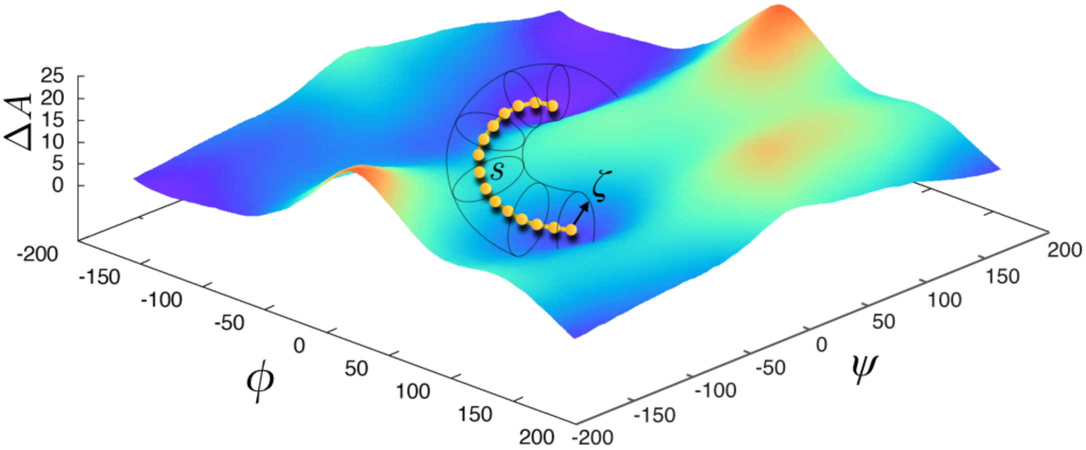
Department of Biochemistry and Molecular Biology
Gordon Center for Integrative Science
The University of Chicago

Centre National de la Recherche Scientifique
Laboratoire International Associé CNRS-UIUC
Université de Lorraine

University of Illinois at Urbana-Champaign
Beckman Institute for Advanced Science and Technology
Theoretical and Computational Biophysics Group

String method with swarms of trajectories: A tutorial for free-energy calculations along a minimum-action path



Mikolai Fajer
Jérôme Hénin
Benoît Roux
Christophe Chipot

September 16, 2017

Please visit www.ks.uiuc.edu/Training/Tutorials/ to get the latest version of this tutorial, to obtain more tutorials like this one, or to join the `tutorial-l@ks.uiuc.edu` mailing list for additional help.

Abstract

This tutorial sets out to determine the free-energy difference between two conformational states of a short, terminally blocked peptide, N–acetyl–N′–methylalaninamide, along a meaningful transition pathway. To reach this objective, use will be made of the string method with swarm of trajectories. Starting from a guess, rectilinear transition pathway connecting the $\mathrm{C}_{7\mathrm{eq}}$ and $\mathrm{C}_{7\mathrm{ax}}$ conformational states in the (ϕ, ψ) backbone-torsion subspace, the most probable, minimum-action path will be sought. The free-energy change along this path will be subsequently estimated, employing path-collective variables in the framework of the adaptive biasing force importance-sampling algorithm.

Contents

1. Introduction

The primary objective of this tutorial is to determine the lowest free-energy, or minimum-action pathway that connects two thermodynamic states of a short peptide, namely N–acetyl–N′–methylalaninamide (NANMA), commonly known a dialanine, and subsequently evaluate the free-energy change along this pathway. Towards this end, use will be made of an approach that pertains to the class of transition-path sampling methods and referred to as string method with swarm of trajectories [1]. This approach builds upon the seminal string method [2], aimed at finding the minimum-action path and associated free energy along the latter in a subspace formed by an appreciably large set of collective variables. In the original formulation, the string — a parametrized curvilinear abscissa mirroring the transition space in high dimension, is evolved as a collection of images by evaluating the local mean force and the metric tensor at each image in constrained molecular dynamics simulations. Assuming that the subspace embraces all relevant degrees of freedom to provide a faithful model of the reaction coordinate, the minimum-action path is an isocommittor path [3, 4] consisting of well-ordered intermediate states describing the progress from one thermodynamic state to the other of the conformational transition. Once the string connecting the two basins of interest of the conformational free-energy surface of NANMA has been optimized, it will be utilized as a transition coordinate to determine the one-dimensional potential of mean force (PMF) along it, employing the adaptive biasing force (ABF) algorithm [5, 6], together with path-collective variables [7].

This tutorial contains advanced material. Do not attempt to tackle the problems therein if you have no preliminary experience with free-energy calculations. The neophyte reader eager to get acquainted with the computation of transition pathways and the free-energy change along it is strongly advised to complete first the introductory tutorial on adaptive-biasing-force calculations — notably the exercise devoted to mapping the two-dimensional free-energy landscape of N–acetyl–N′–methylalaninamide (NANMA), using backbone torsional angles.

The ABF algorithm is implemented as part of the "collective variable caculations" (`colvars`) module of NAMD. The `colvars` module [8, 9] is extensively documented in the NAMD user's guide. A basic working knowledge of VMD and NAMD is highly recommended.

Completion of this tutorial requires

- the various files contained in the archive provided with this document;
- NAMD 2.12 or later (`http://ks.uiuc.edu/Research/namd`);
- VMD 1.9 or later (`http://ks.uiuc.edu/Research/vmd`).

1.1. Theoretical underpinnings

Our objective is to characterize the slow transitions between two thermodynamic states defined by a set of n collective, or coarse variables, $\mathbf{z} \equiv \{z_1, z_2, \ldots, z_n\}$, where $n << N$, the number of atoms of the molecular assembly. A convenient framework towards this end is provided by the concept of most probable transition pathway, i.e., the most probable sequence of configurations visited in the course of the transition between the two thermodynamic states of interest.

The PMF for the coarse variables $\mathbf{z}$ is defined by,

$$\mathrm{e}^{-\beta w(z)} = \frac{\displaystyle\int \mathrm{d}\mathbf{x}\ \delta[\mathbf{z} - \mathbf{z}'(\mathbf{x})]\ \mathrm{e}^{-\beta U(\mathbf{x})}}{\displaystyle\int \mathrm{d}\mathbf{x}\ \mathrm{e}^{-\beta U(\mathbf{x})}} \tag{1}$$

where the Dirac function reflects sampling of the microstates belonging to an iso-$\mathbf{z}$ surface of configurational space.

