

Thermostating in Molecular Dynamics Simulations

1 Velocity Verlet Integrator

The velocity Verlet method has the form [1]

$$\mathbf{r}_i(t + \delta t) = \mathbf{r}_i(t) + \mathbf{v}_i(t)\delta t + \mathbf{a}_i(t)\frac{\delta t^2}{2} + \mathcal{O}(\delta t^3) \quad (1)$$

$$\mathbf{v}_i(t + \delta t) = \mathbf{v}_i(t) + [\mathbf{a}_i(t) + \mathbf{a}_i(t + \delta t)]\frac{\delta t}{2} + \mathcal{O}(\delta t^3) \quad , \quad (2)$$

where $\mathbf{a}_i = \mathbf{F}_i/m_i$, with the mass, m_i , and the force, \mathbf{F}_i of particle i . The forces are derived from the potential which is a function of the positions of an N -particle system. For simplicity, in the following the problem is reduced to one cartesian coordinate only and, hence, the index j is omitted. This algorithm only requires storage of the positions, \mathbf{r}_i , velocities, \mathbf{v}_i , and accelerations, \mathbf{a}_i . Initially, the new positions at time $t + \delta t$ are calculated using Eq. (1) and the velocities at mid-step are computed using

$$\mathbf{v}_i(t + \delta t/2) = \mathbf{v}_i(t) + \mathbf{a}_i(t)\frac{\delta t}{2} \quad (3)$$

Then the forces and accelerations at time $t + \delta t$ are computed and the velocity step is completed with

$$\mathbf{v}_i(t) = \mathbf{v}_i(t + \frac{\delta t}{2}) + \mathbf{a}_i(t + \delta t)\frac{\delta t}{2} \quad (4)$$

At this point, the kinetic energy at time $t + \delta t$ is available. The method uses $9N$ words of storage and is numerical stable, convenient and simple. A pseudocode for performing velocity Verlet dynamics is

1. evolve velocities: $\mathbf{v}_i(t + \delta t/2) = \mathbf{v}_i(t) + \mathbf{a}_i(t)\delta t/2$
2. evolve positions:
 $\mathbf{r}_i(t + \delta t) = \mathbf{r}_i(t) + \mathbf{v}_i(t)\delta t + \mathbf{a}_i(t)\delta t^2/2 = \mathbf{r}_i(t) + \mathbf{v}_i(t + \delta t/2)\delta t$
3. apply the SHAKE procedure if constraints on internal coordinates are present
4. call force $\rightarrow \mathbf{a}_i(t + \delta t)$
5. evolve velocities: $\mathbf{v}_i(t + \delta t) = \mathbf{v}_i(t + \delta t/2) + \mathbf{a}_i(t + \delta t)\delta t/2$
6. apply the RATTLE procedure if constraints on internal coordinates are present

2 Thermostats

In the simplest form of reducing the kinetic energy periodically in a simulated annealing molecular dynamics is to multiply the nuclear velocities by a scaling factor. This approach does work but it has certain disadvantages [2]. The kinetic energy can vary widely over the course of a relatively small amount of simulation time, making the exact amount of kinetic energy lost in a given cooling step essentially random, since it is proportional to the instantaneous kinetic energy. Furthermore, as the particles move to parts of the potential energy surface (PES) which are lower than the instantaneous potential energy, they gain a corresponding amount of kinetic energy as long as the total energy is kept constant. The kinetic energy gained in this manner may be large compared with that possessed by the system before. The result is a ‘runaway’ system which has a temperature much higher than is desired. This runaway behavior is actually quite common at the beginning of a trajectory if a random starting structure is chosen because in the region of interest (bound structures), potential energy surfaces generally tend to be steep over a larger area than they are flat. Both of these features of the velocity scaling approach to simulate annealing make controlling the annealing process difficult. Another disadvantage is that there are no well-defined, conserved quantities.

A temperature based approach to simulated annealing, however, does not suffer from these deficiencies. The temperature is treated explicitly and can be controlled directly instead of indirectly as in the case of velocity scaling.

The entire system treated consists of N real particles (e.g. atoms) and bath particles. The equations of motion (EOM) of the real particles

$$\dot{\mathbf{p}}_i(\mathbf{r}) = \mathbf{F}_i(\mathbf{r}) \quad (5)$$

are extended by an additional term [3]

$$\dot{\mathbf{p}}_i(\mathbf{r}) = \mathbf{F}_i(\mathbf{r}) - \dot{\xi} \mathbf{p}_i(\mathbf{r}) \quad , \quad (6)$$

where \mathbf{r}_i are the positions of the real particles and \mathbf{p}_i are the corresponding momenta. The dot denotes that the corresponding parameters are the first derivative with respect to time. The time development of the positions is unchanged, viz

$$\dot{\mathbf{r}}_i = \frac{\mathbf{p}_i}{m_i} \quad . \quad (7)$$

The second term in Eq. (6) couples the real particles to the bath and contains the first derivative of a thermodynamic friction with respect to time.

2.1 Gaussian Thermostat

One way to define a temperature is to apply Gauss’ principle of nonholonomic constraints [4, 5]. In the nonholonomic case the constraints are given by a function

$g(\mathbf{r}, \mathbf{v}, t) = 0$. The constraint is nonholonomic because it includes velocities. The relation restricting the accelerations, \mathbf{a} , is obtained by a single differentiation of g with respect to time.

$$\begin{aligned}\frac{\partial g(\mathbf{r}_i, \mathbf{v}_i, t)}{\partial t} &= \frac{\partial g}{\partial \mathbf{r}_i} \frac{\partial \mathbf{r}_i}{\partial t} + \frac{\partial g}{\partial \mathbf{v}_i} \frac{\partial \mathbf{v}_i}{\partial t} + \frac{\partial g}{\partial t} \\ &= \frac{\partial g}{\partial \mathbf{r}_i} \mathbf{v}_i + \frac{\partial g}{\partial \mathbf{v}_i} \mathbf{a}_i + \frac{\partial g}{\partial t}\end{aligned}\quad (8)$$

The constant temperature constraint has the form

$$g(\mathbf{r}, \mathbf{v}, t) = \frac{1}{2} \sum_{i=1}^N m_i \mathbf{v}_i^2 - \frac{1}{2} N_f k_B T_{set} = 0 \quad (9)$$

and is nonlinear. In Eq. (9), N_f are the degrees of freedom, k_B is the Boltzmann constant and T_{set} is the desired temperature. Gauss' principle yields

$$\sum_{i=1}^N m_i \mathbf{v}_i \mathbf{a}_i = \sum_{i=1}^N \mathbf{F}_i \mathbf{v}_i = 0 \quad . \quad (10)$$

To derive the Gaussian EOM with the nonholonomic constraint condition, $m_i \mathbf{a}_i$ is substituted by $\mathbf{F}_i - \dot{\xi} m_i \mathbf{v}_i$. The resulting equation is then solved for the time derivative of the friction coefficient, $\dot{\xi}$, which yields

$$\dot{\xi} = \frac{\sum_{i=1}^N \mathbf{F}_i \cdot \mathbf{v}_i}{\sum_{i=1}^N m_i \mathbf{v}_i^2} \quad (11)$$

The conserved energy associated with Eq. (6) and Eq. (11) is simply

$$\mathcal{E}_{Gauss} = \frac{1}{2} \sum_i m_i v_i^2 + E_{pot}[\{\mathbf{r}_i\}] \quad (12)$$

The Gaussian thermostat can be easily combined with the velocity Verlet integrator for the EOM. The computational procedure is the following:

1. calculate the thermostat variable $\dot{\xi}(t) = [\sum_{i=1}^N m_i \mathbf{a}_i(t) \cdot \mathbf{v}_i(t)] / [\sum_{i=1}^N m_i \mathbf{v}_i^2(t)]$
2. evolve velocities: $\mathbf{v}_i(t + \delta t/2) = \mathbf{v}_i(t) + [\mathbf{a}_i(t) - \mathbf{v}_i(t) \dot{\xi}(t)] \delta t/2$
3. evolve positions: $\mathbf{r}_i(t + \delta t) = \mathbf{r}_i(t) + \mathbf{v}_i(t + \delta t/2) \delta t$
4. apply the SHAKE procedure if constraints on internal coordinates are present
5. call force $\rightarrow \mathbf{a}_i(t + \delta t) = \mathbf{F}_i(t + \delta t) / m_i$
6. evolve particle velocities
 $\mathbf{v}_i(t + \delta t) = \mathbf{v}_i(t + \delta t/2) + [\mathbf{a}_i(t + \delta t) - \mathbf{v}_i(t + \delta t) \dot{\xi}(t + \delta t)] \delta t/2$

