Scalable Adaptive Protein Ensemble Refinement Integrating Flexible Fitting

Daipayan Sarkar[**],[†,§] Hyungro Lee[**],[‡] John W. Vant,[¶] Matteo Turilli,[‡,‖] Shantenu Jha,[*,‡,‖] and Abhishek Singharoy[*,¶]

[†]*MSU-DOE Plant Research Laboratory, Michigan State University, East Lansing, MI 48823, USA*

[‡]*Rutgers University, Electrical & Computer Engineering, New Brunswick, NJ 08854, USA*

[¶]*School of Molecular Sciences, Arizona State University, Tempe, Arizona 85281, USA*

[§]*School of Molecular Sciences, Arizona State University, Tempe, Arizona 85281, USA*

[‖]*Brookhaven National Laboratory, Computational Science Initiative, Upton, NY 11973, USA*

E-mail: shantenu.jha@rutgers.edu; asinghar@asu.edu

[**]*D.S. and H.L. contributed equally towards this manuscript.* [*]*Contact authors.*

Abstract

Recent advances in cryo-electron microscopy (cryo-EM) has enabled modeling macromolecular complexes that are essential components of life. The density maps obtained from cryo-EM experiments is often integrated with $ab - initio$, knowledge-driven or first principles-based computational methods to build, fit and refine protein structures inside the electron density maps. Going beyond a single stationary-structure determination scheme, it is becoming more common to interpret the experimental data with a set of multiple physical models all of which contributes to the average observation seen by the experiment. Hence, there is a need to decide on the quality of an ensemble of protein structures on-the-fly, while refining them against the density maps. In this work, we demonstrate such adaptive decision making capabilities during flexible fitting of biomolecules. Our solution uses RADICAL tools (RCT) and we test this new implementation in exascale high performance computing environment for two proteins, Adenylate Kinase (ADK) and Carbon Monoxide Dehydrogenase (CODH). Our results indicate that using multiple replicas in flexible fitting with adaptive decision making improves the overall quality of fit and model by 40 % improvement when compared against the traditional flexible fitting approach. These advances are agnostic to system-size and computing environments.

1 Introduction

Integrative modeling is an area of rapid methodological developments, wherein, atom-resolved structures of biological systems are determined by merging data from multiple experimental sources with physics[1–3] and informatics-based approaches.[4] These elegant fitting,[1–3,5–8] learning[9] and inferencing[10–14] methodologies have been successful in resolving a range of structures, starting with soluble and membrane proteins up to sub-cellular complex architectures.Integrative models routinely make it to top positions at the CASP, EMDB and PDB competitions, serving a diverse cross-section of the Biophysics community.

A key issue in structural or biochemical experiments is heterogeneity of data. The data can be rich, poor and sparse in information depending upon the space or time scales they capture, and yet all of them contribute to the holistic biophysics of the protein under investigation. As a natural consequence of this heterogeneity, a single-model interpretation of the experimental data becomes implausible, opening the door to an ensemble treatment of the data.[15] These ensemble models derived by integrative approaches capture on one hand, the most probable interpretation of the data, while on the other, pinpoints rare-events and hidden conformations, indispensable to biology.

Post the 2017 Nobel Prize, the cryo-EM community has actively sought ways of extracting not just stationary structures, but ensembles and more importantly, molecular dynamics information from electron density data.[16,17] An advantage of the ensemble's interpretation is that, the generation of multiple independent atomic models using an EM density and subsequent analysis of their atomistic agreement statistics provide model quality metrics that directly correlate with global and local EM map quality.[18] This ensemble approach offers essentially both a quantitative and qualitative assessment of the precision of the models and their representation of the density. However, the size of ensembles that collectively describes the diversity in single-particle images (reflecting in the quality of the maps) grows nonlinearly with system-size.[16] For proteins of molecular mass 500 kDa or bigger, composed of 5000 residues or more, a single CPU is expected to take 5000 years of wall-clock time for sampling the collective ensembles using either molecular dynamics (MD) or Monte Carlo (MC) simulations;[19] even the fastest GPUs of the day will not rescue this situation. Data-guided enhanced sampling methodologies, such as MELD[11] (integrated with NAMD via the recently completed CryoFold plugin[12]) or backbone tracing methodologies such as MAINMAST[20] or analogous methods,[9] by themselves, either remain system-size limited, generating ensembles for only local regions within a map, or require further further refinements using conjugate gradient minimization or MD simulation schemes to determine ensemble models.

By leveraging classical force fields (so-called CHARMM[21] energy functions) we have developed a range of molecular dynamics flexible fitting (MDFF) methodologies for integrating X-Ray and Cryo-EM data with MD simulations.[1–3] The simulations are biased towards conforming molecular models into forms consistent with the experimental density maps. These protocols are available through MD simulation engine NAMD,[22] and are also expanded to community codes. As a natural outcome of this fitting procedure, the most probable data-guided models are derived. However, the conformational heterogeneity that contributes to the uncertainty of the the experimental data is lost. Biology often employs such conformational diversity in problems of allostery and recognition, motivating further the need to refine experimental knowledge against an ensemble of models,[12] rather than a single model interpretation. In this article we explore whether, it is possible to recover portions of the conformations lost in brute-force MDFF by running multiple replicas of MDFF in parallel with adaptive decision making based on map-model consistency parameters. Rather than physically enforcing a model into a map, this approach skews the probability of an ensemble of models towards maximizing their consistency with the map. This way, there remains a finite probability of visiting some uncertain models, while still emphasizing determination of the most probable molecular models.

Traditional High Performance Computing (HPC) approaches, however, fail to accommodate the map-model analyses and on-the-fly decision making steps needed within an ensemble workflow.[23] We used the RADICAL-EnsembleToolkit (EnTK)[24] to overcome these challenges, developing multi-replica MDFF as a workflow application. EnTK exposes an application programming interface (API) that enables users to define a workflow in terms of pipelines, stages and tasks, and the resources required by that workflow to execute. EnTK also implements a workflow engine that interprets the given workflow description, acquires the required resources, and manages the execution of the workflow.

Our MDFF workflow application composes individual simulations and supports analysis calculation on intermediate results to perform adaptive sampling. A classical approach runs molecular simulations with long time scales which often require additional time to find interesting regions of the search space. In contrast, adaptive sampling implements an iterative loop that concurrently execute multiple simulations, each with short simulation time.[25–27] Map-model similarities are analyzed at every iteration and to increase the probability of finding models that are most consistent with the data, without getting trapped in any local energy minimum. The decision to focus on the sampling of specific models can be based a number of map-model metrics,[28] such as TM scores,[29] MolProbity,[30] EMRinger,[31] Q-score.[32]

In our first adaptive MDFF workflow, a simple global correlation coefficient (CC) is employed as a criteria to guide the choice of refinement models. This approach iteratively screens model populations based on their CCs with the map, and improves efficiency of computing resource consumption over longer brute-force MD simulations that are notorious for converging to uninteresting local minima. We find that, powered by EnTK's data-staging capabilities and check-pointing of the parallel MD simulations available on NAMD, MDFF trajectories intermittently screened by CC values (see Methods for details) offer ensemble refinement of models.