We will assume that the coarse variables can evolve on the free-energy surface following a random walk, which can be described by a Brownian propagator, discretizing motion by means of time increments, Δt,

$$z_i(\Delta t) = z_i(0) + \sum_j \{\beta D_{ij}[\mathbf{z}(0)]F_j[\mathbf{z}(0)] + \nabla_j D_{ij}[\mathbf{z}(0)]\}\,\Delta t + R_i(0) \tag{2}$$

where D_{ij} is the diffusivity tensor, F_i is the conservative mean force, i.e., $-\nabla_i w(z)$, and $R_i(0)$ is a Gaussian, random noise of zero mean and variance obeying the fluctuation-dissipation theorem, i.e., $\langle R_i(0) R_i(0)\rangle = 2D_{ij}\Delta t$.

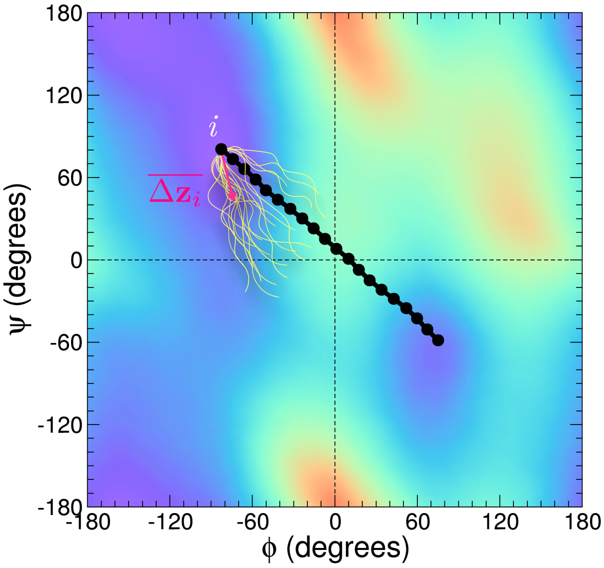


Figure 1: Generation of the swarm of trajectories at image i of the string. From the ensemble of short trajectories, an average drift, $\overline{\Delta z_i}$, is computed, which will subsequently be utilized to evolve the string. In the present example, the string connects two basins of the free-energy landscape of N–acetyl–N′–methylalaninamide defined in the collective-variable subspace formed by the two backbone torsional angles ϕ and ψ.

We will now consider a path, $\mathbf{z}(\kappa)$, connecting the two thermodynamics states, conventionally denoted A and B, of the free-energy landscape, where κ is a progression variable, such that $\kappa = 0$ at state A, and $\kappa = 1$ at state B. An interesting property of the minimum-action path is that evolution starting from anywhere on the latter has the highest probability to remain confined along this particular pathway, which is satisfied when the random noise is nil.

A path is described as an ordered sequence of M images, $\{\mathbf{z}^1, \mathbf{z}^2, \ldots, \mathbf{z}^M\}$. In order to converge towards the minimum-action path, starting from an arbitrary one, the algorithm would evolve each image until propagation would only move the images along the path — in other words, when convergence is attained, the images no longer move in the direction perpendicular to the path. To avoid the undesirable effect of evolving images towards regions of lower free energy, one can enforce at the outcome of each iteration that the distance between consecutive images remain constant. This step is generally referred to as *reparametrization* of the string.

A natural route to evolve the initial, arbitrary path towards the sought minimum action path consists in using the average drift inferred from a ensemble of unbiased trajectories of length Δt generated at each image of the string (see Figure 1). The average drift writes,

$$\overline{\Delta z_i(\Delta t)} = \overline{z_i(\Delta t) - z_i(0)} = \sum_j \{\beta D_{ij}[\mathbf{z}(0)] F_j[\mathbf{z}(0)] + \nabla_j D_{ij}[\mathbf{z}(0)]\} \, \Delta t \tag{3}$$

It ought to be noted that the converged string satisfies the property $\overline{\Delta \mathbf{z}}^{\perp} = 0$, because evolution of the coarse variables obeys Brownian dynamics. The minimum-action path corresponds effectively to a zero-temperature pathway on the free-energy landscape described by its PMF, $w(\mathbf{z})$. It still remains that the degrees of freedom, $\mathbf{x}$, of the molecular assembly evolve a finite temperature.

An important consequence is that since the string is optimized in the collective-variable subspace, where the free-energy landscape is substantially smoother than the complete potential-energy surface of the molecular assembly, convergence is in principle unaffected by possible entrapments in local energy minima.

1.2. The string method with swarms of trajectories

The algorithm of the string method with swarms of trajectories is depicted in Figure 2. In a nutshell, for each image i of the string, the molecular assembly described by its degrees of freedom $\mathbf{x}$ is first thermalized with a set of appropriately chosen geometrical restraints to maintain the coarse variables close to $\mathbf{z}^i$. These restraints are then removed and the unbiased trajectories are then generated, from whence the average drift, $\overline{\mathbf{z}_i}$, is determined to infer the new position of the images in the collective-variable subspace. Next, the pathway is altered to satisfy the reparametrization conditions common to all variants of the string method [2, 4, 10, 11].

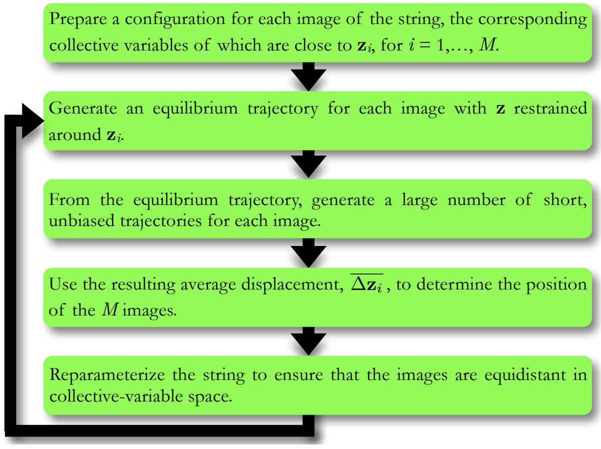

Prepare a configuration for each image of the string, the corresponding collective variables of which are close to $\mathbf{z}_i$, for i = 1,…, M.
Generate an equilibrium trajectory for each image with $\mathbf{z}$ restrained around $\mathbf{z}_i$.
From the equilibrium trajectory, generate a large number of short, unbiased trajectories for each image.
Use the resulting average displacement, $\overline{\Delta \mathbf{z}_i}$, to determine the position of the M images.
Reparameterize the string to ensure that the images are equidistant in collective-variable space.