7. apply the RATTLE procedure if constraints on internal coordinates are present

8. evolve thermostat velocity

$$\dot{\xi}(t + \delta t) = [\sum_{i=1}^N \mathbf{F}_i(t + \delta t) \cdot \mathbf{v}_i(t + \delta t)] / [\sum_{i=1}^N m_i \mathbf{v}_i^2(t + \delta t)]$$

The particle velocities, $\mathbf{v}_i(t + \delta t)$, have to be calculated iteratively, e.g., within a Newton-Raphson procedure (Appendix A) by solving

$$\mathbf{h}_i[\mathbf{v}_i, \dot{\xi}] = b d v'_i + \left[\mathbf{a}_i - \dot{\xi} \mathbf{v}_i \right] \frac{\delta t}{2} - \mathbf{v}_i = \mathbf{0} \quad , \quad 1 \leq i \leq N, \quad (13)$$

where $\mathbf{v}'_i = \mathbf{v}_i(t + \delta t/2)$. The argument of \mathbf{v}_i , of \mathbf{a}_i and $\dot{\xi}$, ($t + \delta t$), is dropped for clarity. The Jacobian matrix is diagonal and its elements are

$$J_{jj} = - \left[\frac{a_j m_j v_j - 2 \dot{\xi} v_j^2 m_j}{\sum_{i=1}^N v_j^2 m_j} + \dot{\xi} \right] \frac{\delta t}{2} - 1 \quad . \quad (14)$$

The initial guess of $\dot{\xi}^0(t + \delta t)$ is usually $\dot{\xi}(t + \delta t/2)$.

2.2 Global Nosé-Hoover Thermostat

Within the global Nosé-Hoover (NH) thermostat [6] the bath particle itself is subject of an equation of motion which is simply a function of the kinetic energy of the system and the desired temperature:

$$\ddot{\xi} = \frac{1}{Q} \left[\sum_{i=1}^N m_i \mathbf{v}_i^2 - N_f k_B T_{set} \right] \quad , \quad (15)$$

where N_f is the number of degrees of freedom of the real particles, k_B the Boltzmann constant and T_{set} the desired temperature. Eq. (15) clearly shows the temperature control mechanism. The term on the right-hand side of Eq. (15) is the kinetic energy difference between the instantaneous value and that at the desired temperature. If the instantaneous temperature is higher than the target value, the friction force will increase and vice versa. In Eq. (15), Q is the bath ‘mass’ parameter

$$Q = N_f k_B T_{set} \tau^2 \quad , \quad (16)$$

where τ represents a characteristic time scale for the motions of the real particles [7]. These time scale can be determined by computing the velocity autocorrelation functions of the particles over a short run and looking at the Fourier transform. The frequencies should then be chosen to correspond to the main peak of the Fourier transform. Another good rule of thumb in choosing τ is that the time step will generally need to be roughly a factor of 20-40 smaller than the shortest period in the system.

Thus, τ can be chosen based on multiples of the time step used [8]. However, one must be cautious not to make Q too small because this leads to high-frequency oscillations in $\dot{\xi}$ and, finally, sampling problems [9].

The conserved energy associated with Eq. (6) and Eq. (15) is [7]

$$\mathcal{E}_{NH} = \frac{1}{2} \sum_i m_i \mathbf{v}_i^2 + E_{pot}[\{\mathbf{r}_i\}] + \frac{1}{2} Q \dot{\xi}^2 + N_f k_B T_{set} \xi \quad (17)$$

The velocity Verlet integrator for the EOM, Eq. (5), Eq. (7) and Eq. (15), is [10]

$$\mathbf{r}_i(t + \delta t) = \mathbf{r}_i(t) + \mathbf{v}_i(t) \delta t + \left[\mathbf{a}_i(t) - \mathbf{v}_i(t) \dot{\xi}(t) \right] \frac{\delta t^2}{2} \quad (18)$$

$$\xi(t + \delta t) = \xi(t) + \dot{\xi}(t) \delta t + \ddot{\xi}(t) \frac{\delta t^2}{2} \quad (19)$$

$$\begin{aligned} \mathbf{v}_i(t + \delta t) &= \mathbf{v}_i(t) + \left[\mathbf{a}_i(t) - \mathbf{v}_i(t) \dot{\xi}(t) \right] \frac{\delta t}{2} \\ &+ \left[\mathbf{a}_i(t + \delta t) - \mathbf{v}_i(t + \delta t) \dot{\xi}(t + \delta t) \right] \frac{\delta t}{2} \end{aligned} \quad (20)$$

$$\dot{\xi}(t + \delta t) = \dot{\xi}(t) + \left[\ddot{\xi}(t) + \ddot{\xi}(t + \delta t) \right] \frac{\delta t}{2} . \quad (21)$$

2.2.1 Solve Equations of Motion Iteratively

The procedure for a constant temperature simulation within the NH algorithm using the velocity Verlet integrator is:

1. update thermostat position, $\xi(t)$, velocity, $\dot{\xi}(t)$, and acceleration, $\ddot{\xi}(t)$:
 - $\ddot{\xi}(t) = 1/Q [\sum_i m_i \mathbf{v}_i(t) - N_f k_B T_{set}]$
 - $\dot{\xi}(t + \delta t/2) = \dot{\xi}(t) + \ddot{\xi}(t) \delta t/2$
 - $\xi(t + \delta t) = \xi(t) + \dot{\xi}(t + \delta t/2) \delta t$
2. evolve particle velocities: $\mathbf{v}_i(t + \delta t/2) = \mathbf{v}_i(t) + [\mathbf{a}_i(t) - \mathbf{v}_i(t) \dot{\xi}(t)] \delta t/2$
3. evolve particle positions: $\mathbf{r}_i(t + \delta t) = \mathbf{r}_i(t) + \mathbf{v}_i(t + \delta t/2) \delta t$
4. apply the SHAKE procedure if constraints on internal coordinates are present
5. call force $\rightarrow \mathbf{a}_i(t + \delta t) = \mathbf{F}_i(t + \delta t)/m_i$
6. evolve particle velocities, $\mathbf{v}_i(t + \delta t)$, and the thermostat velocity, $\dot{\xi}(t + \delta t)$, iteratively (see below)
7. apply the RATTLE procedure if constraints on internal coordinates are present

Steps 1 to 5 can be calculated without any difficulty. In step 6, the particle velocities, $\mathbf{v}_i(t + \delta t)$ and the thermostat velocity, $\dot{\xi}(t + \delta t)$, are given by

$$\mathbf{v}_i(t + \delta t) = \mathbf{v}_i(t + \delta t/2) + \left[\mathbf{a}_i(t + \delta t) - \dot{\xi}(t + \delta t)\mathbf{v}_i(t + \delta t) \right] \frac{\delta t}{2} \quad (22)$$

$$\dot{\xi}(t + \delta t) = \dot{\xi}(t + \delta t/2) + \left[\sum_{i=1}^N m_i \mathbf{v}_i^2(t + \delta t) - N_f k_B T_{set} \right] \frac{\delta t}{2Q} \quad (23)$$

In both, Eq. (22) and Eq. (23), $\mathbf{v}_i(t + \delta t)$ and $\dot{\xi}(t + \delta t)$ appear on the right and left hand sides. In most cases, therefore, these equations cannot be integrated exactly. However, it is relatively straightforward to solve Eq. (22) and Eq. (23) numerically [11]. One possible approach to solve Eq. (24) and Eq. (25) is to use the Newton-Raphson scheme as described in Appendix A. The equations which have to be solved are of the form

$$\mathbf{h}_i[\mathbf{v}_i, \dot{\xi}] = \mathbf{v}'_i + \left[\mathbf{a}_i - \dot{\xi} \mathbf{v}_i \right] \frac{\delta t}{2} - \mathbf{v}_i = \mathbf{0} \quad , \quad 1 \leq i \leq N \quad (24)$$

$$h_{N+1}[\mathbf{v}_i, \dot{\xi}] = \dot{\xi}' + \left[\sum_{i=1}^N m_i \mathbf{v}_i^2 - N_f k_B T_{set} \right] \frac{\delta t}{2Q} - \dot{\xi} = 0 \quad , \quad (25)$$

where $\mathbf{v}'_i = \mathbf{v}_i(t + \delta t/2)$ and $\dot{\xi}' = \dot{\xi}(t + \delta t/2)$. The arguments of \mathbf{v}_i and $\dot{\xi}$, $(t + \delta t)$, are dropped for clarity. **NOTE:** \mathbf{h}_i is a vector that consists of three components, each for the cartesian coordinates x , y and z !

