

# **PLUMED User's Guide**

*Complement for the dAFED and UFED methods*

---

**Version 1.3 – dAFED 0.7 - July 2012**

# Chapter 1

## Running free-energy simulations

### 1.1 Driven Adiabatic Free Energy Dynamics (dAFED)

The driven adiabatic free energy (dAFED) algorithm [1] is also called temperature accelerated molecular dynamics (TAMD) by other authors [2]. In fact, dAFED/TAMD is an improvement over the earlier AFED method [3, 4], which required cumbersome coordinates transformations. In the dAFED/TAMD method, an extra dynamical variable  $S$  is coupled to a collective variable  $s(\mathbf{r})$ , where  $\mathbf{r}$  represents the coordinates of a number  $N$  of atoms in the system. The coupling is mediated by a potential energy function with harmonic constant  $\kappa$ ,

$$V(S, s(\mathbf{r})) = \frac{1}{2}\kappa(S - s(\mathbf{r}))^2. \quad (1.1)$$

The dynamics of the  $S$  meta-variable is adiabatically decoupled from the dynamics of the underlying physical system by choosing a large mass  $m_S \gg \bar{m}$ , where  $\bar{m}$  is a typical mass of the physical system. Thanks to the adiabatic separation, a temperature  $T_S > T$  can be assigned to the  $S$  meta-variable. With this choice of  $m_S$  and  $T_S$ , the physical system will evolve fast at room temperature  $T$  around the instantaneous value of  $s(\mathbf{r}) = S$ . On the other hand,  $S$  will evolve slowly, but have a temperature large enough to drive the system over high free energy barriers.

In the limit of  $\kappa \rightarrow \infty$ , it can be shown that the free energy surface at temperature  $T$  can be recovered from the density  $\rho^{\text{adb}}(S)$  sampled at temperature  $T_S$  during the adiabatic dAFED simulation using

$$G(S) = -k_B T_S \log(\rho^{\text{adb}}(S)). \quad (1.2)$$

This result generalizes well to the case where more than one collective variable is used and  $G(S)$  is a multi-dimensional free energy surface.

The dAFED method requires very efficient thermostating of the meta-variable  $S$ . In the present implementation,  $S$  is coupled to a Generalized Gaussian Moment Thermostat (GGMT) [5] or to a Langevin thermostat [6]. If multiple reaction coordinates are used, one separate GGMT or Langevin thermostat is associated to each of them.

In the case of GGMT, the meta-variable is coupled to two thermostating variables  $p_\eta$  and  $p_\zeta$ , with associated masses  $Q_\eta$  and  $Q_\zeta$ , respectively. Given a typical time scale  $\tau$  of the thermostated system, optimal masses are  $Q_\eta = k_B T_S \tau^2$  and  $Q_\zeta = \frac{8}{3}(k_B T_S)^3 \tau^2$ . The order-2 GGMT dynamics for one degree of freedom is

$$\dot{p}_S = V(S, s(\mathbf{r})) - \frac{p_\eta}{Q_\eta} p_S - \frac{p_\zeta}{Q_\zeta} \left[ k_B T_S + \frac{1}{3} \frac{p_S^2}{m_S} \right] p_S, \quad (1.3)$$

$$\dot{p}_\eta = \frac{p_S^2}{m_S} - k_B T_S, \quad (1.4)$$

$$\dot{p}_\zeta = \frac{1}{3} \left[ \frac{p_S^2}{m_S} \right]^2 - (k_B T_S)^2, \quad (1.5)$$

$$\dot{S} = \frac{p_S}{m_S}, \quad \dot{\eta} = \frac{p_\eta}{Q_\eta}, \quad \dot{\zeta} = \frac{p_\zeta}{Q_\zeta}. \quad (1.6)$$

The implemented integrator for the dynamics above is based on a Trotter decomposition of the corresponding Liouville operator [1]. The quality of the integration can be monitored using the quantity  $H_S$ , which would be conserved if the dynamics of  $S$  was decoupled from the physical system,

$$H_S(S, p_S, \eta, p_\eta, \zeta, p_\zeta) = \frac{p_S^2}{2m_S} + V(S) + \frac{p_\eta^2}{2Q_\eta} + \frac{p_\zeta^2}{2Q_\zeta} + k_B T_S (\eta + \zeta) \quad (1.7)$$

