An interoperable implementation of collective-variable based enhanced sampling methods in extended phase space within the OpenMM package

Shitanshu Bajpai,[a] Brian K. Petkov,[b] Muchen Tong,[b] Charlles R. A. Abreu*[c] Nisanth N. Nair*[a] Mark E. Tuckerman*[b,d,e,f]

Collective variable (CV)-based enhanced sampling techniques are widely used today for accelerating barrier-crossing events in molecular simulations. A class of these methods, which includes Temperature Accelerated Molecular Dynamics (TAMD)/driven-Adiabatic Free Energy Dynamics (d-AFED), Unified Free Energy Dynamics (UFED), and Temperature Accelerated Sliced Sampling (TASS), uses a extended variable formalism to achieve quick exploration of conformational space. These techniques are powerful, as they permit enhancing the sampling of a large number of CVs simultaneously compared to other techniques. Extended variables are kept at a much higher temperature than the physical temperature by ensuring adiabatic separation between the extended and physical subsystems and employing rigorous thermostatting. In this work, we present a computational platform to perform extended phase space enhanced sampling simulations using the open-source molecular dynamics engine `OpenMM`. The implementation allows users to have interoperability of sampling techniques, as well as employ state-of-the-art thermostats and multiple time-stepping. This work also presents protocols for determining the critical parameters and procedures for reconstructing high-dimensional free energy surfaces. As a demonstration, we present simulation results on the high dimensional conformational landscapes of the alanine tripeptide *in vacuo*, tetra-N-methylglycine (tetra-sarcosine) peptoid in implicit solvent, and the Trp-cage mini protein in explicit water.

[a] Department of Chemistry, Indian Institute of Technology Kanpur (IITK), 208016 Kanpur, India
E-mail: nnair@iitk.ac.in

[b] Department of Chemistry, New York University (NYU), New York, New York, 10003, United States
E-mail: mark.tuckerman@nyu.edu

[c] Chemical Engineering Department, Escola de Química, Universidade Federal do Rio de Janeiro, Rio de Janeiro, RJ 21941-909, Brazil
E-mail: abreu@eq.ufrj.br

[d] Courant Institute of Mathematical Sciences, New York University (NYU), New York, New York10003, United States

[e] NYU-ECNU Center for Computational Chemistry at NYU Shanghai, 3663 Zhongshan Road North, Shanghai200062, China

[f] Simons Center for Computational Physical Chemistry, New York University, New York, New York10003, United States

Introduction

Molecular dynamics (MD) simulations are widely used in the exploration of conformational landscapes of chemical and biological systems and for the prediction of free energetics, reaction rates, and mechanisms of physicochemical processes.[1–3] The time step used for the integration of the equations of motion in MD simulations is on the order of femtoseconds while the structural transformations that involve barrier-crossing events occur on much longer timescales. As a result,observing high-barrier-crossing events often requires impractically long MD simulations. To overcome this difficulty, enhanced sampling methods are used in MD simulations. These methods enable the system to explore high free energy regions and accelerate barrier crossing events.[2,4–10] These techniques also provide the means to reconstruct the underlying equilibrium probability distributions and free energy surfaces.

Enhanced sampling in MD simulations can be achieved by enhancing the fluctuations of a set of coarse-grained coordinates of the system, known as collective variables (CVs). CVs are arbitrary functions of physical degrees of freedom. If CVs are chosen appropriately, free energy surfaces computed as a function of these coordinates can be used to predict the free energetics of physicochemical processes; For more details, see Refs.[11–14]. In the following discussions, we consider that a set of CVs, $\{q_\alpha\}$, is *a priori* selected for boosting the sampling of the relevant conformations and for representing the free energy surface; however, the methods presented are not limited to this choice and can be equally applied to CVs derived in other ways such as from machine learning[15,16].

There are several techniques for enhanced sampling employing a boosting of CV motion. Methods such as umbrella sampling[17,18] and metadynamics[19,20], among others[8,21–40], use bias potentials, while other techniques employ high temperature.[2,41–43]Accelerating the sampling of coordinates can also be done without the help of biases or boosts applied to the CVs. Some of these methods directly bias the potential energy surface or use multiple replica/ ensemble simulations, wherein coordinates of the physical system are exchanged with replicas of the system at a high temperature and/or with a high-dimensional potential bias.[2,44–53]

In umbrella sampling MD simulations, a harmonic restraint potential is applied as a bias on one or, at most, two CVs to obtain a biased distribution along these CVs. Disjoint distributions along the CVs from independent umbrella biases are then combined into a single equilibrium distribution by using the Weighted Histogram Analysis Method

(WHAM)[54,55] or alternative approaches.[56,57] Although umbrella sampling permits controlled and steered sampling along a CV, it is severely limited in the dimensionality of the CV space it can sample. Distributions obtained from adjacent umbrella windows must overlap significantly to allow accurate prediction of free energies. This controls the minimum number of umbrella windows required in the US simulations and determines the overall computational cost. Several techniques have been proposed to improve the efficiency and accuracy of the original umbrella sampling method.[58–63]

In metadynamics,[19,29,64] a bias potential is incrementally constructed during the dynamics based on the CV trajectory. Thus, metadynamics permits an efficient self-guided exploration of the conformational space and determination of the free energy landscape.[8,28,65] The method has been applied to a large number of research problems in different domains of science and engineering; see Refs.[14,66–68]. In metadynamics, the simulation time required to flatten the underlying free energy landscape increases exponentially with the dimensionality of the CV space. Thus, the original metadynamics is practically limited to two or three CVs only. Modified versions of metadynamics methods such as bias-exchange metadynamics,[69,70] parallel-bias metadynamics,[71,72] and others[32,73,74] are designed to overcome this problem. In a spirit similar to metadynamics, other sampling techniques such as variationally optimized free-energy flooding,[21] Gaussian mixture-based enhanced sampling,[27] Reweighted autoencoded variational Bayes for enhanced sampling[75], and on-the-fly probability enhanced sampling[76] have been proposed.

Another class of enhanced sampling techniques is based on temperature acceleration of CVs, inspired by the Adiabatic Free Energy Dynamics approach[77]. Temperature Accelerated Molecular Dynamics (TAMD) and driven–Adiabatic Free Energy Dynamics (d–AFED) are two such methods.[42,43] Temperature-accelerated methods offer the advantage that enhanced sampling can be performed on a large number of CVs without incurring the scaling problems inherent to biasing methods like metadynamics.[9,10]

The efficiency of the extended Lagrangian formulation of d-AFED/TAMD has motivated the development of other more powerful methods. In unified free energy dynamics (UFED)[78], a metadynamics-like bias is applied to the extended system variables along with d-AFED/TAMD. This combination of metadynamics and high-temperature boosting of CV dynamics has been found to be very efficient in exploring high dimensional free energy landscapes.[79–84]

Temperature accelerated sliced sampling (TASS)[63] is an approach in which the d-AFED/TAMD Lagrangian is used along with an umbrella bias potential applied to one of the CVs and a metadynamics bias applied to a subset of CVs. The main advantage of having the umbrella bias is to control and steer the exploration of the free energy surface along a specific CV.[62] Entropy-hindered transitions are boosted in this method by the umbrella bias potentials. Moreover, a large number of transverse coordinates are sampled simultaneously with the aid of metadynamics and d-AFED/TAMD. This technique also permits sampling of different transverse coordinates in different umbrella windows. See Refs.[9,10] for detailed reviews on TASS.

Temperature-accelerated methods such as d-AFED/TAMD, UFED, and TASS require careful selection of thermostats and extended Lagrangian parameters in order to achieve an adiabatic separation between the auxiliary degrees of freedom and the physical degrees of freedom. As these techniques handle a large number of CVs (i.e., beyond three), reconstruction of high dimensional free energy surfaces requires special methods (see Ref.[85] for an example). In this paper, we present recipes for the choice of thermostats and simulation parameters and approaches for reconstructing high-dimensional free energy landscapes.

Given the widespread adoption of `OpenMM`[86,87] as an open source and inter-operable engine for MD, we have developed a compatible library called `UFEDMM`[88] to facilitate the usage of all of the techniques in the family of d-AFED/TAMD. We demonstrate the application of this implementation by presenting studies of high-dimensional conformational landscapes of the alanine tripeptide in vacuum, a tetra-N- methylglycine (tetra-sarcosine) oligopeptoid in implicit solvent, and the folding/unfolding landscape of the Trp-cage mini-protein in explicit solvent.

1. Theory and Computational Methods

1.1. Theory of d-AFED/TAMD, UFED, and TASS Methods

The definition of the free energy surface $F(s_1,...,s_n)$ associated with a set of n collective variables (CVs) $q_1(\mathbf{R}),...,q_n(\mathbf{R})$, where $\mathbf{R}$ is a complete set of atomic coordinates, is

$$\begin{aligned} F(s_1,...,s_n) &= -\beta^{-1}\ln\int d\mathbf{R}\, e^{-\beta V(\mathbf{R})}\prod_{\alpha=1}^{n}\delta\left(q_\alpha(\mathbf{R})-s_\alpha\right) \\ &\quad + \text{constant} \\ &\equiv -\beta^{-1}\ln P(s_1,...,s_n) + \text{constant} \end{aligned} \tag{1}$$

where $V(\mathbf{R})$ is the potential energy of the system. The d-AFED/TAMD enhanced sampling approach employs an extended Lagrangian of the form

$$\begin{aligned} \mathcal{L}^{\mathrm{d-AFED}}(\mathbf{R},\dot{\mathbf{R}},s,\dot{s}) &= \mathcal{L}^0(\mathbf{R},\dot{\mathbf{R}}) + \sum_{\alpha=1}^{n}\Big[\frac{1}{2}\mu_\alpha\dot{s}_\alpha^2 \\ &\quad -\frac{\kappa_\alpha}{2}\left(q_\alpha(\mathbf{R})-s_\alpha\right)^2\Big] \end{aligned} \tag{2}$$

where $\mathcal{L}^0(\mathbf{R},\dot{\mathbf{R}}) = T(\dot{\mathbf{R}}) - V(\mathbf{R})$ is the original Lagrangian of the system, $\dot{\mathbf{R}}$ is a complete set of atomic velocities, $T(\dot{\mathbf{R}})$ is the kinetic energy, $\mathbf{s}$ and $\dot{\mathbf{s}}$ are complete sets of auxiliary degrees of freedom and corresponding velocities, respectively, μ_α is the mass-like parameter of the auxiliary degree of freedom s_α, and κ_α is the restraining force of the spring that couples the s_α and the corresponding CV $q_\alpha(\mathbf{R})$. The harmonic coupling in Eq. (2) arises from the replacement of the δ-functions in Eq. (1) with a set of Gaussians using the identity

$$\delta\left(q_\alpha(\mathbf{R})-s_\alpha\right) = \lim_{\kappa_\alpha\to\infty}\left(\frac{\beta\kappa_\alpha}{2\pi}\right)^{1/2} e^{-\beta\kappa_\alpha\left(q_\alpha(\mathbf{R})-s_\alpha\right)^2/2} \tag{3}$$

CVs are constructed over the physical degrees of freedom and are functions of atomic coordinates. The auxiliary degrees of freedom are kept at a higher temperature than the physical degrees of freedom by coupling them with two sets of thermostats. We define, $\beta = (k_\mathrm{B}T)^{-1}$, k_B is the Boltzmann constant, T is the temperature of the physical system,

and $\tilde{\beta} = \left(k_{\rm B}\tilde{T}\right)^{-1}$, where $\tilde{T}$ is the temperature of the auxiliary variables, and $\tilde{T} \gg T$. The auxiliary degrees of freedom are also adiabatically decoupled from the physical degrees of freedom in order to ensure that the free energy landscape is properly sampled [41,43]. The combination of high temperature applied to the auxiliary space and adiabatic decoupling are the key factors permitting enhanced exploration of the CV space.

On the other hand, in the UFED approach, an additional bias potential is added in the auxiliary space to provide a further boost to the sampling of the auxiliary variables. The Lagrangian used in UFED simulations is

$$\mathcal{L}^{\rm UFED}(\mathbf{R}, \dot{\mathbf{R}}, s, \dot{s}) = \mathcal{L}^{\rm d-AFED}(\mathbf{R}, \dot{\mathbf{R}}, s, \dot{s}) - V^{\rm b}(\mathbf{s}, t) \tag{4}$$

where $V^{\rm b}(\mathbf{s}, t)$ is a time-dependent well-tempered metadynamics bias [28,29], given by

$$V^{\rm b}(\mathbf{s}, t) = \sum_{\tau<t} w_\tau \, \exp\left[-\frac{\|\mathbf{s} - \mathbf{s}_\tau\|^2}{2(\delta s)^2}\right] \tag{5}$$

with

$$w_\tau = w_0 \exp\left[-\frac{V^{\rm b}(\mathbf{s}_\tau, \tau)}{k_{\rm B}\Delta T}\right], \tag{6}$$

and $\mathbf{s}_\tau \equiv \mathbf{s}(\tau)$ as in well-tempered metadynamics (WT-MTD). [28,29] Here, τ is a discretized time, and the Gaussian potentials are updated incrementally. In the above, w_τ is the height of the Gaussian deposited at time τ, δs is the width of the Gaussian, and ΔT (in Kelvin) is a parameter that controls the change of the Gaussian height.