The following definitions are introduced: $\kappa_i = \mathbf{v}_i$ for $i = 1, \dots, N$ and $\kappa_i = \dot{\xi}$ for $i = N + 1$. Hence, the elements of the *Jacobian* matrix are

$$J_{jk} = \begin{cases} -1 & j = 3N + 1, k = 3N + 1 \\ m_k v_k \delta t / Q & j = 3N + 1, k \neq 3N + 1 \\ -v_j \delta t / 2 & j \neq 3N + 1, k = 3N + 1 \\ -\dot{\xi} \delta t / 2 - 1 & j \neq 3N + 1, k = j \\ 0 & \text{else} \end{cases} \quad (26)$$

Usually, the initial guess is

$$\dot{\xi}^0(t + \delta t) = \begin{cases} 2\ddot{\xi}(t)\delta t & \text{if } t = 0 \\ \dot{\xi}(t) + 2\ddot{\xi}(t)\delta t & \text{else} \end{cases} \quad (27)$$

$$v_i^0(t + \delta t) = v_i^0(t + \delta t/2) \quad , \quad 1 \leq i \leq 3N \quad (28)$$

Note, if constraints on internal coordinates are present the determination of the velocities is independent of the constraints algorithm used (e.g. RATTLE, SHAKE).¹ This is possible because all particles involved in a common constraint must be coupled to the same thermostat within the present formalism.

¹RATTLE enforces $\sum_i \mathbf{v}_i \cdot \nabla_{\mathbf{r}_i} \sigma_j = 0$ which is independent of a scaling factor on \mathbf{v}_i .

2.2.2 Reversible Velocity Verlet Integrator

In order to derive a reversible integrator (Appendix B) for the NH thermostat combined with the velocity Verlet algorithm the following Liouville operator is applied [12, 13]:

$$\begin{aligned} i\hat{L} &= i\hat{L}_1 + i\hat{L}_2 + i\hat{L}_3 + i\hat{L}_4 + i\hat{L}_5 \\ &= \sum_{i=1}^N \mathbf{v}_i \cdot \frac{\partial}{\partial \mathbf{r}_i} + \sum_{i=1}^N \mathbf{F}_i \cdot \frac{\partial}{\partial \mathbf{p}_i} - \sum_{i=1}^N \dot{\xi} \mathbf{p}_i \cdot \frac{\partial}{\partial \mathbf{p}_i} + \dot{\xi} \cdot \frac{\partial}{\partial \xi} + \ddot{\xi} \cdot \frac{\partial}{\partial \dot{\xi}} \quad . \quad (29) \end{aligned}$$

The approximate discrete time propagator is

$$\begin{aligned} \hat{G}(\delta t) &= \hat{U}_5 \left(\frac{\delta t}{2} \right) \hat{U}_4 \left(\frac{\delta t}{2} \right) \hat{U}_3 \left(\frac{\delta t}{2} \right) \hat{U}_2 \left(\frac{\delta t}{2} \right) \hat{U}_1(\delta t) \\ &\quad \times \hat{U}_2 \left(\frac{\delta t}{2} \right) \hat{U}_3 \left(\frac{\delta t}{2} \right) \hat{U}_4 \left(\frac{\delta t}{2} \right) \hat{U}_5 \left(\frac{\delta t}{2} \right) \end{aligned} \quad (30)$$

where $\hat{U}_i(\tau) = \exp(i\hat{L}_i\tau)$, $\tau = \delta t, \delta t/2$. The resulting algorithm has the following structure [13]:

1. update thermostat velocity $\dot{\xi}(t)$, and thermostat position, $\xi(t)$:
 - $\dot{\xi}(t + \delta t/2) = \dot{\xi}(t) + \ddot{\xi}(t)\delta t/2$
 - $\xi(t + \delta t) = \xi(t) + \dot{\xi}(t + \delta t/2)\delta t$
2. evolve particle velocities:
$$\mathbf{v}_i(t + \delta t/2) = \mathbf{v}_i(t) \times \exp[-\dot{\xi}(t + \delta t/2)\delta t/2] + \mathbf{a}_i(t)\delta t/2$$
3. evolve particle positions:
$$\mathbf{r}_i(t + \delta t) = \mathbf{r}_i(t) + \mathbf{v}_i(t + \delta t/2)\delta t$$
4. apply the SHAKE procedure if constraints on internal coordinates are present
5. call force $\rightarrow \mathbf{a}_i(t + \delta t) = \mathbf{F}_i(t + \delta t)/m_i$
6. evolve particle velocities:
$$\mathbf{v}_i(t + \delta t) = [\mathbf{v}_i(t + \delta t/2) + \mathbf{a}_i(t + \delta t)\delta t/2] \times \exp[-\dot{\xi}(t + \delta t/2)\delta t/2]$$
7. apply the RATTLE procedure if constraints on internal coordinates are present
8. update thermostat velocity:
$$\dot{\xi}(t + \delta t) = \dot{\xi}(t + \delta t/2) + \ddot{\xi}(t + \delta t)\delta t/2$$

The thermostat accelerations, $\ddot{\xi}(t)$, can also be expressed as

$$\ddot{\xi} = \frac{1}{\tau^2} \left[\frac{T(t)}{T_{set}} - 1 \right] \quad . \quad (31)$$

Eq. 31 is obtained by substitution of Eq. (16) into Eq. (15).

2.3 Nosé-Hoover Chain Method

An extension of the Nosé-Hoover thermostat is the Nosé-Hoover chain method (NHC) [14]. Within the NHC method the NH thermostat is coupled to another thermostat or, if necessary, to a whole chain of thermostats. Thus uncontrolled fluctuations in the thermostat degrees of freedom are eliminated and a better temperature control is yielded. It is shown that this generalization of the original NH method still generates a canonical distribution (provided that it is ergodic). Ergodicity requires that the dynamics of the system covers the whole phase which includes the thermostat variables [15]. NHC dynamics gives a very good approximation to the canonical ensemble in pathological cases where the global NH dynamics failed.

For isothermal dynamics the EOM of N particles are [7]:

$$\dot{\mathbf{r}}_i = \frac{\mathbf{p}_i}{m_i} \quad (32)$$

$$\dot{\mathbf{p}}_i = \mathbf{F}_i - \mathbf{p}_i \dot{\xi}_1 \quad (33)$$

$$\ddot{\xi}_1 = \frac{1}{Q_1} \left[\sum_{i=1}^N m_i \mathbf{v}_i^2 - N_f k_B T_{set} \right] - \dot{\xi}_1 \dot{\xi}_2 \quad (34)$$

$$\ddot{\xi}_j = \frac{1}{Q_j} \left[Q_{j-1} \dot{\xi}_{j-1}^2 - k_B T_{set} \right] - \dot{\xi}_j \dot{\xi}_{j+1} \quad , j = 2, \dots, M-1 \quad (35)$$

$$\ddot{\xi}_M = \frac{1}{Q_M} \left[Q_{M-1} \dot{\xi}_{M-1}^2 - k_B T_{set} \right] \quad , \quad (36)$$

where M is the number of thermostats with masses Q_j . The bath ‘mass’ parameters should be [16]

$$Q_1 = N_f k_B T_{set} \tau^2 \quad (37)$$

$$Q_i = k_B T_{set} \tau^2 \quad , \quad 1 < i \leq M. \quad (38)$$

Eq. (32) to Eq. (36) have the conserved energy

$$\begin{aligned} \mathcal{E}_{NHC} = & \frac{1}{2} \sum_i m_i \mathbf{v}_i^2 + E_{pot}[\{\mathbf{r}_i\}] + \sum_{j=1}^M \frac{1}{2} Q_j \dot{\xi}_j^2 \\ & + N_f k_B T_{set} \xi_1 + \sum_{j=2}^M \xi_j k_B T_{set} \end{aligned} \quad (39)$$

The NHC method suffers, however, from the fact that it is only capable of maintaining adequate temperature control in equilibrium [17]. Any perturbation away from equilibrium caused, e.g., by the presence of external fields or by motion over higher barrier, causes the method to break down. One should use a chain length longer than two in general cases where there is a broad distribution of vibrational frequencies [18].

