protein thermodynamics. *arXiv preprint arXiv:2212.07492*, 2022.

[28] Brooke E. Husic, Nicholas E. Charron, Dominik Lemm, Jiang Wang, Adrià Pérez, Maciej Majewski, Andreas Krämer, Yaoyi Chen, Simon Olsson, Gianni de Fabritiis, Frank Noé, and Cecilia Clementi. Coarse graining molecular dynamics with graph neural networks. *The Journal of Chemical Physics*, 153(19):194101, 2020. doi:10.1063/5.0026133. URL `https://doi.org/10.1063/5.0026133`.

[29] Stefan Doerr, Maciej Majewsk, Adrià Pérez, Andreas Krämer, Cecilia Clementi, Frank Noe, Toni Giorgino, and Gianni De Fabritiis. Torchmd: A deep learning framework for molecular simulations, 2020.

[30] Linfeng Zhang, Jiequn Han, Han Wang, Roberto Car, and Weinan E. Deepcg: Constructing coarse-grained models via deep neural networks. *The Journal of Chemical Physics*, 149(3):034101, 2018. doi:10.1063/1.5027645. URL `https://doi.org/10.1063/1.5027645`.

[31] Xiang Fu, Zhenghao Wu, Wujie Wang, Tian Xie, Sinan Keten, Rafael Gomez-Bombarelli, and Tommi Jaakkola. Forces are not enough: Benchmark and critical evaluation for machine learning force fields with molecular simulations. *arXiv preprint arXiv:2210.07237*, 2022.

[32] Frank Noé, Alexandre Tkatchenko, Klaus-Robert Müller, and Cecilia Clementi. Machine learning for molecular simulation. *Annual Review of Physical Chemistry*, 71(1):361–390, 2020. doi:10.1146/annurev-physchem-042018-052331. URL `https://doi.org/10.1146/annurev-physchem-042018-052331`. PMID: 32092281.

[33] Alireza Khorshidi and Andrew A. Peterson. Amp: A modular approach to machine learning in atomistic simulations. *Computer Physics Communications*, 207:310–324, 2016. ISSN 0010-4655. doi:https://doi.org/10.1016/j.cpc.2016.05.010. URL `https://www.sciencedirect.com/science/article/pii/S0010465516301266`.

[34] Stefan Chmiela, Alexandre Tkatchenko, Huziel E. Sauceda, Igor Poltavsky, Kristof T. Schütt, and Klaus-Robert Müller. Machine learning of accurate energy-conserving molecular force fields. *Science Advances*, 3(5):e1603015, 2017. doi:10.1126/sciadv.1603015. URL `https://www.science.org/doi/abs/10.1126/sciadv.1603015`.

[35] Philipp Thölke and Gianni De Fabritiis. Torchmd-net: Equivariant transformers for neural network based molecular potentials. *arXiv preprint arXiv:2202.02541*, 2022.

[36] Claudio Zeni, Andrea Anelli, Aldo Glielmo, and Kevin Rossi. Exploring the robust extrapolation of high-dimensional machine learning potentials. *Phys. Rev. B*, 105:165141, Apr 2022. doi:10.1103/PhysRevB.105.165141. URL `https://link.aps.org/doi/10.1103/PhysRevB.105.165141`.

[37] Paul Robustelli, Stefano Piana, and David E. Shaw. Developing a molecular dynamics force field for both folded and disordered protein states. *Proceedings of the National Academy of Sciences*, 115(21): E4758–E4766, 2018. doi:10.1073/pnas.1800690115. URL `https://www.pnas.org/doi/abs/10.1073/pnas.1800690115`.

[38] Tijana Ignjatovic, Ji-Chun Yang, Jonathan Butler, David Neuhaus, and Kiyoshi Nagai. Structural basis of the interaction between p-element somatic inhibitor and u1-70k essential for the alternative splicing of p-element transposase. *Journal of Molecular Biology*, 351(1):52–65, 2005. ISSN 0022-2836. doi:https://doi.org/10.1016/j.jmb.2005.04.077. URL `https://www.sciencedirect.com/science/article/pii/S0022283605005437`.

[39] Andrew J. Nicoll and Rudolf K. Allemann. Nucleophilic and general acid catalysis at physiological ph by a designed miniature esterase. *Org. Biomol. Chem.*, 2:2175–2180, 2004. doi:10.1039/B404730C. URL `http://dx.doi.org/10.1039/B404730C`.