Figure 2: Algorithm of the string method with swarm of trajectories for the determination of minimum-action paths. Detail of the numerical scheme can be found in reference [1].

From a practical standpoint, running a string calculation with swarms of trajectories is computationally demanding and can become prohibitively expensive for sizable molecular assemblies involving a large number of collective variables. As an illustration, let us consider a string formed by $M = 50$ images, at each one of which a swarm of $N_{\text{traj}} = 50$ independent, $\tau = 20$–ps long trajectories. Let us further assume that convergence occurs in $N_{\text{iter}} = 500$ iterations. It follows that the time required to optimize the string amounts to $t = M \times N_{\text{traj}} \times N_{\text{iter}} \times \tau = 25 \times 10^6$ ps, that is 25 μs. For this reason, string calculations are well suited for massively parallel computer architectures, where the swarms of trajectories can be generated concomitantly for the different images.

1.3. Free energies along path-collective variables

The primary objective of this tutorial is to get familiarized with the concept of transition-path sampling, or, more generally, the search of a relevant model of the reaction coordinate. Towards this end, our first task consists in selecting a set of coarse variables, $\mathbf{z}$, which will define the subspace in which the transition coordinate will be optimized. It ought to be clearly understood that once convergence has been attained, the pathway that we have obtained corresponds to the *minimum-action path in this particular subspace*. A different set of coarse variables could lead to a distinct minimum-action path. While this choice has no bearing on thermodynamics properties, notably on the free-energy difference between the two states A and B accessible to the molecular system — in the limit of a converged pathway and a converged free-energy calculation, it may greatly affect kinetics, in particular the height of the barrier of the PMF that underlies the conformational transition. The reader is invited to verify that the minimum-action path determined with the string method with swarms of trajectories corresponds to a faithful description of the reaction coordinate, for instance, by computing distributions of the committor [3, 12]. This aspect of the problem of modeling reaction coordinates goes well beyond the scope of the present tutorial.

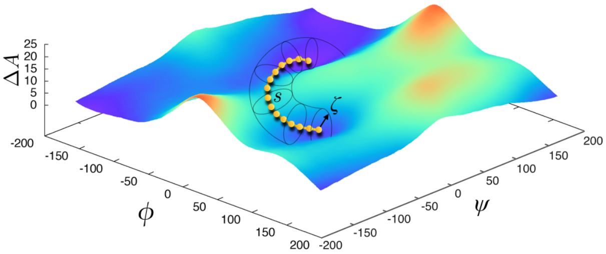


Figure 3: Free-energy landscape of NANMA determined using as coarse variables the backbone torsional angles, ϕ and ψ. The optimized string connects the two deepest basins, namely $\text{C}_{7\text{eq}}$ and $\text{C}_{7\text{ax}}$, going through the lowest possible free-energy barrier. path-collective variable s is introduced to measure the progression along the string, i.e., it varies between 0 and 1 as the conformation of the peptide changes from $\text{C}_{7\text{eq}}$ and $\text{C}_{7\text{ax}}$. ζ is utilized to confine sampling near the string.

The second, ancillary objective of this tutorial is to determine a free-energy change along a minimum-action path. Assuming a converged string, it is desirable to construct the PMF that underlies the conformational transition between states A and B of the free-energy landscape, not only to access the free-energy difference between the corresponding basins, but also to estimate the barrier that separates them. A convenient framework to achieve this goal is provided by the so-called path-collective variables [7], which will be combined here with the adaptive biasing force (ABF) algorithm [5, 6].

We start with our model of the reaction coordinate provided by the coarse variables, $\mathbf{z}$, and introduce two functions of the latter,

$$\begin{cases} s(\mathbf{z}) &= \lim\limits_{\lambda\to\infty} \dfrac{\displaystyle\int_0^1 \mathrm{d}t\; t\; \mathrm{e}^{-\lambda[\mathbf{z}-\mathbf{z}(t)]^2}}{\displaystyle\int_0^1 \mathrm{d}t\; \mathrm{e}^{-\lambda[\mathbf{z}-\mathbf{z}(t)]^2}} \\ \zeta(\mathbf{z}) &= -\lim\limits_{\lambda\to\infty} \dfrac{1}{\lambda}\,\ln \displaystyle\int_0^1 \mathrm{d}t\; \mathrm{e}^{-\lambda[\mathbf{z}-\mathbf{z}(t)]^2} \end{cases} \tag{4}$$

which define isosurfaces of dimension $n-1$, and are related to the free energy through,

$$w(s,\zeta) = -\frac{1}{\beta}\ln\langle\delta[s-s(\mathbf{z})]\delta[\zeta-\zeta(\mathbf{z})]\rangle \tag{5}$$

λ is the inverse of the mean square displacement between consecutive images. We will see that in practice, the infinite limit should not be construed explicitly — an appropriately chosen, large value is generally acceptable, provided that it allows the full range of the path-collective variable to be sampled adequately.

For all intents and purposes, $s(\mathbf{z})$ can be associated to the string — in the neighborhood of $\mathbf{z}(t)$, planes orthogonal to the latter offer a good approximation of the the isosurfaces defined by $s(\mathbf{z}) = s$, with $0 < s < 1$. $\zeta(\mathbf{z}) = \zeta$ mirrors the ensemble of points lying at z^2 from the string, thereby forming a tube wrapping around $\mathbf{z}(t)$ (see Figure 3). In other words, $\zeta(\mathbf{z})$ can be viewed as confinement boundaries, preventing sampling far from the images of the string.

Discretization of the transition coordinate described by the coarse variables $\mathbf{z}$ imposes some rewriting of the set of equations (6) as,

$$\left\{ \begin{array}{lcl} s(\mathbf{z}) & = & \dfrac{1}{M-1} \dfrac{\displaystyle\sum_{k=1}^{M} (k-1)\, \mathrm{e}^{-\lambda(\mathbf{z}-\mathbf{z}_k)^2}}{\displaystyle\sum_{k=1}^{M} \mathrm{e}^{-\lambda(\mathbf{z}-\mathbf{z}_k)^2}} \\ \zeta(\mathbf{z}) & = & -\dfrac{1}{\lambda} \ln \left(\displaystyle\sum_{k=1}^{M} \mathrm{e}^{-\lambda(\mathbf{z}-\mathbf{z}_k)^2} \right) \end{array} \right. \qquad (6)$$

where $(\mathbf{z} - \mathbf{z}_k)^2$ is evaluated as a mean-square displacement, i.e., the square of a root mean-square displacement (RMSD) with respect to the position of image k, which corresponds to one of the collective variables available in the `colvars` module.