2.3.1 Solve Equations of Motion Iteratively

As in the case of the global NH thermostat, the EOM can be solved iteratively using the Newton-Raphson procedure (Appendix A). For a Nosé-Hoover chain consisting of M units, a set of $3N + M$ equations of the form

$$\sum_{k=1}^{3N+M} \frac{\partial h_j(\kappa)}{\partial \kappa_k} \delta \kappa_k = -h_j(\kappa) \quad (40)$$

have to be considered. In Eq. (40), κ_k is defined as one component of the velocity of a particle, v_k for $k \in [1, 3N]$, and for $k \in [3N + 1, N + M]$ as $\dot{\xi}_k$. For $j \in [1, 3N]$, h_k is defined as

$$h_j(v_j, \dot{\xi}_1) = v'_j - v_j + [a_j - \dot{\xi}_1 v_j] \frac{\delta t}{2} \quad , \quad (41)$$

and for $j \in [3N + 1, N + M]$ by

$$h_{3N+1}(v_j, \dot{\xi}_1, \dot{\xi}_2) = \dot{\xi}'_1 + \left[\frac{1}{Q_1} \sum_j^N m_j v_j^2 - N_f k_B T_{set} - \dot{\xi}_1 \dot{\xi}_2 \right] \frac{\delta t}{2} - \dot{\xi}_1 \quad (42)$$

$$h_{3N+k}(\dot{\xi}_{k-1}, \dot{\xi}_k, \dot{\xi}_{k+1}) = \dot{\xi}'_k + \left[\frac{1}{Q_1} Q_{k-1} \dot{\xi}_{k-1}^2 - k_B T_{set} - \dot{\xi}_k \dot{\xi}_{k+1} \right] \frac{\delta t}{2} - \dot{\xi}_k \quad (43)$$

$$h_{3N+M}(\dot{\xi}_{M-1}, \dot{\xi}_M) = \dot{\xi}'_M + \left[\frac{1}{Q_M} Q_{M-1} \dot{\xi}_{M-1}^2 - k_B T_{set} \right] \frac{\delta t}{2} - \dot{\xi}_M \quad (44)$$

This set can be reduced to a set of M equations [11]. In matrix notation these M equations have the form

$$\underline{\mathbf{J}} \cdot \delta \dot{\boldsymbol{\xi}} = -\mathbf{h}' \quad (45)$$

where $\delta \dot{\boldsymbol{\xi}}$ contains the corrections for the thermostat velocities, $\dot{\xi}_i$ ($i = 1, \dots, M$), the elements of the Jacobian matrix are given by

$$J_{ij} = \begin{cases} g_1^* & i = 1, j = i \\ g_M & i = M, j = i \\ g_i & 1 < i < M, j = i \\ e_i & 1 < i < M - 1, j = i + 1 \\ f_i & 2 < i < M, j = i - 1 \\ 0 & \text{elsewhere} \end{cases} \quad (46)$$

and

$$h'_i = \begin{cases} h_{3N+1}^* & i = 1 \\ h_{3N+i} & 1 < i \leq M \end{cases} \quad . \quad (47)$$

In Eq. (47),

$$h_{3N+1}^* = h_{3N+1} - \sum_{j=1}^{3N} \frac{b_j h_j}{d} \quad . \quad (48)$$

The parameters in Eq. (46) and Eq. (48) mean

$$\begin{aligned}
g_1^* &= g_1 - \sum_{j=1}^{3N} \frac{b_j c_j}{d} \\
g_i &= -\dot{\xi}_{i+1} \frac{\delta t}{2} - 1 \\
g_M &= -1 \\
b_j &= m_j v_j \frac{\delta t}{Q_1} \\
c_j &= -v_j \frac{\delta t}{2} \\
d &= -\dot{\xi}_1 \frac{\delta t}{2} - 1 \\
e_i &= -\dot{\xi}_i \frac{\delta t}{2} \\
f_i &= \dot{\xi}_{i-1} \frac{Q_{i-1} \delta t}{Q_i}
\end{aligned}$$

The corrections for the velocities, δv_i , are calculated with

$$\delta v_i = \frac{-h_i - c_i \delta \dot{\xi}_1}{d} \quad (49)$$

The procedure for a constant temperature simulation within the NHC scheme using the velocity Verlet integrator is:

- update thermostat variables $\xi_i(t)$, $\dot{\xi}_i(t)$ and $\ddot{\xi}_i(t)$:
 - $\ddot{\xi}_1(t) = 1/Q_1[\sum_i m_i \mathbf{v}_i(t) - N_f k_B T_{set}] - \dot{\xi}_1(t) \dot{\xi}_2(t)$
 - $\ddot{\xi}_j(t) = 1/Q_j[Q_{j-1} \dot{\xi}_{j-1}^2(t) - k_B T_{set}] - \dot{\xi}_j(t) \dot{\xi}_{j+1}(t)$,
for $1 < j < M$
 - $\ddot{\xi}_M(t) = 1/Q_M[Q_{M-1} \dot{\xi}_{M-1}^2(t) - k_B T_{set}]$
 - $\dot{\xi}_j(t + \delta t/2) = \dot{\xi}_j(t) + \ddot{\xi}_j(t) \delta t/2$, for $j = 1, \dots, M$.
 - $\xi_j(t + \delta t) = \xi_j(t) + \dot{\xi}_j(t + \delta t/2) \delta t$, for $j = 1, \dots, M$.
- evolve velocities: $\mathbf{v}_i(t + \delta t/2) = \mathbf{v}_i(t) + [\mathbf{a}_i(t) - \mathbf{v}_i(t) \dot{\xi}_1(t)] \delta t/2$
- evolve positions: $\mathbf{r}_i(t + \delta t) = \mathbf{r}_i(t) + \mathbf{v}_i(t + \delta t/2) \delta t$
- apply the SHAKE procedure if constraints on internal coordinates are present
- call force $\rightarrow \mathbf{a}_i(t + \delta t) = \mathbf{F}_i(t + \delta t)/m_i$
- apply the RATTLE procedure if constraints on internal coordinates are present

- evolve velocities $\mathbf{v}_i(t + \delta t)$, $\dot{\xi}_j(t + \delta t)$ iteratively via the Newton-Raphson procedure, Eq. (45).

This method to solve the NHC EOM is not very common. Apparently, the iterative procedure lacks of steadiness.

