Identification/selection of E-CAM MD codes for development

E-CAM Deliverable 1.1
Deliverable Type: Report
Delivered in Month 14– November 2016

E-CAM
The European Centre of Excellence for
Software, Training and Consultancy
in Simulation and Modelling

Funded by the European Union under grant agreement 676531

Project and Deliverable Information

Project Title	E-CAM: An e-infrastructure for software, training and discussion in simulation and modelling
Project Ref.	Grant Agreement 676531
Project Website	https://www.e-cam2020.eu
EC Project Officer	Dimitrios Axiotis
Deliverable ID	D1.1
Deliverable Nature	Report
Dissemination Level	Public
Contractual Date of Delivery	Project Month 14(November 2016)
Actual Date of Delivery	30.11.2016

Document Control Information

Document	Title:	Identification/selection of E-CAM MD codes for development
	ID:	D1.1
	Version:	As of November 30, 2016
	Status:	Accepted by Executive Board
	Available at:	https://www.e-cam2020.eu/deliverables
Review	Review Status:	Reviewed
	Action Requested:	No action requested
Authorship	Written by:	Christoph Dellago(University of Vienna)
	Contributors:	David Swenson (University of Amsterdam), Donal MacKernan (University College Dublin), Ralf Everaers (ENS Lyon), Jony Castagna (STFC)
	Reviewed by:	Sara Bonella (CECAM & EPFL), Burkhard Dünweg (MPI for Polymer Research)
	Approved by:	Christoph Dellago (University of Vienna)

Document Keywords

Keywords:	Software development, classical MD, free energy computation, rare event sampling

November 30, 2016

***Disclaimer**:This deliverable has been prepared by the responsible Work Package of the Project in accordance with the Consortium Agreement and the Grant Agreement. It solely reflects the opinion of the parties to such agreements on a collective basis in the context of the Project and to the extent foreseen in such agreements.*

***Copyright notices**: This deliverable was co-ordinated by Christoph Dellago[1] (University of Vienna) on behalf of the E-CAM consortium with contributions from David Swenson (University of Amsterdam), Donal MacKernan (University College Dublin), Ralf Everaers (ENS Lyon), Jony Castagna (STFC) . This work is licensed under the Creative Commons Attribution 4.0 International License. To view a copy of this license, visit http://creativecommons.org/licenses/by/4.0.*

[1] Christoph.Dellago@univie.ac.at

Contents

Executive Summary

Many processes in nature and technology are characterized by rare but important events, which occur on time scales orders of magnitudes longer than basic molecular motions. Such processes, which, for instance, include chemical reactions, protein folding and first order phase transitions, are difficult to simulate with classical molecular dynamics (MD) simply because of the extreme time scales involved. The main goal of Work Package 1 (WP1) is to develop software tools capable of dealing with rare events and complex free energy properties, thus extending the time scales accessible with regular MD. In this report, we will first briefly review current algorithms for the simulation of rare events and related algorithms for the computation of free energies. We will then discuss software packages that make these methods available. Based on this information, we will then give an overview of the software modules to be developed within WP1 of E-CAM. Finally, we will describe how we will benchmark some popular molecular dynamics engines on which the modules to be developed in WP1 will be based.

1 Introduction

Classical molecular dynamics simulation (MD), invented in the 1950's after the advent of the first fast computing machines, is a computational method in which Newton's equations of motion are solved in small time steps to follow the motion of individual atoms and molecules in complex materials. Today, running on powerful computers, MD simulations are routinely used to simulate systems of millions of atoms providing crucial insights on the atomistic level of a variety of processes of interest in physics, materials science, chemistry and biology. For instance, MD simulations are extensively used to study the dynamics and interactions of proteins, understand the properties of solutions or investigate transport in and on solids. From a technological point of view, molecular dynamics simulations play an important role in many fields such as drug development, the discovery of new materials, oil extraction or energy production. Indeed, enormous amounts of data are produced every day by molecular dynamics simulations running on high performance computers around the world and one of the big challenges related to such simulations is to make sense of the data and obtain mechanistic understanding in terms of low-dimensional models that capture the crucial features of the processes under study. Another central challenge is related to the time scale problem often affecting molecular dynamics simulations. More specifically, despite the exponential increase in computing power witnessed during the last decades and the development of efficient molecular dynamics algorithms, many processes are characterized by typical time scales that are still far beyond the reach of current computational capabilities. For instance, the folding of a protein may occur on the time scale of seconds or a liquid can exist in the undercooled state almost indefinitely. However, when the folding or the freezing occurs, it does so quickly. So the event of interest is not slow but rather rare, causing long waiting times before it is observed. Since MD simulations typically need to use a time step in the femtosecond regime as dictated by the timescale of basic atomic motions, simulating such processes would require an impractical number of time steps. Addressing such time scale problems and developing scientific software able to overcome them is one of the central goals of Work Package 1 (WP1) of the ECAM-Project.

Two fundamental problems of statistical mechanics are intimately tied to the time scale problem of classical molecular dynamics simulation:

- The calculation of the populations of metastable states of an equilibrium system. Such populations can be expressed in terms of free energies and hence this problem boils down to the efficient calculation of free energies.
- The sampling of transition pathways between long-lived (meta)stable states and the calculation of reaction rate constants.

Whereas the first problem is a static one and does not necessarily require to follow the dynamics of the system, free energies are often computed using molecular dynamics. Since the understanding of rare but important events also requires the calculation of free energy barriers, which are related to rare configurations, such simulations are affected by the rare event problem. In principle, this problem can be solved by running MD simulations for a very long time. In the best of cases such simulations will be expensive, but often they are simply unfeasible on current computers. Similarly, rare transitions between long-lived states can be found by running an MD simulation until the transition of interest occurs. However, within the accessible computing time the event may never happen. In the past decades, several powerful algorithms have been developed to overcome the time scale problem both for free energy computation and for rare event sampling. Typically, these methods apply an appropriately constructed bias or constraint, which artificially increases the likelihood of the rare event in a way such that it is possible to correct for the bias and restore the true probability of the event. In contrast to straightforward molecular dynamics, for which a number of excellent software packages are available (e.g., Lammps, Charmm, Gromacs, NAMD, etc.), methods for free energy computation and, in particular, for rare event sampling have not yet been implemented, with the required efficiency and scalability, into widespread simulation packages. One objective of WP1 is to close this gap and develop well tested and robust software modules for free energy computation and rare event sampling.

In the following, we will first give an overview of the most widely used algorithms for free energy calculations and the sampling of trajectories involving rare events, followed by a brief discussion of the capabilities of currently available software packages that incorporate (some of) these algorithms. Then, we will give an outlook on the specific software to be developed in WP1 and describe the benchmarking of the codes which will serve as MD engines for the modules to be developed within WP1.The report is concluded with an outlook.