1.4. Practical implementation of the path-collective variables

path-collective variables are implemented in NAMD, within the `colvars` module, in the form of a scripted function. The associated TCL script is sourced in the NAMD configuration file. No particular intervention of the end-user is required in this TCL script.

The different terms of the sums featuring in equation (6) correspond to components, specifically RMSD, which are declared in the `colvars` input file and passed to the TCL script to evaluate both $s(\mathbf{z})$ and $\zeta(\mathbf{z})$. The `colvars` input file, therefore, features M blocks like,

```
rmsd {
   atoms {
      atomnumbers { 1 5 6 7 9 11 15 16 17 19 }
   }
   refpositionsfile    images/string-00.pdb
   componentExp        1
```

where `string-00.pdb` is the PDB file containing the first image of the optimized string.

ABF calculations along a string can be expensive, in particular if both M and the number of atoms involved in the evaluation of the RMSD with respect to the different images are large. It should be observed, however, that only a few terms contribute significantly to $s(\mathbf{z})$ and $\zeta(\mathbf{z})$ in equation (6) — in general most of the M images lie sufficiently far from the current value of the transition coordinate to neglect safely their contribution to the sum. A dynamic mask updated frequently discards from the list those components, which, by virtue of the exponential dependence, carry a negligible weight in the sums of equation (6).

2. Setting up the calculations

The starting point of the calculations proposed in this tutorial is formed by the Cartesian coordinates (PDB) of the two conformational states of NANMA between which the minimum-action path will be determined, namely $\mathrm{C}_{7\mathrm{ax}}$ and $\mathrm{C}_{7\mathrm{eq}}$, together with the structure file (PSF) of the peptide.

It ought to be clearly understood that the objective of the present tutorial is not to look for the most accurate answer, using the latest force-field parameters available, but, instead, to get familiarized with the concept of transition coordinate, minimum-action path and the computation of free-energy changes along the latter. To be consistent with the tutorial dedicated to ABF calculations, wherein the (ϕ, ψ)–two-dimensional free-energy landscape of NANMA is determined, use will be made of the older CHARMM22 force field. The reader will be referred throughout the following sections to the computation of the NANMA free-energy map, which, amongst others, will be utilized to locate the string in the (ϕ, ψ) subspace.

2.1. Definition of the transition coordinate

As has been emphasized previously, search of a minimum-action path is subservient to the choice of coarse variables, $\mathbf{z}$, forming the subspace in which the slow transitions between thermodynamic states will be explored. In the particular instance of NANMA in vacuum, it has been shown that the backbone torsional angles, ϕ and ψ, constitute a good model of the reaction coordinate [3]. These coarse variables will be employed to determine the most probable path that connects the $\mathrm{C}_{7\mathrm{ax}}$ and $\mathrm{C}_{7\mathrm{eq}}$ conformations of the free-energy landscape of NANMA by means of the string method with swarm of trajectories. Subsequently, the free-energy change between the two basins will be calculated along the optimized string, which will serve as the transition coordinate.

The first step of this tutorial is to find the minimum-action path connecting two conformational states of NANMA in vacuum, using the string method with swarm of trajectories.
The second step is to determine the potential of mean force along the optimized string, underlying the conformation transition of NANMA.

2.2. Setting up the string calculation

As a necessary preamble to any string calculation, an initial guess of the pathway has to be proposed. Although the string method with swarm of trajectories is expected to converge towards the most probable pathway, convergence can be hampered by poor guesses. The example of the conformational transition in

NANMA between $C_{7\mathrm{eq}}$ and $C_{7\mathrm{ax}}$ is sufficiently trivial to start with a rectilinear initial string obtained as a linear combination of the values of the backbone torsional angles, ϕ and ψ (see Figure 4). To generate this initial string, a Python script provided in the distribution can be employed,

```
% cd prep
% python prepare.py
```

The Python script will create the relevant `colvars` files for the M images of the string, which, in turn, will be utilized by the string algorithm to optimize the path.

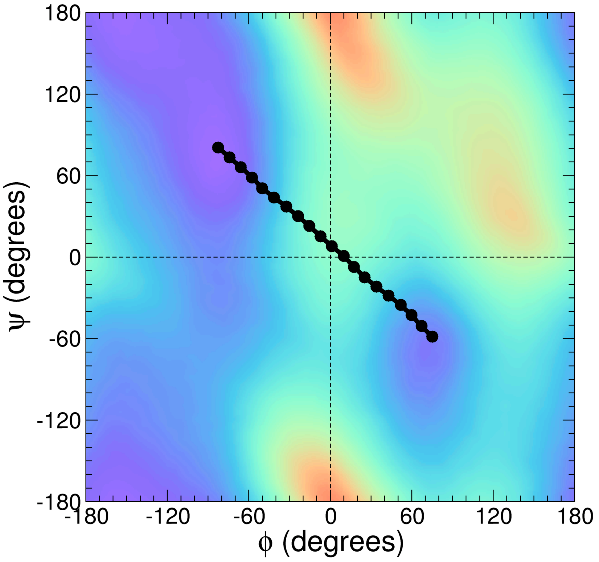


Figure 4: Superimposition of the initial string on the (ϕ, ψ) free-energy landscape of NANMA. The path and the position of the M images is obtained as a linear combination of the values of the torsional angles characteristic of the $C_{7\mathrm{eq}}$ and $C_{7\mathrm{ax}}$ conformational states.

Although it is sequential in nature, the algorithm of the string method with swarms of trajectories is embarrassingly parallelizable. In other words, optimization of the path can proceed in a stepwise fashion, one image after the other, running independent trajectories one after the other to form the swarm. One can also imagine performing the molecular dynamics simulations concurrently, taking advantage of massively parallel computer architectures.

In the particular instance of NANMA in vacuum, optimization of the string carried out on a parallel architecture would require $N_{\mathrm{core}} = M \times N_{\mathrm{traj}}$. Using $M = 20$ images at which $N_{\mathrm{traj}} = 20$ independent trajectories are generated, $N_{\mathrm{core}} = 400$.

