vFEP User Manual Version 0.1

The vFEP algorithm and the vFEP program has been solely developed by **Taisung Lee**, with the financial supports by various projects from **Prof. Darrin York** of Rutgers University. Test cases were performed by **Brian K. Radak** and **Anna Pabis**.

August 21, 2013

Contents

Chapter 1

Current Status

As of August 21, 2013, the vFEP program is a fully functional prototype. It should be able to deliver the expected results. However, the program has not been extensively tested and users should use it with cautions and carefully examine the results.

The documentation is far from complete and is still under construction.

The following reference should be cited if vFEP is utilized:

Lee, T. S., Radak, B..K.; Pabis A.; York, D. M.,
A New Maximum Likelihood Approach for Free Energy Profile Construction from Molecular Simulations
Journal of Chemical Theory and Computation, 2013, 9:153-164
`http://pubs.acs.org/doi/abs/10.1021/ct300703z`

Chapter 2

Install and run the vFEP program

2.1 Installation: Linux

You need to have

- A Linux box (tested on Fedora 16 and 18.)
- A C/C++ compiler installed.
- The cmake utility installed (v2.6 or later, `http://www.cmake.org/`).
  Fedora: *sudo yum install cmake*
  Ubuntu: *sudo apt-get install cmake*
- The boost library installed (v1.47.0 or late, `http://www.boost.org/`).
  Fedora: *sudo yum install boost-devel*
  Ubuntu: *sudo apt-get install boost-devel*
- The wget and unzip utilities for direct downloading AlgLib and dLib libraries.

Steps:

- Download the vFEP package from `http://sites.google.com/site/cancersimulation/software`.
- Untar/unzip the tar file by "*tar -xzvf vFEP.tar.gz*".
- Make sure your boost include files are included in the searching path defined by the environmental variable "INCLUDE", if not, modify the variable "Boost_INCLUDE_DIR" in vFEP/cmake/CMakeLists.txt.
- Run the script "Install". You are done. The resulting binary will be in bin/ .
- The Install script will try to download the AlgLib and the dLib libraries. If the downloading fails, you can manually download them and copy your own AlgLib source files into 3rdParty/alglib/src and dLib source files into 3rdParty/dLib. Rerun the Install script afterward.

* The user manual is in vFEP/doc/Manual.pdf

The Linux build has been tested with gcc version 4.7.2, cmake 2.6, and boost 1.47 and 1.50.

2.2 Installation: Windows

You need to have

* A PC with Windows 7 64 bit OS.

Steps:

* Download the vFEP executable (vFEP.exe) from `http://sites.google.com/site/cancersimu` and you are done.

The Windows build has been compiled and tested with Microsoft Visual C++ 2008 on a PC with Windows 7 Professional 64 bit. It likely will work on PCs with Windows XP (SP2 64 bit) and Windows 8 (64bit).

Please send emails to cancersimulation@gmail.com if you have problems and/or comments.

2.3 Run the vFEP program

Steps:

* Prepare a metafile containing the list of your data files with the following format for each line:

  filename ep1 fc1 [ep2 fc2]

  filename: data filename for each umbrella sampling simulation
  ep1: center position of the first dimension
  fc1: force constant of the first dimension
  ep2: optional, center position of the second dimension
  fc2: optional, force constant of the second dimension

  When ep2 and fc2 *both* are present, the vFEP program will run in the 2D mode otherwise in the 1D mode.

Note that the format is the same as the one used by WHAM for 1D cases BUT different for 2D cases.

* Run the program by

```
vFEP -m ”metafile name” -fep ”output free energy profile”
```

* The output free energy profile has the following format for each line:

```
x [y] energy
```

x: the coordinate of the first dimension
y: the coordinate of the second dimension (for 2D cases only)
energy: the relative free energy in kcal/mole at this point

Other options can be found by *”vFEP -h”* and described in Chapter 3: Program Options.