2 Algorithms for free energy calculations and rare event sampling

A detailed understanding of physical, chemical, and biological processes usually requires knowledge of the underlying free energy surface [13]. For instance, the equilibrium constant of a ligand binding to a protein receptor can be expressed in terms of free energies. Similarly, the rate at which a supercooled liquid transforms into a crystal can be understood in terms of the free energy calculated as a function of the size of the crystalline nucleus forming in the supercooled liquid. Since the advent of fast computing machines, several computer simulation algorithms have been developed for the calculation of free energies in the framework of classical statistical mechanics. In the following we will briefly review the most efficient and widely used of these algorithms.

While free energy landscapes determined from molecular simulations inform us about (meta)stable states and possible pathways for transitions between them, they only provide us with a purely static picture that does not contain any dynamical information. In principle, dynamical properties of complex many body systems in materials science and biology can be studied with straightforward molecular dynamic simulations, in which the equations of classical mechanics are integrated with small time steps. For systems characterized by disparate time scales, however, such brute force methods are extremely expensive and can be applied only in special cases. If long time scales originate from rare but fast transitions between long-lived stable states, rare event simulation methods can be used to simulate processes that are otherwise beyond the possibilities even of today's fastest computers. Below, we will hence provide a brief overview of rare such methods. The choice of algorithms we discuss is guided by the kind of problems we will address within the E-CAM project together with our industrial partners.

2.1 Free energy computation methods

Below we review several free energy methods that can be implemented based on statistical mechanical sampling carried out with classical molecular dynamics algorithms. In many cases,such a sampling requires the use of appropriate thermostatting algorithms that guarantee that the correct statistical mechanical ensemble is sampled. The efficiency of these methods can often be improved using enhanced sampling approaches such as the parallel replica algorithm [38].

2.1.1 Thermodynamic integration

In the framework of statistical mechanics, the free energy F of a system is expressed in terms of the partition function Q,

$$F = -k_{\mathrm{B}} T \ln Q$$

where k_{B} is the Boltzmann constant and T is the temperature. Up to an irrelevant factor, the partition function is the integral of the Boltzmann factor over the typically very high-dimensional configuration space,

$$Q = \int \mathrm{d}x \exp[-U(x)/k_{\mathrm{B}}T].$$

Here, x specifies the microscopic state of the system and $U(x)$ is its total interaction energy. Due to the high dimensionality of configuration space it is practically impossible to compute the partition function with regular numerical integration methods. In the thermodynamic integration algorithm (TI) [29, 26], one circumvents this problem by computing the derivative of the free energy with respect to some control parameter rather than the free energy itself. In contrast to the free energy, its derivative can be expressed as an ensemble average that can be straightforwardly computed in a molecular dynamics or Monte Carlo simulation. By carrying out several of such simulations for different values of the control parameter and integrating the free energy derivative numerically, on finally obtains the free energy difference between two equilibrium states. Since its invention by Kirkwood in 1935 [29], the method of thermodynamic integration, also known as Kirkwood's coupling parameter method, has been applied in numerous applications and currently it is one of the most widely used free energy computation methods.

2.1.2 Free energy perturbation

Also the free energy perturbation method (FEP), introduced by R. Zwanzig in 1954 [48], is based on the computation of free energy differences. In this approach, one samples configuration space for one value of the control parameter and estimates the free energy difference to a state corresponding to a different value of the control parameter through an exponential average. Free energy perturbation works well, if the probability densities of the two states of interest have a large overlap, i.e., if the corresponding potential energy surfaces are similar. If the overlap is insufficient, introduction of intermediate states may be necessary.

2.1.3 Widom particle insertion method

The ability to compute chemical potentials, or, equivalently, the Gibbs free energy, is essential for the determination of phase diagrams of materials. In the Widom insertion method [46], the chemical potential is determined from the work required to insert an a additional particle/molecule to the system or removing it from the system. While the Widom insertion methods works very efficiently for gases and moderately dense liquids, its efficiency dramatically decreases for dense liquids and solids, because the spontaneous formation of cavities large enough to accommodate the inserted particle is a rare event.

2.1.4 Umbrella sampling

As mentioned above, free energy perturbation runs into statistical problems, if the equilibrium densities of the two states, between which one wishes to compute the free energy difference, are too dissimilar. In the umbrella sampling method (US) [40], one solves this problem by introducing a bias function (this is the "umbrella" function giving the methods its name), which ensures that the populated regions of both states are sampled with comparable frequency. The bias introduced by the umbrella function must be removed to obtain the correct free energy, for instance using the weighted histogram analysis methods (WHAM) [25] or the multiple state Bennett acceptance ratio method (MBAR) [5, 37]. Typically, umbrella sampling simulations are used to compute the free energy profile as a function of a particular variable that is believed to encode important information on the behavior of the system. Umbrella sampling simulations for more than one variable are possible, but become cumbersome quickly for increasing dimensionality.

2.1.5 Metadynamics

Metadynamics [31] is a molecular dynamics based method for exploring complex potential energy surfaces. The basic idea of this method is to introduce a history dependent bias building up during the simulation. This bias drives the system away from configuration space regions that have been visited before, thus enabling the system to overcome free energy barriers. In practice, the bias is applied in form of a sum of Gaussian hills to a space spanned by some pre-selected collective variables. As a result of a metadynamics simulation, one obtains the free energy of the system as a function of the collective variables (which is reconstructed from the bias), as well as information on possible mechanisms for transitions between long-lived stable states. More recently, it was demonstrated how metadynamics can also be used to determine rate constants for such transitions [39].

2.1.6 Temperature accelerated molecular dynamics (TAMD) and single sweep method

As metadynamics, temperature accelerated molecular dynamics (TAMD) [33] is a computational approach to sample the free energy landscape in a space spanned by some collective variables selected to capture the important features of the transformation one wants to study. The basic idea of TAMD is to extend configurations space by some extra degrees of freedom coupled to the collective variables. These additional variables are evolved at an artificially high temperature that permits the system to overcome free energy barriers more easily. Provided parameters are chosen appropriately, the method yields the free energy as a function of the collective variables. Improved free energy calculations can be performed with the single sweep method [34], in which TAMD is used to rapidly sample the free energy surface and determine mean forces. The global free energy surface is then reconstructed from the mean forces by applying a variational principle.

