Supplementary Information

PySAGES: flexible, advanced sampling methods accelerated with GPUs

Pablo F. Zubieta Rico[a], Ludwig Schneider[a], Gustavo R. Pérez-Lemus[a], Riccardo Alessandri[a], Siva Dasetty[a], Trung D. Nguyen[b], Cintia A. Menéndez[a], Yiheng Wu[a], Yezhi Jin[a], Yinan Xu[a], Samuel Varner[a,d], John A. Parker[b], Andrew L. Ferguson[a], Jonathan K. Whitmer[c], Juan J. de Pablo[*,a]

[a] *Pritzker School of Molecular Engineering, The University of Chicago, 5640 South Ellis Avenue, Chicago, IL, 60637, USA*
[b] *Research Computing Center, The University of Chicago, 6054 S. Drexel Avenue, Chicago, IL, 60637, USA*
[c] *Department of Chemical and Biomolecular Engineering, University of Notre Dame, 250 Nieuwland Hall, Notre Dame, IN, 46556, USA*
[d] *Division of Chemistry and Chemical Engineering, California Institute of Technology, 1200 East California Boulevard, Pasadena, 91125, CA, USA*

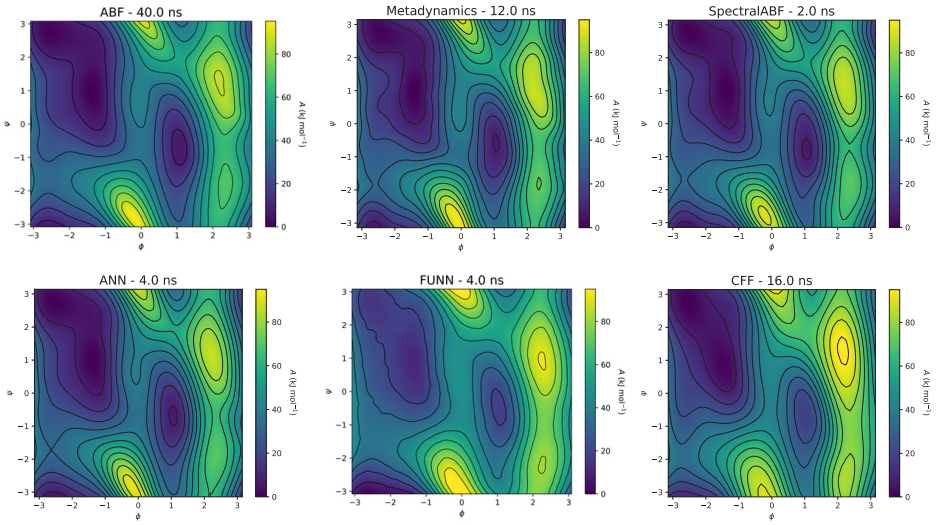
Supplementary Discussion 1

Benchmark test systems

In the following sections, we present the results of the free energy calculation for the benchmark test systems of alanine dipeptide and butane. The details of all the parameters chosen to perform the enhanced sampling simulation of these are summarized in Supplementary Table 1.

Alanine Dipeptide

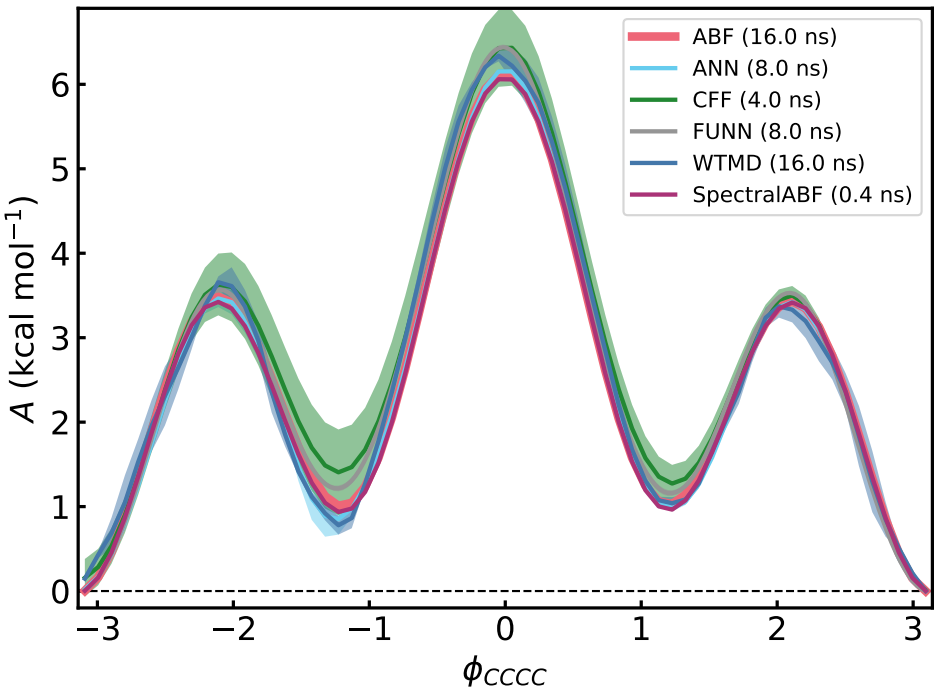
The first test system involves alanine dipeptide in vacuum (Supplementary Figure 1), a benchmark system for enhanced sampling methods that is frequently used in the literature.