[40] Bipasha Barua, Jasper C. Lin, Victoria D. Williams, Phillip Kummler, Jonathan W. Neidigh, and Niels H. Andersen. The Trp-cage: optimizing the stability of a globular miniprotein. *Protein Engineering, Design and Selection*, 21(3):171–185, 01 2008. doi:10.1093/protein/gzm082. URL `https://doi.org/10.1093/protein/gzm082`.

[41] S. Zhang, K. Iwata, M.J. Lachenmann, J.W. Peng, S. Li, E.R. Stimson, Y. a. Lu, A.M. Felix, J.E. Maggio, and J.P. Lee. The alzheimer's peptide aβ adopts a collapsed coil structure in water. *Journal of Structural Biology*, 130(2):130–141, 2000. ISSN 1047-8477. doi:https://doi.org/10.1006/jsbi.2000.4288. URL `https://www.sciencedirect.com/science/article/pii/S1047847700942886`.

[42] L. Molgedey and H. G. Schuster. Separation of a mixture of independent signals using time delayed correlations. *Phys. Rev. Lett.*, 72:3634–3637, Jun 1994. doi:10.1103/PhysRevLett.72.3634. URL `https://link.aps.org/doi/10.1103/PhysRevLett.72.3634`.

[43] Yusuke Naritomi and Sotaro Fuchigami. Slow dynamics of a protein backbone in molecular dynamics simulation revealed by time-structure based independent component analysis. *The Journal of Chemical Physics*, 139(21):12B605_1, 2013.

[44] Steffen Schultze and Helmut Grubmüller. Time-lagged independent component analysis of random walks and protein dynamics. *Journal of Chemical Theory and Computation*, 17(9):5766–5776, 2021. doi:10.1021/acs.jctc.1c00273. URL `https://doi.org/10.1021/acs.jctc.1c00273`. PMID: 34449229.

[45] Sergio Verdú. Total variation distance and the distribution of relative information. In *2014 Information Theory and Applications Workshop (ITA)*, pages 1–3, 2014. doi:10.1109/ITA.2014.6804281.

[46] Taolue Chen and Stefan Kiefer. On the total variation distance of labelled markov chains. CSL-LICS '14, New York, NY, USA, 2014. Association for Computing Machinery. ISBN 9781450328869. doi:10.1145/2603088.2603099. URL https://doi.org/10.1145/2603088.2603099.

[47] Robert B Best and Gerhard Hummer. Optimized molecular dynamics force fields applied to the helix-coil transition of polypeptides. *J. Phys. Chem. B*, 113(26):9004–9015, 2009.

[48] Kresten Lindorff-Larsen, Stefano Piana, Kim Palmo, Paul Maragakis, John L Klepeis, Ron O Dror, and David E Shaw. Improved side-chain torsion potentials for the amber ff99sb protein force field. *Proteins*, 78(8):1950–1958, 2010.

[49] William L Jorgensen, Jayaraman Chandrasekhar, Jeffry D Madura, Roger W Impey, and Michael L Klein. Comparison of simple potential functions for simulating liquid water. *J. Chem. Phys.*, 79(2):926–935, 1983.

[50] Mark James Abraham, Teemu Murtola, Roland Schulz, Szilárd Páll, Jeremy C Smith, Berk Hess, and Erik Lindahl. Gromacs: High performance molecular simulations through multi-level parallelism from laptops to supercomputers. *SoftwareX*, 1:19–25, 2015.

[51] http://www.mdtutorials.com/gmx/lysozyme/index.html.

[52] Nikolas H Chmiel, Donald C Rio, and Jennifer A Doudna. Distinct contributions of kh domains to substrate binding affinity of drosophila p-element somatic inhibitor protein. *Rna*, 12(2):283–291, 2006.

[53] Ruhong Zhou. Trp-cage: folding free energy landscape in explicit water. *Proceedings of the National Academy of Sciences*, 100(23):13280–13285, 2003.

[54] Giovanni Bussi, Davide Donadio, and Michele Parrinello. Canonical sampling through velocity rescaling. *J. Chem. Phys.*, 126(1):014101, 2007.