2.1.7 Non-equilibrium work methods

As demonstrated by C. Jarzynski in 1997 [28], equilibrium free energies can be extracted from the statistics of work carried out in non-equilibrium processes. This unexpected result, valid arbitrarily far far from equilibrium, not only provided a way to analyze single molecule experiments, but also laid the foundation for new algorithms for the computation of free energies based on repeated non-equilibrium transformations. Essentially, these methods exploit the fact that the bias introduced by changing a control parameter in finite time can be expressed in terms of the work accumulated during the transformation, making it possible to un-bias distributions and retrieve equilibrium properties. In a further development, Crooks related the work distribution of a non-equilibrium process to that of the reverse process, in which the control parameter is changed according to the same protocol but in reversed order [14]. Combining the Crooks fluctuation theorem with Bennett's acceptance ratio method [36] results in an efficient way to determine free energy differences from non-equilibrium work transformations, provided the system is removed from equilibrium to a moderate extent.

2.2 Rare event sampling methods

The general goal of rare event sampling methods is to find microscopic transformation pathways between long-lived stable states and determine the rates at which such transformations occur. Knowledge of transition rates is important because it provides a point of contact with experiment but also because it allows one to test the effect of modifications of the material. In the following we will briefly review the most important rare event simulation methods used in the fields of materials science and molecular biology.

2.2.1 Reactive flux method

The rates of chemical reactions can often be understood in the framework of transition state theory (TST) [23], in which one imagines that in order to evolve from reactants to products the system has to cross a transition state, a bottleneck often coinciding with a saddle point of the potential energy surface. In transition state theory, the rate constant for the reaction is then estimated from the free energy required to bring the system to the transition state and the average velocity with which the transition state is crossed. Building on this general idea, in the reactive flux, or Bennett-Chandler, method [12] first one postulates a reaction coordinate, i.e. a function of the coordinates of all atoms that quantifies the progress of the reaction. One then computes the free energy as a function of the reaction coordinate and defines a dividing surface that separates the two states of interest and corresponds to the top of the free energy barrier. In the second step of the calculation one then initiates dynamical trajectories from initial conditions constrained to lie on the dividing surface. Combining a dynamical correction factor calculated from these trajectories with the free energy profile, one obtains the rate constants both for the forward and the backward reaction. The reactive flux method is a very efficient approach for the calculation of reaction rate constants, if the reaction coordinate is properly chosen. If this is not the case, the transmission coefficient related to the probability that a trajectory crossing the dividing surface from reactants to products will indeed relax into the product state will be close to zero, causing large statistical uncertainties in the estimate of the rate constant.

2.2.2 Transition path sampling (TPS)

The problem of the unknown reaction coordinate is avoided in transition path sampling (TPS) [17, 9], a statistical method to sample rare pathways connecting long-lived stable states. Transition path sampling is based on the definition of a probability density of trajectories that obey Markovian dynamics. These dynamical pathways are then sampled using a Monte Carlo procedure in which first a trial path is created and then accepted according to a criterion satisfying detailed balance in trajectory space. As a result of a TPS simulation, one obtains a set of pathways occurring with a frequency proportional to their probability in the transition path ensemble (TPE). The pathways can then be analyzed to yield insights into the transition mechanism. The great advance of TPS is that it does not need an a priori definition of a reaction coordinate, but rather provides the information to identify a good reaction coordinate. By exploiting an isomorphism between time correlation functions and free energies, one can also determine reaction rate constants within the TPS framework [16]. More efficient rate calculations are possible with the transition interface sampling method (TIS) [41] and the partial path transition interface sampling methid (PPTIS) [35], the latter of which can be applied if the dynamics is characterized by rapid memory loss. To date, transition path sampling has been successfully applied to study many processes ranging from chemical reactions and biomolecular isomerizations to crystallization and demixing [10, 7, 18, 15, 8].

2.2.3 Forward flux sampling (FFS)

Forward flux sampling is a rare event method for the simulation of rare events governed by stochastic dynamics [3, 2, 1]. Like the transition interface sampling method, forward flux sampling is based on the definition of a set of non-intersecting interfaces between the initial and the final state. These interfaces, usually defined as iso-surfaces of a particular order parameter, are spaced in a way such that the probability for trajectories to connect neighboring interfaces is non-negligible. Positioning the interfaces in an appropriate way is important for optimizing the statistical efficiency of the method [11]. Once the interfaces are defined, a trajectory is initiated in the initial state and crossing points of the first interfaces are recorded. From these crossing points, new stochastic trajectories re-initiated, some of which will reach the second interface providing the next set of crossing points, which, in turn, will serve as starting points of yet another set of trajectories. Repeating this procedure until the final state is reached, and keeping track of the likelihood to reach one interface from the previous one, allows allows one to compute the reaction rate constant for transitions from the initial to the final state. In addition, by glueing together trajectory segments one obtains also typical transition pathways. In contrast to transition path sampling, forward flux sampling does not require knowledge of the stationary distribution, such that it can be applied to truly non-equilibrium processes. However, the efficiency

of a forward flux sampling simulation strongly depends on the suitable definition of the interfaces, which is equivalent to knowledge of the reaction coordinate. Hence, in equilibrium (or quasi-equilibrium) situations, transition interface sampling is a safer choice than forward flux sampling.

2.2.4 Milestoning

Like TIS and FFS, the milestoning method [24, 45, 21] relies on the definition of interfaces, so-called milestones, to study the kinetics of rare transitions between stable states. In the milestoning method, one first prepares equilibrium distributions of initial conditions constrained to lie on the milestones. Short molecular dynamics trajectories started from these points are then used to determine the probability density for the time needed to reach the next milestone. Under the assumption that times between successive milestone crossings are statistically independent from each other [43] (i.e., there is loss of correlation between the milestones), the total rate for the transition can be extracted from the kinetic information collected locally at the milestones using a generalized master equation. Milestoning is a method particularly well suited to study long diffusive transitions and has been applied to many biomolecular processes [44, 4].

2.2.5 Finite temperature string method

A different and more static perspective is taken in the finite temperature string method. Rooted in transition path theory [20], this method is suitable for the study of transitions in which all transition pathways are localized in a tube around a typical pathway[19, 42]. In the string method, this path at the center of the reactive tube is determined in an iterative fashion by carrying out several simulations constrained to hyperplanes that approximate iso-committor surfaces. The string method can be applied to Cartesian coordinates but also to a set of collective coordinates. In the latter case, the string method yields the minimum free energy path (MFEP),i.e. the most likely transition path in the space spanned by the collective variables [32]. The string method lends itself for the simulation of systems evolving stochastically and is particularly suitable for the study of dynamical processes in the overdamped limit, but cannot be applied to Newtonian dynamics.

2.2.6 Stochastic process rare event sampling (SPRES)

Stochastic process rare event sampling (SPRES) is an interface based method applicable both to equilibrium as well as non-equilibrium situations [6, 30]. In particular, the approach, which is similar to FFS, is suitable for the investigation of aging and driven systems. Short stochastic trajectories are started and, in contrast to FFS, are integrated for a fixed time interval. After each such interval a decision is made, based on the actual value of a reaction coordinate, if a particular trajectory is terminated or one or more new trajectories are spawned from the endpoint. The procedure is carried out in a way that favors trajectories moving forward along the progress variable. Keeping track of the bias applied in this way, one can infer likely pathways and reaction rate constants.