TASS uses another variant of the d-AFED/TAMD Lagrangian:

$$\begin{aligned}\mathcal{L}_h^{\rm TASS}(\mathbf{R}, \dot{\mathbf{R}}, \mathbf{s}, \dot{\mathbf{s}}) = {} & \mathcal{L}^{\rm d-AFED}(\mathbf{R}, \dot{\mathbf{R}}, \mathbf{s}, \dot{\mathbf{s}}) - \\ & W_h^{\rm b}(s_1) - V^{\rm b}(\mathbf{s}^{\rm m}, t), \\ & h = 1, \cdots, M \ .\end{aligned} \tag{7}$$

Here, two kinds of bias potentials $W_h^{\rm b}(s_1)$ and $V^{\rm b}(\mathbf{s}^{\rm m}, t)$, are added to the auxiliary degrees of freedom. The bias $W_h^{\rm b}(s_1)$ is the umbrella bias potential given by

$$W_h^{\rm b}(s) = \frac{1}{2} k_h \left[s_1(\mathbf{R}) - \xi_h\right]^2, \ \ h = 1, \cdots, M \ , \tag{8}$$

and is applied (only) along one auxiliary variable, s_1. The umbrella biases are centered at M different values of the auxiliary variable at values ξ_h, $h = 1, \cdots, M$. A well-tempered metadynamics bias, $V^{\rm b}(\mathbf{s}^{\rm m}, t)$ (Eq. (5)) is applied along a small set of auxiliary variables $\mathbf{s}^{\rm m} \equiv (s_2, \cdots, s_m)$, and $m \leq n$.

1.2. Reconstruction of Free Energy Surfaces in d-AFED/TAMD, UFED, and TASS Methods

1.2.1. d-AFED/TAMD

In d-AFED/TAMD, the free energy surface $F(\mathbf{s})$ at the physical temperature β can be directly computed from the distribution of the auxiliary variables $\mathbf{s}$ from the d-AFED/TAMD simulation using the CV temperature $\tilde{\beta}$ on the $\mathbf{s}$ variables: [42,43]

$$F(s_1, \cdots, s_n) = -\frac{1}{\tilde{\beta}} \ln \tilde{P}(s_1, \cdots, s_n) \ , \tag{9}$$

and

$$\begin{aligned}\tilde{P}(s_1, \cdots, s_n) &= C \int_0^{t_{\rm max}} dt \ \prod_{\alpha=1}^{n} \delta\left(q_\alpha(\mathbf{R}(t)) - s_\alpha\right) \\ &\approx C \int_0^{t_{\rm max}} dt \prod_{\alpha=1}^{n} e^{-\tilde{\beta}\kappa_\alpha (q_\alpha(\mathbf{R}(t)) - s_\alpha)^2/2}\end{aligned} \tag{10}$$

where C is the normalization constant, and $t_{\rm max}$ is the total simulation time. The origin of Eq. (10) is an equating of the time average to the phase-space average

$$\begin{aligned}\tilde{P}(s_1 \cdots, s_n) &= C \left\langle \prod_{\alpha=1}^{n} \delta\left(q_\alpha(\mathbf{R}) - s_\alpha\right) \right\rangle \\ &\approx \left\langle \prod_{\alpha=1}^{n} e^{-\tilde{\beta}\kappa_\alpha (q_\alpha(\mathbf{R}) - s_\alpha)^2/2} \right\rangle\end{aligned} \tag{11}$$

In terms of $\tilde{P}(s_1, ..., s_n)$, the free energy surface $F(s_1, ..., s_n)$ is given by Eq. (9). In practice, $\tilde{P}(s_1, ..., s_n)$ is computed by recording an n-dimensional histogram of the auxiliary variables over the d-AFED/TAMD trajectory and, at the end, normalizing the histogram and shifting it so that the global minimum has zero free energy.

1.2.2. UFED

Similar to d-AFED/TAMD, the free energy surface $F(\mathbf{s})$ as a function of auxiliary variables at temperature T is constructed from the reweighted probability distribution $\tilde{P}^{\rm u}(s_1, ..., s_n)$ of the auxiliary variables $s_1, \cdots, s_n$ at temperature $\tilde{T}$ from a UFED trajectory using [2,43]

$$F(s_1, \cdots, s_n) = -\tilde{\beta}^{-1} \ln \tilde{P}^{\rm u}(s_1, \cdots, s_n) \ . \tag{12}$$

The probability distribution $\tilde{P}^{\rm u}$ is computed from the normalized histogram obtained using the trajectory of the auxiliary variables following, [85,89]

$$\tilde{P}^{\rm u}(s_1, \cdots, s_n) = \frac{\int_0^{t_{\rm max}} dt \, A(t) \, \prod_\alpha^n \delta\left(s_\alpha(t) - s_\alpha\right)}{\int_0^{t_{\rm max}} dt \, A(t)} \tag{13}$$

where

$$A(t) = \exp\left[\tilde{\beta}\left\{V^{\rm b}\left(\mathbf{s}(t), t\right) - c(t)\right\}\right] \ , \tag{14}$$

$$c(t) = \frac{1}{\tilde{\beta}} \ln\left[\frac{\int d\mathbf{s} \, \exp\left[\tilde{\beta}\gamma V^{\rm b}(\mathbf{s}(t), t)\right]}{\int d\mathbf{s} \, \exp\left[\tilde{\beta}(\gamma - 1) V^{\rm b}(\mathbf{s}(t), t)\right]}\right] \ , \tag{15}$$

and

$$\gamma = \frac{T + \Delta T}{\Delta T} \ .$$

Note that the histogram $\tilde{P}^{\rm u}(s_1, ..., s_n)$ arises from the use of the well-tempered biasing potential in Eq. (5) within the UFED scheme. An alternative approach to reconstructing the free energy surface is by computing the derivative of the free energy directly from the ensemble average of the harmonic coupling force $\kappa_\alpha(q_\alpha(\mathbf{R}) - s_\alpha)$ between $q_\alpha(\mathbf{R})$ and s_α in Eq. (2). Integration of the derivatives can then be done using a basis-set representation [90] or by a neural-network representation of free energy. [15] However, such approaches become computationally expensive with increasing dimensionality, and thus they are not used in this work.

1.2.3. TASS

For the reconstruction of free energy surfaces from TASS simulations, we used a mean-force approach as discussed in Ref.[91]. In this method, the one-dimensional projection of the free energy surface along the umbrella coordinate, s_1, is computed first using numerical integration:

$$F_1(s_1') = \int^{s_1'} ds_1 \left(\frac{dF}{ds_1}\right) \approx -\sum_{h=1}^{M_{s'}-1} \Delta\xi_h \frac{1}{2}\left[g_h + g_{h+1}\right] \quad (16)$$

where the mean gradient of free energy g_h is given by,

$$g_h = \langle k_h \left[s_1 - \xi_h\right]\rangle_{\xi_h} \ , \quad (17)$$

and M_s' is the grid point corresponding to the value of s_1, i.e., s_1'. The above is computed from time averaging after a time-dependent reweighting of the bias potential as,

$$g_h = \frac{\int dt\, A(t)\, \left[k\left(s_1(t) - \xi_h\right)\right]}{\int dt\, A(t)} \quad (18)$$

wherein the reweighting factor $A(t)$ is computed following Eq. (14). Then, we construct the multi-dimensional free energy surface as

$$F(s_1, \cdots, s_n) = F_1(s_1) + \Delta F_{s_1}(s_2, \cdots, s_n) \ . \quad (19)$$

Here,

$$\Delta F_{s_1}(s_2, \cdots, s_n) = -\frac{1}{\tilde{\beta}} \ln \tilde{P}^{\mathrm{u}}_{s_1}(s_2, \cdots, s_n) \quad (20)$$

and $\tilde{P}^{\mathrm{u}}_{s_1}$ is the slice of the high-dimensional reweighted probability distribution at s_1. This reweighted probability distribution is computed in the same way as in Eq. (13), for all centers of the umbrella bias, i.e., $\xi_h, \ h = 1, \cdots, M$.

1.3. OpenMM

OpenMM[86] is a free, open-source MD library that operates in diverse hardware platforms, such as multi-core processors and graphics processing units (GPUs). It is highly customizable by design, distinguishing itself from other popular MD software packages. In OpenMM, implementing a new potential energy term or integration algorithm can be as simple as providing character strings with proper algebraic expressions. The program will parse these expressions, optimize and differentiate them, and generate execution kernels for the desired hardware platform. One of the handy features available in OpenMM is the `CustomCVForce` class, which allows the inclusion of collective variables (CVs) in the energy expression. The same customizable classes used for implementing interaction potentials are available for defining such CVs, and OpenMM will automatically apply the chain rule for computing the forces that depend on them.

We implemented a Python library that extends OpenMM to facilitate the use of extended phase-space dynamics methods for enhanced sampling of collective variables. This library, called `UFEDMM`, is able to launch simulations applying UFED (Figure 1), d-AFED/TAMD (Figure 2), metadynamics, well-tempered metadynamics, or other types of biased extended phase-space simulation. It is also flexible enough to allow the execution of related methods such as TASS (Figure 3) and λ-AFED[92] or λ-metadynamics[93]

```
import openmm
import ufedmm
from numpy import pi
from openmm import app, unit
from sys import stdout
######## Alanine Dipeptide System ********************
pdb = app.PDBFile("alanine-dipeptide.pdb")
pdb.topology.setUnitCellDimensions([2.5*unit.nanometers]*3)
atoms = [f"{a.name}:{a.residue.name}" for a in pdb.topology.atoms()
    ]
dihedral_atoms = {
    "phi": map(atoms.index, ["C:ACE", "N:ALA", "CA:ALA", "C:ALA"]),
    "psi": map(atoms.index, ["N:ALA", "CA:ALA", "C:ALA", "N:NME"]),
}
system = app.ForceField("amber03.xml").createSystem(
    pdb.topology, nonbondedMethod=app.NoCutoff,
    constraints=app.HBonds, removeCMMotion=False,
)
######## Thermostat Parameters ************************
temp, gamma = 300*unit.kelvin, 10/unit.picoseconds
######## Extended Space Parameters ********************
mass = 30*unit.Dalton*(unit.nanometer/unit.radians)**2 # Mass
Ks=1000*unit.kilojoules_per_mole/unit.radians**2 #Coupling constant
Ts = 1500*unit.kelvin                                  # Temperature
######## Metadynamics Parameters **********************
sigma, height, period = pi/10, 2*unit.kilojoules_per_mole, 200
######## Define Collective Variables ******************
phi = ufedmm.CollectiveVariable("phi", openmm.CustomTorsionForce("
    theta"))
phi.force.addTorsion(*dihedral_atoms["phi"], [])
psi = ufedmm.CollectiveVariable("psi", openmm.CustomTorsionForce("
    theta"))
psi.force.addTorsion(*dihedral_atoms["psi"], [])
######## Define Auxiliary Variables *******************
s_phi = ufedmm.DynamicalVariable("s_phi", -pi, pi, mass, Ts, phi,
    Ks, sigma=sigma)
s_psi = ufedmm.DynamicalVariable("s_psi", -pi, pi, mass, Ts, psi,
    Ks, sigma=sigma)
######### Set up and Perform UFED Simulation ***********
ufed = ufedmm.UnifiedFreeEnergyDynamics([s_phi, s_psi], temp,
    height, period)
integrator = ufedmm.GeodesicLangevinIntegrator(temp, gamma, 2*unit.
    femtoseconds)
platform = openmm.Platform.getPlatformByName("CPU")
simulation = ufed.simulation(pdb.topology, system, integrator,
    platform)
simulation.context.setPositions(pdb.positions)
simulation.context.setVelocitiesToTemperature(temp)
simulation.reporters.append(ufedmm.StateDataReporter(
    stdout, 100, step=True, multipleTemperatures=True, variables=
    True, speed=True,
))
simulation.step(1000000)
```

Figure 1. UFED code snippet for alanine dipeptide system

```
constants = {
    "Ks": 1000*unit.kilojoules_per_mole/unit.radians**2, # Coupling
     constant Real-Aux
}

#############d-AFED/TAMD ***********************************
s_phi = ufedmm.DynamicalVariable("s_phi", -pi, pi, mass, Ts, phi,
    Ks, sigma=None)
s_psi = ufedmm.DynamicalVariable("s_psi", -pi, pi, mass, Ts, psi,
    Ks, sigma=None)
#########################################################
```

Figure 2. TAMD code snippet to couple real and auxiliary space for alanine dipeptide system

```
constants = {
    "Ks": 1000*unit.kilojoules_per_mole/unit.radians**2, # Coupling
     constant Real-Aux
    "Ku": 50*unit.kilojoules_per_mole/unit.radians**2,   # Umbrella
     coupling constant
    "Umb_center": -3.2*unit.radians,                     # Umbrella
     center
}
potential = (
    f"0.5*Ks*min(dphi, {2*pi}-dphi)^2 + 0.5*Ku*min(dUmb, {2*pi}-
    dUmb)^2;"
    "dphi=abs(phi-s_phi);"
    "dUmb=abs(s_phi-Umb_center)"
)
############## Umbrella Bias along phi ************
s_phi = ufedmm.DynamicalVariable(
    "s_phi", -pi, pi, mass, Ts, phi, potential, **constants
)
############ MTD bias along psi******************
s_psi = ufedmm.DynamicalVariable("s_psi", -pi, pi, mass, Ts, psi,
    Ks, sigma=sigma)
```

Figure 3. TASS code snippet to add umbrella bias along ϕ and WTMTD bias along ψ in alanine dipeptide system

with minimal extra setup effort. A repository of CVs commonly used in the study of biochemical systems, such as radius of gyration, coordination number, and helical content, is provided together with UFEDMM. However, it is possible to define other CVs using OpenMM's custom force classes or a plugin available for linking OpenMM with PLUMED [94], a well-known library that can treat a wide spectrum of CV types.