[55] Michele Parrinello and Aneesur Rahman. Polymorphic transitions in single crystals: A new molecular dynamics method. *J. App. Phys.*, 52(12):7182–7190, 1981.

[56] Naveen Michaud-Agrawal, Elizabeth J Denning, Thomas B Woolf, and Oliver Beckstein. Mdanalysis: a toolkit for the analysis of molecular dynamics simulations. *Journal of computational chemistry*, 32(10):2319–2327, 2011.

[57] Richard J Gowers, Max Linke, Jonathan Barnoud, Tyler JE Reddy, Manuel N Melo, Sean L Seyler, Jan Domanski, David L Dotson, Sébastien Buchoux, Ian M Kenney, et al. Mdanalysis: a python package for the rapid analysis of molecular dynamics simulations. In *Proceedings of the 15th python in science conference*, volume 98, page 105. SciPy Austin, TX, 2016.

[58] Fastforward github. https://github.com/fgrunewald/fast_forward.

[59] Siewert J. Marrink, Alex H. de Vries, and Alan E. Mark. Coarse grained model for semiquantitative lipid simulations. *The Journal of Physical Chemistry B*, 108(2):750–760, 2004. doi:10.1021/jp036508g. URL https://doi.org/10.1021/jp036508g.

[60] Siewert J. Marrink, H. Jelger Risselada, Serge Yefimov, D. Peter Tieleman, and Alex H. de Vries. The martini force field: Coarse grained model for biomolecular simulations. *The Journal of Physical Chemistry B*, 111(27):7812–7824, 2007. doi:10.1021/jp071097f. URL https://doi.org/10.1021/jp071097f. PMID: 17569554.

[61] Aram Davtyan, Nicholas P. Schafer, Weihua Zheng, Cecilia Clementi, Peter G. Wolynes, and Garegin A. Papoian. Awsem-md: Protein structure prediction using coarse-grained physical potentials and bioinformatically based local structure biasing. *The Journal of Physical Chemistry B*, 116(29):8494–8503, 2012. doi:10.1021/jp212541y. URL https://doi.org/10.1021/jp212541y. PMID: 22545654.

[62] Wei Lu, Carlos Bueno, Nicholas P Schafer, Joshua Moller, Shikai Jin, Xun Chen, Mingchen Chen, Xinyu Gu, Aram Davtyan, Juan J de Pablo, et al. Openawsem with open3spn2: A fast, flexible, and accessible framework for large-scale coarse-grained biomolecular simulations. *PLoS computational biology*, 17(2):e1008308, 2021.

[63] Adolfo B. Poma, Marek Cieplak, and Panagiotis E. Theodorakis. Combining the martini and structure-based coarse-grained approaches for the molecular dynamics studies of conformational transitions in proteins. *Journal of Chemical Theory and Computation*, 13(3):1366–1374, 2017. doi:10.1021/acs.jctc.6b00986. URL https://doi.org/10.1021/acs.jctc.6b00986. PMID: 28195464.

[64] Martini3 tutorial. http://cgmartini.nl/index.php/2021-martini-online-workshop/tutorials/564-2-proteins-basic-and-martinize-2#GoProteins.

[65] Openawsem github. https://github.com/npschafer/openawsem.

[66] Giovanni Ciccotti, Raymond Kapral, and Eric Vanden-Eijnden. Blue moon sampling, vectorial reaction coordinates, and unbiased constrained dynamics. *ChemPhysChem*, 6(9):1809–1814, 2005.

[67] Andreas Krämer, Aleksander P Durumeric, Nicholas E Charron, Yaoyi Chen, Cecilia Clementi, and Frank Noé. Statistically optimal force aggregation for coarse-graining molecular dynamics. *arXiv preprint arXiv:2302.07071*, 2023.

[68] Leonard E Baum and Ted Petrie. Statistical inference for probabilistic functions of finite state markov chains. *The annals of mathematical statistics*, 37(6):1554–1563, 1966.

[69] L.R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989. doi:10.1109/5.18626.