The heat transfer  $W_S$  from the meta-variable to the physical system can be calculated,

$$W_S = \int_0^t dt' \kappa(S - s(\mathbf{r})) \frac{p_S}{m_S} \quad (1.8)$$

The effective adiabaticity of the coupling can thus be asserted. In addition, for each extended variable  $S$ , the quantity  $H_S + W_S$  should be strictly conserved, which provides a quality check for the simulation. In addition, Let  $\mathcal{H}$  be the pseudo-energy of the physical system including the associated thermostats and barostats. Then the total energy of the simulation,  $\mathcal{H} + \sum_{j=1}^d H_{S_j}$ , should be conserved as well. As a corollary, considering the physical system only, the quantity  $\mathcal{H} - \sum_{j=1}^d W_{S_j}$  should be conserved.

A different way to assess the adiabaticity of the simulation is through the configurational temperature of the collective variable  $s$ ,

$$T_{\text{config}} = \frac{1}{k_B} \frac{\langle |\nabla U(s)|^2 \rangle}{\langle \nabla^2 U(s) \rangle} = \frac{\kappa}{k_B} \langle (s - S)^2 \rangle. \quad (1.9)$$

For a system at equilibrium the configurational temperature should be equivalent to the kinetic temperature and the heat bath temperature. A higher configurational temperature is a signature of nonequilibrium dynamics, in which a significant heat flow would take place between extended and physical system.

In practice, the choice of  $m_S$  is subject to some pragmatic considerations. The value of  $m_S$  should be as high as possible to ensure good adiabatic separation. However, given the limited planned simulation time, the evolution of  $S$  has to be fast enough to correctly sample the CV range of interest. By running a short dAFED simulation and plotting the evolution of  $S$ , one can estimate an average diffusion speed. From that, a maximum admissible value for  $m_S$  can be deduced, such that  $S$  can cross many time the CV range during the simulation.

The choice of the coupling constant  $\kappa$  determines the resolution of the observed free energy surface  $G(S)$ . Ideally,  $\kappa$  should be very large, but its value is limited by the requirement of integrating accurately the coupling term, Eq. 1.1. The typical period of that harmonic oscillator is given by  $\tau = 2\pi\sqrt{\mu/\kappa}$ , where  $\mu = m_S m_{\text{eff}} / (m_S + m_{\text{eff}})$  is the reduced mass, with  $m_{\text{eff}}$  the effective mass of the CV  $s(r)$ . For a one-dimensional CV,  $m_{\text{eff}}$  can be expressed as

$$m_{\text{eff}} = \left[ \sum_{j=1}^{3N} \left( \frac{ds}{dr_j} \right)^2 \frac{1}{m_j} \right]^{-1}. \quad (1.10)$$

Note that if the CV is multidimensional the situation is slightly more complicated and  $m_{\text{eff}}$  is in fact a tensor, whose diagonal elements are given

by Eq. 1.10. Nevertheless,  $m_{\text{eff}}$  can be used to estimate an order of magnitude for the period  $\tau$  for each CV, and thus estimate which time step is appropriate for a given  $m_S$  (or vice versa). It is recommended to check that the time step is appropriate for the chosen  $\kappa$  by plotting the evolution of  $s(r)$  and  $S$  from a short simulation in which the dAFED variables are saved very frequently.

### 1.1.1 Input for dAFED

For each CV, a DAFED directive is used to define the parameters of the corresponding dynamics. On the same line, the number of the CV to which the directive applies is specified after the keyword `CV`. The temperature  $T_S$  in K is given after keyword `TEMPERATURE`. The thermostat time constant  $\tau$  is given in ps after keyword `TAUTHERMO`. The mass  $m_S$  and harmonic constant  $\kappa$ , are given after the keywords `MASS` and `KAPPA`, respectively. The units of  $\kappa$  and  $m_S$  depend on the nature of the CV. They should always be such that  $\kappa S^2$  and  $m\dot{S}^2$  are both in units of energy (kJ/mol= amu nm<sup>2</sup> / ps<sup>2</sup>), see the example below.

In addition, two optional keywords can be used with the DAFED directive. First, for periodic CVs such as torsion angles, the  $S$  variable should also evolve on a periodic interval. This is specified by the keyword `PERIODIC`, followed by two numbers for the lower and upper bounds. The numbers can be replaced by `MINUS_PI`, `PLUS_PI`, or `PLUS_2PI` to specify  $-\pi$ ,  $+\pi$ , or  $+2\pi$ , respectively.