We have tested the MDFF workflow with up to 100 replicas with 16 iterations across different resolutions, 1.8 Å, 3.0 Å, 5.0 Å and achieved around 40 % ($= \frac{0.8-0.57}{0.57} \times 100$) improvement of converging cross-correlation over the course of workflow lifetime. Specifically, we architect, design and implement an integrated pipeline for ensemble refinement with EnTK and MDFF to support cryo-EM modeling across intermediate to high-quality density maps between 2 to 5 Å resolution. The pipeline is tested for using up to 400 replica with (1 node/replica). In all these cases, we find that an ensemble approach with adaptive decision making offers more diverse ensembles than brute force MDFF, and still offers the most data-consistent model. Going beyond traditional MDFF, these ensembles capture on one hand, the 'best' model, while simultaneously the uncertainty in the assignments on the other. The performance of the pipeline improves with system-size (3341 ADK and 11452 CODH atom counts), and remains robust to computing platforms. Taken together, our implementation breaks free of the traditional high-performance computing execution model that assumes singular jobs and static execution of tasks and data, to one that is fundamentally designed for data-integration and assimilation across different scales, quality and sparsity.

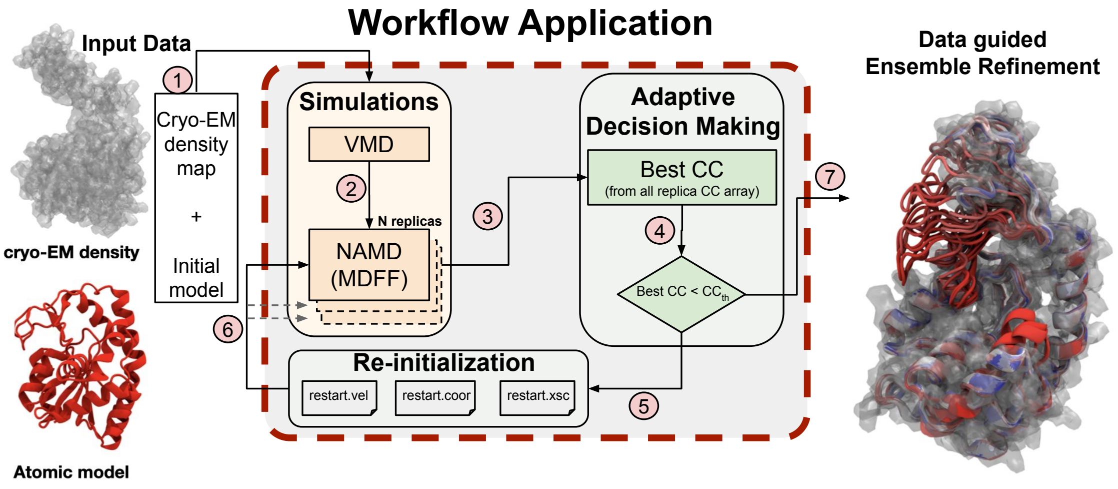


Figure 1: Overview of the workflow application showing how NAMD/VMD is used to perform flexible fitting iteratively. Internal boxes with annotation numbers indicate the sequence of the workflow: (1) Input data (2) Simulation preparation and execution using VMD and NAMD respectively (3) Building CC matrix and sort through CC matrix to select best CC (4) Check if best CC is lower than threshold CC (5) Use the current state of the molecular sytem corresponding to the best CC using the restart files (6) Re-seed all replicas with the restart files and perform the next iteration of flexible fitting (7) Data guided ensemble refined models.

2 Data guided ensemble refinement using MDFF and EnTK

The adaptive decision-making for MD ensemble refinement with 3-dimensional density data sets is implemented as a recursive simulation-analysis workflow application enabled by EnTk (Figure 1 and Algorithm 1). This workflow consists of an 'ensemble' of simulation and analysis pipelines that executes concurrently on HPC resources. Each constituent pipeline consists of seven serial tasks: **(1)** load an empirically determined density map or generate a simulated map. Then convert this map to an MDFF potential. Independently, examine an initial search model quality in terms of stereochemical properties, and perform rigid body docking to place this search model inside the EM density map. **(2)** Define the secondary structure restraints. Visual Molecular Dynamics (VMD) then prepares the necessary input files required by NAMD to deploy MDFF. In this step, multiple replicas of the system are prepared by individual EnTK pipeline for which MDFF is performed. Finally the multi-replica MDFF simulations are performed in parallel. **(3)** VMD is re-used to calculate the interim cross-correlation value between the atomic model (corresponding to replica window) and the EM density map. The CC values from different replicas are then combined together to construct a matrix of CC values. EnTK uses data staging area to move the CC values in files from flexible fitting to the adaptive decision making block across multiple replicas. **(4)** Here, a decision is made on whether the flexible fitting simulations will be continued or terminated if the computed CC is $\geq$ a user-defined threshold cross-correlation. This on-the-fly map-model analysis enables an adaptive flexible fitting algorithm (such as MDFF) to run iteratively inside EnTK, without user intervention. For scenarios which require multiple iterations **(5)**, all the replicas are re-seeded with the atom coordinates, velocities and periodic system information corresponding to MDFF model with the best cross-correlation from the previous iteration, and the next round of multi-replica MDFF proceeds *6.* Again, EnTK uses data staging area to store these information in files and provide them to the replicas. This feature does not only make the algorithm adaptive but also enables intelligent decision making, with scope of improvement in future with advanced decision-making or inferencing algorithms. Finally, the application converges to yield data-guided ensemble refined models **(7)** which exits EnTK workflow application and downloads results to the end-user's home directory.

Algorithm 1: Cross-correlation exchanging scheme.

```
begin
    while $CC\ replica\ resolution \leq CC\ threshold$ do
        generate N replicas wih selected coordination
        repeat simulation stage (selected coordination, new density map)
        repeat analysis stage (last frame coordination, replica index)
        increase iteration by 1
        end
```

EnTK's application programming interface (API) is implemented as a Python module, loaded into the workflow application's code. The API exposes classes for pipeline, stage and task, allowing to directly map the workflow description to the logical representation of an ensemble of simulations. Each task object exposes a set of variables with which to configure input, output files, executable, resource requirements and pre/post execution directives. Finally, an appmanager object is used to contain the workflow description and execute it with a single `AppManager.run()` method. The iteration logic to change the workflow description and issue another `AppManager.run()` is written in pure Python as part of the workflow application. The entire MDFF workflow application of this paper required only 500 lines of Python code.

As already described in,[24] EnTK complements the ensemble simulation paradigm with decision-making through real-time workflow and parameter changes, based on the results of the analysis stages. In the present context, this feature enables iterative workflow executions with a single HPC batch-job submission, avoiding costly manual evaluation of cross-correlation coefficient, workflow editing and re-submission. EnTK also abstracts from the users the need to explicitly manage data-flow and task execution. It manages data staging so that each task of each stage has either a copy or a link to all the NAMD input files it requires, allowing the users to focus on the MDFF simulation and VMD analysis methods, without having to explicitly manage data sourcing, saving and exchange. Furthermore, EnTK schedules and executes the workflow's tasks, managing the mapping of tasks to available resources on each compute node allocated to the workflow execution. Users have only to specify the amount of CPU cores/GPUs needed by each task and whether the task is (Open)MPI.