[70] Martin K. Scherer, Benjamin Trendelkamp-Schroer, Fabian Paul, Guillermo Pérez-Hernández, Moritz Hoffmann, Nuria Plattner, Christoph Wehmeyer, Jan-Hendrik Prinz, and Frank Noé. PyEMMA 2: A Software Package for Estimation, Validation, and Analysis of Markov Models. *Journal of Chemical Theory and Computation*, 11:5525–5542, October 2015. ISSN 1549-9618. doi:10.1021/acs.jctc.5b00743. URL http://dx.doi.org/10.1021/acs.jctc.5b00743.

[71] Robert Zwanzig. From classical dynamics to continuous time random walks. *Journal of Statistical Physics*, 30(2):255–262, 1983.

[72] Vijay S. Pande, Kyle Beauchamp, and Gregory R. Bowman. Everything you wanted to know about markov state models but were afraid to ask. *Methods*, 52(1):99–105, 2010. ISSN 1046-2023. doi:https://doi.org/10.1016/j.ymeth.2010.06.002. URL https://www.sciencedirect.com/science/article/pii/S1046202310001568. Protein Folding.

[73] Brooke E. Husic and Vijay S. Pande. Markov state models: From an art to a science. *Journal of the American Chemical Society*, 140(7):2386–2396, 2018. doi:10.1021/jacs.7b12191. URL https://doi.org/10.1021/jacs.7b12191. PMID: 29323881.

[74] Christopher Kolloff and Simon Olsson. Machine learning in molecular dynamics simulations of biomolecular systems. *arXiv preprint arXiv:2205.03135*, 2022.

[75] Frank Noé, Illia Horenko, Christof Schütte, and Jeremy C Smith. Hierarchical analysis of conformational dynamics in biomolecules: Transition networks of metastable states. *The Journal of chemical physics*, 126(15):04B617, 2007.

[76] Frank Noé. Probability distributions of molecular observables computed from markov models. *The Journal of chemical physics*, 128(24):244103, 2008.

[77] John D Chodera and Frank Noé. Probability distributions of molecular observables computed from markov models. ii. uncertainties in observables and their time-evolution. *The Journal of chemical physics*, 133(10):09B606, 2010.

[78] Nicolae-Viorel Buchete and Gerhard Hummer. Coarse master equations for peptide folding dynamics. *The Journal of Physical Chemistry B*, 112(19):6057–6069, 2008.

[79] Christof Schütte, Frank Noé, Jianfeng Lu, Marco Sarich, and Eric Vanden-Eijnden. Markov state models based on milestoning. *The Journal of chemical physics*, 134(20):05B609, 2011.

[80] Guillermo Pérez-Hernández, Fabian Paul, Toni Giorgino, Gianni De Fabritiis, and Frank Noé. Identification of slow molecular order parameters for markov model construction. *The Journal of Chemical Physics*, 139(1):015102, 2013. doi:10.1063/1.4811489. URL https://doi.org/10.1063/1.4811489.

[81] S. Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982. doi:10.1109/TIT.1982.1056489.

[82] Hao Wu and Frank Noé. Variational approach for learning markov processes from time series data. *Journal of Nonlinear Science*, 30(1):23–66, 2020.

[83] Jan-Hendrik Prinz, Hao Wu, Marco Sarich, Bettina Keller, Martin Senne, Martin Held, John D. Chodera, Christof Schütte, and Frank Noé. Markov models of molecular kinetics: Generation and validation. *The Journal of Chemical Physics*, 134(17):174105, 2011. doi:10.1063/1.3565032. URL https://doi.org/10.1063/1.3565032.

[84] A Papoulis. Bayes' theorem in statistics and bayes' theorem in statistics (reexamined). *Probability, random variables, and stochastic processes. 2nd ed. New York, NY: McGraw-Hill*, pages 38–114, 1984.

[85] Emmanouil Christoforou, Hari Leontiadou, Frank Noé, Jannis Samios, Ioannis Z. Emiris, and Zoe Cournia. Investigating the bioactive conformation of angiotensin ii using markov state modeling

revisited with web-scale clustering. *Journal of Chemical Theory and Computation*, 18(9):5636–5648, 2022. doi:10.1021/acs.jctc.1c00881. URL `https://doi.org/10.1021/acs.jctc.1c00881`. PMID: 35944098.

[86] KT Schutt, Pan Kessel, Michael Gastegger, KA Nicoli, Alexandre Tkatchenko, and K-R Muller. Schnetpack: A deep learning toolbox for atomistic systems. *Journal of Chemical Theory and Computation*, 15(1):448–455, 2018.