Multiple strategies exist for simulating an extended Lagrangian system in OpenMM. Before describing two of them, we point out that a `Context` object, responsible for storing the current state of a system (particle positions and velocities, box shape, etc.), can contain global variables that affect its potential energy and forces. Also, `CustomIntegrator` objects can include steps that modify the values of such variables. Hence, if we use them to store each auxiliary variable's mass-position-velocity triad, we can define a custom integrator for solving the appropriate equations of motion. Depending on the thermostats applied to the auxiliary variables, other triads will also be necessary. This strategy can become cumbersome to implement in a general form and inefficient for involving too many custom-integrator steps to update global variables.

The strategy we adopt here is suited for a fixed-volume simulation box with periodic boundary conditions. For every auxiliary variable s_α present in the extended Lagrangian, we add a new particle to the system and define the relation

$$s_\alpha = s_\alpha^{\min} + (s_\alpha^{\max} - s_\alpha^{\min})\lambda_\alpha\left(\frac{x_{N+\alpha}}{L_x}\right), \qquad (21)$$

where x_i is the x coordinate of the i-th particle, N is the number of physical particles, L_x is the box size in the x direction, $[s_\alpha^{\min}, s_\alpha^{\max})$ is a range specified for s_α, and λ_α is a function that depends on the periodicity status of s_α. If it is a periodic quantity, then

$$\lambda_\alpha(w) = w - \lfloor w \rfloor, \qquad (22)$$

where $\lfloor w \rfloor$ is the greatest integer less than or equal to w. The function above transfers the periodic boundary conditions from the box's x dimension to s_α, and keeps the latter's value inside the specified range. Otherwise, if s_α is non-periodic, then

$$\lambda_\alpha(w) = 2\min\left(w - \lfloor w \rfloor, \lceil w \rceil - w\right), \qquad (23)$$

where $\lceil \cdot \rceil$ is the least integer greater than or equal to w. The function above imposes reflective boundary conditions at $s_\alpha = s_\alpha^{\min}$ and $s_\alpha = s_\alpha^{\max}$. When the $(N+\alpha)$-th particle crosses a plane $x = kL_x/2$, for any integer k, its related variable s_α bounces back while $\dot{s}_\alpha$ reverses sign as would occur if s_α had collided elastically with a hard wall.

The relations above make s_α amenable to treatment via `CustomCVForce`. Fortunately, the minimum, floor, and ceiling functions are available for custom potential definitions in OpenMM. In the extended-Lagrangian framework, the actual dynamical variables are now the x coordinates of the added particles. Note that their y and z coordinates are irrelevant and can be left immobile in practice. The added particles interact with the original ones through the potential energy extension, which depends on $\mathbf{s}$ and the other CVs in $\mathbf{q}(\mathbf{R})$. To specify the mass of each new particle, we apply the chain rule to express the kinetic energy extension as a function of $\dot{x}_{N+\alpha}$ instead of $\dot{s}_\alpha$, which makes

$$m_{N+\alpha} = \mu_\alpha\left(\frac{\mathrm{d}s_\alpha}{\mathrm{d}x_{N+\alpha}}\right)^2 = \mu_\alpha\nu_\alpha\left(\frac{s_\alpha^{\max} - s_\alpha^{\min}}{L_x}\right)^2, \qquad (24)$$

where $\nu_\alpha = (\mathrm{d}\lambda_\alpha/\mathrm{d}w)^2$. In practice, if we exclude the boundary points, we set $\nu_\alpha = 1$ for a periodic variable or $\nu_\alpha = 4$ for a non-periodic variable.

Implementing d-AFED/TAMD requires two different thermostats coupled separately to the original and added particles. OpenMM's built-in class `NoseHooverIntegrator` has this capability. For greater flexibility, we implemented custom integrators in UFEDMM employing the massive thermostatting algorithm, in which each degree of freedom is separately thermostatted. Massive thermostats can control the temperature of the system more strictly than can global thermostats, in which a single thermostat is connected to the entire system. This is particularly important for d-AFED/TAMD and related methods, as they can more effectively prevent heat flow from the hot auxiliary variables to the cold physical degrees of freedom, in much the same way as is done in the Car-Parrinello method [95,96]. The integrators available in UFEDMM follow the middle splitting scheme [97,98]. In the "middle" scheme, at every time step, thermostats act on the particle velocities between two half-step displacements of the particle coordinates. The integrators also allow multiple time-stepping via the Reference System Propagation Algorithm (RESPA) [99]. For cases in which the interaction forces comprise two distinct components with different characteristic time scales, the splitting formula that provides the numerical integrator is

$$\begin{aligned} e^{\Delta t\mathcal{L}} \;=\; & e^{\frac{1-\ell}{2}\Delta t\mathcal{L}_v^{[2]}}\Big[e^{\frac{1-\ell}{2}\frac{\Delta t}{n_1}\mathcal{L}_v^{[1]}} \\ & \Big(e^{\frac{1}{2}\frac{\Delta t}{n_b n_1}\mathcal{L}_r}e^{\frac{\Delta t}{n_b n_1}\mathcal{L}_{\mathrm{bath}}}e^{\frac{1}{2}\frac{\Delta t}{n_b n_1}\mathcal{L}_r}\Big)^{n_b} \\ & e^{\frac{1+\ell}{2}\frac{\Delta t}{n_1}\mathcal{L}_v^{[1]}}\Big]^{n_1} e^{\frac{1+\ell}{2}\Delta t\mathcal{L}_v^{[2]}}, \end{aligned} \qquad (25)$$

where the Liouville operator $\mathcal{L}$ is decomposed as

$$\mathcal{L} = \mathcal{L}_v + \mathcal{L}_r + \mathcal{L}_{\mathrm{bath}} \qquad (26)$$

such that

$$\begin{aligned} \mathcal{L}_r \;&=\; \dot{\mathbf{R}}\cdot\frac{\partial}{\partial\mathbf{R}} + \dot{\mathbf{s}}\cdot\frac{\partial}{\partial\mathbf{s}} \\ \mathcal{L}_v \;&=\; \mathrm{M}^{-1}\mathbf{F}_{\mathbf{R}}\cdot\frac{\partial}{\partial\dot{\mathbf{R}}} + \sum_{\alpha=1}^{n}\frac{\mathcal{F}_\alpha}{\mu_\alpha}\frac{\partial}{\partial\dot{s}_\alpha} \end{aligned} \qquad (27)$$

Here, M is a diagonal matrix of physical masses, $\mathbf{F}$ is the full set of physical forces, and $\mathcal{F}_\alpha$ is the harmonic force on s_α. Dividing $\mathbf{F}_{\mathbf{R}}$ into fast and slow components, $\mathbf{F}_{\mathbf{R}}^{[1]} + \mathbf{F}_{\mathbf{R}}^{[2]}$, respectively, and assuming $\mathcal{F}_\alpha$ is a fast force, we obtain two contributions to $\mathcal{L}_v = \mathcal{L}_v^{[1]} + \mathcal{L}_v^{[2]}$. In Eq. (25), $e^{\Delta t\mathcal{L}_v^{[j]}}$ denotes an impulse caused by the forces of group j, $e^{\Delta t\mathcal{L}_r}$ denotes a particle displacement with constant velocity, and $e^{\Delta t\mathcal{L}_{\mathrm{bath}}}$ denotes the action of thermostats. The size of each complete time step is Δt, and this is the interval between evaluations of the forces in group 2. However, those in group 1 are evaluated more frequently, at an interval $\Delta t/n_1$. In general, we can split the forces into M groups and define an array `respa_loops` $= [n_1, \cdots, n_{M-1}]$, so that the evaluation interval for forces in each group j becomes $\Delta t/\Pi_{k=j}^{M-1} n_k$. Finally, it is possible to further reduce the interval for the displacement-thermostat-displacement sequence by defining another parameter `bath_loops` $= n_{\mathrm{b}}$, where n_{b} is the exponent in the central operator sequence in Eq. 25. To our knowledge, using a splitting like

$$(e^{\frac{1}{2}\frac{\Delta t}{n_{\mathrm{b}} n_1}\mathcal{L}_r}e^{\frac{\Delta t}{n_{\mathrm{b}} n_1}\mathcal{L}_{\mathrm{bath}}}e^{\frac{1}{2}\frac{\Delta t}{n_{\mathrm{b}} n_1}\mathcal{L}_r})^{n_{\mathrm{b}}}$$

instead of the more trivial

$$e^{\frac{1}{2}\frac{\Delta t}{n_1}\mathcal{L}_r}\left(e^{\frac{\Delta t}{n_b n_1}\mathcal{L}_{\text{bath}}}\right)^{n_b}e^{\frac{1}{2}\frac{\Delta t}{n_1}\mathcal{L}_r}$$

has not been previously reported. It can increase integration accuracy and has a small computational overhead, since it does not require additional force evaluations.

1.4. Computational Details

1.4.1. Alanine Tripeptide In Vacuo

The alanine tripeptide *in vacuo* is modeled using the AMBER14SB force field; see Figure 4. MD simulations were carried out using the OpenMM-UFEDMM interface. A time step of 1 fs was used to integrate the equations of motion. Three separate classes of the simulation were run with physical (T=300 K) and extended ($\tilde{T}$=3000 K) system temperatures controlled using massive thermostatting with the Generalized Gaussian Moment Thermostat (GGMT),[100] Nosé-Hoover Chain Thermostat (NHC),[101] and Regulated-Nosé-Hoover-Langevin (R-NHL) thermostat[102]. We took the time constant (τ) to be 40 fs for NHC, R-NHL, and GGMT thermostats, and a friction coefficient (γ) of 1 ps^{-1} and regulation parameter $n = 1$ for R-NHL thermostat. As described in[102], with $n = 1$ the R-NHL thermostat is equivalent to the Stochastic Isokinetic Nosé-Hoover method[103] with a single thermostat per degree of freedom (i.e. $L = 1$). We set the multiple time stepping parameter `respa_loops = [4,1]` for the NHC integrator, which implies that the fast (harmonic bond and harmonic angle) force components were integrated with time step $\delta t = \Delta t/4$ and the remaining slow force components were integrated with a time step $\Delta t = 1$ fs. We took the Ramachandran angles ϕ_1,ϕ_2,ψ_1,ψ_2 of the peptide as the CVs for enhanced sampling. Optimal values of μ and κ were explored in this study (Section 3). In the UFED simulations, a well-tempered metadynamics[29] bias was applied along all four CVs. We took w_0=0.5 kcal mol^{-1}, δs = 0.05 rad, ΔT = 21000 K to construct the well-tempered metadynamics bias.

1.4.2. Tetrasarcosine in Implicit Solvent

We performed d-AFED/TAMD and UFED simulations to compute the conformational landscape of tetrasarcosine in implicit solvent; see Figure 1 (b). We used the GAFF2 force field[104–106] for these simulations. The generalized Born implicit solvent Model was used, with parameters were taken from Ref.[107]. Eight collective variables ϕ_1, ϕ_2, ϕ_3, ψ_1, ψ_2, ψ_3, ω_2, ω_3 were taken for the enhanced conformational sampling of tetrasarcosine. In the d-AFED/TAMD and UFED simulations, we chose κ_α=2.8×10^3 kcal mol^{-1} rad^{-2} and μ_α = 6.0 Da nm^2 rad^{-2} for all the extended space variables. We considered both R-NHL and GGMT thermostats in our simulations. In the UFED simulations, we applied well-tempered metadynamics bias along the ϕ_1 and ϕ_2 angles. All the other simulation parameters, including the temperatures of the physical and the auxiliary variables, were the same as those used for the alanine tripeptide (Section 1.4.1).

1.4.3. Trp-cage in Explicit Water

Trp-cage is a 20 amino acid mini-protein (NLYIQ WLKDG GPSSG RPPPS) first synthesized by Neidigh *et al.*[108]. The protein is known to fold in 4 μs at 300 K and pH 7.0.[109,110] The initial structure was prepared from the folded NMR structure PDB ID IL2Y (Chain A).[108] The protein structure was solvated in a periodic box of 43×43×43 Å^3 containing 2676 flexible TIP3P[111] water molecules and one Cl$^-$ anion to neutralize the protein. We used the AMBER99SB[112] force field for the protein. Long-range electrostatic interactions were evaluated using the Particle Mesh Ewald method. After initial energy minimization, the structure of the solvated protein was equilibrated for 1 ns at 1 bar and 300 K using the R-NHL[102,113,114] thermostat and a Monte Carlo Barostat[115] until the density fluctuations of the system were stabilized. From the equilibrated folded structure, we performed 1 ns of equilibration in the NVT ensemble to generate the initial structures for various umbrella windows of a TASS simulation. The extended variables in TASS were kept at $\tilde{T}$=3000 K, while the physical system was kept at T = 300 K.

Based on earlier reports,[93,116–118] we chose eight CVs for the enhanced conformational sampling: (a) the radius of gyration (Rg) of backbone C$_\alpha$ atoms; (b) root mean square deviation (RMSD) of backbone C$_\alpha$ atoms from the native structure; (c) root mean square deviation (RMSD$_{\text{Helix}}$) of C$_\alpha$ atoms of residues 2-8 (Leu2, Tyr3, Ile4, Gln5, Trp6, Leu7, Lys8) from the native structure; (d) root mean square deviation(RMSD$_{\text{Hcore}}$) of Trp6, Pro12 and Pro17-19 atoms from the native structure; (e) Salt-Bridge (Sb) is defined as the distance between guanidino carbon of Arg16 and Asp9 C$_\gamma$; (f) End-to-End (e2e) defined as the distance between C$_\alpha$ of residue Asn1 and Ser20 ; (f) Alpha-helical content (α_{H}) in the structure; (g) Dihedral-Correlation (Dih$_{\text{corr}}$). Definitions of these CVs are given in SI Section 1.1; see also Table 1. For reasons that will be explained in Section 3.1, we found that scaling up the values of CVs is important in certain cases to facilitate CV oscillation about the corresponding auxiliary variables. The parameters μ and κ chosen for the (scaled) CVs are listed in Table 1.