2.2.7 Weighted ensemble method (WE)

The basic strategy of splitting and propagting re-weighted trajectories is also applied in the weighted ensemble (WE) method [27], which can be used to study rare events in systems evolving according to Brownian dynamics. In the weighted ensemble method, trajectories can spawn hierarchies of daughter trajectories each of which is assigned a weight that takes into account the bias introduced into the trajectory generation by favoring motion towards the reactants. The weighted ensemble method has mainly been used to study conformational transitions in proteins [47] and molecular association processes [50].

3 Software packages for free energy computation and rare event sampling

There is a wide array of classical molecular simulation software available, ranging from free open source code, proprietary code that is free for academic use, and purely proprietary code, not to mention in-house codes of individual laboratories or consortia. In many cases such codes include full simulation engines – but some are used as molecular builders – and others focus on visualization. Much of this is not suitable for E-CAM within the context of Work Package 1 Classical Molecular Dynamics for various reasons. E-CAM software modules need to be: open source; free for academic use; scale well when ported to massively parallel platforms; be such that high quality force-fields are available for the applications important to our academic and industrial partners; frequently updated to the most recent parallel environments and platforms; ideally well accepted by the community; and, finally, the software needs to be reasonably readable at a source code level. More concretely, E-CAM needs to be able to:

- modify available existing code to build E-CAM modules;
- exploit available code for input, testing & benchmarks;
- exploit available code where E-CAM modules are in effect plug-ins; and,
- use the module creation also for advanced training in the development and application of advanced scientific software.

An additional desirable attribute is that E-CAM modules should be able to function, where appropriate, across more than one of the scientific work-packages of E-CAM.

3.1 Available packages

3.1.1 Molecular dynamics engines for massively parallel platforms

When all of these criteria are applied, one finds that only LAMMPS, GROMACS, and arguably NAMD are suitable simulation engines for our purposes. Codes which are primarily focused on other scientific work-packages as their characteristics and suitability are reported in the corresponding E-CAM deliverables (D2.1 and D4.1). LAMMPS is by far the most readable and easy to modify of these three C++ codes, and is mostly used for non-biological type applications, in part because until this year it could not be applied with the force-fields most suitable for life science applications. GROMACS and NAMD are ideally suited to biological applications. All three codes scale very well on PRACE type platforms and have strong and active communities of developers as well as users and can be used in hybrid multi-scale environments. All three also have a wide variety of advanced statistical mechanics modules "hard wired" at a source code level, which is an important consideration for computational speed/efficiency but which can be cloned and modified to implement E-CAM modules. The modules already present can be loosely divided into two categories: biased sampling/perturbations of underlying systems through the addition of potentials defined with respect to suitable collective variables, and, in the case of thermodynamic integration, particle types; and, multiple trajectory methods, such as replica exchange. This variety of module types is advantageous, not only for their direct use, but also because they serve as a template of prototypes which can be cloned and transformed into novel E-CAM modules. There are also two plug-in rare-event method libraries that can be used with LAMMPS, GROMACS and NAMD (and other codes too): PLUMED and COLVARS. PLUMED and COLVARS were conceived primarily to facilitate free energy calculations where a bias is applied using user defined sets of collective variables. They allow many different biasing schemes to be used, and if one is not already present, an existing scheme can typically be easily cloned and modified. Let us now examine each of these codes in turn, to see in what way they are indeed suitable for our purposes.

3.1.2 LAMMPS (Large-scale Atomic/Molecular Massively Parallel Simulator)

LAMMPS can model systems with only a few particles up to millions or billions, and is a freely-available open-source code, distributed under the terms of the GNU Public License. It uses a distributed-memory message-passing parallelism (MPI), and spatial-decomposition of the simulation domain for parallelism. It can also be used with GPU (CUDA and OpenCL) and Intel(R) Xeon Phi(TM) co processors. It can be used to run one or multiple simulations simultaneously (in parallel) from one script. It can be used directly, or built as a library allowing it to be invoked through a library interface or a Python wrapper. It can be easily coupled with other codes, where LAMMPS calls the other code, or the other code calls LAMMPS, or through an umbrella code calling several codes including LAMMPS, thus making LAMMPS very suitable for multi-scale and multi-paradigm simulations. About 20 % of LAMMPS is core code, and the rest consists of modules having a very consistent structure, which allows users to easily create their own in-house modules largely by cloning and modifying existing modules. In the context of an advanced user/developer

three classes are of key interest: "fixes" which in the E-CAM context can be used for implementing bias forces; "computes" which can be used to compute the expectation values of user defined observables/collective variables; and forces/energies which can be used to realize user defined force-fields.

3.1.3 GROMACS

Gromacs is widely used for simulations of biological system of up to a million atoms. This code has a large number of users, particularly in Europe. It is exceptionally fast through its use of parallelization algorithms working on almost every machine level: SIMD registers inside cores, multi-threading, heterogeneous CPU–GPU acceleration, state-of-the-art 3D domain decomposition, and ensemble-level parallelization through built-in replica exchange. GROMACS developers have been gradually transforming their code from being primarily C based, to one based on C++. It is anticipated that in keeping with this, GROMACS will become much more object oriented, legible, logical and well-structured. This should give to users (and developers) increased control to: run simulations, manipulate configurations and typologies, implement custom post-processing tools, and finally, of great importance to E-CAM, implement advanced sampling algorithms. For now the GROMACS code is still much less modular, and therefore less easy to modify by the expert user than LAMMPS. That said, the source code is readable, and modifiable, albeit requiring more effort than LAMMPS. The large number of modules it has to assist in preparing initial conditions, perform sophisticated free energy calculations as well as multi-scale multi-paradigm and mesoscale simulations, and an extensive tool kit for statistical analysis, mean that it is a very useful platform for E-CAM in the development of specialized modules. GROMACS is Free Software, available under the GNU Lesser General Public License (LGPL), version 2.1.

3.1.4 NAMD

NAMD is a parallel molecular dynamics code designed for high-performance simulation of large bio-molecular systems, scales to hundreds of cores for typical simulations and beyond 500,000 cores for the largest simulations. It is used internationally, particularly in the US, and is open source and free for non-commercial use. It is designed in an object-oriented style with C++ for extensibility and maintainability through a modular design. There is a general perception that only the most expert developers can write NAMD modules, in contrast with, for instance LAMMPS. This is perhaps due in part to the very extensive user friendly TCL toolkit of NAMD, which allows a wide variety of scripts driving various modules to be written by an active community of users. Thus, while NAMD is very useful as a code of reference, the lack of a well documented developer manual means that for now, it is not a suitable platform for E-CAM module development.