2.3.2 Explicit Reversible Velocity Verlet Integrator

An integration scheme for NHC dynamics can be formulated also using the approach described in Appendix B. The Liouville operator for the EOM is [16]

$$\begin{aligned}
i\hat{L} &= i\hat{L}_1 + i\hat{L}_2 + i\hat{L}_{NHC} \\
&= \sum_{i=1}^N \mathbf{v}_i \cdot \frac{\partial}{\partial \mathbf{r}_i} + \sum_{i=1}^N \mathbf{F}_i \cdot \frac{\partial}{\partial \mathbf{p}_i} \\
&\quad - \sum_{i=1}^N \dot{\xi}_i \mathbf{p}_i \cdot \frac{\partial}{\partial \mathbf{p}_i} + \sum_{i=1}^M \dot{\xi}_i \cdot \frac{\partial}{\partial \xi_i} + \sum_{i=1}^{M-1} (\ddot{\xi}_i - \dot{\xi}_i \dot{\xi}_{i+1}) \cdot \frac{\partial}{\partial \dot{\xi}_i} + \ddot{\xi}_M \cdot \frac{\partial}{\partial \dot{\xi}_M} \quad (50)
\end{aligned}$$

Using the simple generalization of the Trotter formula, Eq. (B-5), the evolution operator to can be written as

$$\begin{aligned}
\hat{G}(\delta t) &= \exp(i\hat{L}_{NHC} \frac{\delta t}{2}) \times \exp(i\hat{L}_1 \frac{\delta t}{2}) \times \exp(i\hat{L}_2 \delta t) \\
&\quad \times \exp(i\hat{L}_1 \frac{\delta t}{2}) \times \exp(i\hat{L}_{NHC} \frac{\delta t}{2}) + \mathcal{O}(\delta t^3) \quad (51)
\end{aligned}$$

where \hat{L}_1 and \hat{L}_2 retain their definitions in Eq. (29) and $i\hat{L}_{NHC} = i\hat{L} - i\hat{L}_1 - i\hat{L}_2$ consists of the remaining terms in Eq. (50). The NHC part of the evolution operator, $\exp(i\hat{L}_{NHC})$, is modified such that multiple time steps (Appendix B.2) can be used [16]:

$$\begin{aligned}
\exp(i\hat{L}_{NHC} \frac{\delta t}{2}) &= \prod_{j=1}^{n_c} \exp(i\hat{L}_{NHC} \frac{\delta t}{2n_c}) \quad (52) \\
\exp(i\hat{L}_{NHC} \frac{\delta t}{2n_c}) &= \exp\left(\frac{\delta t}{4n_c} \ddot{\xi}_M \frac{\partial}{\partial \dot{\xi}_M}\right) \times \exp\left(-\frac{\delta t}{8n_c} \dot{\xi}_M \dot{\xi}_{M-1} \frac{\partial}{\partial \dot{\xi}_{M-1}}\right) \\
&\quad \times \exp\left(\frac{\delta t}{4n_c} \ddot{\xi}_{M-1} \frac{\partial}{\partial \dot{\xi}_{M-1}}\right) \times \exp\left(-\frac{\delta t}{8n_c} \dot{\xi}_M \dot{\xi}_{M-1} \frac{\partial}{\partial \dot{\xi}_{M-1}}\right) \\
&\quad \times \exp\left(-\frac{\delta t}{2n_c} \sum_{i=1}^N \dot{\xi}_i \mathbf{p}_i \frac{\partial}{\partial \mathbf{p}_i}\right) \times \exp\left(-\frac{\delta t}{2n_c} \sum_{i=1}^M \dot{\xi}_i \frac{\partial}{\partial \xi_i}\right) \quad (53) \\
&\quad \times \dots \times \exp\left(-\frac{\delta t}{8n_c} \dot{\xi}_M \dot{\xi}_{M-1} \frac{\partial}{\partial \dot{\xi}_{M-1}}\right) \times \exp\left(\frac{\delta t}{4n_c} \ddot{\xi}_{M-1} \frac{\partial}{\partial \dot{\xi}_{M-1}}\right) \\
&\quad \times \exp\left(-\frac{\delta t}{8n_c} \dot{\xi}_M \dot{\xi}_{M-1} \frac{\partial}{\partial \dot{\xi}_{M-1}}\right) \times \exp\left(\frac{\delta t}{4n_c} \ddot{\xi}_M \frac{\partial}{\partial \dot{\xi}_M}\right)
\end{aligned}$$

In Eq. (53), a multiple time, $n_c > 1$, has been used. For typical simulations, n_c can be set to one. However, if the time scale associated with the NHC, τ , is small, n_c must be taken rather large to generate accurate trajectories. Substantial improvement can be obtained by using a higher order integration algorithm, e.g. the Yoshida Suzuki integration scheme (Appendix B.3):

$$\exp\left(i\hat{L}_{NHC}\frac{\delta t}{2}\right) = \prod_{i=1}^{n_c} \left[\prod_{j=1}^m \exp\left(i\hat{L}_{NHC}\frac{w_j\delta t}{2n_c}\right) \right], \quad (54)$$

where w_j are chosen such that when $\exp[i\hat{L}_{NHC}\delta t/(2n_c)]$ is approximated as in Eq. (53) and the error is $\mathcal{O}[\delta t/(2n_c)]^5$ (Appendix B.3). The values of w_j are depending on m and are summarized in the table below.

Yoshida Suzuki Integration Parameters [16, 8].

m	w_j
1	1
3	$w_1 = w_3 = 1/(2 - 2^{1/3}), w_2 = 1 - 2w_1$
5	$w_j = p_{2,j}^{yosu}$
7	$w_1 = w_7 = -1.17767998417887$ $w_2 = w_6 = 0.235573213359357$ $w_3 = w_5 = 0.78451361047756$ $w_4 = 1 - 2(w_1 + w_2 + w_3)$
25	$w_j = p_{2,k}^{yosu} p_{3,l}^{yosu}, k, l = 1, \dots, 5$

$$p_{ij}^{yosu} = \begin{cases} 1 - \frac{4}{4-4^{b_i}} & \text{if } i \bmod 3 = 0 \\ \frac{1}{4-4^{b_i}} & \text{else} \end{cases} \quad (55)$$

where $b_i = \frac{1}{2i-1}$ and $i = 2, \dots, 5$ and $j = 1, \dots, 5$ [8].

The approach described appears to be rather complicated. However, it is straightforward and computationally inexpensive to implement. First the operator $\exp(i\hat{L}_{NHC}\delta t/2)$ is applied to update $\{\xi, \dot{\xi}, \mathbf{v}\}$. Next, the updated particle velocities as used as input to the usual velocity Verlet step, Eq. (2). Then the operator $\exp(i\hat{L}_{NHC}\delta t/2)$ is applied once more to the output of the velocity Verlet step.

The direct translation technique, Eq. (B-9)–Eq. (B-11), can be used to convert the operator $\exp[i\hat{L}_{NHC}w_j\delta t/(2n_c)]$ into a set of instructions analogous to Eq. (52):

```

get akin =  $\sum_{i=1}^N m_i \mathbf{v}_i$ 

do k = 1,  $n_c$ 

  do j = 1,  $m$ 
     $\delta t_s = w_j \delta t / n_c$ 
    – calculate thermostat acceleration
  
```

```

 $\ddot{\xi}_1 = (akin - N_f k_B T_{set}) / Q_1$ 
do  $i = 2, M$ 
   $\ddot{\xi}_i = (Q_{i-1} \dot{\xi}_{i-1}^2 - k_B T_{set}) / Q_i$ 
enddo
– update the thermostat velocities
 $\dot{\xi}_M = \dot{\xi}_M + \delta t_s / 4 \cdot \ddot{\xi}_M$ 
do  $i = 1, M - 1$ 
   $aa = \exp(-\delta t_s / 8 \cdot \dot{\xi}_{M+1-i})$ 
   $\dot{\xi}_{M-i} = \dot{\xi}_{M-i} \cdot aa^2 + \delta t_s / 4 \cdot \ddot{\xi}_{M-i} \cdot aa$ 
enddo
– update the particle velocities
 $aa = \exp(-\delta t_s / 2 \cdot \dot{\xi}_1)$ 
do  $i = 1, N$ 
   $\mathbf{v}_i = \mathbf{v}_i \cdot aa$ 
enddo
– update the thermostat positions
do  $i = 1, M$ 
   $\xi_i = \xi_i + \delta t_s / 2 \cdot \dot{\xi}_i$ 
enddo
– update thermostat accelerations
enddo
 $akin = aa^2 \cdot akin$ 
 $\ddot{\xi}_1 = (akin - N_f k_B T_{set}) / Q_1$ 
do  $i = 2, M$ 
   $\ddot{\xi}_i = (Q_{i-1} \dot{\xi}_{i-1}^2 - k_B T_{set}) / Q_i$ 
enddo
– update thermostat velocities
do  $i = 1, M - 1$ 
   $aa = \exp(-\delta t_s / 8 \cdot \dot{\xi}_{i+1})$ 
   $\dot{\xi}_i = \dot{\xi}_i \cdot aa^2 + \delta t_s / 4 \cdot \ddot{\xi}_i \cdot aa$ 
   $\ddot{\xi}_{i+1} = (Q_i \dot{\xi}_i^2 - k_B T_{set}) / Q_{i+1}$ 
enddo
 $\dot{\xi}_M = \dot{\xi}_M + \delta t_s / 4 \cdot \ddot{\xi}_M$ 
enddo
enddo