3 Methods

Modern adaptive sampling frameworks are dynamic, extensible, scalable and robust to facilitate hundreds or thousands of experiments for searching different structures, and specialized features can be added to solve existing problems through the framework. We developed a workflow application using RADICAL cybertools[33] that provides a scalable workflow framework for implementing ensemble refinement with cross correlation calculation on HPC computing resources. MDFF-EnTK (Molecular Dynamics Flexible Fitting using Ensemble ToolKit), depicted in Figure 1, supports adaptive decision making algorithms to iterate between molecular dynamics flexible fitting simulation and cross-correlation analysis. Our workflow application is portable to explore the space of experimental configurations and support various use cases, so that the ensemble refinement produces results on different dimensions of a physical system; resolution density, simulation length, replica count, and HPC resource. The full integration is explained in the following sections: (a) MDFF simulation, (b) CC analysis, (c) RADICAL cybertools.

3.1 Molecular Dynamics Flexible Fitting simulation:

In the simulation stage of the pipeline, MDFF-EnTK uses the conventional MDFF algorithm, as described in.[3] Briefly, MDFF requires, as input data, an initial structure and a cryo-EM density map. A potential map is generated from the density and subsequently used to bias a MD simulation of the initial structure. The structure is subject to the EM-derived potential while simultaneously undergoing structural dynamics as described by the MD force field.

Let the Coulomb potential associated with the EM map be $\Phi(\mathbf{r})$. Then the MDFF potential map is given by,

$$V_{EM}(\mathbf{r}) = \begin{cases} \zeta \left[\frac{\phi((r)) - \phi_{th}}{\phi_{max} - \phi_{th}} \right], & \text{if } \phi(r) \geq \phi_{th} \\ \zeta, & \text{if } \phi(r) < \phi_{th} \end{cases} \tag{1}$$

where ζ is a scaling factor that controls the strength of the coupling of atoms to the MDFF potential, ϕ_{th} is a threshold for disregarding noise, and $\phi_{max} = max(\phi(r))$. The potential energy contribution from the MDFF forces is then

$$U_{EM}(\mathbf{r} = \sum_i w_i V_{EM}(r_i) \tag{2}$$

where i labels the atoms in the structure and w_i is an atom-dependent weight, usually the atomic mass. During the simulation, the total potential acting on the system is given by,

$$U_{total} = U_{MD} + U_{EM} + U_{SS} \tag{3}$$

where U_{MD} is the MD potential energy as provided by MD force fields (e.g. CHARMM) and U_{SS} is a secondary structure restraint potential that prevents warping of the secondary structure by the potentially strong forces due to U_{EM}. A detailed description of the MDFF methodology is presented in. Specific simulation parameters for the example cases of ADK and CODH are provided on the GitHub page.[34]

3.2 Cross-correlation analysis

For analysis as part of ensemble refinement, mainly towards adaptive decision making we calculate the cross correlation (CC) value for all replicas at a given time (t) in the MDFF-EnTK pipeline. Note, the time (t) here is not the number of MD steps. Instead the analysis is performed every iteration, for N MD steps and for M replicas. So, based on this the total simulation time is equal to $t_{steps/iteration} \times N_{iteration} \times M_{replicas}$. At the end of each iteration, the highest value of CC is analyzed and corresponding atomic coordinates are used to seed the M replicas for the next iteration.

3.3 Ensemble ToolKit (RADICAL Cybertools)

In order to implement the pipeline, we have extended an open-source, Python framework that facilitates adaptive ensemble biomolecular simulations at scale, RADICAL-EnsembleToolkit (EnTK). The first step of writing the EnTK workflow code is to construct a task parallel execution of MDFF simulation using NAMD, and to connect the analysis stage to find highest CC values among replicas. While all the necessary information such as NAMD checkpoints and CC values are kept under EnTK's data staging area, distributed computing resources are coordinated to ensure the workflow performance over CPUs and GPUs from heterogeneous HPC platforms. In addition, several features have been added to the application by utilizing existing capabilities of RADICAL tools. Tcl scripting is interfaced with EnTK APIs to interact with VMD software directly and the partitioned scheduling is introduced to assign a single node per replica exclusively for the best performance of NAMD simulations. Usability and productivity have been addressed to automate resource configurations and experiment settings as well as ensuring reproducibility of scientific data. With the MDFF-EnTK application, replacing MD engines or analysis methods needs to change a few lines of settings in a workflow management file without source code modifications. The application, MDFF-EnTK is available on GitHub (https://github.com/radical-collaboration/MDFF-EnTK) and implemented to support adaptive decision making for ensemble-based simulations and to enable the novel analysis method, MDFF or others on HPC resources.

4 Results

In what follows, we have conducted a series of experiments for different replica numbers and compare the flexible fitting and computational performance of different experiment settings on ADK and CODH proteins in this section. Two HPC facilities were used where Oak Ridge Leadership Computing Summit has two IBM Power9 processors and six NVIDIA V100 GPU accelerators and Pittsburgh Supercomputer Center Bridges2 has two AMD EPYC 7742 processors.

4.1 Adaptive Decision Making provides a variance in Ensemble Refinement at High Resolution Density Maps

In Figure 2 we show how the cross correlation coefficient (CCC) changes across iterations for different ensemble members (replicas) and resolutions of EM density maps for the protein adenylate kinase (ADK). Here, we perform molecular dynamics flexible fitting (MDFF) to fit ADK at three resolutions - high (1.8 Å), intermediate (3.0 Å) and low (5.0 Å) density maps. The simulation length, which is defined as the number of steps in a MD simulation times the total number of iterations, remains constant for different replicas. The CCC basically provides a measure on the quality of fit of the atomic model to density map, where a higher number means a better fit. As shown, for intermediate and low resolutions, the CCC is $\approx 90\%$, while at high resolution the CCC is $\approx 80\%$. This occurs mainly because there are many ways to fit in an intermediate or low resolution density maps, and not so many ways to fit inside a higher resolution density map. However, it is important to note that, as the replica count increases the distribution of CCC traces across iterations gets wider for high resolution density map. The wider distribution physically means, variance in the ensemble of protein structures. Conventionally, MDFF and it's variants - cascade MDFF (cMDFF), resolution exchange MDFF (ReMDFF) generate ensemble of protein structures with very low variance. MDFF-EnTK, during cryoEM ensemble refinement now enables us a measure on how far a set of protein structures are from their mean, which is a structure one obtains from traditional MDFF.

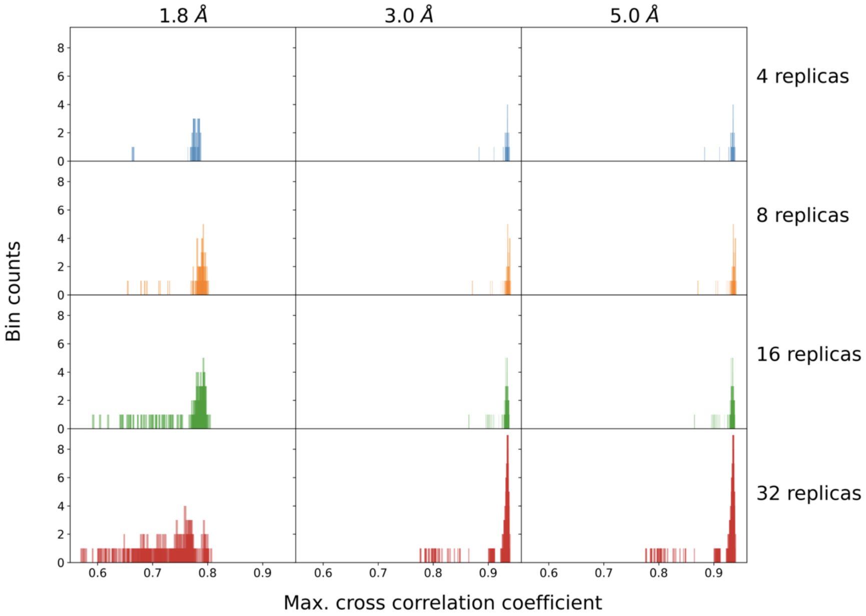


Figure 2: CC vs iteration comparison for high resolution (1.8 Å, intermediate resolution (3.0 Å) and low resolution (5.0 Å) cryo-Electron microscopy density maps, for different ensemble members, 4 (blue), 8 (orange), 16 (green) and 32 (red) respectively.