3.1.5 PLUMED and COLVARS plug-in codes

PLUMED is a plug-in to several molecular simulation codes including LAMMPS, NAMD, GROMACS, CP2K (a DFT based MD code), and was developed primarily to facilitate a large variety of advanced free energy calculations to be performed on different simulation engines and machine platforms. It can run as a stand alone code to analyse simulations results, or as a plug-in to codes such as those mentioned above. It is an open source C++ code, and is object oriented, legible, logical and well-structured, with detailed manuals both for users and developers, with the express purpose to facilitate advanced users/developers to create where necessary their own code, using appropriate PLUMED modules as templates. When used as a plug-in for a simulation engine, at each time step, it reads the coordinate data of the constituent atoms of the engine, performs various operations on that data, for example computing bias energies and forces associated with a wide variety of collective variables, and updates as appropriate the corresponding force fields on the simulation engine. It is also involved in the initialization process, and writes results to files if required. For systems that are not very large requiring a modest number of computing cores, PLUMED is excellent, and extremely useful. However, as it is really a plug-in, it is cannot directly exploit in its own calculations specialized features in simulation engines developed to perform the calculation of energy and forces using message passing on distributed platforms across multiple nodes, as well as over the processor cores of each node. This means that for simulations over large numbers of nodes, plugins such as PLUMED may produce significant overheads, coming from the costs of transmitting atomic coordinates to one node and of processing them. Thus PLUMED is not appropriate to E-CAM needs when performing production runs on large systems. However, E-CAM can exploit the simple and concise PLUMED syntax in the user definition of free energy calculations, as well as use its results in the development of test cases.

The COLVARS plugin, developed in the US, and for use especially with LAMMPS and NAMD is very similar in spirit to PLUMED, but performs better on massively parallel platforms, particularly when the associated simulation engine is LAMMPS, although it too can have difficulties with communication bottle necks associated with the reading of atomic positions required to determine, for instance, bias energies and forces. This difficulty can be significant in

cases where a collective variable depends on a large number of atomic coordinates. LAMMPS has automatic features that can reduce this difficulty, although not entirely, as does NAMD for center of mass coordinates.

3.2 System preparation and force-field/topology translators

The most widely used visualizer for particle based simulations is VMD. Its use extends beyond visualization of the results of simulations and includes: statistical analysis; preparation of initial conditions; translation of force-field parameters from one code to another appropriate to different simulation engines using the TCL module topo tools. Another very useful tool for life-science applications is the CHARMM-GUI. It is normally used as an internet service – and through topo-tools and other scripting facilitates benchmark comparisons between different codes (LAMMPS, GROMACS, NAMD and CHARMM (a proprietary code from the lab of noble laureate Martin Karplus) for a variety of systems. While bio-informatics is not strictly speaking within the purview of E-CAM, bio-informatics tools are extensively used and needed in the preparation of initial conditions for large life-science systems. As such, they are relevant to life-science industries associated with E-CAM, for instance pharmaceuticals, and food science. Internet based services are provided by, for example, I-TASSER, and INTFOLD (bio-informatics), RCSB Protein data bank (equilibrium and in some cases dynamical structures via X-RAY/NMR). Codes for linux platforms are available from I-TASSER, and the widely used homology mapping tools known as modeller. Another suite of codes developed primarily for advanced materials research through an NSF funded program is called OPENKIM, but is a community initiative with participants and developers world-wide. It is in effect a database of force-fields developed for different materials, and includes test data and comparisons with experiment.

3.3 Commercial codes - industry practice

Before focusing on our immediate concerns, it is worth mentioning two vendors of simulation software: Biovia/Dassault (Materials Studio and Discover for life sciences) and Schrödinger. The Biovia codes are widely used in industry, and to some extent by experimentalists, because of their ease of use, particularly for the non-expert simulator. Thus, these codes provide a useful indication of industrial needs of modelling as it is currently perceived. Interestingly many of their core modules were first developed in academia, and often remain free for non-commercial use albeit without user friendly GUIs. That said, academic codes are usually far better tuned for scalability and high performance on massively parallel platforms.

3.4 Challenges for free energy computation

As alluded to earlier, free energy calculations can be divided into two major categories - those which employ perturbations to either bias a system with respect to a suitable set of collective variables or perform alchemical transformations, and trajectory methods such as replica exchange. Or for that matter combinations of these approaches. The free energy methods available in PLUMED and COLVARS are focused on the use of bias, but also include statistical analysis tools. LAMMPS, GROMACS and NAMD have modules relevant to both categories. But despite this large choice of methods, significant obstacles still have to be faced when performing free energy calculations of realistic systems. For example, molecular mechanisms play a central role in the functioning of GPCR proteins, and yet their equilibrium structures are very difficult to determine in experiment or simulation due to their trans-membrane nature. GPCR proteins play a crucial role in inter and intra cellular signaling, and are as a result the target of some 30 % of drugs. Another example are the effects on large molecular complexes of changing the conditions of a solvent, for example by adding a co-solvent, or an impurity or a salt. At physiological densities, not to mention very high densities, this is very difficult to model accurately due to the presence of tricky singularities associated with particle insertion that have to be avoided. Even more challenging is modelling the effect of changes in the pH, which is an important industrial issue given that pH and salt levels are two of the most accessible control parameters to the experimentalist or chemical engineer working in pharmaceutical processing or food science for example. GROMACS and NAMD have constant pH codes, but their accuracy is still frequently far poorer (and computationally more expensive) than semi-empirical approaches.

3.5 Rare event sampling

Rare event methods generally fall into two classes: (1) approaches that obtain the free energy by modifying the underlying potential energy surface, such as metadynamics and some forms of umbrella sampling; and (2) trajectory-based approaches, which aim to capture correct kinetics by leaving the dynamics unchanged. LAMMPS, GROMACS, NAMD, and the plug-in codes have modules to implement many of the potential-modifying methods. However, the central

difficulty of determining reaction paths between dominant metastable and equilibrium states is still very challenging, and of great importance in biology, chemistry and physics. This has led to an increased focus on trajectory-based rare events methods.

Since trajectory-based rare events methods leave the dynamics of the system unchanged, the primary computational cost is in the underlying dynamics engine, and packages that implement these methods usually wrap around some other molecular dynamics package. This also means that the question of scalability can be placed either on the wrapping package (by running many trajectories simultaneously) or on the underlying engine (by parallelizing as discussed in the section on molecular dynamics codes).

Currently, there are no packages that dominate the field of trajectory-based rare event simulations. Most such simulations are still performed using scripts that are only shared among a small number of research groups, often because their authors do not consider the scripts sufficiently usable to be made public.