```

Note, a time-saving feature has been employed. The effect of the operator $\exp [i\hat{L}_{NHC}w_j\delta t/(2n_c)]$ is to scale the particle velocities by the factor $\exp [-\dot{\xi}_1w_j\delta t/(2n_c)]$. The only coupling of the particle velocities to the thermostat variables occurs through the total atomic kinetic energy $\sum_{i=1}^N m_i v_i$ which appears in the acceleration on the first thermostat, $\ddot{\xi}_1$. In addition, the total particle kinetic energy can be evolved by multiplying by the factor $\exp [-\dot{\xi}_1w_j\delta t/(2n_c)]$ at each step in the iteration. The entire propagator may be implemented by performing the procedure defined in the instructions above, before and after performing the velocity Verlet integrator.

When the bath particles have small masses, relative to the rest of the system, a shorter time step must be used in the integration of the EOM. However, the use of the multiple time step approach alleviates this problem [2].

2.4 “Massive” Nosé-Hoover Chains

One attractive feature of the Nose Hoover thermostat methods is that they may be easily extended to include multiple thermostats. The ability to couple different degrees of freedom with different energy flow characteristics to separate thermostats is very useful in various situations. For example, separate thermostats can be coupled to solvent and proteins degrees of freedoms to keep them at the same temperature in simulations of a protein in solution. This often a problem in NVE simulations. Separate thermostats can also be used to keep nuclear degrees of freedom hot and electronic degrees of freedom cold, i.e., on the Born-Oppenheimer surface, a requirement that is crucial in CPMD simulations. As a rapid equilibration technique for proteins each degree of freedom is coupled to a separate NHC. This technique is referred as “massive” Nosé-Hoover chains (MNHC) [15].

Considered are n chains where the j th chain is coupled to g_j degrees of freedom and

$$g = \sum_{j=1}^n g_j \quad (56)$$

is the total number of degrees of freedom in the system. Newton’s EOM for the i th degree of freedom coupled to the j th chain is

$$a_{ij} = \frac{F_{ij}}{m_{ij}} - v_{ij}\dot{\xi}_{1j} \quad (57)$$

where $\dot{\xi}_{1j}$ is the velocity of first thermostat in the chain. The EOM for the thermostat

variables in the j th chain containing M thermostats at a temperature $T_{j,set}$ are

$$\ddot{\xi}_{1j} = \frac{1}{Q_{1j}} \left[\sum_{i=1}^{g_j} m_{ij} \mathbf{v}_{ij}^2 - g_j k_B T_{j,set} \right] - \dot{\xi}_{1j} \dot{\xi}_{2j} \quad (58)$$

$$\ddot{\xi}_{kj} = \frac{1}{Q_{kj}} \left[Q_{k-1,j} \dot{\xi}_{k-1,j}^2 - k_B T_{j,set} \right] - \dot{\xi}_{kj} \dot{\xi}_{k+1,j} \quad , k = 2, \dots, M-1 \quad (59)$$

$$\ddot{\xi}_{Mj} = \frac{1}{Q_{Mj}} \left[Q_{M-1,j} \dot{\xi}_{M-1,j}^2 - k_B T_{j,set} \right] \quad , \quad (60)$$

where Q_{kj} is the mass of the k th thermostat variable in the chain. It can be shown that the extended system has the conserved quantity

$$\begin{aligned} \mathcal{E}_{MNHC} = & \frac{1}{2} \sum_{j=1}^N m_j \mathbf{v}_j^2 + E_{pot}[\{\mathbf{r}_j\}] + \frac{1}{2} \sum_{j=1}^n \sum_{i=1}^M Q_{ij} \dot{\xi}_{ij}^2 \\ & + \sum_{j=1}^n g_j \xi_{1j} k_B T_{j,set} + \sum_{j=1}^n \sum_{i=2}^M \xi_{ij} k_B T_{j,set} \quad . \end{aligned} \quad (61)$$

Note, that the ‘‘massive’’ NH method corresponds to $M = 1$ in the above equations.

The rate of convergence of a system to the canonical ensemble can depend on the choice of the thermostat mass parameters, Q_{kj} . In the case of ‘‘massive’’ chains all the masses in a given chain are the same (since $g_j = 1$) and equal to

$$Q_{kj} = \frac{m_j}{m_H} k_B T_{j,set} \tau^2 \quad (62)$$

where m_H is the mass of a hydrogen atom, so that relative rates of thermostat fluctuations are inversely proportional to the masses, m_j , of the particles to which the thermostats are coupled [15].

MNHC simulations are quite sensitive to initial conditions. However, this sensitivity could be eliminated by coupling an additional chain with a thermostat time scale τ' to the whole system [8].

The use of multiple NHC with ‘‘massive’’ chains as an extreme example appears especially promising for controlling the temperature in MD simulations of heterogeneous biological systems with weak couplings between intramolecular and intermolecular degrees of freedom (e.g., solvated proteins or biological membranes).

Note that, in a system with constraints and multiple chains of thermostats on the particle degrees of freedom (e.g., all atoms of type X thermostated with one chain and all atoms of type Y with another), atoms involved in a common constraints must be thermostated by the same chain, independent of the integration algorithm applied. If this is not the case the number of degrees of freedom (N_f) associated with each individual chain becomes a complicated function of the particle positions [16].

2.5 Generalized Gaussian Moment Thermostat

A relatively new method for generating the canonical ensemble via continuous dynamics is to control the fluctuations of an arbitrary number of the multidimensional Gaussian moment distribution function [17]. This method is called generalized Gaussian momentum thermostat (GGMT). Comparison of the GGMT to NHC dynamics shows that it performs as well or better in equilibrium calculations and leads to superior results in non-equilibrium situations, including, e.g., barrier crossing. Therefore, the GGMT method emerges a useful new molecular dynamics tool which should be applicable to a variety of different kinds of calculations including equilibration, simulated annealing, classical and quantum canonical averages, non-equilibrium simulations and *ab initio* molecular dynamics.

The EOM within the GGMT approach have the form

$$\dot{\mathbf{r}}_i = \frac{\mathbf{p}_i}{m_i} \quad (63)$$

$$\dot{\mathbf{p}}_i = \mathbf{F}_i(\{\mathbf{r}_i\}) - \sum_{n=1}^M \sum_{k=1}^n \dot{\xi}_n \frac{(k_B T_{set})^{n-k}}{C_{k-1}} S^{k-1} \mathbf{p}_i \quad (64)$$

$$\dot{\xi}_n = \left[(k_B T_{set})^{n-1} + \sum_{k=2}^n \frac{(k_B T_{set})^{n-k}}{N_f C_{k-2}} S^{k-1} \right] \dot{\xi}_n \quad (65)$$

$$\ddot{\xi}_n = \frac{S^n}{C_{n-1}} - N_f (k_B T_{set})^n, \quad n = 1, \dots, M \quad (66)$$

for an N -particle system coupled to a set of $2M$ additional variables, where M is the total number of non-vanishing, i.e., even, moments of the Gaussian distribution in momentum. The dynamics is designed to control the fluctuations in these Gaussian moments. In Eq. (63)–Eq. (66), the quantity $S = \sum_{i=1}^N m_i \mathbf{v}_i^2$ and the constants, C_n , are defined to be $C_n = \prod_{k=1}^n (N_f + 2k)$ with $C_0 = 1$. The conserved energy attributed to Eq. (63)–Eq. (66) is

$$\mathcal{E}_{GGMT} = \frac{1}{2} \sum_i m_i \mathbf{v}_i^2 + E_{pot}[\{\mathbf{r}_i\}] + \sum_{n=1}^M \frac{p_{\xi_n}^2}{2Q_n} + N_f k_B T_{set} \sum_{n=1}^M \xi_n \quad (67)$$

It is clear from the EOM that the thermostat variables, ξ_n , are all dimensionless, while their corresponding momenta, p_{ξ_n} , are dimensioned as $(\text{energy})^n \times (\text{time})$. Therefore, the time scale parameters, Q_n , have dimensions $(\text{energy})^{2n-1} \times (\text{time})^2$ and should be chosen according to $Q_n = A_n (k_B T_{set})^{2n-1} \tau^2$, where $A_n = [N_f / C_{n-1}] \sum_{k=1}^n (C_{n+k-2} / C_{k-1})$ and τ is some appropriate time scale of the system. This choice of masses has been obtained by performing a harmonic analysis of the EOM in a manner similar to that presented in Appendix B of Ref. [14].