4.2 Statistics of map-model fits improve with larger replica simulations

In addition, to the variance in protein structure during ensemble refinement, the ensemble refinement algorithm presented here uses adaptive decision making to improve model quality as shown in Figure 4 (A) for ADK at 1.8 Å resolution density map. In this figure, results indicate for higher replicas the CCC value improves by ≈ 20% over a single long MDFF trajectory (CCC=0.67), while a fixed simulation length is maintained for all replicas. Subsequently, in Figure 4 (B) we test if the simulation length has an effect on improving the quality of fit for larger number of replicas. Based on the Figure 4 (A), one would notice that the CCC value grows and then drops and forks as the replicas increase. We anticipate here, since there are higher number replicas, the simulation time per replica is shorter to maintain the same simulation length as that of a single MDFF trajectory.

To test this hypothesis, for replicas 64, 100, 200, 400 we increase the simulation length per replica to match that of 16 replicas. We choose the simulation length per replica from 16 replicas, as that provides the best CCC value. Here, in Figure 4 (B) we see that simulation length per replica does have an impact on the overall model quality, as now the CCC value improves by ≈ 31% over a single long MDFF trajectory, providing a net increase of 11% over a MDFF trajectory with shorter simulation length per replica. As a control experiment, we also performed the 64-replica simulation without decision-making. The the CCC values still remain around the 50 % mark, reminiscent of the single long simulation, reinforcing the need for adaptive decision-making.

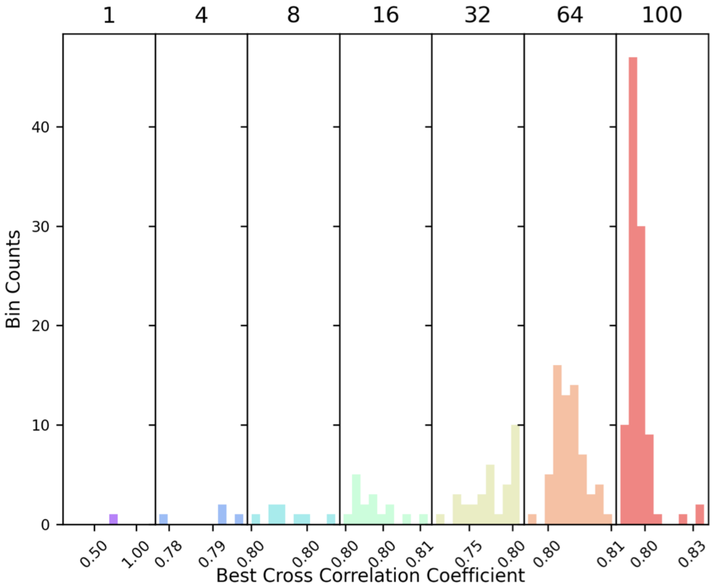


Figure 3: Phase Diagram for best cross-correlation (CC) at high resolution 1.8 Å map for Adenylate kinase (ADK) for different ensemble members (1 - 400), illustrating the effect of simulation time on the quality of fit. (A) Performance of best CC when the total length of the simulation equals the length of a single long MDFF trajectory (red circle) where 16 ensemble members provide the best model quality (B) Performance for best CC where the simulation length per ensemble member was increased for 64, 100, 200 and 400 replicas, specifically to match the same simulation length per ensemble member to that of the 16 ensemble member (80 ps per ensemble member per iteration). Results show an increase in quality of fit with increase in number of ensemble members, showing the dependence of trajectory length for each ensemble member over successive iterations.

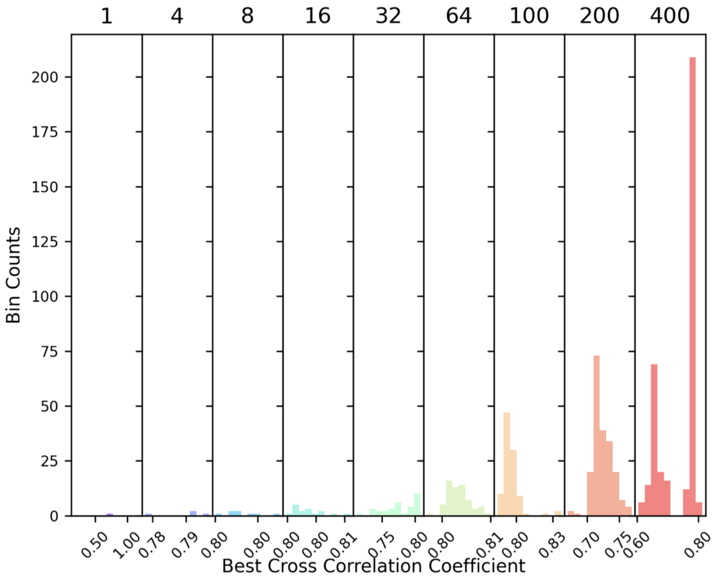


Figure 4: Phase Diagram for best cross-correlation (CC) at high resolution 1.8 Å map for Adenylate kinase (ADK) for different ensemble members (1 - 400), illustrating the effect of simulation time on the quality of fit. (A) Performance of best CC when the total length of the simulation equals the length of a single long MDFF trajectory (red circle) where 16 ensemble members provide the best model quality (B) Performance for best CC where the simulation length per ensemble member was increased for 64, 100, 200 and 400 replicas, specifically to match the same simulation length per ensemble member to that of the 16 ensemble member (80 ps per ensemble member per iteration). Results show an increase in quality of fit with increase in number of ensemble members, showing the dependence of trajectory length for each ensemble member over successive iterations.

4.3 Computational ensemble refinement is robust to system size

Figure 5 shows the application of the performance of MDFF-EnTK pipeline for a larger system, carbon monoxide dehydrogenase (CODH). The objective here is to evaluate the dependence of system size on MDFF-EnTK parameters estimated from our multi-replica ADK simulations. For this purpose, we use the same parameter values obtained from 64 replicas of ADK at 1.8 Å and perform flexible fitting simulations of CODH at 1.8 Å and 3.0 Å. The results indicate that at both high and intermediate resolutions we find multiple populations of an ensemble of structures fitted to a density map. The overall maximum CC value improves in successive iterations, establishing that the present algorithm scales well with system size, for different ensemble members, specifically 16 (5 A) and 100 replicas (5 B).