Supplementary Figure 1: Free energy landscape of alanine dipeptide (Amber ff99SB [1]) in vacuum as a function of the dihedral angles ϕ and ψ obtained with PySAGES and OpenMM via different enhanced sampling methods: ABF, Metadynamics, Spectral ABF, ANN, FUNN, CFF. Each panel also indicates the length of the simulation necessary for the free energy to converge. The long ABF simulations represent the ground truth.

Butane

As a second test system, we compute the free energy profile along the C-C-C-C dihedral angle, ϕ_{CCCC}, of a butane molecule (in vacuum), Figure Supplementary Figure 2.



Supplementary Figure 2: Free energy profile along the dihedral angle of a butane molecule (using an OPLS [2]-based force field) obtained via different enhanced sampling methods with PySAGES and HOOMD-blue: adaptive biasing force, artificial neural networks sampling, combined force frequency sampling, force-biasing using neural networks, well-tempered metadynamics, and Spectral adaptive biasing force. The legend also indicates the length of the simulation. Each free energy profile is the mean of three independent runs and the standard deviation is reported as shaded area (note that in some cases it is too small to be noticed). The long adaptive biasing force simulations represent the ground truth.

Example System Details

Supplementary Table 1: Parameters and methods details for the various examples. For all methods but Metadynamics, we used a grid with 50 points along each collective variable (CV) for alanine dipeptide (ADP) and with 64 points along the collective variable (CV) for butane. The parameter N for ABF is the threshold before accounting for the full average of the adaptive biasing force.

System	Backend	collective variable (CV)	Method	Settings	Fig.
ADP	OpenMM	ϕ and ψ	ABF	$N = 500$ (default)	Supplementary Figure 1
			ANN	topology = $(8, 8)$	
			CFF	topology = $(14,)$	
			FUNN	topology = $(14,)$ $\sigma = 0.35$ rad	
			Metadynamics	$h = 1.2$ kJ/mol stride = 500	
			Spectral ABF	—	
Butane	HOOMD-blue	ϕ_{CCCC}	ANN	topology = $(8, 8)$	Supplementary Figure 2
			CFF	topology = $(8,)$	
			FUNN	topology = $(8,)$ $\sigma = 0.10$ rad	
			WTMD	$h = 0.01$ kJ/mol stride = 50 $\Delta T = 5000$	
			Spectral ABF	—	

Supplementary Discussion 2

Collective variable for the distance to an interface

Implementation of the CV described in Section "Example applications of enhanced sampling with PySAGES," that is, the distance between a group of atoms to an interface defined by another group of atoms.

```
class DistanceToInterface(TwoPointCV):
    def __init__(self, indices, axis, sigma, scope, bins=100, coeff=1):
        super().__init__(indices)
        self.axis = axis
        self.sigma = sigma
        self.scope = scope
        self.bins = bins
        self.coeff = coeff

    @property
    def function(self):
        return lambda r1, r2: distance_to_interface(
            r1, r2, axis=self.axis,
            sigma=self.sigma, scope=self.scope,
            bins=self.bins, coeff=self.coeff
        )

def distance_to_interface(p1, p2, axis, sigma, scope, bins, coeff):
    mobile_axis = barycenter(p1)[axis]
    positions_axis = p2.flatten()[axis::3]
    centers = np.linspace(scope[0], scope[1], bins)
    centers = np.expand_dims(centers, 1)
    positions_axis = np.expand_dims(positions_axis, 0)
    diff = positions_axis - centers
    mass = np.exp(-0.5 * (diff / sigma) ** 2)
    mass = np.sum(mass, axis=1)
    mass_diff = np.abs(mass[1:] - mass[:-1])
    centers = np.squeeze(centers)
    centers_mean = (centers[1:] + centers[:-1]) / 2
    probability = nn.softmax(mass_diff * coeff)
    interface = np.sum(probability * centers_mean)
    return mobile_axis - interface
```

Supplementary References

1. Hornak, V. *et al.* Comparison of multiple amber force fields and development of improved protein backbone parameters. *Proteins: Struct., Funct., Bioinf.* **65**, 712–725 (2006).

2. Jorgensen, W. L., Maxwell, D. S. & Tirado-Rives, J. Development and testing of the OPLS all-atom force field on conformational energetics and properties of organic liquids. *J. Am. Chem. Soc.* **118**, 11225–11236 (1996).