The optional keyword `JACOBIAN_FORCE` causes a bias force  $F = -2k_B T/S$  to be applied to the dynamics of  $S$ . This is useful with distance CVs in order to counterbalance the effect of the Jacobian factor and sample a more uniform distribution along the CV.

**Example.**

The following lines couple CV 1 (a distance in nm) to a meta-variable of mass  $10^5$  amu with a harmonic constant of  $10^6$  kJ/mol/nm<sup>2</sup> and CV 2 (a unitless number) to a meta-variable with mass  $10^3$  amu\*nm<sup>2</sup> with a harmonic constant of  $10^4$  kJ/mol. For both CV, the dAFED temperature is 600 K and the GGMT thermostat time constant is 0.2 ps. See text for the optional keywords `JACOBIAN_FORCE` and `PERIODIC`.

```
DISTANCE LIST 1 34
TORSION LIST 5 15 29 36

DAFED CV 1 TEMPERATURE 600 MASS 1e5 KAPPA 1e6 TAUTHERMO 0.2 JACOBIAN_FORCE
DAFED CV 2 TEMPERATURE 600 MASS 1e3 KAPPA 1e4 TAUTHERMO 0.2 PERIODIC MINUS_PI PLUS_PI

DAFED_CONTROL RESTART checkpoint_file WRITE_STATE -1 N_RESPA 1

PRINT W_STRIDE 100
ENDMETA
```

A separate `DAFED_CONTROL` directive contains general controls for the dAFED simulation. The dAFED dynamics, including all variables described in Eqs. 1.3 - 1.6 can be restarted exactly from a previous run using a checkpoint file. Following the `WRITE_STATE` keyword appears the number of steps after which a checkpoint file is saved. A value of  $-1$  implies that a checkpoint file is written only when GROMACS saves its own checkpoint file, i.e. at regular wall clock time intervals. The checkpoint file is saved in the current directory with default name `DAFED.STATE`. The optional keyword `RESTART` is used to specify the path to the checkpoint file from which to restart.

The integrator for  $S$  can be selected on the `DAFED_CONTROL` line with the keyword `INTEGRATOR`, followed by either `GGMT` (default) or `LANGEVIN_EM` for Langevin evolution with the simple Euler-Maruyama integrator, or `LANGEVIN_CV` for Langevin evolution with the integrator of Ciccotti-Vanden Eijnden [6]. Specifying `LANGEVIN` defaults to `LANGEVIN_EM`.

With a high value of  $\kappa$  (as required by the dAFED method), oscillations of  $s(\mathbf{r})$  can become faster than the fastest mode in the physical system. This would in principle require choosing a smaller time step, at the expense of sampling efficiency. Instead, following a multiple time step approach, the dAFED force can be integrated more often than the forces in the physical system. This feature is implemented only with GROMACS and the GGMT thermostat. The user has to divide the general MD time step by a number  $N_{\text{RESPA}}$  (typically between 2 and 10). The optional keyword `N_RESPA` followed by the number  $N_{\text{RESPA}}$  in the `DAFED_CONTROL` directive instructs GROMACS to evaluate the physical forces only every  $N_{\text{RESPA}}$  steps. Together with the `md-vv` integrator of GROMACS, this should produce a correct RESPA [7]

scheme in the NVT ensemble. With this, `nstcalcenergy = 1` has to be set in the GROMACS input file. Note that this feature is still experimental and energy conservation should be checked.

### 1.1.2 Typical output for dAFED

With the dAFED method, the COLVAR file will contain the following data, if  $d$  collective variables are used:

- time step
- value of the collective variable  $s_1(\mathbf{r})\dots s_d(\mathbf{r})$

Then for each of the  $S_j$ ,  $j = 1\dots d$ , appears a set of 5 columns with :

- the meta-variable  $S$
- the instantaneous temperature of  $S$  in K
- the conserved quantity  $H_S$ , see Eq. 1.7, in kJ/mol
- the work  $W_S$  from  $S$  to the physical system, see Eq. 1.8, in kJ/mol
- the effective mass  $m_{\text{eff}}$  according to Eq. 1.10, in a.m.u.

These fields are labeled `sj`, `T_sj`, `E_sj`, `W_sj`, and `M_effj` respectively, in the COLVAR header line,  $j = 1\dots d$ . Additional collective variables can be monitored during a dAFED run, in which case more columns will appear before the first set of dAFED fields. For long production runs, the user can choose a more compact form of output in which only the CV  $s_j(\mathbf{r})$  and the position  $S_j$  of each corresponding extended variable are printed. This is achieved by adding the keyword `PRINT_NO_DETAILS` on the `DAFED_CONTROL` input line.