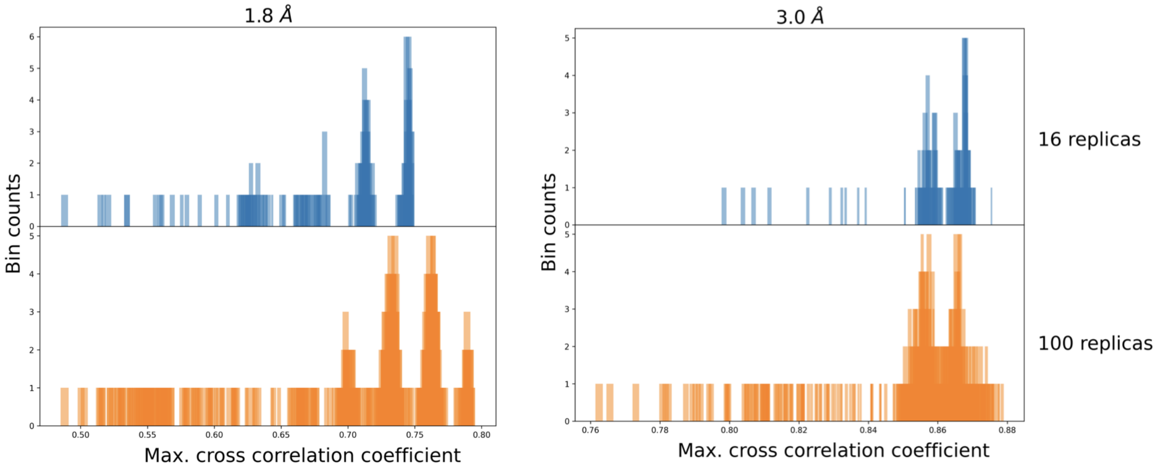


Figure 5: CODH cross correlation vs iteration index for different ensemble members using the optimal parameters from ADK phase plot at different resolutions - 1.8 Åand 3.0 Å.

5 Performance characterization of MDFF and EnTK

The goal of this section is to assess the computing performance of the workflow application on HPC resources and provide evidence that MDFF-EnTK would manage computing resources efficiently and have the overhead of running multiple replicas restrained while using the integrated RADICAL environment.

5.1 Experiment Configuration

We designed 11 experiments to evaluate the efficiency of EnTK, measured in terms of overhead and resource utilization when executing MDFF-EnTK. We discuss two biological systems in the experiments: adenylate kinase (ADK) and carbon monoxide dehydrogenase (CODH).

We use 16 nanoseconds (ns) for the MD simulation as a baseline (t) for our performance characterization. Our experiments compare t to iterative MD simulations with 16 MD steps (N), showing the computational efficiency of adaptive ensemble refinement. Each experiment is configured with parameters of atomic resolution, number of replicas and simulation length (see Table 1). We also measure how performance varies across two HPC platforms: ORNL Summit and PSC Bridges2.

Our performance characterization uses two metrics: overhead (OVH) and resource utilization (RU). OVH is the amount of time in which compute nodes are available but not used to execute tasks, while RU is the percentage of compute nodes being used for executing tasks. We measure OVH through the parameter settings (Table 1) and provide resource utilization of compute nodes so that the performance behavior of MDFF-EnTK can be identified across experiments. Experiment 9 uses the same parameter settings as Experiment 5 , and Experiment 10 shows the result with the intermediate resolution 3.0Å. Experiment 11 uses 100 Summit nodes with the same settings as Experiment 9.

Table 1: Experiments to characterize EnTK performance. System: biological system name; Rep. (M): Total number of replicas between 2 and 100; Sim. Len. (ps): timescale per iteration in picoseconds; Res. (Å): resolutions in Angstrom (high 1.8Å and intermediate 3.0Å); Resource: GPUs and CPU cores on OLCF Summit and CPU cores only on PSC Bridges2; Tasks: number of tasks for each experiment; OVH(s): Overhead of EnTK in second.

Exp. ID	**System**	**Rep. (M)**	**Sim. Len. (ps)**	**Res. (Å)**	**Resource**	**Tasks**	**OVH (s)**
1	ADK	2	64	1.8	Bridges (CPU)	256	81.0 ± 10
2	ADK	4	32	1.8	Bridges (CPU)	512	126.0 ± 10
3	ADK	4	250	1.8	Summit (GPU&CPU)	512	92.0
4	ADK	8	160	1.8	Summit (GPU&CPU)	1024	105.27 ± 18
5	ADK	16	80	1.8	Summit (GPU&CPU)	2048	114.06 ± 16
6	ADK	32	40	1.8	Summit (GPU&CPU)	4096	109.33 ± 10
7	ADK	64	20	1.8	Summit (GPU&CPU)	8092	158.87 ± 57
8	ADK	100	10	1.8	Summit (GPU&CPU)	12800	266.98 ± 245
9	CODH	16	80	1.8	Summit (GPU&CPU)	2048	93.34 ± 17
10	CODH	16	80	3.0	Summit (GPU&CPU)	2048	99.44 ± 20
11	CODH (long)	100	80	1.8	Summit (GPU&CPU)	12800	113.61 ± 20

We provide templates to allow users to replicate the experiments presented in this paper. The source code and configuration parameters of the experiments are published on the MDFF-EnTK Github repository.[34] Users can use those templates as a starting point to create and run their own experiments. The templates, written in YAML, store user-defined attributes for experiments and HPC resources separately, ensuring flexible analysis on diverse computing platforms.

We utilize up to 4 compute nodes on Bridges2 and 100 on Summit, executing each replica on a full compute node. On Bridges2 we run the NAMD MD engine on 128 cores (AMD EPYC 7742 with of 256GB DDR4 memory), without GPU acceleration. Note that Bridges2 offers 24 compute nodes, each with 8 GPU V100 accelerators but we decided to use only CPU resources due to their limited availability. On Summit, we run the CUDA-enabled NAMD MD engine on 6 NVIDIA V100 GPU accelerators per node. Different hardware platforms show wide performance gaps in time to solution but the cross-correlation is similar when using the same configurations.

5.2 EnTK Overhead Steady Across Different HPC Platforms

We measured the time spent by EnTK to bootstrap and clean up the execution environment. Those are overheads as they measure the time spent before and after the execution of the workflow's tasks, when computing resources are already available. We measured the overheads across two HPC platforms and with an increasing number of compute nodes, offering a characterization of the cost of using the tool to execute MDFF-EnTK at different scales and on two HPC platforms.

Note that both bootstrap and clean up overheads are independent of the workflow scale as the time taken to manage the execution environment does not depend on the number of tasks executed in it. However, bootstrap overhead can vary, depending on the performance of the filesystem that serves packages and files during the bootstrapping process. Using a pre-configured environment may reduce the bootstrapping overhead.

We explore the scalability of the iterative workflow MDFF-EnTK on Bridges2 and Summit. The overhead is between 3% and 5% of the total execution time of the workflow presented in §3. Figure 6 and Table 1 show that such overhead is invariant of the number of replicas executed on Summit (150.90 ± 115 seconds) and on Bridges (103.5 ± 22.5 seconds) when running from 2 replicas to 100 replicas. The overhead varies across HPC platforms mainly due to differences in filesystem performance, network latency and the use of static environments to initialize.