For most systems $M = 2, 3$ is already generally sufficient [8]. A good rule of thumb in choosing τ is that this time step will generally need to be roughly a factor of 20–40 smaller than the shortest period in your system. One logical choice might be to

choose a lower frequency in your spectrum. Thus, one can choose τ based on multiples of the time step (which helps considerably to understand the time scales chosen and correlate them with the power spectrum).

Consider the fourth-moment ($M = 2$) GGMT coupled to an N -particle system in three dimensions. The corresponding Liouville operator is

$$i\hat{L} = \sum_{i=1}^N \left[\mathbf{v}_i \cdot \frac{\partial}{\partial \mathbf{r}_i} + \mathbf{F}_i \cdot \frac{\partial}{\partial \mathbf{p}_i} \right] + i\hat{L}_{GGMT} \quad (68)$$

$$\begin{aligned} i\hat{L}_{GGMT} = & - \lambda(\dot{\xi}_1, \dot{\xi}_2) \sum_{i=1}^N \mathbf{p}_i \frac{\partial}{\partial \mathbf{p}_i} - \frac{\dot{\xi}_2}{3N+2} \left(\sum_{k=1}^N \frac{\mathbf{p}_k^2}{m_k} \right) \left(\sum_{i=1}^N \mathbf{p}_i \frac{\partial}{\partial \mathbf{p}_i} \right) \\ & + \dot{\xi}_1 \frac{\partial}{\partial \xi_1} + \bar{g}(\mathbf{p}_i) \dot{\xi}_2 \frac{\partial}{\partial \xi_2} + \bar{G}_1(\mathbf{p}_i) \frac{\partial}{\partial \dot{\xi}_1} + \bar{G}_2(\mathbf{p}_i) \frac{\partial}{\partial \dot{\xi}_2} \end{aligned} \quad (69)$$

where $\lambda(\dot{\xi}_1, \dot{\xi}_2)$ is defined as

$$\lambda(\dot{\xi}_1, \dot{\xi}_2) = \dot{\xi}_1 + k_B T_{set} \dot{\xi}_2 \quad (70)$$

and

$$\begin{aligned} \bar{g}(\mathbf{p}_i) &= k_B T_{set} + \frac{1}{N_f} \sum_{k=1}^N m_k \mathbf{v}_k^2 \\ \bar{G}_1(\mathbf{p}_i) &= \sum_{k=1}^N m_k \mathbf{v}_k^2 - N_f k_B T_{set} \\ \bar{G}_2(\mathbf{p}_i) &= \frac{1}{N_f + 2} \left(\sum_{k=1}^N m_k \mathbf{v}_k^2 \right)^2 - N_f (k_B T_{set})^2 \end{aligned} \quad (71)$$

The basic factorization of the propagators, $\exp(i\hat{L}\delta t)$ and $\exp(i\hat{L}_{GGMT})$, is similar to that of Eq. (51) and Eq. (52), respectively. The operator $\exp(i\hat{L}_{GGMT}\delta t_j/2)$ is

factorized according to

$$\begin{aligned}
\exp(i\hat{L}_{GGMT}\frac{\delta t_j}{2}) &= \exp\left(\frac{\delta t_j}{4}\bar{G}_1(\mathbf{p}_i)\frac{\partial}{\partial\dot{\xi}_1}\right) \times \exp\left(\frac{\delta t_j}{4}\bar{G}_2(\mathbf{p}_i)\frac{\partial}{\partial\dot{\xi}_2}\right) \\
&\times \exp\left(-\frac{\delta t_j}{8}\lambda(\dot{\xi}_1, \dot{\xi}_2)\sum_{i=1}^N \mathbf{v}_i\frac{\partial}{\partial\mathbf{v}_i}\right) \\
&\times \exp\left[-\frac{\delta t_j}{4}\dot{\xi}_2\frac{1}{N_f+2}\left(\sum_{k=1}^N m_k\mathbf{v}_k^2\right)\left(\sum_{i=1}^N \mathbf{v}_i\frac{\partial}{\partial\mathbf{v}_i}\right)\right] \\
&\times \exp\left(-\frac{\delta t_j}{8}\lambda(\dot{\xi}_1, \dot{\xi}_2)\sum_{i=1}^N \mathbf{v}_i\frac{\partial}{\partial\mathbf{v}_i}\right) \\
&\times \exp\left(\frac{\delta t_j}{2}\dot{\xi}_1\frac{\partial}{\partial\dot{\xi}_1}\right) \times \exp\left(\frac{\delta t_j}{2}\bar{g}(\mathbf{p}_i)\dot{\xi}_2\frac{\partial}{\partial\dot{\xi}_2}\right) \\
&\times \exp\left(-\frac{\delta t_j}{8}\lambda(\dot{\xi}_1, \dot{\xi}_2)\sum_{i=1}^N \mathbf{v}_i\frac{\partial}{\partial\mathbf{v}_i}\right) \\
&\times \exp\left[-\frac{\delta t_j}{4}\dot{\xi}_2\frac{1}{N_f+2}\left(\sum_{k=1}^N m_k\mathbf{v}_k^2\right)\left(\sum_{i=1}^N \mathbf{v}_i\frac{\partial}{\partial\mathbf{v}_i}\right)\right] \\
&\times \exp\left(-\frac{\delta t_j}{8}\lambda(\dot{\xi}_1, \dot{\xi}_2)\sum_{i=1}^N \mathbf{v}_i\frac{\partial}{\partial\mathbf{v}_i}\right) \\
&\times \exp\left(\frac{\delta t_j}{4}\bar{G}_2(\mathbf{p}_i)\frac{\partial}{\partial\dot{\xi}_2}\right) \times \exp\left(\frac{\delta t_j}{4}\bar{G}_1(\mathbf{p}_i)\frac{\partial}{\partial\dot{\xi}_1}\right) \quad (72)
\end{aligned}$$

(73)

Using the simple generalization of the Trotter formula, Eq. (B-5), the evolution operator to can be written as

$$\begin{aligned}
\hat{G}(\delta t) &= \exp(i\hat{L}_{GGMT}\frac{\delta t}{2}) \times \exp(i\hat{L}_1\frac{\delta t}{2}) \times \exp(i\hat{L}_2\delta t) \\
&\times \exp(i\hat{L}_1\frac{\delta t}{2}) \times \exp(i\hat{L}_{GGMT}\frac{\delta t}{2}) + \mathcal{O}(\delta t^3) \quad (74)
\end{aligned}$$

where \hat{L}_1 and \hat{L}_2 retain their definitions in Eq. (29) and $i\hat{L}_{GGMT} = i\hat{L} - i\hat{L}_1 - i\hat{L}_2$

Since, according to Eq. (74), the thermostat operator is applied before and after a full velocity Verlet step, standard SHAKE and RATTLE procedures for enforcing constraints on internal coordinates can be applied in the usual manner, i.e., SHAKE after the position update and RATTLE just before the second application of $\exp(i\hat{L}_{GGMT}\delta t/2)$. In addition, for N particles in three dimensions subject to n_{constr} , the number of degrees of freedom is no longer N_f but, in general, $N_f - n_{constr}$. For an N -atomic molecule in vacuum, for which the translational motion of the center of mass and the

angular momentum relative to the center of mass is removed and on which n_{constr} constraints have been imposed, $N_f = 3N - n_{constr} - 6$ [19].