Chapter 3

Program options

Available options

option	arg.	default	description
-h			help

File options

option	arg.	default	description
-m	string	(required)	the metafile for umbrella sampling (also turn on umbrella sampling type calculation)
-b	string		the metafile for BEDAM (also turn on BEDAM type calculation)
-fep	string		Output the free energy profile to file.
-log	string		Output the program progress to file.
-Z	string		Output the relative free energy shifts between windows.

Data options

option	arg.	default	description
-s	integer		dataFrequency: must be an integer =0,1,-1: use all data points (default). >1: only pickup data points every dataFrequence, e.g., dataFrequency=10 will cause that only the 1st, 11th, 21th , 31th... data points will be used. <-1: only 1/(-dataFrequency) portion of data points will be used (randomly pickup), e.g., if dataFrequence=-10, 1/10 of total data points will be randomely picked up and used.
-x0,-y0	*double*		Specify the reference point of x (and y for 2D) where the output function value will be adjusted to zero.
-s1	*double*	1.0	Specify the scalar factor for the coordinate for the 1st dimension
-s2	*double*	1.0	Specify the scalar factor for the coordinate for the 2st dimension
-jr1		(off)	Add the polar coordinate Jocabian correction to the first dimension.
-jr1		(off)	Add the polar coordinate Jocabian correction to the second dimension.
-sym		(off)	Create the missing symmetrical windows in a 2D simulation.
-nsplit	integer	1	Split each window into nsplit (per dimension) child widows. Only applicable to windows with zero biasing potentials.
-no_pb		(on)	Turn off the automatic addition of periodic boundary window pairs.

Numerical options

option	arg.	default	description
-grid	*integer*	200	Number of grid points (of each dimension) of the free energy profile.

-t	*string*	MLE	the method to be used for umbrella sampling data:
			method=MLE: use maximum likelihoond method, i.e., vFEP. (default, 1D and 2D).
			method=Derivative, UI: use umbrella integration. (1D only)
			method=WHAM, Wham: use WHAM. (1D only)

-i	*string*	cubic	the interpolation type used in vFEP interpolation=cubic, spline, c, s: use cubic spline (default, 1D and 2D). interpolation=rational, r: use rational interpolation. (1D only)
-n1	*integer*		Number of spline nodes used in the first dimension (Default: automatically determined by the program.)
-n2	*integer*		Number of spline nodes used in the second dimension (Default: automatically determined by the program.)
-nq	*integer*	12 or 48	Number of quadrature points used for integration (default: 48 for Gauss-Hermite quadrature and 12 for Gauss-Legendre quadrature).
-qgl		off	Use Gauss-Legendre quadrature rules (defauls: use Gauss-Hermite quadrature rules with non-zero biasing potential, otherwise Gauss-Legendre.).

Path Analysis (2D only)

option	arg.	default	description
-path	*string*		filename : Output the path points to file.
-st	*string*		filename : Output the stationary points to file.
-pd1	*double*	1.0 / (number grids)	Resolutions for creating the paths and the stationary points for the first dimension. Default: 1.0/(number grids).
-pd2	*double*	1.0 / (number grids)	Resolutions for creating the paths and the stationary points for the second dimension. Default: 1.0/(number grids).
-max		(off)	In the path analysis, maximum points are included(i.e., the points with 2nd derivatives¡0). Default: only minimum points.

Other conditions

option	arg.	default	description
-x	*double*	input data	The minimal value of the range of output free energy profile for the first dimension.
-X	*double*	input data	The maximal value of the range of output free energy profile for the first dimension.
-y	*double*	input data	The minimal value of the range of output free energy profile for the second dimension.
-Y	*double*	input data	The maximal value of the range of output free energy profile for the second dimension.
			-x, -X, -y -Y: Default values: The range(s) will be the input data range(s).
-dx	*double*	input data	The minimal value of the range of data to be used in analysis (the first dimension.)
-dX	*double*	input data	The maximal value of the range of data to be used in analysis (the first dimension.)
-dy	*double*	input data	The minimal value of the range of data to be used in analysis (the second dimension.)
-dY	*double*	input data	The maximal value of the range of data to be used in analysis (the second dimension.)
			-dx, -dX, -dy -dY: Default values: The range(s) will be the input data range(s), i.e., all data will be used in analysis.
-r1			turn on periodical boundary conditions for the first dimension: requires setting min and max x by -x xmin and -X xmax
-r2			turn on periodical boundary conditions for the second dimension: requires setting min and max y by -y xmin and -Y xmax
-r			equivalent to " -r1 -r2 ", i.e. turn on periodical boundary conditions for both dimensions.
-n	*integer*		must be a positive integer: Maximum number of iterations in optimization. (default: no limit)
-tol	*double*	1e-6	The tolerance (of likelihood in unit of RT) to stop optimization.