However, there are several packages which have started to fill this gap. In particular, we have identified four, all of which satisfy the E-CAM requirements of being open source and freely available to use or modify:

- FRESHS (Flexible Rare Event Sampling Harness System) [30], written in Python and shell scripts, can perform FFS and SPRES simulations, using Gromacs, LAMMPS, or ESPResSo.
- WESTPA (Weighted Ensemble Simulation Toolkit with Parallelization and Analysis) [49], written in Python and shell scripts, can perform weighted ensemble simulations and finite temperature string simulations, using NAMD, OpenMM, Gromacs, or Amber.
- MOIL (Molecular Operations In Life) [22], written in Fortran, can perform milestoning simulations and finite temperature string simulations, using its own internal MD engine.
- OPS (OpenPathSampling), written in Python, can perform TPS and TIS simulations (including replica exchange TIS), using OpenMM or an internal toy engine. Development is underway to support other MD engines, including LAMMPS and Gromacs.

Each of these packages implements different rare events methods, and supports different engines. Therefore, one of the important considerations is how flexible the overall framework is, i.e., which package makes it easier to implement the methods from the other packages.

On that point, OpenPathSampling's support for replica exchange transition interface sampling makes it stand out. This method requires tracking the entire trajectory (not just the final point) as well as tracking a replica identifier and the path ensemble currently associated with the trajectory. It also requires simultaneously sampling from several ensembles, and being able to perform replica exchange between them. Implementing this in any of the other codes would require a major overhaul of the core code. On the other hand, OpenPathSampling's data structures can handle the sorts of sampling used by the other codes, which have fewer requirements.

Another aspect to consider is the support of various underlying molecular dynamics engines. MOIL only works with its own engine, which largely excludes it from consideration for E-CAM development. However, the other packages are designed to support arbitrary engines. While OpenPathSampling currently lags in the number of engines supported, its approach to supporting external engines differs from the others, and will be more efficient. FRESHS and WESTPA work by running some fixed number of time steps per iteration. For fixed path length methods, this is a good approach. However, for flexible path length methods, this approach has additional computational cost from starting and stopping simulations, and from overshooting the target. OpenPathSampling's external engine module launches the external engine once, and reads the trajectory from the file system during the simulation. It then kills the trajectory when it reaches a stopping point, thus reducing the costs of restarts and overshooting. In addition, if a direct API is available (as with OpenMM), OpenPathSampling uses that instead, and therefore never overshoots.

OpenPathSampling also already includes extensive unit tests and in-code documentation, as the E-CAM software standards require. FRESHS lacks formal in-code documentation, and the test suites for both FRESHS and WESTPA are less extensive that those in OpenPathSampling.

4 Modules to be developed in Work Package 1

Bilateral and multilateral discussions with industrial partners have highlighted several problems where additional software modules are sorely needed. These include:

a. solubility (pharma and food science)

b. hydration and drying (pharma and food science)

c. effects of impurities (pharma and advanced materials)

d. effect of mutations on proteins

e. control of pH and salt levels (pharma, food science and advanced materials)

f. bio-availability, that is the transport of drugs and bioactives through various membranes (stomach wall, blood-brain barrier, cell walls to biological targets) (pharma and food science)

g. non-equilibrium (driven) systems (pharma)

h. in silico design of biosensors (bioscience and pharma, environmental testing)

i. nucleation kinetics and crystallization (pharma and advanced materials)

j. storage in complex matrices (pharma and advanced materials)

The software tools required to address topics a,b,c, and d are similar, even if the scientific scope is very large, ranging from advanced materials to molecular biology - and can be grouped together as alchemical methods. Simulating conditions of constant pH and salt levels and their effects on complexes is extremely challenging, and relevant to topics f, h ,i and j. In most of the above topics, rare-event methods are needed- either in the context of free energy, or to estimate kinetic effects and rate constants, the study of which can be facilitated by OpenPathSampling. While addressing all of these themes is beyond the current scope of E-CAM given the number of personnel, our objective is to identify common underlying methods for development as part of the E-CAM software infrastructure which will facilitate further development by the larger community. Note that in addition to the modules listed below additional modules will be developed based on the needs arising from the industrial collaborations of WP1.

4.1 Free energies

4.1.1 Alchemical methods

Alchemical methods can be also termed particle insertion and deletion methods. At low density, this can often by performed using simple Monte Carlo methods, but not at high density pertaining to themes a-d. NUI-UCD is developing a novel method for particle insertion - the porting of corresponding software modules to the E-CAM library is planned in the latter half of 2017, including benchmarking, and comparison with other approaches.

4.1.2 Constant pH methods

While there are several codes to simulate constant pH and salt levels, they remain computationally expensive and not very accurate. NUI-UCD in conjuction with collaborators is completing a benchmark study including semi-empirical and more detailed molecular simulation methods. It is possible to use alchemical approaches here too - but the computational cost has to date been considered prohibitive. Software modules are not expected earlier than late 2017 and more likely 2018.

4.2 Path Sampling

Software modules for trajectory sampling will be developed that contribute to several aspects of the OpenPathSampling software package. These modules will facilitate use of the code in the pilot project developed in collaboration with our industrial partner Biki Technologies on protein-ligand binding. Modules will be developed that add support for new molecular dynamics engines, such as adding support for Gromacs. Established path sampling methods that are missing from OpenPathSampling will also be added, such as two-way shooting and aimless shooting. Other algorithms that are related to path sampling can also be added, such as reactive flux (Bennett-Chandler) and calculation of the committor. Finally, additional analysis tools for OpenPathSampling, such as path density plots, will be developed

as software modules. Some of these modules are already in development and will be delivered in January 2017, and several others will be delivered through the rest of 2017 and into 2018.

5 Benchmarking

The codes we will develop in WP1 will focus on statistical sampling for free energy computation and rare event trajectories and will use established software packages as MD engines to generate the dynamics of the system. Here, we will provide a description of the benchmarks we will carry out to assess the performance of Gromacs and Lammps, two of the most popular MD packages currently available.

To obtain a performance envelope under different hardware architectures and parallel methodologies, the following test sets will be run for each code:

- Strong scaling without Input Output (IO)
- Weak scaling without IO
- Strong scaling with IO
- Strong scaling without IO using the Intel XeonPhi (KNC) coprocessor
- Strong scaling without IO using NVidia GPU accelerators

The strong scaling without IO will give and idea of the performance vs number of cores keeping the size of the problem constant (like fixed number of atoms or particles). The test will also give an idea of the ratio between computing and communication time.

The second set, weak scaling without IO, will better identify the effect of the communication between cores. In this case, the ratio between problem size and number of cores is kept constant.