[87] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. URL `http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf`.

[88] Bart M. H. Bruininks, Paulo C. T. Souza, and Siewert J. Marrink. *A Practical View of the Martini Force Field*, pages 105–127. Springer New York, New York, NY, 2019. ISBN 978-1-4939-9608-7. doi:10.1007/978-1-4939-9608-7_5. URL `https://doi.org/10.1007/978-1-4939-9608-7_5`.

[89] Siewert J Marrink and Alan E Mark. The mechanism of vesicle fusion as revealed by molecular dynamics simulations. *Journal of the American Chemical Society*, 125(37):11144–11145, 2003.

[90] Siewert J Marrink and Alan E Mark. Molecular dynamics simulation of the formation, structure, and dynamics of small phospholipid vesicles. *Journal of the American Chemical Society*, 125(49):15233–15242, 2003.

[91] Luca Monticelli, Senthil K Kandasamy, Xavier Periole, Ronald G Larson, D Peter Tieleman, and Siewert-Jan Marrink. The martini coarse-grained force field: extension to proteins. *Journal of chemical theory and computation*, 4(5):819–834, 2008.

[92] Djurre H de Jong, Gurpreet Singh, WF Drew Bennett, Clement Arnarez, Tsjerk A Wassenaar, Lars V Schafer, Xavier Periole, D Peter Tieleman, and Siewert J Marrink. Improved parameters for the martini coarse-grained protein force field. *Journal of chemical theory and computation*, 9(1):687–697, 2013.

[93] César A López, Giovanni Bellesia, Antonio Redondo, Paul Langan, Shishir PS Chundawat, Bruce E Dale, Siewert J Marrink, and S Gnanakaran. Martini coarse-grained model for crystalline cellulose microfibers. *The Journal of Physical Chemistry B*, 119(2):465–473, 2015.

[94] Jaakko J Uusitalo, Helgi I Ingólfsson, Parisa Akhshi, D Peter Tieleman, and Siewert J Marrink. Martini coarse-grained force field: extension to dna. *Journal of chemical theory and computation*, 11(8):3932–3945, 2015.

[95] Djurre H De Jong, Nicoletta Liguori, Tom Van Den Berg, Clement Arnarez, Xavier Periole, and Siewert J Marrink. Atomistic and coarse grain topologies for the cofactors associated with the photosystem ii core complex. *The Journal of Physical Chemistry B*, 119(25):7791–7803, 2015.

[96] Xun Chen, Mingchen Chen, and Peter G. Wolynes. Exploring the interplay between disordered and ordered oligomer channels on the aggregation energy landscapes of α-synuclein. *The Journal of Physical Chemistry B*, 126(28):5250–5261, 2022. doi:10.1021/acs.jpcb.2c03676. URL `https://doi.org/10.1021/acs.jpcb.2c03676`. PMID: 35815598.

[97] Xun Chen, Mingchen Chen, Nicholas P Schafer, and Peter G Wolynes. Exploring the interplay between fibrillization and amorphous aggregation channels on the energy landscapes of tau repeat isoforms. *Proceedings of the National Academy of Sciences*, 117(8):4125–4130, 2020.

[98] Weihua Zheng, Min-Yeh Tsai, Mingchen Chen, and Peter G Wolynes. Exploring the aggregation free energy landscape of the amyloid-β protein (1–40). *Proceedings of the National Academy of Sciences*, 113(42):11835–11840, 2016.

[99] Heidi Klem, Glen M Hocky, and Martin McCullagh. Size-and-shape space gaussian mixture models for structural clustering of molecular dynamics trajectories. *Journal of chemical theory and computation*, 2022.

[100] Sina Stocker, Johannes Gasteiger, Florian Becker, Stephan Günnemann, and Johannes T Margraf. How robust are modern graph neural network potentials in long and hot molecular dynamics simulations? *Machine Learning: Science and Technology*, 3(4):045010, 2022.

[101] Michael Schaarschmidt, Morgane Riviere, Alex M Ganose, James S Spencer, Alexander L Gaunt, James Kirkpatrick, Simon Axelrod, Peter W Battaglia, and Jonathan Godwin. Learned force fields are ready for ground state catalyst discovery. *arXiv preprint arXiv:2209.12466*, 2022.