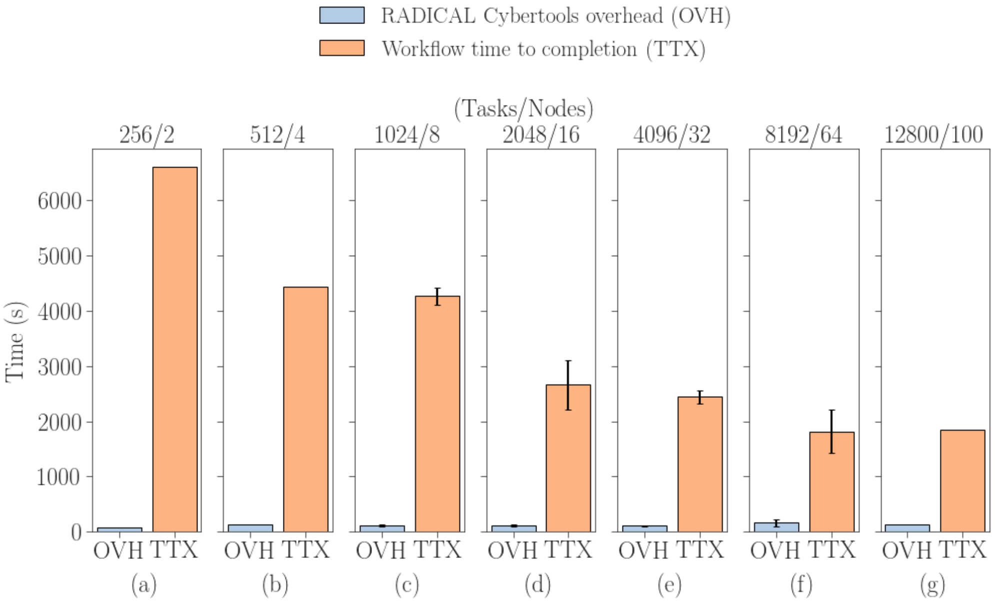


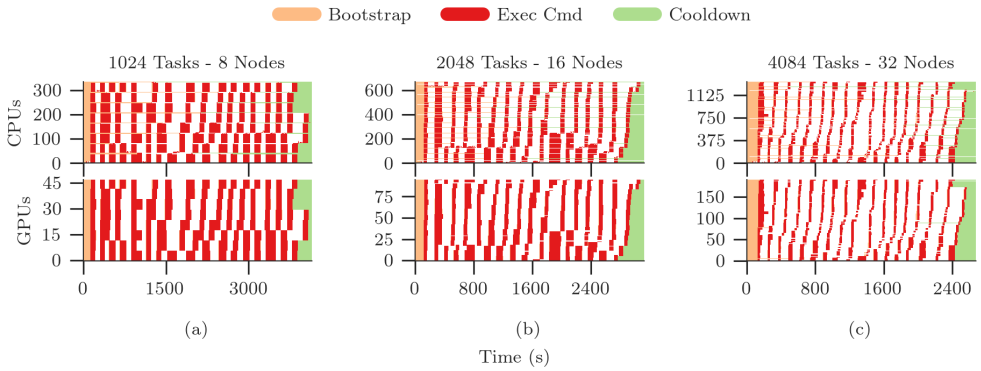
Figure 6: EnTK Overheads on PSC Bridges2 (a, b) and Summit (c - g) compute nodes. The total simulation time (equal to $t_{steps/iteration} \times N_{iteration} \times M_{replicas}$) is 16ns. TTX tends to decrease with the increasing of the number of compute nodes.

Bridges2 shows three times larger overhead compared to Summit, mainly due to the different performance of the parallel filesystems: Lustre on Bridges2, GPFS on Summit. On Lustre, the initial access to files takes longer than continuous access because Lustre has to retrieve a location of the actual storage device over the network. The additional results from both platforms are reported in the Supporting Information (SI) 6.

5.3 EnTK Resource Utilization Is Independent of Five Dimensions

Our experiments measure resource utilization at different scales and across five dimensions: HPC platform, replica sets, simulation timesteps, system size, and resolution. For each experiment, we changed the number of replicas with the same total simulation time (16ns)—equal to $t_{steps/iteration} \times N_{iteration} \times M_{replicas}$—and varied the HPC platform and the number of used compute nodes.

Figure 7 depicts the CPU/GPU utilization for ADH at high resolution (1.8A) for 8, 16 and 32 replicas. The regions with a red color indicate compute resources used by the replicas whereas other regions with a orange, white, and green color represent resources are not utilized. The workflow needs to initialize, wait for tasks to become available for an execution and to terminate in these regions repectively. The figure 7 shows the resource utilization with computing units divided into CPU (upper half) and GPU (bottom half). The bottom halves shows the utilization of between 48 and 192 GPUs (NVIDIA V100) on Summit while executing 16 iterations of the pipeline; the upper halves the utilization of between 336 and 1344 CPUs (IBM Power9) on Summit.



(a) (b) (c)

Figure 7: Resource Utilization of MDFF-EnTK. CPU (top) and GPU (bottom) resources are visualized for experiments 3, 4, and 5 with the ADK system, 1.8Å high resolution, 16ns (t) and 16 iterations (N) on Summit. 8 (left), 16 (middle), and 32 (right) replicas (M), executed at different scales with a 1:1 node/replica ratio.

Figure 7 shows that the utilization of CPU and GPU resources remains 73.3% in average across scales

(83.2% for 8 replicas, 74.9% for 16 replicas and 62% for 32 replicas), and the slightly increased overheads are negligible as the increased quality of fit with a large number of replicas is shown according to Section 4. For a, b, and c, the number of executed tasks is increased and the average overall execution time is reduced for the same simulation time. We identified the regions with a white color in the figure 7 in which tasks spent time waiting for other tasks to be finished with a global barrier in the iterative workflow, e.g., exchange cc value and then continue simulations. It may reduce these white regions by optimizing task executables. Another observation is that the overheads of bootstrap and cooldown stages remain constant across replicas which helps maintaining resource utilization at scale. Ensemble-style runs may suffer from these overheads unless the workflow framework is capable of dispatching many tasks quickly without sacrificing overall resource utilization.

6 Conclusions

Cryo-EM data of a protein represents an average of many two-dimensional images transformed to a three-dimensional density map. Classical methods in statistical mechanics such as MD fail to determine such an ensemble in finite length simulations, as structures remain trapped in deep potential wells corresponding to local dense points in density maps . To circumvent this algorithmic bottleneck of importance sampling and to decide the quality of an ensemble of protein structures on-the-fly, we present a framework for ensemble refinement of protein structures with adaptive decision making to improve both the quality of model and fit. An ensemble model offers, on one hand, the most probable structural representation based on available density information, while capturing protein conformational dynamics that are often ignored in traditional single-model interpretation.

Our ensemble refinement workflow allows adaptive decision-making for molecular dynamics flexible fitting simulations by the the integration of correlation analysis with MD simulations via the EnTK pipeline. This pipeline is implemented in multiple national resources. The pipeline performs an user-defined number of iterative fitting and analysis tasks. This multi-replica scheme improves with statistical significance, the quality of models over those derived from the traditional scheme of performing a single long MDFF simulation. Consequently, the new scheme arrives not just at the best-fit but a population of models with varied ranges of data-consistency. In addition, we show that MDFF integrated with EnTK is well suited for exascale high-performance computing environments[35] by managing resource utilization of GPU and CPU computing units and the workflow overhead for increased ensemble members. We also show that our approach would have a similar computational cost as the traditional single long MDFF simulation, but with a quick turnaround time (shorter wall time of workload), while exploring interesting regions in the density map. We continue to extend the capability of MDFF-EnTK in complex applications in exascale high-performance computing environments.