Chapter 4

Introduction

The vFEP method is intended to be used for the following purposes (for 1- and 2-D cases):

* To obtain the corresponding free energy profile from a set of umbrella sampling simulations;

* To obtain the underlining free energy profile of set of observed data, maybe a set of plain molecular dynamics (MD) or Monte Carlo (MC) simulations, or just simply some data distribution.

The detailed list of its functionality, along with their usage, can be found in the chapter ”Program Options”.

Free energy is a key concept in modern physical sciences and offers a wealth of insights into complex molecular problems [1]. One dimensional free energy profiles are routinely employed to study various molecular systems with respect to a certain variable/coordinate and many enhanced sampling methods for accurately accomplishing this task have been developed in the past decades[2].

One of the most widely used methods for determining free energy surfaces for chemical reactions, where often there are geometric coordinates that are known to be aligned with the overall reaction coordinate, is the “umbrella sampling” technique [3, 4]. Combining stratification with equilibrium and statically biased sampling, umbrella sampling is particularly amenable to parallel execution, especially in high performance distributed environments[5, 6], as well as extension or combination with replica exchange[7, 8] and alchemical simulation techniques[9].

Although there are numerous well-developed and widely used methods to construct 1D free energy profiles from umbrella sampling simulations, such as the weighted histogram analysis method [10, 11], the Umbrella integration method (UI)[12], the multistate Bennett acceptance ratio method (MBAR) [13, 14] and others [15, 16, 17, 18], publicly available methods/programs for 2D cases are still very limited. Both WHAM and MBAR algorithms have been extended and implemented to 2D cases, and are publicly available at `http://membrane.urmc.rochester.edu/content/wham` and `http://simtk.org/home/pymbar`, respectively. [19, 14]. 2D-UI [20] and GAMUS[17, 21] implementations and applications have also been reported but they are not publicly available. Nevertheless, due to the two key difficulties in umbrella sampling methods, the problem of "data re-weighting" and of "data representation", the cost of such calculations can still be quite prohibitive, especially in two or higher dimensions. Here we briefly review these two major problems just described:

The need of overlap in data re-weighting: Traditional methods to construct free energy profiles, such as WHAM[10] and MBAR[13, 14], rely on the knowledge of overlap between umbrella windows to properly re-weight data for each window, although this can be practically and exactly solved when there is only one sample set (*i.e.*, one umbrella sampling window) by the free energy perturbation (FEP)/Zwanzig relation and the related expression for arbitrary mechanical observables [22, 3, 23]. This type of approach inevitably requires significant overlap between windows in order to sensibly construct the global free energy profile.

An alternative approach assumes the smoothness of the free energy profile between nearby windows. The Umbrella Integration (UI) approach of Kästner and Thiel [24, 25, 20] uses a Gaussian distribution to model the un-weighted probability density for each umbrella window (or equivalently, quadratic functions for the free energy profile) from which the analytic derivatives are calculated and integrated in order to recover the global probability density. Hence no explicit re-weighting is necessary. This approach is equivalent to assuming continuous first derivatives of the free energy profile between windows. Instead of fix-positioned quadratic functions, the 1D-vFEP method [26] utilizes cubic spline functions to model the free energy profile, equivalent to assuming continuous first and second derivatives of the free energy profile between windows. It has been demonstrated[26] in 1D cases that UI and 1D-vFEP require less of a degree of overlap between windows compared to WHAM and histogram-based MBAR.