Umbrella bias potentials were placed along the RMSD CV from 0.25 to 7 Å at a gap of 0.25 Å. The parameter κ_h=0.23×10^{-2} kcal mol^{-1} Å^{-2} was chosen for all the umbrella windows. A well-tempered metadynamics bias was applied along the Rg CV, and the bias parameters w_0 = 0.5 kcal mol^{-1}, δs = 0.05 Å, ΔT=27000 K for windows between 0.5 to 3 Å, and ΔT=117000 K for the windows ranging from 3.25 Å to 7 Å. The starting structure for the NVT equilibration of each window is taken from the (preceding) neighboring equilibrated window.

2. Results

2.1. Optimal Choice of Extended System Parameters and Thermostats

For a successful application of d-AFED/TAMD and related methods, a correct choice of extended system parameters is crucial. These include the coupling constants $\{\kappa_\alpha\}$, the mass-like parameters $\{\mu_\alpha\}$, and the auxiliary temperature $\tilde{T}$. For UFED, one must also specify the Gaussian width δs, the Gaussian height w_0, and the deposition rate. Finally, if a well-tempered bias is employed, then we also need the temperature interval ΔT.

In particular, the parameters $\{\kappa_\alpha\}$ and $\{\mu_\alpha\}$ determine the nature of the harmonic coupling and timescale on which the auxiliary variables move. These two sets of parameters should be chosen such that $q_\alpha(\mathbf{R})$ and s_α are tightly coupled and track each other during the simulation. Intuitively, this

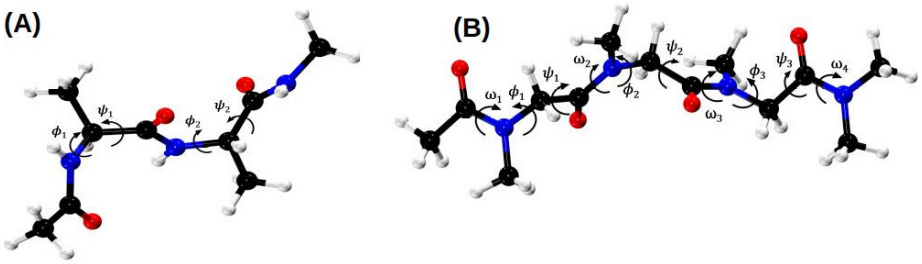

Figure 4. (A) Alanine tripeptide (B) Tetrasarcosine. Ramachandran angles are labeled. Atom colors: O(red), N(blue), C(black), H(white)

Table 1. List of CVs used in the simulation of Trp-cage, the scaling factors to enhance the fluctuation of the CVs, and the μ and the κ parameters corresponding to the scaled-CVs are provided.

CV	Scaling Factor	μ (Da)	κ (kcal mol^{-1} Å^{-2})
Rg	100	2.3	9.08
RMSD	500	0.05	1.19
RMSD$_{\mathbf{Helix}}$	200	2.0	8.0×10^{2}
RMSD$_{\mathbf{Hcore}}$	200	1.5	2.39
Sb	100	0.004	0.47
e2e	100	0.004	0.47
		μ (Da Å^2)	κ (kcal mol^{-1})
α_{H}	10	0.05	8.60
Dih$_{\mathrm{corr}}$	5	0.01	4.89

is consistent with the objective that the high-temperature extended variables represented by s_α effectively "drive" the corresponding system variables to which they are coupled. Further, an adiabatic separation between the physical ($q_\alpha(\mathbf{R})$) and the auxiliary subsystems (s_α) must be ensured through the choice of parameters μ_α for an accurate determination of the free energy surface. The two subsystems are thermostatted at two different temperatures (at least) in d-AFED/TAMD, UFED, and TASS methods. If the temperature of the auxiliary variables is not kept substantially higher than the physical variables, then the advantage of d-AFED/TAMD is lost. Any flow of heat from the hot auxiliary system to the cold physical degrees of freedom can result in incorrect sampling. Lack of an adiabatic separation violates the condition under which we can reweight the sampled probability distribution. To ensure an adiabatic separation, s_α is taken much heavier than the effective mass, $M_{\mathrm{eff},\alpha}$, of q_α ; i.e., $\mu_\alpha >> M_{\mathrm{eff}\alpha}$. On the other hand, μ_α should be small enough to permit sufficient diffusion. A very high value of $\{\kappa_\alpha\}$ can result in high-frequency oscillations, warranting the use of a small time step for accurate integration of the equations of motion, a problem that can be addressed using multiple time-stepping techniques [119]. Thus a balance in accuracy and efficiency should be the aim in choosing the parameters $\{\kappa_\alpha\}$ and $\{\mu_\alpha\}$.

It was reported that a good choice of μ_α is ∼100 times the effective mass of the corresponding CV, which can be computed as [90]

$$M_{\mathrm{eff},\alpha} = \left[\sum_{i=1}^{N} \frac{1}{M_I} \nabla_{\mathbf{R}_i} q_\alpha \cdot \nabla_{\mathbf{R}_i} q_\alpha\right]^{-1} . \qquad (28)$$

A function named `effective_mass()` for computing $M_{\mathrm{eff},\alpha}$ is included in the UFEDMM distribution. [88,120]

A simple test of these parameters, which should be performed before launching a production simulation, consists in plotting $q_\alpha(t)$ and $s_\alpha(t)$ together over a short MD run. Figure 5(a) presents an ideal case, where q has fast oscillations compared to s, yet these fast oscillations clearly occur about the trajectory of s. This pattern was originally shown by Rosso *et al.* [41]. However, Figure 5(b) shows a case in which κ_α is not large enough to restrain q_α along the s_α trajectory, indicating that κ_α needs to be increased. In Figure 5, each MD step is indicated by points; thus, by a visual inspection of these plots, we can also conclude that the chosen time step is appropriate to integrate the fast oscillations of $q_\alpha(t)$.

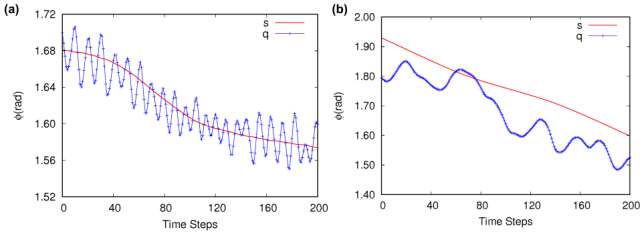

Figure 5. (a) Plot of the physical variable, $q = \phi$, along with the corresponding auxiliary variable s during a d-AFED/TAMD simulation of alanine dipeptide using an ideal set of κ and μ parameters; (b) The same plot of q and s when a much lower value of κ was chosen.

In practice, while an optimal choice of parameters depends on the system configuration and the chosen collective variables, there is a sufficiently broad domain of parameter values over which the temperature acceleration of auxiliary variables is effective. To demonstrate this, we computed the four-dimensional free energy surface $F(\phi_1, \psi_1, \phi_2, \psi_2)$ of the

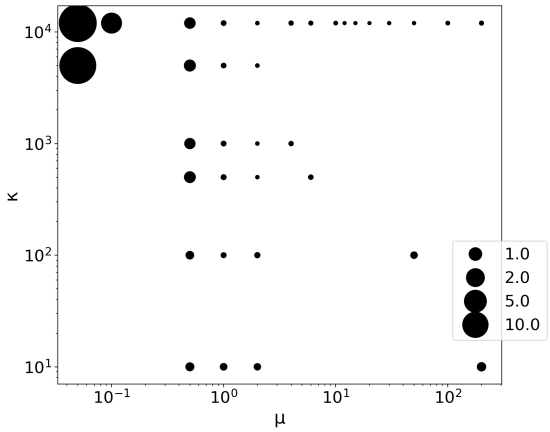

Figure 6. L^2 error (kcal mol^{-1}) of the free energy surface $F(\phi_1, \psi_1, \phi_2, \psi_2)$ of alanine tripeptide *in vacuo* computed for various values of κ (kJ mol^{-1}) and μ (Da nm^2 rad^{-2}). The radius of the circle is proportional to the L^2 error. Here, the L^2 error was computed after 5 ns simulation taking the data after 10 ns for $\kappa = 1200$ kJ mol^{-1} rad^{-2} and $\mu = 6$ Da nm^2 rad^{-2}.

alanine tripeptide *in vacuo* using UFED for a range of values of κ_α and μ_α; see Figure 6. In this example, all four κ_α, $\alpha = 1, ..., 4$ are the same, and we denote the single value as κ; the same is true for μ_α, $\alpha = 1, ..., 4$, and we denote the single value as μ. The L^2 error of a free energy surface $F(\mathbf{s})$ with respect to a reference surface, $F_{\text{ref}}(\mathbf{s})$, is computed as

$$L^2 = \sqrt{\frac{1}{N_{\text{g}}} \sum_{i=1}^{N_{\text{g}}} \left[F(s_i) - F_{\text{ref}}(s_i)\right]^2} \ . \tag{29}$$

Here N_{g} is the total number of grid points, and s_i specifies a grid point in the CV space. L^2 errors were measured after 5 ns for each set of parameters, taking the free energy surface after 10 ns as the reference. There exists a broad domain of parameter values, namely, the high-μ and high-κ regime, for which the L^2 error is the lowest. As mentioned earlier, high κ results in fast oscillations, and thus a small time step is to be used, while high μ results in slow diffusion of the CVs. Small μ and small κ values result in poor adiabatic separation between $q_\alpha(\mathbf{R})$ and s_α. Moreover, $q_\alpha(\mathbf{R})$ may not follow the dynamics of s_α. Thus the small-μ and small-κ parameter space has significant error in the free energy estimates. From Figure 6, we confirm that a good choice of the parameters is $2 \leq \mu \leq 12$ (Da nm^2 rad^{-2}) and $10^3 \leq \kappa \leq 10^4$ (kJ mol^{-1} rad^{-2}), for which L^2 errors are less than 0.3 kcal mol^{-1}.

For certain CV choices, the natural fluctuation of q_α is small. In such cases, a large k_α, and a small time step are required to obtain a proper dynamics of the auxiliary variable. Examples of such CVs are root mean square deviation (RMSD) and coordination number; see Figure 7.In such cases, a time step much smaller than 1 fs is required for accurate integration of the equations of motion. Use of a multiple-time stepping integration scheme can help to ameliorate this problem, and we have implemented the RESPA algorithm [99] as well as a general-purpose multiple-timestep integration class in UFEDMM. However, we also propose an alternative approach to overcome this issue, which is to scale the q_α so as to amplify the oscillations. Extended variables $\{s_\alpha\}$ are coupled to the scaled q_α. The proposed scaling is applied until a proper oscillation of q_α about the trajectory of s_α is observed, and a time step of 1 fs is adequate to integrate the fast oscillations of q_α; see Figure 7b. The values of $\mathbf{s}$ are then scaled back to their original values while computing the free energy surface, which does not affect the final result. Since the ϕ and ψ CVs are of the same type, scaling is therefore not required. This procedure was needed for the CVs used in the case of Trp-cage simulation; see Table 1.

In order to maintain separate temperatures and minimize energy flow between the physical and auxiliary systems, rigorous thermostatting is indispensable. A massive thermostat is recommended for quick thermalization of all the degrees of freedom and to alleviate the errors due to energy leaks between the physical and the auxiliary subsystems. In this way, the required adiabaticity can be maintained. In order to explore the performance of various thermostats, we considered three methods here, namely the GGMT, NHC, and R-NHL algorithms. We carried out this benchmark study using alanine tripeptide *in vacuo*.

First, we monitored the running average of the temperature of the auxiliary (s_α) and real degrees of freedom as a function of time; see Figure 8. The correct system temperature is achieved quickly with the GGMT and R-NHL thermostats, whereas the NHC thermostat is not as effective as can be expected given the thermostat properties. Next, we monitored the convergence of the free energy surface achieved by the different thermostat approaches. In particular, we computed the four-dimensional free energy surface of the alanine tripeptide *in vacuo* as a function of the Ramachandran angles $(\phi_1, \psi_1, \phi_2, \psi_2)$. The L^2-error of this four-dimensional free energy surface was computed as well. We studied the internal convergence by taking the four-dimensional free energy surface with the same thermostat as the reference after a 300 ns UFED simulation (Figure 9). It is clear that using the three thermostats, the L^2 error converges to less than 0.5 kcal mol^{-1} within 50 ns. The projected free energy surfaces after 300 ns are provided in Figure 10. The results indicate that both GGMT and R-NHL thermostats are better for d-AFED/TAMD/UFED/TASS simulations.

As a further benchmark, we computed the exploration efficiency ϵ defined to be [121]

$$\epsilon = \frac{\text{number of visited bins}}{\text{total number of bins}} \times 100\% \ . \tag{30}$$

Here we took the four-dimensional free energy surface represented in 26×26×26×26 grid points for the computation of ϵ. In Figure 9(b), we plot ϵ as a function of simulation length for R-NHL and GGMT thermostats from UFED simulations. It can be clearly seen that both thermostats perform very well, and nearly 70% of the four-dimensional free energy space was explored in 50 ns of the simulation. Nearly 100% exploration was achieved in ∼100 ns of the simulation. We also find that the efficiency of exploration in d-AFED/TAMD and UFED simulations is equally good.