Acknowledgement

Singharoy acknowledges start-up funds from the SMS and CASD at Arizona State University, CAREER award from NSF (MCB-1942763), and an NDEP grant by the Department of Defense. This work used the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by National Science Foundation grant number ACI-1548562. Benchmarks were also performed using the resources of the OLCF at the Oak Ridge National Laboratory, which is supported by the Office of Science at DOE under Contract No. DE-AC05-00OR22725, made available via the INCITE program.

References

(1) Trabuco, L. G.; Villa, E.; Mitra, K.; Frank, J.; Schulten, K. Flexible Fitting of Atomic Structures into Electron Microscopy Maps Using Molecular Dynamics. *Structure* **2008**, *16*, 673–683, DOI: 10.1016/j.str.2008.03.005.

(2) Trabuco, L. G.; Villa, E.; Schreiner, E.; Harrison, C. B.; Schulten, K. Molecular dynamics flexible fitting: A practical guide to combine cryo-electron microscopy and X-ray crystallography. *Methods* **2009**, *49*, 174–180, DOI: 10.1016/j.ymeth.2009.04.005.

(3) Singharoy, A.; Teo, I.; McGreevy, R.; Stone, J. E.; Zhao, J.; Schulten, K. Molecular dynamics-based refinement and validation for sub-5 Å cryo-electron microscopy maps. *eLife* **2016**, *5*, 1–33, DOI: 10.7554/eLife.16105.

(4) Leaver-Fay, A.; Tyka, M.; Lewis, S. M.; Lange, O. F.; Thompson, J.; Jacak, R. et al. *Methods in Enzymology*; Academic Press, 2011; Vol. 487; pp 545–574, DOI: 10.1016/B978-0-12-381270-4.00019-6.

(5) McGreevy, R.; Teo, I.; Singharoy, A.; Schulten, K. Advances in the molecular dynamics flexible fitting method for cryo-EM modeling. *Methods* **2016**, *100*, 50–60, DOI: 10.1016/j.ymeth.2016.01.009.

(6) Igaev, M.; Kutzner, C.; Bock, L. V.; Vaiana, A. C.; Grubmüller, H. Automated cryo-EM structure refinement using correlation-driven molecular dynamics. *eLife* **2019**, *8*, 1–33, DOI: 10.7554/eLife.43542.

(7) Kim, D. N.; Moriarty, N. W.; Kirmizialtin, S.; Afonine, P. V.; Poon, B.; Sobolev, O. V. et al. Cryo_fit: Democratization of flexible fitting for cryo-EM. *Journal of Structural Biology* **2019**, *208*, 1–6, DOI: 10.1016/j.jsb.2019.05.012.

(8) Costa, M. G.; Fagnen, C.; Vénien-Bryan, C.; Perahia, D. A New Strategy for Atomic Flexible Fitting in Cryo-EM Maps by Molecular Dynamics with Excited Normal Modes (MDeNM-EMfit). 2020; https://pubs.acs.org/doi/abs/10.1021/acs.jcim.9b01148.

(9) Pfab, J.; Phan, N. M.; Si, D. DeepTracer for fast de novo cryo-EM protein structure modeling and special studies on cov-related complexes. *Proceedings of the National Academy of Sciences of the United States of America* **2021**, *118*, DOI: 10.1073/pnas.2017525118.

(10) Perez, A.; MacCallum, J. L.; Dill, K. A. Accelerating molecular simulations of proteins using Bayesian inference on weak information. *Proceedings of the National Academy of Sciences* **2015**, *112*, 11846–11851, DOI: 10.1073/pnas.1515561112.

(11) Perez, A.; Morrone, J. A.; Brini, E.; MacCallum, J. L.; Dill, K. A. Blind protein structure prediction using accelerated free-energy simulations. *Science Advances* **2016**, *2*, DOI: 10.1126/sciadv.1601274.

(12) Shekhar, M.; Terashi, G.; Gupta, C.; Sarkar, D.; Debussche, G.; Sisco, N. J. et al. CryoFold: Determining protein structures and data-guided ensembles from cryo-EM density maps. *Matter* **2021**, *4*, 3195–3216, DOI: 10.1016/j.matt.2021.09.004.

(13) Bonomi, M.; Camilloni, C.; Cavalli, A.; Vendruscolo, M. Metainference: A Bayesian inference method for heterogeneous systems. *Science Advances* **2016**, *2*, 1501177, DOI: 10.1126/sciadv.1501177.

(14) Bonomi, M.; Hanot, S.; Greenberg, C. H.; Sali, A.; Nilges, M.; Vendruscolo, M. et al. Bayesian Weighing of Electron Cryo-Microscopy Data for Integrative Structural Modeling. *Structure* **2019**, *27*, 175–188.e6, DOI: 10.1016/j.str.2018.09.011.

(15) Balasubramanian, V.; Jensen, T.; Turilli, M.; Kasson, P.; Shirts, M.; Jha, S. Adaptive ensemble biomolecular applications at scale. *SN Computer Science* **2020**, *1*, 1–15.

(16) Frank, J.; Ourmazd, A. Continuous changes in structure mapped by manifold embedding of single-particle data in cryo-EM. *Methods* **2016**, *100*, 61–67, DOI: 10.1016/j.ymeth.2016.02.007.

(17) Ourmazd, A. Cryo-EM, XFELs and the structure conundrum in structural biology. *Nature Methods* **2019**, *16*, 941–944, DOI: 10.1038/s41592-019-0587-4.

(18) Herzik, M. A.; Fraser, J. S.; Lander, G. C. A Multi-model Approach to Assessing Local and Global Cryo-EM Map Quality. *Structure* **2019**, *27*, 344–358.e3, DOI: 10.1016/j.str.2018.10.003.

(19) Netz, R. R.; Eaton, W. A. Estimating computational limits on theoretical descriptions of biological cells. *Proceedings of the National Academy of Sciences* **2021**, *118*, e2022753118, DOI: 10.1073/pnas.2022753118.

(20) Terashi, G.; Kihara, D. De novo main-chain modeling for em maps using MAINMAST. *Nature Communications* **2018**, *9*, 1–11, DOI: 10.1038/s41467-018-04053-7.

(21) Huang, J.; Mackerell, A. D. CHARMM36 all-atom additive protein force field: Validation based on comparison to NMR data. *Journal of Computational Chemistry* **2013**, *34*, 2135–2145, DOI: 10.1002/jcc.23354.

(22) Phillips, J. C.; Hardy, D. J.; Maia, J. D. C.; Stone, J. E.; Ribeiro, J. V.; Bernardi, R. C. et al. Scalable molecular dynamics on CPU and GPU architectures with NAMD. *The Journal of Chemical Physics* **2020**, *153*, 044130, DOI: 10.1063/5.0014475.

(23) Ahn, D. H.; Bass, N.; Chu, A.; Garlick, J.; Grondona, M.; Herbein, S. et al. Flux: Overcoming scheduling challenges for exascale workflows. *Future Generation Computer Systems* **2020**, *110*, 202–213.

(24) Balasubramanian, V.; Turilli, M.; Hu, W.; Lefebvre, M.; Lei, W.; Modrak, R. et al. Harnessing the power of many: Extensible toolkit for scalable ensemble applications. 2018 IEEE International Parallel and Distributed Processing Symposium (IPDPS). 2018; pp 536–545.

(25) Dakka, J.; Farkas-Pall, K.; Turilli, M.; Wright, D. W.; Coveney, P. V.; Jha, S. Concurrent and adaptive extreme scale binding free energy calculations. 2018 IEEE 14th International Conference on e-Science (e-Science). 2018; pp 189–200.