Data representation: In order to extract a distribution function from a set of data, it is often necessary to employ a certain type of representation of that function. This is commonly formulated as the density estimation problem[27, 28]. Perhaps the simplest method of data representation is to use a histogram estimator of the probability density [10, 9, 29]. However, this approach is frequently not numerically stable, especially when the data is too sparse such that the width of histogram bins cannot be made sufficiently small. A useful alternative approach can be to apply a more robust kernel density estimator, but this too will fail with extremely sparse data sets. A completely different type of approach is to fit the overall density distribution through a pre-defined model[30] by optimizing the model parameters according to a merit function. Maragakis, *et al.* suggested a maximum likelihood approach utilizing the Gaussian-Mixture Model on umbrella sampling (GAMUS) for the global probability density based on the re-weighted data [17, 31]. Similarly, Basner and Jarzynski proposed a binless estimator based upon the optimal correction to an arbitrary reference distribution[32]. UI[24, 25, 20] uses Gaussian models for the un-weighted probability densities and has also recently been extended to higher order densities (*i.e.* skewed Gaussians)[33]. It is well known that such parametric approaches lead to a significant reduction in the number of data points needed to obtain a converged result. However this often comes at the expense of increased bias depending on the inherent accuracy of the parameteric form. For example, the approximations/assumptions of UI require near-quadratic (or near quartic) behavior of the local free energy surface for individual windows and this has been demonstrated to be inaccurate in simulations with weak biasing potentials [26]. This problem may be reduced by imposing stronger harmonic biasing potentials but this often leads to lower overlap between windows and hence the same kind of failures associated with sparsely populated histogram estimators[4]. GAMUS has also been shown not to be ideal for quantitatively describing details of the free energy surface[17].

The current vFEP implementation[26] for constructing 1 and 2-dimensional free energy profiles demonstrated that the method is able to effectively address the above two difficulties, and outperforms other methods in terms of the amount and sparsity of the data needed to construct the overall free energy profiles.

Chapter 5

Theory

Here we briefly describe the maximum likelihood method utilized in the present work, beginning with a clarification of what is the difference between the terms "probability" and "likelihood" used in this context. In statistical modeling, *probability* refers to the possible outcome of data, and is usually modeled by a fixed functional form and a variable set of parameters. On the other hand, *likelihood* refers to how likely a given model can describe a set of observed outcome data. [34] Hence,

- **Probability**: $p(\{x_n\}|\{\theta_m\})$ is the probability model, defined by a fixed functional form and variable set of parameters $\{\theta_m\}$, that returns the probability of observing the data set $\{x_n\}$; i.e., for a given set of model parameters $\{\theta_m\}$, $p(\{x_n\}|\{\theta_m\})$ predicts the outcome for the set of data $\{x_n\}$: $\{\theta_m\} \rightarrow \{x_n\}$.

- **Likelihood**: $\mathcal{L}(\{\theta_m\}|\{x_n\})$ is the likelihood that the observed data set $\{x_n\}$ was generated by the probability distribution model defined by the set of parameters $\{\theta_m\}$; i.e., $\mathcal{L}(\{\theta_m\}|\{x_n\})$, for a given set of observed data $\{x_n\}$, provides an assessment of the goodness of the model parameters: $\{x_n\} \rightarrow \{\theta_m\}$.

The maximum likelihood method, or maximum likelihood estimation (MLE), [35, 36] is the procedure of finding the optimal set of parameters that maximize the likelihood of the model probability distribution function to represent a given set of observed data.

MLE begins with the definition of the likelihood function of the sample data. The likelihood function of a set of data is the probability of obtaining that particular set of data, given the probability distribution model function defined by a chosen functional form along with a set of trial model parameters. Here we consider the probability, $p(x)$, of observing a molecular system at a particular value of a single generalized coordinate x (the extension to multiple dimensions is straight forward). This probability is given by