2.2. Conformational Landscape of the Sarcosine Tetrapeptoid in Implicit Solvent

We now demonstrate an application of the new `UFEDMM` module to the generation of a high-dimensional conformational landscape of the tetrasarcosine peptoid in implicit solvent. Peptoids are a class of peptidomimetic oligomers composed

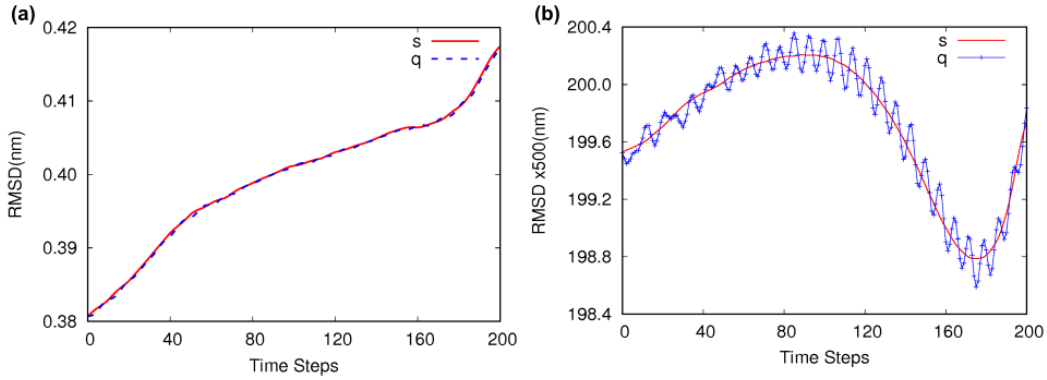

Figure 7. Plot of physical (q_α) and auxiliary variables (s_α) corresponding to the C_α-backbone RMSD CV used in d-AFED/TAMD simulation of Trp-cage. (a) The fluctuations of q_α are very small, and thus the chosen value of κ and μ are unable to ensure an ideal oscillation of q_α about s_α.(b) The fluctuations of q_α are amplified by scaling up the CV by 500, after which oscillation of q_α about s is observed. Each point in the plot of q_α corresponds to an integration timestep. It is clear from the plot that the timestep is appropriate for integrating the fast oscillations of q_α.

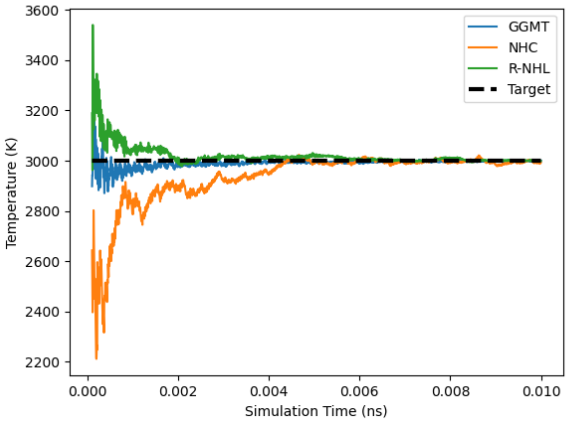

Figure 8. Running average of the temperature of the auxiliary variables during the alanine tripeptide *in vacuo* simulations using the GGMT, R-NHL, and NHC thermostats with optimal thermostat parameters. Here the dotted black line labels the target temperature of 3000 K.

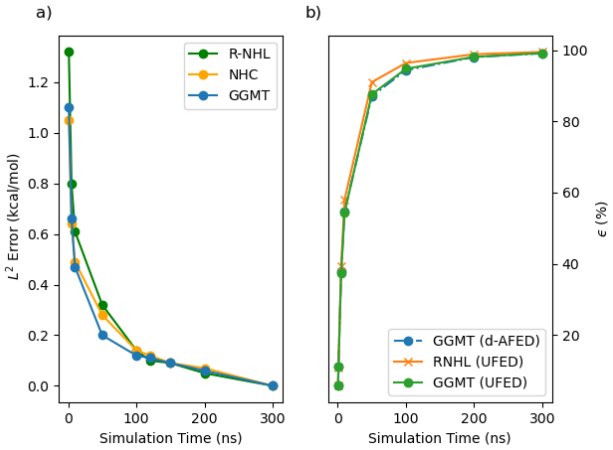

Figure 9. (a) L^2 error (kcal/mol) computed for $F(\phi_1, \psi_1, \phi_2, \psi_2)$ for alanine tripeptide *in vacuo* with respect to the free energy surface after 300 ns of UFED simulation. (b) Exploration efficiency (ϵ) over $F(\phi_1, \psi_1, \phi_2, \psi_2)$ surface for alanine tripeptide as a function of simulation time.

of N-substituted glycine units. Despite their inability to form hydrogen-bond networks, they adopt stable 3D structures not accessible by standard peptides. Peptoids exhibit notable characteristics, such as the ability to introduce diverse side-chain functionalities and resistance to hydrolytic degradation by proteases. As a result, they have become potential candidates for biomedical applications with superior biocompatibility and potent biological activities.[122–128] For this study, we employed the d-AFED/TAMD and UFED methods. Parameters for the simulation and the thermostats were chosen using protocols discussed earlier; see Section 2 for details.

The eight-dimensional free energy surface

$$F(\phi_1, \phi_2, \phi_3, \psi_1, \psi_2, \psi_3, \omega_2, \omega_3)$$

for tetrasarcosine obtained from UFED simulations was projected onto (ϕ_1, ψ_1) (ϕ_2, ψ_2) (ϕ_3, ψ_3) space; see Figure 11(a-f). The d-AFED/TAMD results are presented in SI Section 1.3. From the L^2 error plots in Figure 11(e-h), it is clear that both d-AFED/TAMD and UFED simulations have converged below 0.5 kcal mol^{-1} within 50 ns. We also find that both R-NHL and GGMT thermostats are effective in maintaining the temperature of the peptoid system in implicit solvent, in agreement with the observations we had earlier for alanine tripeptide.

In all the free energy surfaces (Figure 11(a-f)), the minimum energy conformation occurs at $\phi = \pm\pi$ and $\psi = \pm\pi/2$, indicating that the backbone of the peptide prefers a *trans* conformation. These results agree with Ref.[129]. The ϕ_2, ψ_2 dihedral angles show slightly higher energy barriers than ϕ_1, ψ_1 and ϕ_3, ψ_3. This is expected, as in these two dihedral angles, substituents attached to the central bond have a larger steric hindrance, which makes it difficult for these angles to rotate.

2.3. Conformational Landscape of Trp-cage in Explicit Water

We now demonstrate the application of the `UFEDMM` module to perform TASS simulations. We chose to investigate the folding/unfolding free energy landscape of the Trp-cage mini-protein in explicit water. This is an ideal problem for TASS, as a steered exploration of the landscape along the "(un)folding" coordinate should improve the efficiency

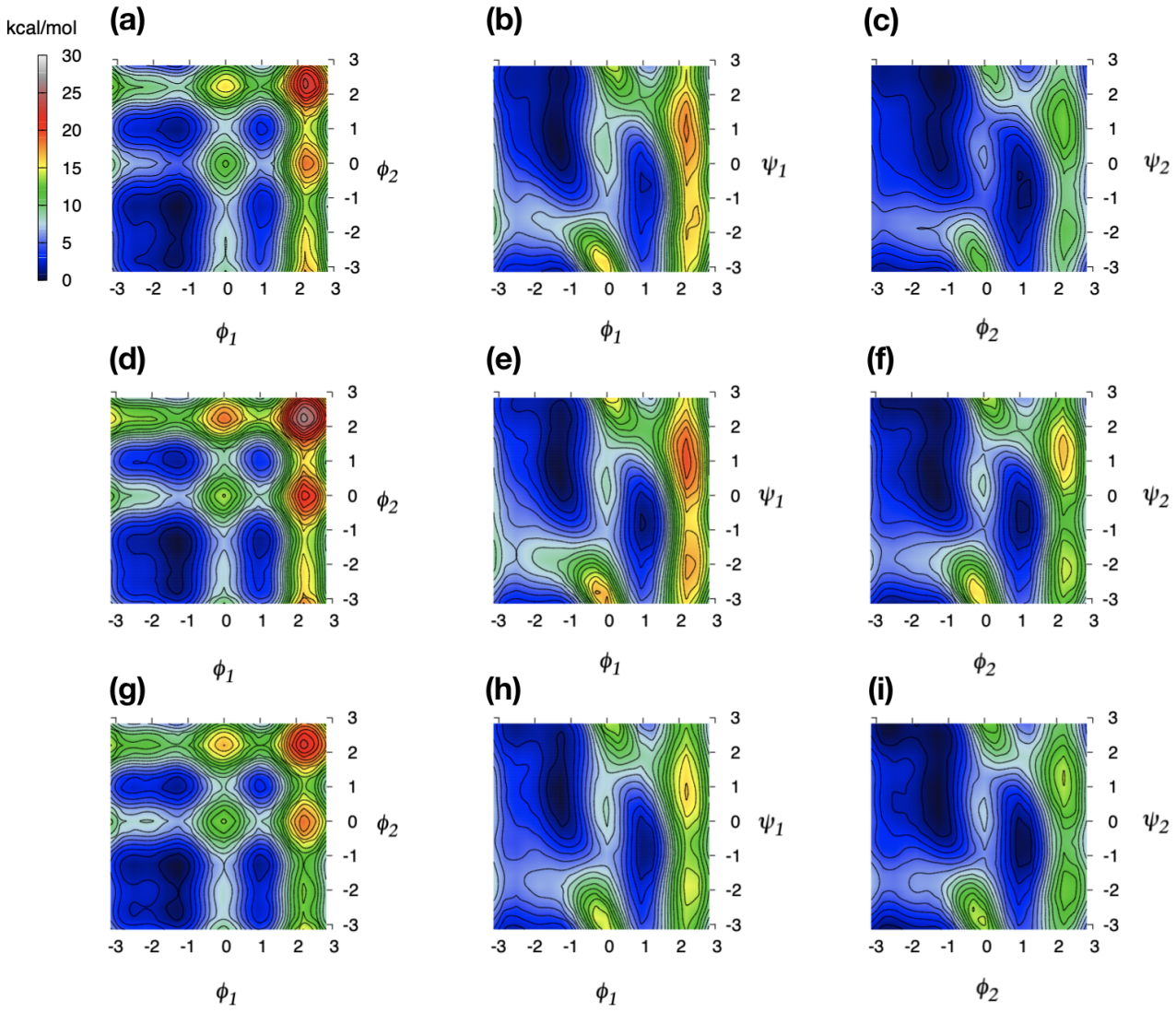

Figure 10. $F(\phi_1, \phi_2)$, $F(\phi_1, \psi_1)$, and $F(\phi_2, \psi_2)$ computed after 300 ns using GGMT (a,b,c), R-NHL (d,e,f), and NHC (g,h,i) thermostats. Free energy is in kcal/mol and angles are in radians.

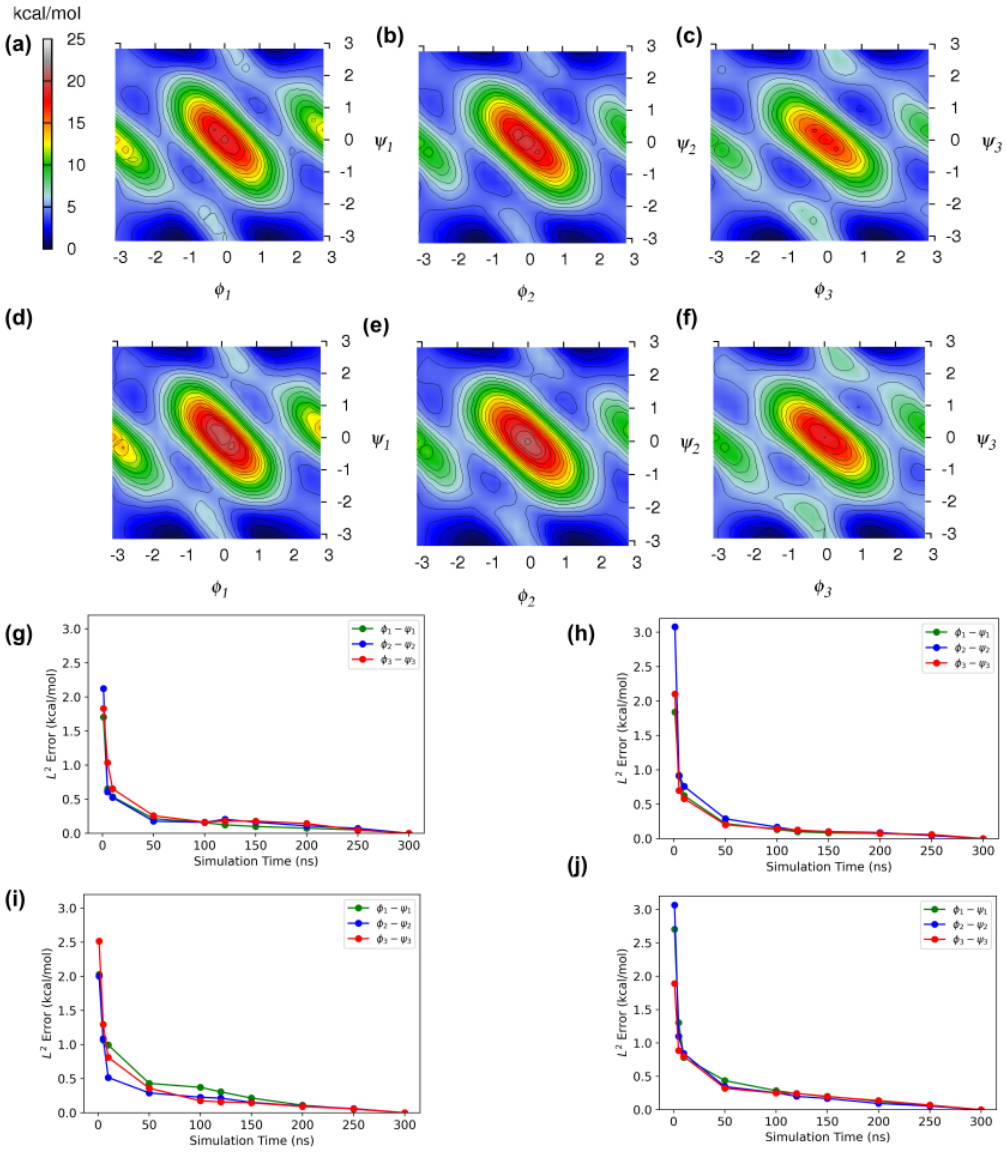

Figure 11. Projected free energy surfaces of tetrasarcosine, $F(\phi_1, \psi_1)$, $F(\phi_2, \psi_2)$, and $F(\phi_3, \psi_3)$, computed from UFED simulation with GGMT (a,b,c) and R-NHL thermostats (d,e,f). L^2 error computed for $F(\phi_1, \psi_1)$ (green), $F(\phi_2, \psi_2)$ (blue), and $F(\phi_3, \psi_3)$ (red) are shown here using d-AFED/GGMT (g), d-AFED/R-NHL (h), UFED/GGMT (i), UFED/R-NHL (j).

of sampling. The system can take a large number of conformations [116,117] in the intermediate and unfolded states, and this fact makes the exploration of the free energy landscape slow unless we employ controlled sampling. A self-guided exploration can be inefficient, and multiple folding-unfolding transitions will be rarely seen using conventional sampling techniques due to the entropy of the unfolded and intermediate states.