Should the reader have access to a massively parallel computer architecture, it is advisable to evolve the images of the string concurrently for the sake of time. Under these premises, running the string calculation with swarms of trajectories would look like,

```
% cd parallel
%./create_output 400
% mpirun -np 400 namd2 +replicas 400 initial.conf +stdout output/%02d/job00.%02d.log
```

Depending on how NAMD is set up, additional options might be needed in the above command line, e.g., `-machinefile nodelist` to specify the list of nodes on which the job will be run. The reader is referred to the specific CHARM++ and NAMD documentations for further detail. In this example, use will be made of 400 cores, handling each one trajectory at any given iteration — which means that, since we have set `-np 400`, only one core is utilized to simulate the molecular system (for an appreciably larger assay formed by 100,000 atoms, ideally run on say 100 cores, the same string calculation would require `-np 40000`).

The command line `./create_output 400` prepares the relevant number of subdirectories in which intermediate results will be written, specifically the drift and the position of the images at each iteration.

Setting for NANMA in vacuum $M = 20$ images at which $N_{\rm traj} = 20$ independent trajectories are generated, directory `output` contains 400 subdirectories, ranging from `000` to `399`, and organized as follows. Subdirectories `000` to `019` gather the data accrued through the $N_{\rm iter}$ iterations for the $N_{\rm traj}$ trajectories spawned from the first image. The next block of twenty subdirectories gathers the data for the second image, and so forth.

Under most circumstances, in particular for tutorial purposes, the end-user is expected to have access to limited resources, consisting in general of a single– or dual-core laptop, thus, calling for a sequential version of the algorithm. In this case, running the string calculation would look like,

```
% cd sequential
%./create_output 400
% mpirun -np 20 namd2 +replicas 20 initial.conf +stdout output/%02d/job00.%02d.log
```

It should be noted in the above example that while the script handles the swarms for each image in a sequential fashion, the images themselves are still processed in parallel. This limitation is a direct consequence of NAMD being to this date unable to load new structures on the fly — in other words, if we were to run the string algorithm on a single core, we would start with the first image and generate its swarm of trajectories sequentially, yet without the possibility to switch to the next image of the string.

2.3. Setting up the free-energy calculation along the string

The second part of this tutorial consists in computing the free-energy difference between the $C_{7\rm eq}$ and $C_{7\rm ax}$ conformational states of NANMA along the optimized string. path-collective variables are coded within the `colvars` module of NAMD as scripted functions. Such TCL scripted functions are sourced in the NAMD configuration file,

```
source            pathCVsz-Comp.tcl
```

Scripted functions often feature a number of parameters, which can be passed directly from the NAMD configuration file. In the particular instance of `pathCVsz-Comp.tcl`, these parameters are declared in the following TCL block,

```
namespace eval pathCV {
  set lambda     20.0
  set tolerance  1e-7
  set freq        500
  set min_images   10
}
```

where `lambda` is the inverse of the mean square displacement between two consecutive images. As has been alluded to previously, performance of NAMD can be markedly affected by the number of components defined in the `colvars` configuration file — as a consequence of `colvars` being handled by `node0`. For this reason, a dynamic mask switches off components corresponding to images lying too far from the current value of the transition coordinate. `tolerance` is the threshold below which components are discarded and `freq` is the frequency at which the list of components forming the dynamic mask is updated. At each molecular dynamics steps, a minimum of `min_images` images will always be considered for the calculation of $s(\mathbf{z})$ and $\zeta(\mathbf{z})$.

The TCL scripted function is also instantiated by the `colvars` configuration file,

```
scriptedFunction      pathCVs
```

where `pathCVs` is the name given to $s(\mathbf{z})$. As usual, components are declared in the `colvars` configuration file,

```
colvar {
   name                    s
   width                   0.01
   lowerboundary           0.0
   upperboundary           1.0
   lowerwallconstant       10.0
   upperwallconstant       10.0
   scriptedFunction        pathCVs
   extendedLagrangian      on
   extendedFluctuation     0.01
   rmsd {
     atoms {
       atomnumbers { 1 5 6 7 9 11 15 16 17 19 }
     }
     refpositionsfile      images/string-00.pdb
     componentExp          1 }

...

   rmsd {
     atoms {
       atomnumbers { 1 5 6 7 9 11 15 16 17 19 }
      }
     refpositionsfile      images/string-19.pdb
     componentExp          20 }
}
```

As has been mentioned previously, the components are RMSDs with respect to the M images forming the string. It ought to noted that the free-energy calculation is carried out in the framework of the extended adaptive biasing force (eABF) algorithm, wherein the transition coordinate, denoted `s` in the configuration file, is connected to a fictitious particle by means of a stiff spring. In eABF, in lieu of following the transition coordinate as a function of time, the location of the fictitious particle is being tracked down — for further detail, the reader is referred to the reference article 6.

> The reader will verify that differences in the trajectory of the transition coordinate and of the fictitious particle to it are Gaussian distributed. Optimum choice of eABF parameters `extendedFluctuation` and `extendedTimeConstant` is discussed in reference 13.

It should be remembered that $s(\mathbf{z})$ varies between 0 and 1, corresponding to thermodynamic states A and B of the conformational transformation, i.e., $\mathrm{C}_{7\mathrm{eq}}$ and $\mathrm{C}_{7\mathrm{ax}}$ in the example of NANMA in vacuum. Fine discretization is, therefore, crucial, especially when $s(\mathbf{z})$ approaches the boundaries — the free-energy tends to rise abruptly when $s(\mathbf{z}) \longrightarrow 0$, or 1.

> The choice of `lambda` constitutes a thorny question. Too small a value will result in hampered sampling near the boundaries, i.e., 0 and 1, and, thus, an incomplete free-energy landscape. Conversely, too large a value will allow spilling beyond the boundaries — a common symptom observed in eABF calculations, yielding unrealistic gradients. Whilst $\lambda \sim 1/|\mathbf{z}_{i+1} - \mathbf{z}_i|^2$, it ought to be remembered that images are equidistant in the subspace formed by the coarse variables, they may not be so from the standpoint of mean-square displacements, i.e., the root mean-square displacement between consecutive images along the string is not necessarily constant.