$$p(x) = \frac{e^{-F(x)}}{\int e^{-F(x')}\mathrm{d}x'} \tag{5.1}$$

where $F(x) \equiv \mathrm{F}(x)/(k_\mathrm{B}T)$ is the unitless scaled free energy profile, $\mathrm{F}(x)$ is the free energy profile, k_B is the Boltzmann constant and T is the absolute temperature. Consider now a parametric model for the scaled free energy profile $F(x|\{\theta_m\})$ where $\{\theta_m\}$ is the set of parameters. The probability distribution model, $p(x|\{\theta_m\})$, also contains the set of parameters, due to its relation to $F(x|\{\theta_m\})$. Now considering the probability, $p(\{x_n\}|\{\theta_m\})$ of a sampled data set $\{x_n\}$, if the sampling data points are independent to each other, then:

$$p(\{x_n\}|\{\theta_m\}) = p(x_1, x_2, \ldots, x_N|\{\theta_m\}) = p(x_1|\{\theta_m\}) \cdot p(x_2|\{\theta_m\}) \cdots p(x_n|\{\theta_m\}). \tag{5.2}$$

The likelihood $\mathcal{L}$ of the trial free energy profile $F\{\theta_m\}$ with the given observed data set $\{x_n\}$ is:

$$\mathcal{L}(F\{\theta_m\}|\, x_1, \ldots, x_N) = \mathcal{L}(\{\theta_m\}|\, x_1, \ldots, x_N) = \prod_{i=1}^{N} p(x_i|\{\theta_m\}). \tag{5.3}$$

In the present work, instead of dealing with individual windows, we attempt to find the optimal solution of the above equation by defining a global function $F(x)$ with a set of defined parameters $\{\theta_m\}$. It is practical to use the logarithm of the likelihood function, called the log-likelihood $\hat{\ell}$:

$$\hat{\ell}(\{\theta_m\}|\, x_1, \ldots, x_N) = \frac{1}{N}\ln\mathcal{L} = \frac{1}{N}\sum_{n=1}^{N} \ln p(x_n|\{\theta_m\}). \tag{5.4}$$

Since the likelihood is always positive and monotonic, there is no loss of generality in formulating a variational principle based on the log-likelihood, which offers some advantages in terms of numerical stability and is conventional in the literature. Hereafter, we use the term "likelihood" generically to refer to both the likelihood or the log-likelihood, and will reference specific equations when the mathematical distinction is necessary. The MLE method estimates $\{\theta_m\}$ by finding the values of $\{\theta_m\}$ that maximize $\hat{\ell}$:

$$\hat{\ell}(\{\theta_m^*\}|\, x_1, \ldots, x_N) = \underset{\{\theta_m\}\in\Theta}{\arg\max}\, \hat{\ell}(\{\theta_m\}|\, x_1, \ldots, x_n) = \underset{\{\theta_m\}\in\Theta}{\arg\max} \frac{1}{N}\sum_{n=1}^{N} \ln p(x_n|\{\theta_m\}) \tag{5.5}$$

where Θ defines the space that $\{\theta_m\}$ can span. If a biasing potential $W^\alpha(x)$ is applied in the αth window in a set of umbrella sampling simulations, the probability of finding the system with a certain coordinate value x is:

$$p^\alpha(x) = \frac{1}{Z^\alpha}\exp\{-[F(x){+}W^\alpha(x)]\}, \text{where } Z^\alpha = \int_{-\infty}^{\infty} \exp\{-[F(x){+}W^\alpha(x)]\}dx \tag{5.6}$$

Suppose that for the simulation of the αth window, there are N^α points observed with coordinate values $\{x_i^\alpha\}$. Since they are observed points, the probability of each point is equal with value $1/N^\alpha$. The likelihood of the whole system with an overall free energy profile $F(x)$ can be expressed as the combination of the likelihood of individual windows obtained from eq. (5.4) and eq. (5.6) as:

$$\begin{aligned}\hat{\ell}(F) &\equiv \sum_{\alpha}^{\text{windows}} c^\alpha \hat{\ell}^\alpha(\{\theta_m\}|\{x_n^\alpha\}) \\ &= -\sum_{\alpha}^{\text{windows}} c^\alpha \left\{ \ln Z^\alpha + \frac{1}{N^\alpha} \sum_{i}^{\text{datapoints}} [F(x_i^\alpha) + W^\alpha(x_i^\alpha)] \right\}\end{aligned} \tag{5.7}$$