Several experimental [108–110,130–145] and computational studies [69,93,119,146–170] have investigated the folding of Trp-cage. The presence of intermediate metastable states has been observed computationally [152,171] and experimentally [132]. Two major folding pathways have been identified for this protein [116,117,172,173]. The free energy difference between the folded and unfolded states of the protein at 298 K is known experimentally to be 0.77 kcal mol^{-1}.[136] From the earlier works, it is also known that the free energy estimate is sensitive to the force fields used and the quality of the simulation; [117,154] for e.g., using the OPLS-AA force field, the native folded state was found to be 1.3 kcal mol^{-1} higher than the unfolded state at 300 K. [116,117] Thus the problem is quite challenging and is an ideal testbed for demonstrating the efficiency of TASS using `UFEDMM`.

We followed the protocols presented earlier to determine suitable extended system parameters and we use the R-NHL thermostat for all Trp-cage simulations. We chose the RMSD of C_α atoms as the "folding/unfolding" coordinate to drive the conformations from one end state to the other. Therefore, to achieve controlled sampling with TASS, we applied an umbrella bias along this coordinate. To further enhance the sampling of the conformational space, seven other CVs were also considered in the simulation. Among them is the radius of gyration, whose associated auxiliary variable was biased via well-tempered metadynamics. All eight CVs were biased by high temperature; see Table 1 for details.

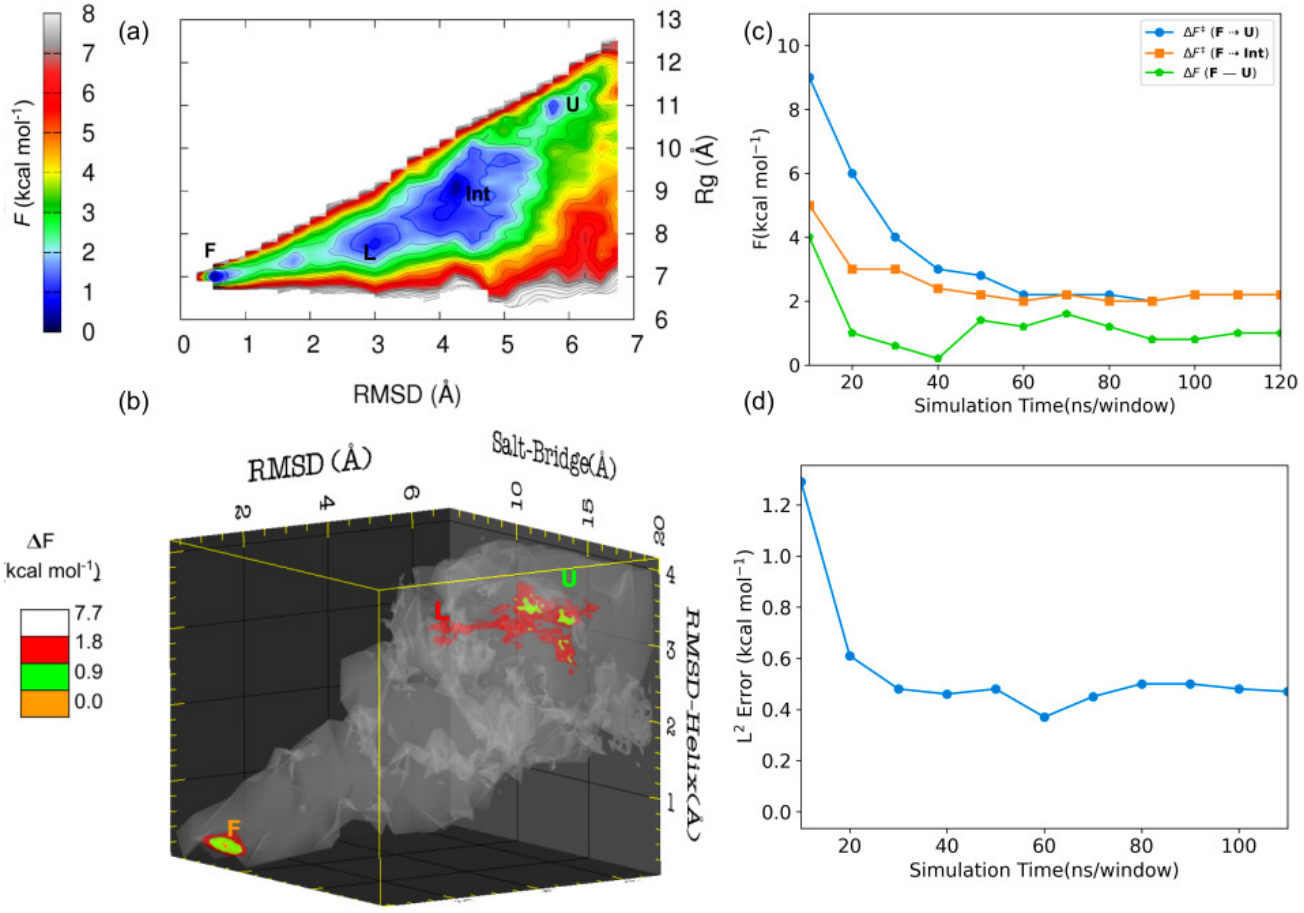

Figure 12. (a) Free energy surface for Trp-cage unfolding projected in (a) (RMSD,Rg), and (b) (RMSD,$\text{RMSD}_{\text{Helix}}$,Sb) CV spaces. (c) Convergence of free energy barriers **F**→**Int** ($\Delta F^{\ddagger}(\mathbf{F} \rightarrow \mathbf{Int})$), and **F**→**U** ($\Delta F^{\ddagger}(\mathbf{F} \rightarrow \mathbf{Int})$), as well as free energy difference between **F** and **U** states ($\Delta F(\mathbf{F} - \mathbf{U})$) are shown. (d) Internal convergence of the L^2 error computed for the F(RMSD,$\text{RMSD}_{\text{Helix}}$,Sb) three dimensional surface is shown.

We computed the eight-dimensional free energy surface after 120 ns/window, as well as its projections onto the (RMSD,Rg) and (RMSD, $\text{RMSD}_{\text{Helix}}$, Sb) spaces. These projections are shown in Figure 12(a,b). The reconstructed surfaces for a selection of intermediate lengths of the simulation are provided in SI Section 1.2. When compared with the previous reports [116,117,146,174], our simulations correctly identify the folded (**F**), unfolded (**U**) and intermediate (**Int**) states on the F(RMSD,Rg) surface. As expected, the **F** state (Figure 13(a)) appears for RMSD below 1 Å with respect to the NMR structure and is the lowest energy minimum on the free energy landscape. The broad basin of **Int** in Figure 12(a) is between RMSD of 1.5 Å and 5.5 Å, while the **U** state is at RMSD of ∼5.9 Å. The **F** state is 0.9 kcal mol^{-1} lower than the **U** state and at same level with **Int** states. The time evolution of the free energy estimates and the L^2 error (Figure 12(c,d)) indicates convergence to reasonable accuracy after 70 ns/window. The free energy barriers as obtained from our TASS simulations for the **F** → **Int** and **F** → **U** transitions are both 2.2 kcal mol^{-1} and in agreement with other reports [146,149,174]. Our estimate of the free energy difference between the folded and unfolded states (0.9 kcal mol^{-1}) is also in excellent agreement with the experimental value (0.77 kcal mol^{-1} at 298 K) [136] and with previously reported simulation studies. [146]

It has been reported that the reaction coordinate for the folding of Trp-cage depends mainly on $\text{RMSD}_{\text{Helix}}$ and on the RMSD of C_α atoms. [116,117] The presence of a salt bridge between Asp16 and Asp9 is characteristic of several conformations of the **Int** and **U** states. We observed the **L**, **I**, and **Pd** conformations, as reported earlier, [116,117] in the TASS trajectories; see Figure 13. Here the **Pd** state is characterized by the separation of Pro12 from the polyproline helix (composed of Pro17-19) as well as Trp6 (Figure 13c). The **I** state is characterized by the detachment of Trp6 from the polyproline helix. Both the **I** and the **Pd** states retain the alpha-helical character of the native protein, while the **L** state has a smaller alpha-helical content. In the **L** state, the interactions between Pro12, Pro18, and Trp6 are retained as in the native state. From snapshots extracted from the TASS trajectories, we carried out short unbiased MD simulations to further probe the stability of these states. We found that both the **Pd** and **I** states are metastable, leading to the formation of the native folded **F**, unfolded states or **Int** states during unbiased simulation. The **L** was found to be relatively more stable than **Pd** and the **I** states in the ten independent unbiased MD simulations performed. These observations support the presence of two distinct folding pathways as reported earlier. [116–119,172]

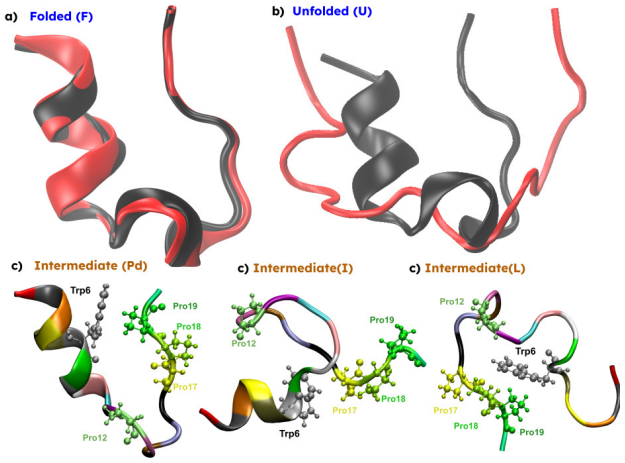

Figure 13. Conformations of Trp-cage: (a) Folded (**F**) native state as in the X-ray structure (black) and obtained from TASS (red) are overlapped; (b) Unfolded (**U**) state obtained from TASS (red) is compared with the folded X-ray structure (black); (c) A representative structure for the intermediate **Pd**, **I** and **L** states seen in the TASS trajectories. Some of the critical residues are highlighted in ball-stick format.

3. Conclusions

We have presented the `UFEDMM` library as an open-source extension of `OpenMM` facilitating extended phase-space methods for enhanced molecular dynamics-based sampling. This library makes available a selection of different extended phase-space methods including d-AFED/TAMD, UFED, and TASS. A number of state-of-the-art thermostats, multiple-time-step integration schemes, a large number of CVs for biomolecular systems, and pre-/post-processing scripts are made available.

We used the reweighted probability distribution of the chosen CVs for each example system presented in order to construct high-dimensional free energy surfaces. For TASS, a mean-force-based formalism for computing free energy surfaces was employed. The accuracy and convergence of the free energy estimates were studied by calculating L^2 error as a function of simulation time. The accuracy of the extended phase-space methods depends on the parameters $\{\kappa_\alpha\}$ and $\{\mu_\alpha\}$, and recipes for determining appropriate parameters for the chosen CVs are provided here. Our formula for determining these parameters is to determine first μ_α as $100 \times M_{\text{eff}}$, where M_{eff} is given by Eq. (28) for each CV, which in turn can be determined by the `effective_mass` program distributed with `UFEDMM`. The parameter κ_α can then be determined by examining the dynamics of auxiliary variables and the CVs, as shown in Figure 5. For some CV types, scaling their values is required in order to achieve proper adiabatically decoupled motion of the auxiliary degrees of freedom. The R-NHL and GGMT thermostats are good choices for thermostatting the extended and physical degrees of freedom. Thermostatting all the degrees of freedom is vital, and thus bond constraints cannot be used in these simulations. The performance of the thermostat can be verified by monitoring the running average of the temperature, as in Figure 8.

We presented the d-AFED/TAMD and UFED studies in computing the four-dimensional conformational free energy landscape of alanine tripeptide *in vacuo* and an eight-dimensional free-energy surface of the tetrasarcosine in implicit solvent. Using the TASS method, we also explored the eight-dimensional free energy landscape of solvated Trp-cage. The results of these simulations agree with the previously reported data.