3. Running and analyzing the simulations

Before embarking on the present tutorial, the reader ought to be reminded that the calculations proposed herein, especially the string method with swarms of trajectories, are computationally intensive and ideally necessitate access to the parallel architecture of a commodity cluster. The second part of the tutorial requires, in sharp contrast, far more modest resources owing to the limited size of the molecular object examined here.

3.1. Optimization of the string

The main parameters controlling the string optimization are declared in `swarms.conf`, invoked by `initial.conf`, mentioned earlier in this tutorial,

```
set temperature              300
set num_iter                 500
set num_images                20
set num_swarm_steps           20
set num_equil_steps        20000
set smooth_param             0.1
set swarms_force_constant    0.5
```

Here, `num_iter` is the number of iterations of the optimization, and `num_images`, the number of images of the string, i.e., M. At each image, `num_equil_steps` equilibration steps are generated prior to spawning multiple short trajectories, `num_swarm_steps` long, forming the swarm.

In the case of NANMA in vacuum, `num_swarm_steps` ought to be sufficiently small to avoid images from drifting far away from their previous position as a result of the fast, spontaneous isomerization of the peptide about its backbone torsional angles.

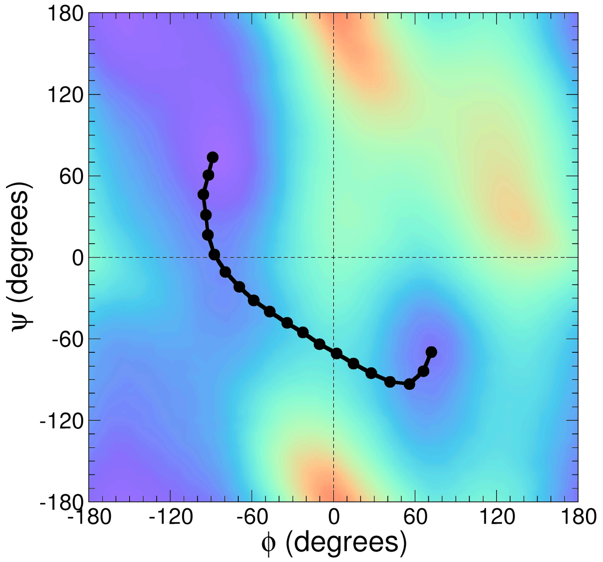


Figure 5: Superimposition of the optimized string on the (ϕ, ψ) free-energy landscape of NANMA. The path and the position of the M images has been obtained from a calculation involving 500 iterations and swarms of 20 trajectories. Each trajectory consisted of 20 molecular dynamics steps prefaced by a restrained equilibration of 20,000 steps.

After `num_iter` iterations, the raw results can be found in the subdirectories of `output`. A number of scripts are provided in the distribution of the tutorial to extract the relevant information from the large amount of files generated in the course of the optimization.

```
% ./collect-configuration number of images
```

will extract the last Cartesian coordinates of the different images and parse them in a series of M `coor` files — namely, `string-00.coor`, ..., `string-`M`.coor`, which can be readily visualized using VMD. For subsequent use, notably in the path-collective variable free-energy calculations, it is advised that the binary-formatted coordinate files be converted into PDB files, i.e., `string-00.pdb`, ..., `string-`M`.pdb`. Furthermore,

```
% ./parse-phi-psi number of images
```

will generate a series of `phi-`i`.dat` and `psi-`i`.dat` files, where $i = 0, \ldots, M - 1$, containing the value of backbone angles ϕ and ψ at each iteration of the string optimization (see Figure 5).

The reader is invited to check the convergence of the string optimization by considering the (ϕ, ψ) values of the different images at each iteration, using as a reference the string after `num_iter` iterations.

3.2. Free-energy change along the string

In this second part of the tutorial, two routes will be followed, exploring the role of the ancillary collective variable $\zeta(\mathbf{z})$. In the first route, this variable will be grossly ignored, which implies that sampling of configurational space can proceed in regions far from the string. In the second route, $\zeta(\mathbf{z})$ is introduced in the form of a geometric restraint,

```
harmonic {
   colvars                z
   centers                0.0
   forceConstant          0.5
}
```

aimed at confining sampling within a tube wrapping around the string. As has been shown by Branduardi et al, the most probable pathway does not necessarily coincide with $\zeta(\mathbf{z}) = 0$ [7]. It is, therefore, recommended that the force constant acting on $\zeta(\mathbf{z})$ be sufficiently soft to allow excursions from the optimized string.

Not too surprisingly, the effect of the harmonic potential acting on $\zeta(\mathbf{z})$ improves convergence of the free-energy calculation. Still, repeated simulations indicate that, either with or without a geometric restraint, a reliable and consistent PMF can be obtained within 5 ns.

It ought to be reminded that with eABF, the gradient generated in the course of the simulation (i.e., `grad` file) reflects the biasing force acting on the fictitious particle rather than on the atoms involved in the path collective variable itself.

The `grad` and `pmf` files written by Colvars in an eABF calculation reflect the average force acting on the extended, generalized coordinate — not on the actual collective variable, and should not be used as is for analysis purposes.

Access to the *true* gradient imposes a deconvolution of the contribution due to the harmonic spring from that of the potential energy function. This deconvolution is performed on the fly in Colvars, employing either the Zheng and Yang estimator [14], or the corrected z-averaged restraint estimator [15]. The first estimator is based on an umbrella integration (UI) [16],

$$G'(\xi') = \left(\frac{\mathrm{d}G}{\mathrm{d}\xi}\right)_{\xi'} = \frac{\displaystyle\sum_{\Xi'} N(\xi',\Xi')\left[\frac{(\xi' - \langle\xi_{\Xi'}\rangle)}{\beta\sigma^2_{\Xi'}} - K_\xi(\xi' - \Xi')\right]}{\displaystyle\sum_{\Xi'} N(\xi',\Xi')} \tag{7}$$

where $\xi(\mathbf{x})$ is the collective variable, function of the positions of the real particles of the molecular assembly and is restrained through potential $1/2\ K_\xi(\xi' - \xi_{\Xi'})^2$ to a one-dimensional fictitious particle, Ξ, moving dynamically . $N(\xi',\Xi')$ is the number of samples ξ' collected from the Ξ'-restrained ensemble, which is assumed to be Gaussian.