where $\{c^\alpha\}$ are the combination weights defining the relative contribution of likelihood from different windows when combining the local likelihood into a global likelihood. When assuming all windows contribute equally, the c^α can simply be set to be equal, *i.e.*, $c^\alpha = 1$. It can also be shown that, in the exact sampling limit, the global optimal F is also the optimal F for each individual window, *i.e.*, the choice of $\{c^\alpha\}$ does not affect the resulting optimal $F(x)$ (see Supporting Information). In practice, for finite sampling, we observe that the overall result is largely insensitive to the choice of c^α, and for the present work, we choose $c^\alpha = 1$ for all windows (also see Supporting Information). In the above equation for the global likelihood function, we have used F as the argument to emphasize that optimization of the likelihood function is with respect to the free energy profile F (by varying the $\{\theta_m\}$ parameters).

There remains the task of finding the F that maximizes $\hat{\ell}(F)$. Note that in the above equation the term $W^\alpha(x_i^\alpha)$ is constant and does not need to be evaluated if the goal is to maximize the likelihood. Also, the term $-\ln Z^\alpha$ is equivalent to the relative free energies (or free energy shifts) between windows in other re-weighting schemes. In the present VFEP approach, the "re-weighting" procedure is implicitly accomplished through the normalization against the global trial function F.

An alternate strategy is to model $F(x)$ locally in the region of each window, $F^\alpha(x)$, and construct the global $F(x)$ using the $F^\alpha(x)$ with the observed data density as weighting. The only variable parameters in this approach are the relative free energy shifts between every window $\{f^\alpha\}$ (the reference free energy being arbitrary) that establish the relative weights for each window. Thus, the global $F(x)$ is defined by the parameter set $\{f^\alpha\}$ and a set of fixed local free energy profiles $F^\alpha(x)$. Applying the MLE procedure to $F(x)$ with respect to the parameter $\{f^\alpha\}$ leads to the WHAM and the MBAR equations [16, 13, 37, 14]. Note that within such a context, MBAR is also a parametric procedure where the relative free energy shifts of windows are the MLE parameters and local free energy profiles are pre-defined in data fitting procedures, whereas the proposed VFEP uses MLE parameters to construct the detailed overall free energy profiles. In summary, the WHAM and MBAR formula are equivalent to the MLE results when the global free energy profile is constructed from the local free energy profiles and the relative free energies are used as the parameters to optimize the likelihood.

In the present work, instead of dealing with individual windows, we attempt to find the optimal solution of eq. (5.7) by defining a global function $F(x)$ with a set of defined parameters $\{\theta_m\}$ (*i.e.* $F(x) \equiv F(x|\{\theta_m\})$). The procedure is as follows:

1. Choose a trial function $F(x)$ with a initial parameter set $\{\theta_m\}$.
2. Evaluate the likelihood $\hat{\ell}(F)$ of the trial function $F(x)$ according to eq. (5.7).
3. Vary the parameter set $\{\theta_m\}$ until the maximum of $\hat{\ell}(F)$ is reached.

4. The trial $F(x)$ with the maximal $\hat{\ell}(F)$ is the desired overall free energy profile.

For the 1D implementation, two types of analytic functions were selected to model the overall free energy profile: a cubic spline function[38] and a rational interpolation function[39]. Both were originally designed for interpolation usage. Nevertheless, one could treat the interpolation input data as the variable parameters; for example, a cubic spline function needs to have the $\{x_i, y_i\}$ data nodes defined in order to build the desired cubic spline interpolation, where x_i is the independent variable and y_i is the corresponding observed function value. In this work, we select fixed x_i and treat y_i as the MLE parameters to be optimized, *e.g.* a cubic spline function defined by $\{x_i, y_i\}$ will be the trial free energy function in eq. (5.7) and the optimal free energy profile is reached through changing $\{y_i\}$. This is equivalent to assuming that the free energy profile varies slower than a cubic polynomial between windows or that the first and second derivatives of free energy profile are continuous between windows.