We hope this work will facilitate researchers to perform extended system-based exploration and computation of the high-dimensional free energy landscape of physicochemical processes.

Supporting Information Available

Supporting Information has details about the CVs used for Trp-cage TASS simulations, free energy landscapes of Trp-cage computed for different simulation lengths, and d-AFED/TAMD results for alanine tripeptide and trisarcosine.

Acknowledgments

SB thanks IIT Kanpur for his Ph.D. scholarship. SB and NNN thank ParamSanganak (IIT Kanpur) for computational resources. M.E.T. acknowledges support from the National Science Foundation, grant no. CHE-1955381. M.E.T. and M.T. also acknowledge support from the U.S. Department of Energy, grant no. DE-SC0020971 M0003. NNN and MET gratefully acknowledge the support of the Indo-US Science and Technology Forum for funding the RARE symposia, which promoted this collaborative work.

Data Availability Statement

The data published in this study and the reweighting program are available from the corresponding author upon relevant request. The `UFEDMM` git repository link https://github.com/craabreu/ufedmm.

References

[1] C. Dellago, P. G. Bolhuis, F. S. Csajka, D. Chandler, *J. Chem. Phys.* **1998**, *108*, 1964.
[2] M. E. Tuckerman, *Statistical Mechanics: Theory and Molecular Simulation*, Oxford University Press, Oxford, 1st edition **2010**.
[3] D. Frenkel, B. Smit, *Understanding Molecular Simulation: From Algorithms to Applications*, Academic Press **2002**.
[4] B. Peters, *Reaction Rate Theory and Rare Events*, Elsevier, Amsterdam, Netherlands **2017**.
[5] E. Vanden-Eijnden, *J. Comput. Chem.* **2009**, *30*, 1737.
[6] C. D. Christ, A. E. Mark, W. F. van Gunsteren, *J. Comput. Chem.* **2010**, *31*, 1569.
[7] S. Bonella, S. Meloni, G. Ciccotti, *Eur. Phys. J. B* **2012**, *85*, 97.
[8] O. Valsson, P. Tiwary, M. Parrinello, *Annu. Rev. Phys. Chem.* **2016**, *67*, 159.
[9] S. Awasthi, N. N. Nair, *Wiley Interdiscip. Rev. Comput. Mol. Sci.* **2019**, *9*, e1398.
[10] S. Paul, N. N. Nair, V. Harish, *Mol. Sim.* **2019**, *45*, 1273.
[11] M. A. Rohrdanz, W. Zheng, C. Clementi, *Annu. Rev. Phys. Chem.* **2013**, *64*, 295, pMID: 23298245.

[12] B. Peters, *Annu. Rev. Phys. Chem.* **2016**, *67*, 669, pMID: 27090846.
[13] F. Pietrucci, *Rev. Phys.* **2017**, *2*, 32.
[14] G. Bussi, A. Laio, *Nat Rev Phys* **2020**, *2*, 200.
[15] E. Schneider, L. Dai, R. Q. Topper, C. Drechsel-Grau, M. E. Tuckerman, *Phys. Rev. Lett.* **2017**, *119*, 150601.
[16] D. Wang, Y. Wang, J. Chang, L. Zhang, H. Wang, W. E., *Nature Computational Science* **2021**, *2*, 20.
[17] G. M. Torrie, J. P. Valleau, *Chem. Phys. Lett.* **1974**, *28*, 578.
[18] G. Torrie, J. Valleau, *J. Comput. Phys.* **1977**, *23*, 187.
[19] A. Laio, M. Parrinello, *Proc. Natl. Acad. Sci.* **2002**, *99*, 12562.
[20] R. Martoňák, A. Laio, M. Parrinello, *Phys. Rev. Lett.* **2003**, *90*, 075503.
[21] O. Valsson, M. Parrinello, *Phys. Rev. Lett.* **2014**, *113*, 090601.
[22] O. Valsson, M. Parrinello, *J. Chem. Theory Comput.* **2015**, *11*, 1996.
[23] P. Shaffer, O. Valsson, M. Parrinello, *Proc. Natl. Acad. Sci.* **2016**, *113*, 1150.
[24] M. Nava, R. Quhe, F. Palazzesi, P. Tiwary, M. Parrinello, *J. Chem. Theory Comput.* **2015**, *11*, 5114.
[25] R. Quhe, M. Nava, P. Tiwary, M. Parrinello, *J. Chem. Theory Comput.* **2015**, *11*, 1383.
[26] V. Limongelli, M. Bonomi, M. Parrinello, *Proc. Natl. Acad. Sci.* **2013**, *110*, 6358.
[27] J. Debnath, M. Parrinello, *J. Phys. Chem. Lett.* **2020**, *11*, 5076, pMID: 32510225.
[28] J. F. Dama, M. Parrinello, G. A. Voth, *Phys. Rev. Lett.* **2014**, *112*, 240602.
[29] A. Barducci, G. Bussi, M. Parrinello, *Phys. Rev. Lett.* **2008**, *100*, 020603.
[30] G. Bussi, A. Laio, M. Parrinello, *Phys. Rev. Lett.* **2006**, *96*, 090601.
[31] D. Branduardi, G. Bussi, M. Parrinello, *J. Chem. Theory Comput.* **2012**, *8*, 2247.
[32] G. Bussi, F. L. Gervasio, A. Laio, M. Parrinello, *J. Am. Chem. Soc.* **2006**, *128*, 13435.
[33] P. Raiteri, A. Laio, F. L. Gervasio, C. Micheletti, M. Parrinello, *J. Phys. Chem. B* **2006**, *110*, 3533.
[34] J. F. Dama, G. Rotskoff, M. Parrinello, G. A. Voth, *J. Chem. Theory Comput.* **2014**, *10*, 3626.
[35] E. Darve, A. Pohorille, *J. Chem. Phys.* **2001**, *115*, 9169.
[36] C. B. Barnett, K. J. Naidoo, *Mol. Phys.* **2009**, *107*, 1243.
[37] D. Mendels, J. J. de Pablo, *J. Phys. Chem. Lett.* **2022**, *13*, 2830.
[38] B. Pampel, O. Valsson, *J. Chem. Theory Comput.* **2022**, *18*, 4127.
[39] D. Wang, Y. Wang, J. Chang, L. Zhang, H. Wang, W. E., *Nat. Comput. Sci.* **2022**, *2*, 20.
[40] M. Invernizzi, P. M. Piaggi, M. Parrinello, *Physical Review X* **2020**, *10*, 041034.
[41] L. Rosso, P. Mináry, Z. Zhu, M. E. Tuckerman, *J. Chem. Phys.* **2002**, *116*, 4389.
[42] L. Maragliano, E. Vanden-Eijnden, *Chem. Phys. Lett.* **2006**, *426*, 168.
[43] J. B. Abrams, M. E. Tuckerman, *J. Phys. Chem. B* **2008**, *112*, 15742.
[44] A. F. Voter, *Phys. Rev. Lett.* **1997**, *78*, 3908.
[45] R. Affentranger, I. Tavernelli, E. E. Di Iorio, *J. Chem. Theory Comput.* **2006**, *2*, 217.
[46] P. Liu, B. Kim, R. A. Friesner, B. J. Berne, *Proc. Nat. Acad. Sci.* **2005**, *102*, 13749.
[47] Y. Q. Gao, *J. Chem. Phys.* **2008**, *128*, 064105.
[48] H. Grubmüller, *Phys. Rev. E* **1995**, *52*, 2893.
[49] D. Hamelberg, J. Mongan, J. A. McCammon, *J. Chem. Phys.* **2004**, *120*, 11919.
[50] A. Mitsutake, Y. Sugita, Y. Okamoto, *Peptide Science* **2001**, *60*, 96.
[51] S. G. Itoh, H. Okumura, *J. Chem. Theory Comput.* **2012**, *9*, 570.
[52] T. Morishita, S. G. Itoh, H. Okumura, M. Mikami, *Phys. Rev. E* **2012**, *85*, 066702.
[53] T. Hayami, J. Higo, H. Nakamura, K. Kasahara, *J. Comput. Chem.* **2019**, *40*, 2453.
[54] A. M. Ferrenberg, R. H. Swendsen, *Phys. Rev. Lett.* **1989**, *63*, 1195.
[55] S. Kumar, J. M. Rosenberg, D. Bouzida, R. H. Swendsen, P. A. Kollman, *J. Comput. Chem.* **1992**, *13*, 1011.
[56] J. Kästner, W. Thiel, *J. Chem. Phys.* **2005**, *123*, 144104.
[57] J. Kästner, *WIREs Comput. Mol. Sci.* **2011**, *1*, 932.
[58] C. Bartels, M. Karplus, *J. Comput. Chem.* **1997**, *18*, 1450.
[59] R. Hooft, B. Vaneijck, J. Kroon, *J. Chem. Phys.* **1992**, *97*, 6690.
[60] M. Mezei, *J. Comput. Phys.* **1987**, *68*, 237.
[61] W. Wojtas-Niziurski, Y. Meng, B. Roux, S. Bernéche, *J. Chem. Phys.* **2013**, *9*, 1885.
[62] S. Awasthi, V. Kapil, N. N. Nair, *J. Comput. Chem.* **2016**, *37*, 1413.
[63] S. Awasthi, N. N. Nair, *J. Chem. Phys.* **2017**, *146*, 094108.
[64] M. Iannuzzi, A. Laio, M. Parrinello, *Phys. Rev. Lett.* **2003**, *90*, 238302.
[65] M. Bonomi, A. Barducci, M. Parrinello, *J. Comput. Chem.* **2009**, *30*, 1615.
[66] A. Barducci, M. Bonomi, M. Parrinello, *WIREs Comput. Mol. Sci.* **2011**, *1*, 826.
[67] G. Bussi, D. Branduardi, *Free-Energy Calculations with Metadynamics: Theory and Practice*, chapter 1, pages 1–49, Wiley-Blackwell **2015**.
[68] L. Sutto, S. Marsili, F. L. Gervasio, *Wiley Interdiscip Rev Comput Mol Sci* **2012**, *2*, 771.
[69] S. Piana, A. Laio, *J. Phys. Chem. B* **2007**, *111*, 4553.
[70] N. Todorova, F. Marinelli, S. Piana, I. Yarovsky, *J. Phys. Chem. B* **2009**, *113*, 3556.
[71] J. Pfaendtner, M. Bonomi, *J. Chem. Theory Comput.* **2015**, *11*, 5062.
[72] S. Alamdari, J. Sampath, A. Prakash, L. D. Gibson, J. Pfaendtner, Efficient Sampling of High-Dimensional Free Energy Landscapes: A Review of Parallel Bias Metadynamics, in E. J. Maginn, J. Errington (Editors), *Foundations of Molecular Modeling and Simulation*, Springer Singapore, Singapore **2021** pages 123–141.
[73] A. Gil-Ley, G. Bussi, *J. Chem. Theory Comput.* **2015**, *11*, 1077.
[74] C. Camilloni, D. Provasi, G. Tiana, R. A. Broglia, *Proteins: Struct. Funct. Genet.* **2008**, *71*, 1647.
[75] J. M. L. Ribeiro, P. Bravo, Y. Wang, P. Tiwary, *J. Chem. Phys.* **2018**, *149*, 072301.
[76] M. Invernizzi, M. Parrinello, *J. Phys. Chem. Lett.* **2020**, *11*, 2731, pMID: 32191470.
[77] L. Rosso, J. B. Abrams, M. E. Tuckerman, *J. Phys.*