The second estimator is the corrected z–averaged restraint (CZAR) estimator [15],

$$G'(z') = \left(\frac{\mathrm{d}G}{\mathrm{d}z}\right)_{z'} = -\frac{1}{\beta}\frac{\mathrm{d}\ln\tilde{\rho}(z')}{\mathrm{d}z'} + k\left(\langle\lambda\rangle_{z'} - z'\right) \tag{8}$$

where $z = \xi(q)$ is the is the collective variable, λ is the extended variable harmonically coupled to z by means of a stiff spring of force constant k, and $\tilde{\rho}(z)$ is the observed distribution of collected variable z, upon which the time-dependent eABF bias is exerted.

The end-user may choose at the level of the Colvars configuration file between the CZAR (`CZARestimator`) and the Zheng and Yang (`UIestimator`) estimator. The former is the default for eABF calculations. Depending upon the choice of the estimator, NAMD will generate the usual potential-of-mean-force (for one-dimensional free-energy profiles only), gradient and histogram files with a distinctive, `czar`, or `UI` suffix.

The PMFs of Figure 6 show that conformational transition of NANMA from $\mathrm{C_{7eq}}$ to $\mathrm{C_{7ax}}$ is accompanied by an appreciable free-energy barrier amounting to about 9.5–10.2 kcal/mol, comparable to that found by Branduardi et al. [7].

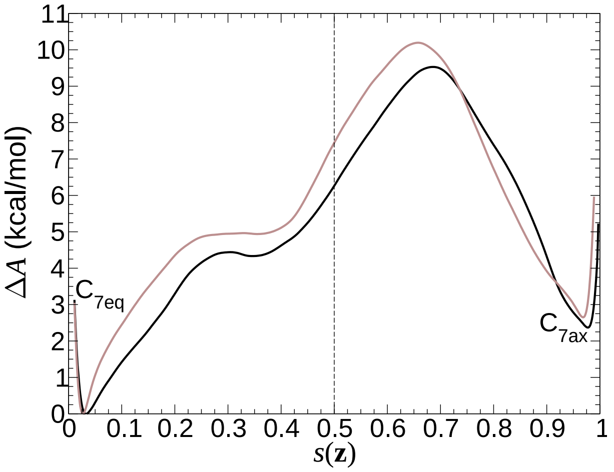


Figure 6: Potentials of mean force obtained along path-collective variable $s(\mathbf{z})$, characterizing the $\mathrm{C_{7eq}}$ to $\mathrm{C_{7ax}}$ conformational transition of NANMA in vacuum. Light curve: With a soft harmonic potential acting on ancillary variable $\zeta(\mathbf{z})$. Black curve: without geometric restraint on $\zeta(\mathbf{z})$.

The free-energy difference between the two thermodynamic states, of 2.3–2.6 kcal/mol, agrees well with the previous estimates of Rosso et al. [17] and Hénin et al. [8], following different computing strategies. The reader is invited to compare the estimate obtained here with that inferred from a (ϕ, ψ) two-dimensional free-energy calculation, integrating over the relevant basins,

$$\Delta A = -\frac{1}{\beta} \ln \frac{\displaystyle\int_{\mathrm{C_{7ax}}} \mathrm{d}\phi \mathrm{d}\psi \; \mathrm{e}^{-\beta \Delta A(\phi,\psi)}}{\displaystyle\int_{\mathrm{C_{7eq}}} \mathrm{d}\phi \mathrm{d}\psi \; \mathrm{e}^{-\beta \Delta A(\phi,\psi)}} \tag{9}$$

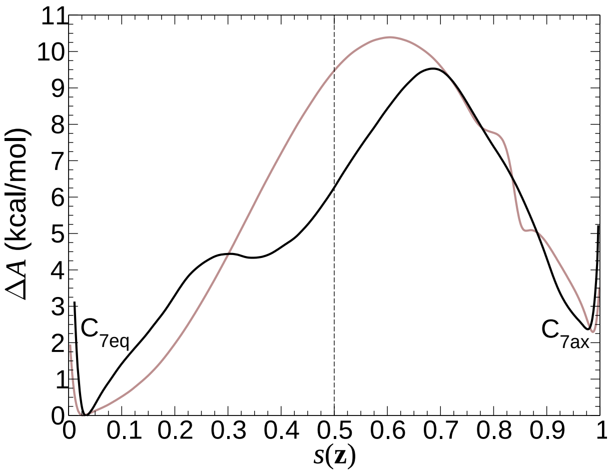


Figure 7: Potentials of mean force obtained along path-collective variable $s(\mathbf{z})$, characterizing the $\mathrm{C_{7eq}}$ to $\mathrm{C_{7ax}}$ conformational transition of NANMA in vacuum. Light curve: The path-collective variable is applied on the guess, initial string, i.e., a rectilinear path in the (ϕ, ψ) subspace shown in Figure 4. Black curve: The path collective variable is applied on the optimized string depicted in Figure 5.

One might wonder what the PMF would look like, were it computed along a non-converged string. To address this question, the reader is invited to repeat the free-energy calculation, using as a reference the guess string of Figure 4, that is the starting point of the optimization procedure. As can be observed in Figure 7, compared to the PMF obtained using the converged string of Figure 5, that corresponding to the rectilinear path in the (ϕ, ψ) subspace — see Figure 4, features a free-energy barrier close to one kcal/mol higher, while the free-energy difference between $\mathrm{C_{7eq}}$ and $\mathrm{C_{7ax}}$ is nearly identical in both cases.

Using strings corresponding to different stages of the optimization procedure, e.g., at $k = 0$, $k = N_{\mathrm{iter}}/2$ and $k = N_{\mathrm{iter}}$, show that the latter is a minimum-action path by repeating the path-collective variable free-energy calculation and estimating the height of the barrier separating the $\mathrm{C_{7eq}}$ to $\mathrm{C_{7ax}}$ conformational states of NANMA.

4. Concluding remarks and extensions of the tutorial

It is not superfluous to reemphasize that this document contains advanced material, which, in glaring contrast with introductory tutorials, requires that the reader has grasped the theoretical underpinnings of the methodology to perform the proposed string and free-energy calculations with utmost efficiency. The string method with swarms of trajectories, even for as mundane a molecular object as NANMA, is computationally demanding and requires appropriate resources, in the form of commodity clusters. Furthermore, contrary to introductory tutorials, the archive does not include all the files, input and output, but only the bare necessities for an end-user proficient with NAMD and VMD to complete the present study gracefully.