For the 2D implementation, we utilize 2D cubic spline functions to model F and search the optimal spine coefficients to maximize the likelihood function $\hat{\ell}(F)$. Using 2D cubic splines is equivalent to assuming that the free energy profile varies slower than a cubic polynomial between windows or that the first and second derivatives of the free energy profile are continuous between windows for both of the coordinates.

Chapter 6

Where/How to use vFEP

Bibliography

[1] Free Energy Calculations. Springer Series in Chemical Physics. Springer, Berlin, 2007.

[2] Daniel M. Zuckerman. Equilibrium sampling in biomolecular simulations . *Annu. Rev. Biophys*, 40:41–62, 2011.

[3] G. M. Torrie and J. P. Valleau. Nonphysical sampling distributions in Monte Carlo free-energy estimation: Umbrella sampling. *J. Comput. Phys.*, 23:187–199, 1977.

[4] Johannes Kästner. Umbrella sampling. *WIREs Comput. Mol. Sci.*, 1(6):932–942, 2011.

[5] Andre Luckow, Lukas Lacinksi, and Shantenu Jha. *The 10th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*, chapter SAGA BigJob: An Extensible and Interoperable Pilot-Job Abstraction for Distributed Applications and Systems, pages 135–144. ACM, 2010.

[6] Andre Luckow, Mark Santcroos, Ole Weidner, Andre Merzky, Sharath Maddineni, and Shantenu Jha. *Proceedings of the 21st International Symposium on High-Performance Parallel and Distributed Computing,HPDC'12*, chapter Towards a common model for pilot-jobs. ACM, 2012.

[7] Wei Jiang and Benoit Roux. Free Energy Perturbation Hamiltonian Replica-Exchange Molecular Dynamics (FEP/H-REMD) for Absolute Ligand Binding Free Energy Calculations. *J. Chem. Theory Comput.*, 6(9):2559–2565, 2010.

[8] Emilio Gallicchio and Ronald M. Levy. Advances in all atom sampling

methods for modeling protein-ligand binding affinities. *Curr. Opin. Struct. Biol.*, 21:161–166, 2011.

[9] Marc Souaille and Benoît Roux. Extension to the weighted histogram analysis method: combining umbrella sampling with free energy calculations. *Comput. Phys. Commun.*, 135:40–57, 2001.

[10] S. Kumar, D. Bouzida, R.H. Swendsen, P.A. Kollman, and J.M. Rosenberg. The weighted histogram analysis method for free-energy calculations on biomolecules 1: The method . *J. Comput. Chem.*, 13:1011–1021, 1992.

[11] Benoît Roux. The calculation of the potential of mean force using computer simulations. *Comput. Phys. Commun.*, 91:275–282, 1995.

[12] Johannes Kstner and Walter Thiel. Bridging the gap between thermodynamic integration and umbrella sampling provides a novel analysis method: "umbrella integration". *J Chem Phys*, 123(14):144104, Oct 2005.

[13] Michael R. Shirts, Eric Bair, Giles Hooker, and Vijay S. Pande. Equilibrium free energies from nonequilibrium measurements using maximum-likelihood methods. *Phys. Rev. Lett.*, 91(14):140601, 2003.

[14] Michael R. Shirts and John D. Chodera. Statistically optimal analysis of samples from multiple equilibrium states. *J. Chem. Phys.*, 129:124105, 2008.

[15] Michiel Sprik and Giovanni Ciccotti. Free energy from constrained molecular dynamics. *J. Chem. Phys.*, 109:7737–7744, 1998.

[16] Christian Bartels. Analyzing biased Monte Carlo and molecular dynamics simulations. *Chem. Phys. Lett.*, 331(5-6):446–454, 2000.

[17] Paul Maragakis, Arjan van der Vaart, and Martin Karplus. Gaussian-mixture umbrella sampling. *J. Phys. Chem. B*, 113(14):4664–4673, Apr 2009.