The purpose of the third test is to understand the impact of IO operations on the performance of the codes. The frequency of the IO operations will be chosen accordingly to typical values used in scientific applications.

The last two sets of tests are driven by the exascale target within the E-CAM project: coprocessor and accelerators will be the future paths to achieve this target and preliminary benchmarks will help to understand how far the codes are from those architectures and how much effort will be needed to eventually adapt the modules developed within E-CAM to run efficiently on them.

All tests will be run accordingly to the parallelism methodology coded (distributed memory, shared memory, task parallelism, etc.). The different impact of the different methodologies for the given architecture will not be analyzed (i.e. MPI on a shared memory on a node vs OpenMP), but ideal combination only will be tested (i.e. one MPI processes per node and OpenMP threads matching the maximum number of logical core per nodes).

For hybrid architectures (processor/coprocessor) the tests will be run only according to the optimization set (i.e. the split between processor and coprocessor work) suggested by the developer of the codes.

Finally, across all tests the wall time of each core is considered equal across all cores (no imbalance analysis will be carried out).

6 Conclusion

One of the central objectives of WP1 of E-CAM is the creation of software modules for the calculation of free energies and the sampling and analysis of rare event trajectories. This report gives an overview of currently available algorithms, provided in Section 2, as well as a discussion of relevant software packages, given in Section 3. Based on the analysis of algorithms and software, we define, in Section 4, a core set of modules to be developed in WP1. Note that additional modules may be developed based on the needs of the industrial partners of WP1. The report also includes, in Section 5, a description of the benchmarks we will carry out to assess the performance of the software packages that will be used as molecular dynamics engines employed to generate dynamics underlying the sampling routines developed in WP1.

References

URLs referenced

Page ii

https://www.e-cam2020.eu ... https://www.e-cam2020.eu
https://www.e-cam2020.eu/deliverables ... https://www.e-cam2020.eu/deliverables
Christoph.Dellago@univie.ac.at ... mailto:Christoph.Dellago@univie.ac.at
http://creativecommons.org/licenses/by/4.0 ... http://creativecommons.org/licenses/by/4.0

Page 7

LAMMPS ... http://lammps.sandia.gov/
GROMACS ... http://www.gromacs.org/
NAMD ... http://www.ks.uiuc.edu/Research/namd/
PLUMED ... http://www.plumed.org/
COLVARS ... http://colvars.github.io/COLVARS

Page 8

CP2K ... https://www.cp2k.org/

Page 9

VMD ... http://www.ks.uiuc.edu/Research/vmd/
topo tools ... https://sites.google.com/site/akohlmey/software/topotools
CHARMM-GUI ... http://www.charmm-gui.org/
CHARMM ... https://www.charmm.org/
I-TASSER ... http://zhanglab.ccmb.med.umich.edu/I-TASSER/
INTFOLD ... http://www.reading.ac.uk/bioinf/IntFOLD/index.html
RCSB ... http://www.rcsb.org/pdb/home/home.do
modeller ... https://salilab.org/modeller/
OPENKIM ... https://openkim.org/
Materials Studio ... http://accelrys.com/products/collaborative-science/biovia-materials-studio/
Discover for life sciences ... http://accelrys.com/products/collaborative-science/biovia-discovery-studio/
Schrödinger ... https://www.schrodinger.com/

Page 10

FRESHS ... http://www.freshs.org
WESTPA ... https://westpa.github.io/westpa/
MOIL ... http://clsb.ices.utexas.edu/web/moil.html
OPS ... http://openpathsampling.org

Citations

[1] Rosalind J. Allen, Daan Frenkel, and Pieter Rein ten Wolde. Forward flux sampling-type schemes for simulating rare events: Efficiency analysis. *The Journal of Chemical Physics*, 124(19), 2006.

[2] Rosalind J. Allen, Daan Frenkel, and Pieter Rein ten Wolde. Simulating rare events in equilibrium or nonequilibrium stochastic systems. *The Journal of Chemical Physics*, 124(2), 2006.

[3] Rosalind J. Allen, Patrick B. Warren, and Pieter Rein ten Wolde. Sampling rare switching events in biochemical networks. *Phys. Rev. Lett.*, 94:018104, Jan 2005.

[4] Juan M. Bello-Rivas and Ron Elber. Simulations of thermodynamics and kinetics on rough energy landscapes with milestoning. *Journal of Computational Chemistry*, 37(6):602–613, 2016.

[5] Charles H Bennett. Efficient estimation of free energy differences from monte carlo data. *Journal of Computational Physics*, 22(2):245 – 268, 1976.

[6] Joshua T. Berryman and Tanja Schilling. Sampling rare events in nonequilibrium and nonstationary systems. *The Journal of Chemical Physics*, 133(24), 2010.

[7] P. G. Bolhuis, D. Chandler, C. Dellago, and P. L. Geissler. Transition Path Sampling: Trowing Ropes Over Rough Mountain Passes, in the Dark. *Annu. Rev. Phys. Chem.*, 53:291, 2002.

[8] P. G. Bolhuis and C. Dellago. Practical and conceptual path sampling issues. *Eur. Phys. J. Special Topics*, pages doi:10.1140/epjst/e2015–02419–6, 2015.

[9] P. G. Bolhuis, C. Dellago, and D. Chandler. Sampling ensembles of deterministic transition pathways. *Faraday Discuss.*, 110:412, 1998.

[10] P. G. Bolhuis, C. Dellago, P. L. Geissler, and D. Chandler. Transition path sampling: throwing ropes over mountains in the dark. *J. Phys.: Condens. Matter*, 12:A147, 2000.

[11] Ernesto E. Borrero and Fernando A. Escobedo. Optimizing the sampling and staging for simulations of rare events via forward flux sampling schemes. *The Journal of Chemical Physics*, 129(2), 2008.

[12] David Chandler. Statistical mechanics of isomerization dynamics in liquids and the transition state approximation. *The Journal of Chemical Physics*, 68(6):2959–2970, 1978.

[13] C. Chipot and A. Pohorille. *Free energy calculations*. Springer, 2007.

[14] Gavin E. Crooks. Nonequilibrium measurements of free energy differences for microscopically reversible markovian systems. *Journal of Statistical Physics*, 90(5):1481–1487, 1998.

[15] C. Dellago and P. G. Bolhuis. Transition Path Sampling and other Advanced Simulation Techniques for Rare Events. In C. Holm and K. Kremer, editors, *Advanced Computer Simulation Approaches for Soft Matter Sciences III*, volume 221 of *Advances in Polymer Science*, page 167. Springer-Verlag, Berlin Heidelberg, 2009.

[16] C. Dellago, P. G. Bolhuis, and D. Chandler. On the calculation of reaction rate constants in the transition path sampling. *J. Chem. Phys.*, 110:6617, 1999.