### 1.1.3 Unified Free Energy Dynamics (UFED)

UFED is a recent extension [8] of the dAFED/TAMD method which allows to combine the benefits of high temperature extended variables with adaptive bias potentials similar to those used in metadynamics. The UFED method rests on the fact that the free energy surface can be reconstructed from the

thermodynamic force  $\mathcal{F}(S)$  instead of the histogram  $\rho^{\text{adb}}(S)$ . Indeed, in the spirit of the well-known thermodynamic integration technique, we can write

$$\mathcal{F}(S) = -\nabla_S G_\beta(S) \quad (1.11)$$

$$= \frac{1}{Z(S)} \int d\mathbf{r} \kappa(s(\mathbf{r}) - S) e^{-\beta[U(\mathbf{r}) + \frac{\kappa}{2}(s(\mathbf{r}) - S)^2]} \quad (1.12)$$

$$= \langle f_{\mathbf{r} \rightarrow S} \rangle_S \quad (1.13)$$

The last line represents the force exerted by the physical system on the extended variable, averaged at a fixed position of  $S$ . This average can easily be obtained in a post processing phase from the values of  $S$  and  $s(\mathbf{r})$  stored in the COLVAR file using a grid in the  $S$  space. Note that due to the fast oscillations of  $s(\mathbf{r})$ , samples should be collected at high frequency. Finally,  $\mathcal{F}(S)$  is integrated numerically to get the free energy profile. In dimensions greater than one,  $\mathcal{F}(S)$  will not exactly be a consistent multidimensional gradient, due to statistical noise. The PMF can however easily be reconstructed as the surface  $G_\beta(S)$  whose discrete derivative best fits  $\mathcal{F}(S)$  in the least-squares sense. This postprocessing step provides the additional benefit of producing a smooth PMF.

The second fundamental ingredient for the UFED method is that, if the adiabatic separation is effective,  $\mathcal{F}(S)$  does not depend on the actual distribution of  $s$  [8]. If this holds, we can introduce a bias potential of any kind acting on  $S$ . We introduce a Gaussian-based adaptive potential

$$V_{\text{bias}}(S, t) = h \sum_{k\tau < t} \exp \left( - \sum_{i=1}^d \frac{(S - S^{(0)}(k\tau))^2}{2\sigma^2} \right), \quad (1.14)$$

which is similar to the metadynamics bias potential, except that it acts on the extended variables  $S$  instead of the CV  $s(\mathbf{r})$ . UFED has several advantages over its parent methods. First, the adaptive bias allows using a lower temperature than dAFED, which facilitates obtaining effective adiabatic separation. Second, the use of the force to construct the free energy surface (instead of the sum of hills in metadynamics) makes the final accuracy of UFED independent of the hill size, and does not require that all basins are filled up with hills.

In order to activate UFED, the user only needs to add a line with the directive `UFED_HILLS` in addition to the `DAFED` and `DAFED_CONTROL` directives described above (see example below). For most aspects, `UFED_HILLS` works

just as the HILLS directive of metadynamics. It must be followed by a keyword HEIGHT after which the value of  $h$  is specified, see Eq. (1.14). The hill deposition stride is specified after the keyword W\_STRIDE. The Gaussian widths  $\sigma_i$  are taken from the keywords SIGMA specified on the line of each CV,  $i = 1, \dots, d$ .

Some collective variables have intrinsic domain limitations (such as the number of H-bonds that cannot be smaller than zero), or the user might want to impose limitations (such as the maximum distance to which a ligand can be separated from its host). In these cases, it is useful to impose the limitations to the domain of the  $S$  variables, which are otherwise unbounded. This is especially necessary when a bias potential is used. One way to do this without perturbing the distribution of  $S$  within the range of interest is to use reflective walls at which the momentum  $P_S$  is inverted.

Reflective walls are activated with the directives LREFLECT and UREFLECT, corresponding to a lower or upper limit, respectively. The CV on which the reflective wall acts is specified after the keyword CV and the limit value is given after the keyword LIMIT. If UFED\_HILLS and LREFLECT or UREFLECT are active, extra hills are added at a symmetrical position on the other side of the wall as soon as  $S_i$  is closer than  $3\sigma_i$  to the wall. This prevents the formation of an artificial ditch in the bias potential close to the wall [9].