[18] Niels Hansen, Jo?ica Dolenc, Matthias Knecht, Sereina Riniker, and Wilfred F. van Gunsteren. Assessment of enveloping distribution sampling to calculate relative free enthalpies of binding for eight

netropsin-DNA duplex complexes in aqueous solution. *J. Comput. Chem.*, 33(6):640–651, 2012.

[19] Alan Grossfield. WHAM: the weighted histogram analysis method, version 2.0.4, 6 2005.

[20] Johannes Kstner. Umbrella integration in two or more reaction coordinates. *J Chem Phys*, 131(3):034109, Jul 2009.

[21] Justin Spirit, Jennifer K. Binder, Marcia Levitus, and Arjan van der Vaart. Cy3-DNA stacking interactions strongly depend on the identity of the terminal basepair. *Biophys. J.*, 100:1049–1057, 2011.

[22] Robert W. Zwanzig. High-temperature equation of state by a perturbation method. I. Nonpolar gases. *J. Chem. Phys.*, 22:1420–1426, 1954.

[23] M. P. Allen and D. J. Tildesley. *Computer Simulation of Liquids.* Oxford Science Publications, New York, 1987.

[24] Johannes Kästner and Walter Thiel. Bridging the gap between thermodynamic integration and umbrella sampling provides a novel analysis method: "Umbrella integration". *J. Chem. Phys.*, 123:144104, 2005.

[25] Johannes Kästner and Walter Thiel. Analysis of the statistical error in umbrella sampling simulations by umbrella integration. *J. Chem. Phys.*, 124:234106, 2006.

[26] Tai-Sung Lee, Brian Radak, Anna Pabis, and Darrin M. York. A new maximum likelihood approach for free energy profile construction from molecular simulations. *J. Chem. Theory Comput.*, 9:153–164, 2013.

[27] B. W. Silverman. On the Estimation of a Probability Density Function by the Maximum Penalized Likelihood Method. *Ann. Stat.*, 10(3):795–810, 1982.

[28] Simon J. Sheather. Density Estimation. *Statist. Sci.*, 19(4):588–597, 2004.

[29] John D. Chodera, William C. Swope, Jed W. Pitera, Chaok Seok, and Ken A. Dill. Use of the weighted histogram analysis method for the analysis of simulated and parallel tempering simulations. *J. Chem. Theory Comput.*, 3(1):26–41, 11/09/ 2007.

[30] Dhruva K Chakravorty, Malika Kumarasiri, Alexander V Soudackov, and Sharon Hammes-Schiffer. Implementation of umbrella integration within the framework of the empirical valence bond approach. *J Chem Theory Comput*, 4(11):1974–1980, 2008.

[31] Justin Spiriti, Hiqmet Kamberaj, and Arjan Van Der Vaart. Development and application of enhanced sampling techniques to simulate the long-time scale dynamics of biomolecular systems. *Int. J. Quantum Chem.*, 112(1):33–43, 2012.

[32] Jodi E. Basner and Christopher Jarzynski. Binless estimation of the potential of mean force. *J. Phys. Chem. B*, 112(40):12722–12729, 2008.

[33] Johannes Kästner. Umbrella integration with higher-order correction terms. *J. Chem. Phys.*, 136:234102, 2012.

[34] A.W.F. Edwards. *Likelihood.* Cambridge University Press, Cambridge, 1972.

[35] R. A. Fisher. On the Mathematical Foundations of Theoretical Statistics. *Phil. Trans. R. Soc. Lond. A*, 222:309–368, 1922.

[36] John Aldrich. R. A. Fisher and the making of maximum likelihood 1912–1922. *Statist. Sci.*, 12(3):162–176, 1997.

[37] Paul Maragakis, Martin Spichty, and Martin Karplus. Optimal estimates of free energies from multistate nonequilibrium work data. *Phys. Rev. Lett.*, 96(10):100602, 2006.

[38] Hiroshi Akima. A New Method of Interpolation and Smooth Curve Fitting Based on Local Procedures. *J. ACM*, 17(4):589–602, 1970.

[39] Michael S. Floater and Kai Hormann. Barycentric Rational Interpolation with no Poles and High Rates of Approximation. *Numerische Mathematik*, 107(2):315–331, 2007.