As has been seen throughout this tutorial, obtaining the final free-energy difference between the $\mathrm{C}_{7\mathrm{eq}}$ to $\mathrm{C}_{7\mathrm{ax}}$ conformations of NANMA from a guess path connecting in linear fashion the two basins in the $(\phi,\ \psi)$ subspace is a multistep process, involving a number of tunable parameters likely to impact the ultimate result, alongside the convergence of the calculations. As a possible extension of this tutorial, the reader is invited to explore the role played by these different parameters in the PMF underlying the conformational transition. Here, we suggest those parameters that ought to be examined with appropriate care,

– The number of images, M, forming the string. How coarse can the string be without sacrificing the reliability of the final free-energy difference?

– The number of iterations, N_{iter}, of the string optimization procedure. It is crucial that the reader verify convergence of the string prior to initiating the free-energy calculation along the path-collective variable.

– The number of trajectories forming the swarms. Will an increase of this number provide a more accurate estimate of the drift at each image?

– The force constant of the spring connecting the fictitious particle to the transition coordinate in the eABF algorithm. To which extent does it affect the PMF? The reader will ensure that the distribution of the variation between the trajectories of the fictitious particle and that of the path-collective variable obeys a normal law.

– The inverse of the mean-square displacement, λ, between consecutive images. How does this parameter impact sampling of the transition coordinate? In particular, how does it truncate sampling?

– The minimum number of images involved in the path-collective variable free-energy calculation. Compared to M, what is the minimum number of images taken into account that still guarantees an accurate reproduction of the PMF?

Appendix: Archive

This tutorial is provided with all the files necessary towards the calculation of the free-energy change along the most probable pathway that connects the $\mathrm{C}_{7\mathrm{eq}}$ and $\mathrm{C}_{7\mathrm{ax}}$ conformational states of NANMA. The archive is organized in two directories, namely `String with Swarms of Trajectories` and `Path Collective Variables`. The first directory gathers all the files required for the string calculation of the minimum-action path in the $(\phi,\ \psi)$ subspace. The second directory contains all the files required for the free-energy calculation along the optimized transition pathway, utilizing path-collective variable. Directory `String with Swarms of Trajectories` features four subdirectories. `prep` handles the generation of the guess transition pathway, utilized subsequently in `parallel` and `sequential`, the subdirectories where the string optimization is performed in a parallel or in a sequential fashion, respectively. Subdirectory `toolkit` contains scripts for setting up and analyzing the string calculation. Directory `Path Collective Variables` gathers the different files necessary for conducting the free-energy calculation along the optimized string, which is provided in the form of a collection of PDB files. As mentioned previously, the path-collective variable is encoded as a scripted function, supplied in the distribution, alongside with the `colvars` components.

Caveat emptor. Though the tutorial includes all the relevant files needed to perform the different free-energy calculations described herein, the reader is strongly advised to not use these files blindly, without checking first their contents.

References

[1] Pan, A. C.; Sezer, D.; Roux, B., Finding transition pathways using the string method with swarms of trajectories, *J. Phys. Chem. B* **2008**, *112*, 3432–3440.

[2] Maragliano, L.; Vanden-Eijnden, E., A temperature accelerated method for sampling free energy and determining reaction pathways in rare events simulations, *Chem. Phys. Lett.* **2006**, *426*, 168–175.

[3] Bolhuis, P. G.; Dellago, C.; Chandler, D., Reaction coordinates of biomolecular isomerization, *Proc. Natl. Acad. Sci. U. S. A.* **2000**, *97*, 5877–5882.

[4] E, W.; Ren, W.; Vanden-Eijnden, E., Transition pathways in complex systems: Reaction coordinates, isocommittor surfaces, and transition tubes, *Chem. Phys. Lett.* **2005**, *413*, 242–247.

[5] Darve, E.; Pohorille, A., Calculating free energies using average force, *J. Chem. Phys.* **2001**, *115*, 9169–9183.

[6] Comer, J.; Gumbart, J. C.; Hénin, J.; Lelièvre, T.; Pohorille, A.; Chipot, C., The adaptive biasing force method: Everything you always wanted to know, but were afraid to ask, *J. Phys. Chem. B* **2015**, *119*, 1129–1151.

[7] Branduardi, D.; Gervasio, F. L.; Parrinello, M., From A to B in free energy space, *J. Chem. Phys.* **2007**, *126*, 054103.

[8] Hénin, J.; Fiorin, G.; Chipot, C.; Klein, M. L., Exploring multidimensional free energy landscapes using time-dependent biases on collective variables, *J. Chem. Theor. Comput.* **2010**, *6*, 35–47.

[9] Fiorin, G.; Klein, M. L.; Hénin, J., Using collective variables to drive molecular dynamics simulations, *Mol. Phys.* **2013**, *111*, 3345–3362.

[10] E, W.; Ren, W.; Vanden-Eijnden, E., String method for the study of rare events, *Phys. Rev. B* **2002**, *66*, 052301.

[11] E, W.; Ren, W.; Vanden-Eijnden, E., Simplified and improved string method for computing the minimum energy paths in barrier-crossing events, *J. Chem. Phys.* **2007**, *126*, 164103.

[12] Bolhuis, P. G.; Chandler, D.; Dellago, C.; Geissler, P., Transition path sampling: Throwing ropes over mountain passes, in the dark, *Ann. Rev. Phys. Chem.* **2002**, *59*, 291–318.

[13] Fu, H.; Shao, X.; Chipot, C.; Cai, W., Extended adaptive biasing force algorithm. An on–the–fly implementation for accurate free–energy calculations, *J. Chem. Theory Comput.* **2016**, *12*, 3506–3513.

[14] Zheng, L.; Yang, W., Practically efficient and robust free energy calculations: Double-integration orthogonal space tempering, *J. Chem. Theor. Comput.* **2012**, *8*, 810–823.

[15] Lesage, Adrien; LeliÃ¨vre, Tony; Stoltz, Gabriel; Hénin, Jérôme, Smoothed biasing forces yield unbiased free energies with the extended-system adaptive biasing force method., *J. Phys. Chem. B* **2017**, *121*, 3676–3685.

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

[17] Rosso, L.; Abrams, J. B.; Tuckerman, M. E., Mapping the backbone dihedral free-energy surfaces in small peptides in solution using adiabatic free-energy dynamics., *J Phys Chem B* **Mar 2005**, *109*, 4162–4167.