**Example.**

The following example shows how to setup a UFED run. CV 1 (a distance in nm) is coupled with a harmonic constant of  $10^5$  kJ/mol/nm<sup>2</sup> to a meta-variable of mass  $10^6$  a.m.u. at temperature 400 K. In this case we have specified a Langevin integrator for this meta-variable and we print a compact COLVAR file. For UFED, we deposit every 1000 steps a hill of height 0.5 kJ/mol and width 0.1 nm. In addition, we have restricted the space of the meta-variable with a lower reflection wall at 0.1 nm and an upper reflection wall at 1.5 nm.

```
DISTANCE LIST 1 2 SIGMA 0.1

DAFED CV 1 TEMPERATURE 400 MASS 1e6 KAPPA 1e5 TAUTHERMO 0.5
DAFED.CONTROL WRITE_STATE -1 INTEGRATOR LANGEVIN PRINT_NO_DETAILS

UFED_HILLS HEIGHT 0.5 W_STRIDE 1000

LREFLECT CV 1 LIMIT 0.1
UREFLECT CV 1 LIMIT 1.5

PRINT W_STRIDE 100
ENDMETA
```

We note that, in addition to  $G(S)$ , it is possible to calculate the ensemble average of any observable  $A(\mathbf{r})$  in during a dAFED or UFED simulation, even

if the distribution of states is different from the canonical ensemble at low (physical) temperature. It can be shown [10] that if the adiabatic separation is effective, we have

$$\langle A \rangle = \frac{\int ds \langle A \rangle_S e^{-\beta G(S)}}{\int ds e^{-\beta G(S)}}, \quad (1.15)$$

where  $\langle A \rangle_S$  is the average value of  $A$  accumulated on a grid of fixed  $S$  positions. Here,  $G(S)$  is obtained from the same simulation by integrating values of  $\langle f_{\mathbf{r} \rightarrow s} \rangle_S$  accumulated on the same grid. The integral in Eq. 1.15 is performed numerically *a posteriori*. Using Eq. 1.15, dAFED can for example be combined [10] with thermodynamic integration ( $A = dH/d\lambda$ ) or free energy perturbation ( $A = \exp[-\beta \Delta H(\lambda)]$ ) to calculate alchemical free energy differences in flexible molecules.

# Bibliography

- [1] J. B. Abrams, M. E. Tuckerman, Efficient and direct generation of multidimensional free energy surfaces via adiabatic dynamics without coordinate transformations, *J. Phys. Chem. B* 112 (2008) 15742–15757.
- [2] L. Maragliano, E. Vanden-Eijnden, A temperature accelerated method for sampling free energy and determining reaction pathways in rare events simulations, *Chem. Phys. Lett.* 426 (2006) 168–175.
- [3] L. Rosso, M. E. Tuckerman, An adiabatic molecular dynamics method for the calculation of free energy profiles, *Mol. Simul.* 28 (1) (2002) 91–112.
- [4] L. Rosso, P. Minari, Z. Zhu, M. E. Tuckerman, On the use of adiabatic molecular dynamics to calculate free energy profiles, *J. Chem. Phys.* 116 (2002) 4389–4402.
- [5] Y. Liu, M. E. Tuckerman, Generalized Gaussian moment thermostating: A new continuous dynamical approach to the canonical ensemble, *J. Chem. Phys.* 112 (4) (2000) 1685–1700.
- [6] E. Vanden-Eijnden, G. Ciccotti, Second-order integrators for Langevin equations with holonomic constraints, *Chem. Phys. Lett.* 429 (1-3) (2006) 310–316.
- [7] G. J. Martyna, M. E. Tuckerman, D. J. Tobias, M. L. Klein, Explicit reversible integrators for extended systems dynamics, *Mol. Phys.* 87 (5) (1996) 1117.
- [8] M. Chen, M. A. Cuendet, M. E. Tuckerman, Heating and flooding: A unified approach for rapid generation of free energy surfaces, *J. Chem. Phys.* 137 (2012) 024102.

- [9] Y. Crespo, F. Marinelli, F. Pietrucci, A. Laio, Metadynamics convergence law in a multidimensional system, *Physical Review E* 81 (5) (2010) 55701.
- [10] M. A. Cuendet, M. E. Tuckerman, Alchemical free energy differences in flexible molecules from thermodynamic integration or free energy perturbation combined with driven adiabatic dynamics, *J. Chem. Theory Comput.* (2012) in press, doi:10.1021/ct300090z.