(26) Hruska, E.; Balasubramanian, V.; Lee, H.; Jha, S.; Clementi, C. Extensible and scalable adaptive sampling on supercomputers. *Journal of Chemical Theory and Computation* **2020**, *16*, 7915–7925.

(27) Zwier, M. C.; Adelman, J. L.; Kaus, J. W.; Pratt, A. J.; Wong, K. F.; Rego, N. B. et al. WESTPA: An interoperable, highly scalable software package for weighted ensemble simulation and analysis. *Journal of chemical theory and computation* **2015**, *11*, 800–809.

(28) Lawson, C. L.; Kryshtafovych, A.; Adams, P. D.; Afonine, P. V.; Baker, M. L.; Barad, B. A. et al. Cryo-EM model validation recommendations based on outcomes of the 2019 EMDataResource challenge. *Nature Methods* **2021**, *18*, 156–164, DOI: 10.1038/s41592-020-01051-w.

(29) Zhang, Y.; Skolnick, J. Scoring function for automated assessment of protein structure template quality. *Proteins: Structure, Function, and Bioinformatics* **2004**, *57*, 702–710, DOI: 10.1002/prot.20264.

(30) Williams, C. J.; Headd, J. J.; Moriarty, N. W.; Prisant, M. G.; Videau, L. L.; Deis, L. N. et al. MolProbity: More and better reference data for improved all-atom structure validation. *Protein Science* **2018**, *27*, 293–315, DOI: 10.1002/pro.3330.

(31) Barad, B. A.; Echols, N.; Wang, R. Y. R.; Cheng, Y.; Dimaio, F.; Adams, P. D. et al. EMRinger: Side chain-directed model and map validation for 3D cryo-electron microscopy. *Nature Methods* **2015**, *12*, 943–946, DOI: 10.1038/nmeth.3541.

(32) Pintilie, G.; Zhang, K.; Su, Z.; Li, S.; Schmid, M. F.; Chiu, W. Measurement of atom resolvability in cryo-EM maps with Q-scores. *Nature Methods* **2020**, *17*, 328–334, DOI: 10.1038/s41592-020-0731-1.

(33) Turilli, M.; Balasubramanian, V.; Merzky, A.; Paraskevakos, I.; Jha, S. Middleware Building Blocks for Workflow Systems. *Computing in Science Engineering* **2019**, *21*, 62–75, DOI: 10.1109/MCSE.2019.2920048.

(34) MDFF Integration with EnTK. https://github.com/radical-collaboration/MDFF-EnTK, 2019.

(35) Merzky, A.; Turilli, M.; Titov, M.; Al-Saadi, A.; Jha, S. Design and performance characterization of radical-pilot on leadership-class platforms. *arXiv preprint arXiv:2103.00091* **2021**,

Supporting Information Available

6.1 Performance Results of CODH

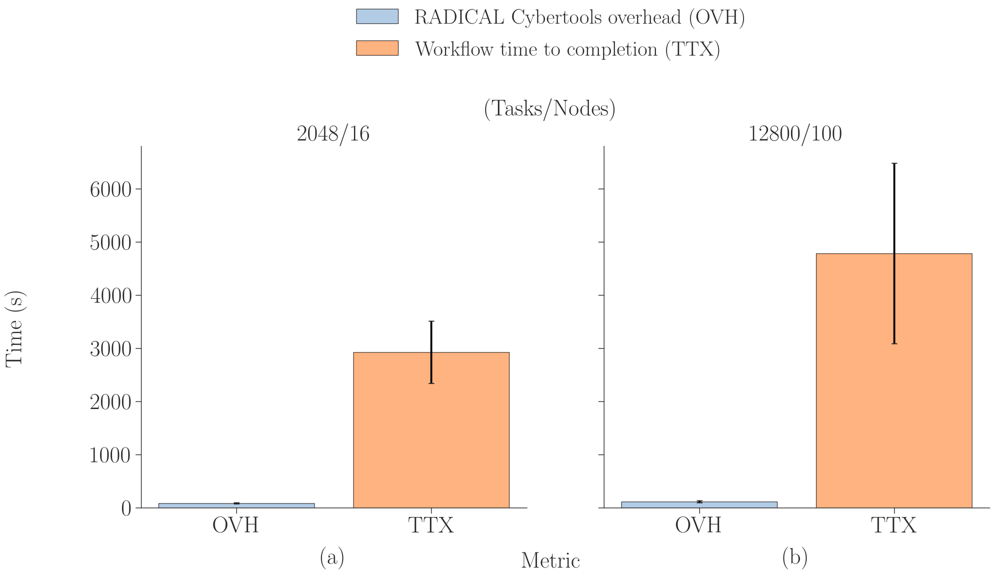
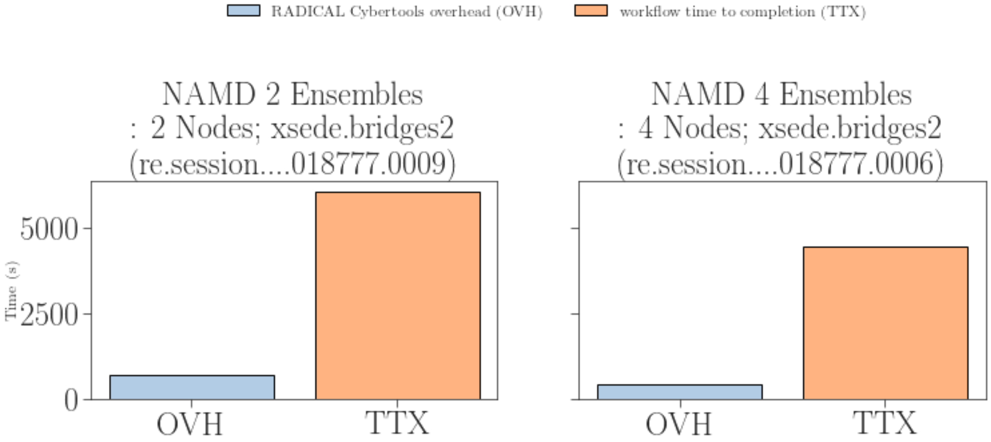


Figure 8: RADICAL Overheads for CODH on 16 and 100 Summit compute nodes. The simulation time per iteration is 80ps and TTX tends to decrease with the increasing of the number of compute nodes.

Figure 8 shows the RCT overheads of running CODH with 16 replicas and 100 replicas in which 99.44±20 seconds and 113.61 ± 20 seconds are measured respectively. The bar plots with a light blue color (OVH label on X-axis) indicate the overhead in seconds of completing the MDFF-EnTK workflow, and the bars with orange color (TTX label on X-axis) report estimated workflow time to completion for sixteen iterations. The experiments of id 7, 8 and 9 in the table 1 are corresponding to these plots.

Figure 9 shows the RCT overheads of running 2 and 4 replicas on PSC Bridges in which 636.18 ± 10 seconds and 332.39 ± 10 seconds are measured respectively. These overheads include 87 seconds and 42 seconds startup time (bootstrap) for 2 and 4 replicas and we believe that initial access delay performance on Lustre filesystem is fluctuated.



(a)

(b)

Figure 9: RADICAL Overheads on PSC Bridges2 compute nodes. (ADK, 1.8A, 2/4 replicas, 2048ps timescale)