Chem. B **2005**, *109*, 4162.
[78] M. Chen, M. A. Cuendet, M. E. Tuckerman, *J. Chem. Phys.* **2012**, *137*, 024102.
[79] D. Wiczew, N. Szulc, M. Tarek, *Bioelectrochemistry* **2021**, *141*, 107869.
[80] X. Ding, X. Lin, B. Zhang, *Nat. Commun.* **2021**, *12*, 1091.
[81] J.-M. Escoffre, P. Campomanes, M. Tarek, A. Bouakaz, *Ultrason Sonochem* **2020**, *64*, 104998.
[82] M. Chen, T.-Q. Yu, M. E. Tuckerman, *Proc. Natl. Acad. Sci.* **2015**, *112*, 3235.
[83] A. Samanta, M. Chen, T.-Q. Yu, M. Tuckerman, W. E, *J. Chem. Phys.* **2014**, *140*, 164109.
[84] J. R. Cendagorta, J. Tolpin, E. Schneider, R. Q. Topper, M. E. Tuckerman, *J. Phys. Chem. B* **2020**, *124*, 3647.
[85] S. Awasthi, S. Gupta, R. Tripathi, N. N. Nair, *J. Phys. Chem. B* **2018**, *122*, 4299.
[86] P. Eastman, J. Swails, J. D. Chodera, R. T. McGibbon, Y. Zhao, K. A. Beauchamp, L.-P. Wang, A. C. Simmonett, M. P. Harrigan, C. D. Stern, R. P. Wiewiora, B. R. Brooks, V. S. Pande, *PLoS Comput. Biol.* **2017**, *13*, e1005659.
[87] *OpenMM, version 7.7.0* **2022**, https://github.com/craabreu/ufedmm (accessed on Nov 16, 2022).
[88] C. Abreu, UFEDMM, version 1.0 **2022**, https://github.com/craabreu/ufedmm (accessed on Nov 16, 2022).
[89] P. Tiwary, M. Parrinello, *J. Phys. Chem. B* **2014**, *119*, 736.
[90] M. A. Cuendet, M. E. Tuckerman, *J. Chem. Theory Comput.* **2014**, *10*, 2975.
[91] A. Pal, S. Pal, S. Verma, M. Shiga, N. N. Nair, *J. Comput. Chem.* **2021**, *42*, 1996.
[92] J. B. Abrams, L. Rosso, M. E. Tuckerman, *J. Chem. Phys.* **2006**, *125*, 074115.
[93] X. Wu, G. Yang, Y. Zu, Y. Fu, L. Zhou, X. Yuan, *Mol. Simul.* **2012**, *38*, 161.
[94] M. Bonomi, D. Branduardi, G. Bussi, C. Camilloni, D. Provasi, P. Raiteri, D. Donadio, F. Marinelli, F. Pietrucci, R. A. Broaglia, M. Parrinello, *Comput. Phys. Commun.* **2009**, *180*, 1961.
[95] R. Car, M. Parrinello, *Physical Review Letters* **1985**, *55*, 2471.
[96] M. E. Tuckerman, M. Parrinello, *The Journal of Chemical Physics* **1994**, *101*, 1302.
[97] Z. Zhang, X. Liu, Z. Chen, H. Zheng, K. Yan, J. Liu, *The Journal of Chemical Physics* **2017**, *147*, 034109.
[98] Z. Zhang, X. Liu, K. Yan, M. E. Tuckerman, J. Liu, *The Journal of Physical Chemistry A* **2019**, *123*, 6056.
[99] M. Tuckerman, B. J. Berne, G. J. Martyna, *The Journal of Chemical Physics* **1992**, *97*, 1990.
[100] Y. Liu, M. E. Tuckerman, *J. Chem. Phys.* **2000**, *112*, 1685.
[101] G. J. Martyna, M. L. Klein, M. Tuckerman, *J. Chem. Phys.* **1992**, *97*, 2635.
[102] C. R. A. Abreu, M. E. Tuckerman, *Mol. Phys.* **2021**, *119*, e1923848.
[103] B. Leimkuhler, D. T. Margul, M. E. Tuckerman, *Molecular Physics* **2013**, *111*, 3579.
[104] J. Wang, R. M. Wolf, J. W. Caldwell, P. A. Kollman, D. A. Case, *J. Comput. Chem.* **2004**, *25*, 1157.
[105] J. Wang, R. M. Wolf, J. W. Caldwell, P. A. Kollman, D. A. Case, *J. Comput. Chem.* **2005**, *26*, 114.
[106] X. He, V. H. Man, W. Yang, T.-S. Lee, J. Wang, *J. Chem. Phys.* **2020**, *153*, 114502.
[107] H. Nguyen, D. R. Roe, C. Simmerling, *J. Chem. Theory Comput.* **2013**, *9*, 2020, pMID: 25788871.
[108] J. W. Neidigh, R. M. Fesinmeyer, N. H. Andersen, *Nat. Struct. Mol. Biol.* **2002**, *9*, 425.
[109] L. Qiu, S. A. Pabit, A. E. Roitberg, S. J. Hagen, *J. Am. Chem. Soc.* **2002**, *124*, 12952.
[110] B. Barua, J. C. Lin, V. D. Williams, P. Kummler, J. W. Neidigh, N. H. Andersen, *Protein Eng. Des. Sel.* **2008**, *21*, 171.
[111] A. D. MacKerell, D. Bashford, M. Bellott, R. L. Dunbrack, J. D. Evanseck, M. J. Field, S. Fischer, J. Gao, H. Guo, S. Ha, D. Joseph-McCarthy, L. Kuchnir, K. Kuczera, F. T. K. Lau, C. Mattos, S. Michnick, T. Ngo, D. T. Nguyen, B. Prodhom, W. E. Reiher, B. Roux, M. Schlenkrich, J. C. Smith, R. Stote, J. Straub, M. Watanabe, J. Wiórkiewicz-Kuczera, D. Yin, M. Karplus, *J. Phys. Chem. B* **1998**, *102*, 3586, pMID: 24889800.
[112] V. Hornak, R. Abel, A. Okur, B. Strockbine, A. Roitberg, C. Simmerling, *Proteins: Struct. Funct. Genet.* **2006**, *65*, 712.
[113] B. Leimkuhler, E. Noorizadeh, F. Theil, *J. Stat. Phys.* **2009**, *135*, 261.
[114] B. Leimkuhler, C. Matthews, *J. Chem. Phys.* **2013**, *138*, 174102.
[115] J. Aqvist, P. Wennerström, M. Nervall, S. Bjelic, B. O. Brandsdal, *Chem. Phys. Lett.* **2004**, *384*, 288.
[116] J. Juraszek, P. G. Bolhuis, *Proc. Natl. Acad. Sci.* **2006**, *103*, 15859.
[117] J. Juraszek, P. G. Bolhuis, *Biophys. J.* **2008**, *95*, 4246.
[118] J. Strahan, A. Antoszewski, C. Lorpaiboon, B. P. Vani, J. Weare, A. R. Dinner, *J. Chem. Theory Comput.* **2021**, *17*, 2948.
[119] W. Chen, A. L. Ferguson, *J. Comput. Chem.* **2018**, *39*, 2079.
[120] *UFEDMM-Docs, Version 1.0* **2022**, https://ufedmm.readthedocs.io/en/latest/pythonapi/ufedmm.html (accessed on Jan 13, 2022).
[121] A. Gupta, S. Verma, R. Javed, S. Sudhakar, S. Srivastava, N. N. Nair, *J. Comput. Chem.* **2022**, *43*, 1186.
[122] J. Sun, Z. Li, Peptoid applications in biomedicine and nanotechnology, in *Peptide Applications in Biomedicine, Biotechnology and Bioengineering*, pages 183–213, Woodhead Publishing, Buckingham, England, UK **2018**.
[123] K. Moehle, H. J. Hofmann, *Biopolymers* **1996**, *38*, 781.
[124] V. A. Voelz, K. A. Dill, I. Chorny, *Biopolymers* **2011**, *96*, 639.
[125] G. L. Butterfoss, P. D. Renfrew, B. Kuhlman, K. Kirshenbaum, R. Bonneau, *J. Am. Chem. Soc.* **2009**, *131*, 16798.
[126] A. Prakash, M. D. Baer, C. J. Mundy, J. Pfaendtner, *Biomacromolecules* **2018**, *19*, 1006.
[127] J. L. Kessler, G. Kang, Z. Qin, H. Kang, F. G. Whitby, T. E. Cheatham, C. P. Hill, Y. Li, S. M. Yu, *J. Am. Chem. Soc.* **2021**, *143*, 10910.
[128] J. Sun, R. N. Zuckermann, *ACS Nano* **2013**, *7*, 4715.
[129] R. K. Spencer, G. L. Butterfoss, J. R. Edison, J. R. Eastwood, S. Whitelam, K. Kirshenbaum, R. N. Zuckermann, *Biopolymers* **2019**, *110*, e23266.
[130] H. Meuzelaar, K. A. Marino, A. Huerta-Viga, M. R.

Panman, L. E. J. Smeenk, A. J. Kettelarij, J. H. van Maarseveen, P. Timmerman, P. G. Bolhuis, S. Woutersen, *J. Phys. Chem. B* . **2013**, *117*, 11490.
[131] Z. Ahmed, I. A. Beta, A. V. Mikhonin, S. A. Asher, *J. Am. Chem. Soc.* **2005**, *127*, 10943.
[132] H. Neuweiler, S. Doose, M. Sauer, *Proc. Natl. Acad. Sci.* **2005**, *102*, 16650.
[133] A. T. Iavarone, J. H. Parks, *J. Am. Chem. Soc.* **2005**, *127*, 8606.
[134] M. R. Bunagan, X. Yang, J. G. Saven, F. Gai, *J. Phys. Chem. B* . **2006**, *110*, 3759.
[135] A. T. Iavarone, A. Patriksson, D. van der Spoel, J. H. Parks, *J. Am. Chem. Soc.* **2007**, *129*, 6726.
[136] W. W. Streicher, G. I. Makhatadze, *Biochemistry* **2007**, *46*, 2876.
[137] K. H. Mok, L. T. Kuhn, M. Goez, I. J. Day, J. C. Lin, N. H. Andersen, P. J. Hore, *Nature* **2007**, *447*, 106.
[138] P. Hudáky, P. Stráner, V. Farkas, G. Váradi, G. Tóth, A. Perczel, *Biochemistry* **2008**, *47*, 1007.
[139] R. M. Culik, A. L. Serrano, M. R. Bunagan, F. Gai, *Angew. Chem.* **2011**, *50*, 10884.
[140] P. Rovó, V. Farkas, O. Hegyi, O. Szolomájer-Csikós, G. K. Tóth, A. Perczel, *J. Pept. Sci.* **2011**, *17*, 610.
[141] P. Rogne, P. Ozdowy, C. Richter, K. Saxena, H. Schwalbe, L. T. Kuhn, *PLOS ONE* **2012**, *7*, e41301.
[142] A. Hałabis, W. Żmudzińska, A. Liwo, S. Ołdziej, *J. Phys. Chem. B* **2012**, *116*, 6898.
[143] P. Rovó, P. Stráner, A. Láng, I. Bartha, K. Huszár, L. Nyitray, A. Perczel, *Chem. Eur. J.* **2013**, *19*, 2628.
[144] C. M. Adams, F. Kjeldsen, A. Patriksson, D. van der Spoel, A. Gräslund, E. Papadopoulos, R. A. Zubarev, *Int. J. Mass Spectrom.* **2006**, *253*, 263.
[145] F. Chalyavi, A. J. Schmitz, M. J. Tucker, *J. Phys. Chem. Lett.* **2020**, *11*, 832.
[146] A. B. Kapakayala, N. N. Nair, *J. Comput. Chem.* **2021**, *42*, 2233.
[147] S. Shityakov, E. V. Skorb, M. Nosonovsky, *R. Soc. open sci.* **2022**, *9*, 220160.
[148] C. D. Snow, B. Zagrovic, V. S. Pande, *J. Am. Chem. Soc.* **2002**, *124*, 14548.
[149] R. Appadurai, J. Nagesh, A. Srivastava, *Nat. Commun.* **2021**, *12*, 958.
[150] C. Simmerling, B. Strockbine, A. E. Roitberg, *J. Am. Chem. Soc.* **2002**, *124*, 11258.
[151] S. Chowdhury, M. C. Lee, G. Xiong, Y. Duan, *J. Mol. Biol.* **2003**, *327*, 711.
[152] R. Zhou, *Proc. Natl. Acad. Sci.* **2003**, *100*, 13280.
[153] Z. Hu, Y. Tang, H. Wang, X. Zhang, M. Lei, *Arch. Biochem. Biophys.* **2008**, *475*, 140.
[154] Q. Shao, J. Shi, W. Zhu, *J. Chem. Phys.* **2012**, *137*, 125103.
[155] L. Bò, E. Milanetti, C. G. Chen, G. Ruocco, A. Amadei, M. D'Abramo, *ACS Omega* **2022**, *7*, 13448.
[156] P. Wu, X. Hu, W. Yang, *The Journal of Physical Chemistry Letters* **2011**, *2*, 2099.
[157] K. A. Marino, P. G. Bolhuis, *J. Phys. Chem. B* **2012**, *116*, 11872.
[158] Z. Lai, N. K. Preketes, S. Mukamel, J. Wang, *J. Phys. Chem. B* **2013**, *117*, 4661.
[159] W. Xu, Y. Mu, *Biophys. Chem.* **2008**, *137*, 116.
[160] M. Gupta, D. Nayar, C. Chakravarty, S. Bandyopadhyay, *Phys. Chem. Chem. Phys.* **2016**, *18*, 32796.
[161] S. B. Kim, D. R. Gupta, P. G. Debenedetti, *Sci. Rep.* **2016**, *6*, 25612.
[162] H. W. Hatch, F. H. Stillinger, P. G. Debenedetti, *J. Phys. Chem. B* **2014**, *118*, 7761.
[163] S. Kannan, M. Zacharias, *PLOS ONE* **2014**, *9*, e88383.
[164] A. Schug, W. Wenzel, U. H. E. Hansmann, *J. Chem. Phys.* **2005**, *122*, 194711.
[165] H. Sidky, W. Chen, A. L. Ferguson, *J. Phys. Chem. B* **2019**, *123*, 7999.
[166] A. Patriksson, C. M. Adams, F. Kjeldsen, R. A. Zubarev, D. van der Spoel, *J. Phys. Chem. B* **2007**, *111*, 13147.
[167] J. W. Pitera, W. Swope, *Proc. Natl. Acad. Sci.* **2003**, *100*, 7587.
[168] D. Paschek, R. Day, A. E. García, *Phys. Chem. Chem. Phys.* **2011**, *13*, 19840.
[169] A. Byrne, D. V. Williams, B. Barua, S. J. Hagen, B. L. Kier, N. H. Andersen, *Biochemistry* **2014**, *53*, 6011.
[170] A. Bille, B. Linse, S. Mohanty, A. Irbäck, *J. Chem. Phys.* **2015**, *143*, 175102.
[171] K. Lindorff-Larsen, S. Piana, R. O. Dror, D. E. Shaw, *Science* **2011**, *334*, 517.
[172] S. B. Kim, C. J. Dsilva, I. G. Kevrekidis, P. G. Debenedetti, *J. Chem. Phys.* **2015**, *142*, 085101.
[173] F. Marinelli, F. Pietrucci, A. Laio, S. Piana, *PLoS Comput. Biol.* **2009**, *5*, e1000452.
[174] Y. Miao, F. Feixas, C. Eun, J. A. McCammon, *J. Comput. Chem.* **2015**, *36*, 1536.

Keywords: Temperature Accelerated Molecular Dynamics • Unified Free Energy Dynamics • driven-Adiabatic Free Energy Dynamics • Temperature Accelerated Sliced Sampling • Umbrella Sampling