[17] C. Dellago, P. G. Bolhuis, F. S. Csajka, and D. Chandler. Transition path sampling and the calculation of rate constants. *J. Chem. Phys.*, 108:1964, 1998.

[18] C. Dellago, P. G. Bolhuis, and P. L. Geissler. Transition Path Sampling. *Adv. Chem. Phys.*, 123:1, 2002.

[19] Weinan E, Weiqing Ren, and Eric Vanden-Eijnden. Finite temperature string method for the study of rare events. *The Journal of Physical Chemistry B*, 109(14):6688–6693, 2005. PMID: 16851751.

[20] Weinan E. and Eric Vanden-Eijnden. Towards a theory of transition paths. *Journal of Statistical Physics*, 123(3):503, 2006.

[21] Ron Elber. Perspective: Computer simulations of long time dynamics. *The Journal of Chemical Physics*, 144(6), 2016.

[22] Ron Elber, Adrian Roitberg, Carlos Simmerling, Robert Goldstein, Haiying Li, Gennady Verkhivker, Chen Keasar, Jing Zhang, and Alex Ulitsky. MOIL: A program for simulations of macromolecules. *Computer Physics Communications*, 91(1-3):159–189, September 1995.

[23] Henry Eyring. The activated complex in chemical reactions. *The Journal of Chemical Physics*, 3(2):107–115, 1935.

[24] Anton K. Faradjian and Ron Elber. Computing time scales from reaction coordinates by milestoning. *The Journal of Chemical Physics*, 120(23):10880–10889, 2004.

[25] Alan M. Ferrenberg and Robert H. Swendsen. Optimized monte carlo data analysis. *Phys. Rev. Lett.*, 63:1195–1198, Sep 1989.

[26] D. Frenkel and B. Smit. *Understanding Molecular Simulations: From Algorithms to Applications*. Academic Press, 2002.

[27] G.A. Huber and S. Kim. Weighted-ensemble brownian dynamics simulations for protein association reactions. *Biophysical Journal*, 70(1):97 – 110, 1996.

[28] C. Jarzynski. Nonequilibrium equality for free energy differences. *Phys. Rev. Lett.*, 78:2690–2693, Apr 1997.

[29] John G. Kirkwood. Statistical mechanics of fluid mixtures. *The Journal of Chemical Physics*, 3(5):300–313, 1935.

[30] Kai Kratzer, Joshua T. Berryman, Aaron Taudt, Johannes Zeman, and Axel Arnold. The flexible rare event sampling harness system (freshs). *Computer Physics Communications*, 185(7):1875 – 1885, 2014.

[31] Alessandro Laio and Michele Parrinello. Escaping free-energy minima. *Proceedings of the National Academy of Sciences*, 99(20):12562–12566, 2002.

[32] Luca Maragliano, Alexander Fischer, Eric Vanden-Eijnden, and Giovanni Ciccotti. String method in collective variables: Minimum free energy paths and isocommittor surfaces. *The Journal of Chemical Physics*, 125(2), 2006.

[33] Luca Maragliano and Eric Vanden-Eijnden. A temperature accelerated method for sampling free energy and determining reaction pathways in rare events simulations. *Chemical Physics Letters*, 426(1–3):168 – 175, 2006.

[34] Luca Maragliano and Eric Vanden-Eijnden. Single-sweep methods for free energy calculations. *The Journal of Chemical Physics*, 128(18), 2008.

[35] D. Moroni, P. G. Bolhuis, and T. S. van Erp. Rate constants for diffusive processes by partial path sampling. *J. Chem. Phys.*, 120:4055, 2004.

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

[37] Michael R. Shirts and John D. Chodera. Statistically optimal analysis of samples from multiple equilibrium states. *The Journal of Chemical Physics*, 129(12), 2008.

[38] Robert H. Swendsen and Jian-Sheng Wang. Replica monte carlo simulation of spin-glasses. *Phys. Rev. Lett.*, 57:2607–2609, Nov 1986.

[39] Pratyush Tiwary and Michele Parrinello. From metadynamics to dynamics. *Phys. Rev. Lett.*, 111:230602, Dec 2013.

[40] G.M. Torrie and J.P. Valleau. Nonphysical sampling distributions in monte carlo free-energy estimation: Umbrella sampling. *Journal of Computational Physics*, 23(2):187 – 199, 1977.

[41] T. S. van Erp, D. Moroni, and P. G. Bolhuis. A novel path sampling method for the calculation of rate constants. *J. Chem. Phys.*, 118:7762, 2003.

[42] Eric Vanden-Eijnden and Maddalena Venturoli. Revisiting the finite temperature string method for the calculation of reaction tubes and free energies. *The Journal of Chemical Physics*, 130(19), 2009.

[43] Eric Vanden-Eijnden, Maddalena Venturoli, Giovanni Ciccotti, and Ron Elber. On the assumptions underlying milestoning. *The Journal of Chemical Physics*, 129(17), 2008.

[44] Shruthi Viswanath, Steven M. Kreuzer, Alfredo E. Cardenas, and Ron Elber. Analyzing milestoning networks for molecular kinetics: Definitions, algorithms, and examples. *The Journal of Chemical Physics*, 139(17), 2013.

[45] Anthony M. A. West, Ron Elber, and David Shalloway. Extending molecular dynamics time scales with milestoning: Example of complex kinetics in a solvated peptide. *The Journal of Chemical Physics*, 126(14), 2007.

[46] B. Widom. Some topics in the theory of fluids. *The Journal of Chemical Physics*, 39(11):2808–2812, 1963.

[47] Bin W. Zhang, David Jasnow, and Daniel M. Zuckerman. Efficient and verified simulation of a path ensemble for conformational change in a united-residue model of calmodulin. *Proceedings of the National Academy of Sciences*, 104(46):18043–18048, 2007.

[48] Robert W. Zwanzig. High-temperature equation of state by a perturbation method. i. nonpolar gases. *The Journal of Chemical Physics*, 22(8):1420–1426, 1954.

[49] Matthew C Zwier, Joshua L Adelman, Joseph W Kaus, Adam J Pratt, Kim F Wong, Nicholas B Rego, Ernesto Suárez, Steven Lettieri, David W Wang, Michael Grabe, Daniel M Zuckerman, and Lillian T Chong. WESTPA: An Interoperable, Highly Scalable Software Package for Weighted Ensemble Simulation and Analysis. *Journal of chemical theory and computation*, 11(2):800–809, January 2015.

[50] Matthew C. Zwier, Joseph W. Kaus, and Lillian T. Chong. Efficient explicit-solvent molecular dynamics simulations of molecular association kinetics. *Journal of Chemical Theory and Computation*, 7(4):1189–1197, 2011. PMID: 26606365.