A C code for the implementation of N particles coupled to the GGMT for the case of $M = 2$ as well as the case of arbitrary M can be found in the EPAPS material (see EPAPS Document No. E-JCPSA6-112-513948. This document may be retrieved via the EPAPS homepage, <http://www.aip.org/pubserv/epaps.html>, or from <ftp.aip.org> in the directory /epaps/. See the EPAPS homepage for more information.)

A Newton-Raphson Procedure [20]

A typical problem gives $3N + 1$ functional relations to be zeroed, involving variables $\kappa_i, i = 1, 2, \dots, 3N + 1$:

$$h_i(\kappa_1, \kappa_2, \dots, \kappa_{3N+1}) = 0 \quad , \quad i = 1, 2, \dots, 3N + 1 \quad . \quad (\text{A-1})$$

Let $\boldsymbol{\kappa}$ denote the entire vector of values κ_j and \mathbf{h} denote the entire vector of functions h_j . In the neighborhood of $\boldsymbol{\kappa}$, each of the functions h_j can be expanded in Taylor series

$$h_j(\boldsymbol{\kappa} + \delta\boldsymbol{\kappa}) = h_j(\boldsymbol{\kappa}) + \sum_{k=1}^{3N+1} \frac{\partial h_j(\boldsymbol{\kappa})}{\partial \kappa_k} \delta\kappa_k + \mathcal{O}(\delta\kappa^2) \quad , \quad j, k = 1, \dots, 3N + 1 \quad (\text{A-2})$$

The matrix of partial derivatives appearing in Eq. (A-2) is the *Jacobian* matrix $\underline{\mathbf{J}}$:

$$J_{jk} = \frac{\partial h_j(\boldsymbol{\kappa})}{\partial \kappa_k} \quad (\text{A-3})$$

In matrix notation, Eq. (A-2) is

$$\mathbf{h}(\boldsymbol{\kappa} + \delta\boldsymbol{\kappa}) = \mathbf{h}(\boldsymbol{\kappa}) + \underline{\mathbf{J}} \cdot \delta\boldsymbol{\kappa} + \mathcal{O}(\delta\boldsymbol{\kappa}^2) \quad (\text{A-4})$$

By neglecting terms to the order of $\delta\boldsymbol{\kappa}^2$ and higher a linear set of equations for the corrections $\delta\boldsymbol{\kappa}$ is obtained which move each function h_i closer to zero simultaneously, namely

$$\underline{\mathbf{J}} \cdot \delta\boldsymbol{\kappa} = -\mathbf{h} \quad (\text{A-5})$$

This equation can be solved by *LU* decomposition [20]. The corrections, $\delta\boldsymbol{\kappa}$, are added to the solution vector,

$$\boldsymbol{\kappa}_{new} = \boldsymbol{\kappa}_{old} + \delta\boldsymbol{\kappa} \quad , \quad (\text{A-6})$$

and the process is iterated to convergence.

B Reversible Reference System Propagator Algorithms [12]

B.1 The Trotter Expansion of the Liouville Propagator

The Liouville operator \hat{L} for a system of N_f degrees of freedom is defined in cartesian coordinates as

$$i\hat{L} = \sum_{j=1}^{N_f} \left[\dot{x}_j \frac{\partial}{\partial x_j} + F_j \frac{\partial}{\partial p_j} \right] \quad (\text{B-1})$$

where $\Gamma = \{x_j, p_j\}$ are the positions and conjugate momenta of the system and F_j is the force on the j th degree of freedom. \hat{L} is a linear Hermitian operator on the space of square integrable functions of Γ . The classical propagator is the

$$\hat{U}(t) = \exp(i\hat{L}t) \quad (\text{B-2})$$

and the state of the system at time t is given by

$$\Gamma(t) = \hat{U}(t)\Gamma(0) \quad . \quad (\text{B-3})$$

The operator \hat{U} is unitary, i.e., $\hat{U}(-t) = \hat{U}^{-1}(t)$. In the following the Liouville operator is decomposed into two parts such that

$$i\hat{L} = i\hat{L}_1 + i\hat{L}_2 \quad . \quad (\text{B-4})$$

For this decomposition the Trotter theorem [21] yields

$$\begin{aligned} \exp[i(\hat{L}_1 + \hat{L}_2)t] &= \left[\exp[i(\hat{L}_1 + \hat{L}_2)t/P] \right]^P \\ &= \left[\exp\left(i\hat{L}_1 \frac{\delta t}{2}\right) \times \exp(i\hat{L}_2 \delta t) \times \exp\left(i\hat{L}_1 \frac{\delta t}{2}\right) \right]^P \\ &+ \mathcal{O}(t^3/P^2) \quad , \end{aligned} \quad (\text{B-5})$$

where $\delta t = t/P$. From this the discrete time propagator is defined as

$$\begin{aligned} \hat{G}(\delta t) &= \hat{U}_1\left(\frac{\delta t}{2}\right) \times \hat{U}_2(\delta t) \times \hat{U}_1\left(\frac{\delta t}{2}\right) \\ &= \exp\left(i\hat{L}_1 \frac{\delta t}{2}\right) \times \exp(i\hat{L}_2 \delta t) \times \exp\left(i\hat{L}_1 \frac{\delta t}{2}\right) \quad . \end{aligned} \quad (\text{B-6})$$

The three factors in $\hat{G}(\delta t)$ are separately unitary and, therefore, $\hat{G}(\delta t)$ is also unitary, i.e., $\hat{G}^{-1}(t) = \hat{G}^\dagger(t) = \hat{G}(-t)$. This means that any integrator based on this Trotter factorization will be time reversible.

The formal solution for any decomposition of the Liouvillian can be generated as follows: Defined are the propagators

$$\Gamma_1[\delta t; \Gamma(0)] = \hat{U}_1(\delta t)\Gamma(0) \quad (\text{B-7})$$

and

$$\Gamma_2[\delta t; \Gamma(0)] = \hat{U}_2(\delta t)\Gamma(0) \quad (\text{B-8})$$

which are the state at time δt when the system is propagated by $\hat{U}_1(\delta t)$ or $\hat{U}_2(\delta t)$, respectively, starting from the state $\Gamma(0)$. Then by applying the operators in Eq. (B-6) serially to $\Gamma(0)$ yields

$$\begin{aligned} \Gamma(\delta t) &= \hat{U}_1\left(\frac{\delta t}{2}\right)\hat{U}_2(\delta t)\hat{U}_1\left(\frac{\delta t}{2}\right)\Gamma(0) \\ &= \hat{U}_1\left(\frac{\delta t}{2}\right)\hat{U}_2(\delta t)\Gamma_1\left[\frac{\delta t}{2}; \Gamma(0)\right] \end{aligned} \quad (\text{B-9})$$

$$= \hat{U}_1\left(\frac{\delta t}{2}\right)\Gamma_2\left\{\delta t; \Gamma_1\left[\frac{\delta t}{2}; \Gamma(0)\right]\right\} \quad (\text{B-10})$$

$$= \Gamma_1\left(\frac{\delta t}{2}; \Gamma_2\left\{\delta t; \Gamma_1\left[\frac{\delta t}{2}; \Gamma(0)\right]\right\}\right) . \quad (\text{B-11})$$

This scheme can be translated into a computer code very simply:

1. Start with the initial state $\Gamma(0)$ and generate the motion under the propagator $\exp(i\hat{L}_1\delta t/2)$. This yields the state $\Gamma_1[\delta t/2; \Gamma(0)]$.
2. Use the state just generated as the initial state and generate the motion using the propagator $\exp(i\hat{L}_2\delta t)$. This yields the state $\Gamma_2\{\delta t; \Gamma_1[\delta t/2; \Gamma(0)]\}$.
3. Start with state generated in step 1 as the initial condition and generate the motion using $\exp(i\hat{L}_1\delta t/2)$. This yields the final state $\Gamma(\delta t)$ specified in Eq. (B-11).

The solution for a more sophisticated propagation proceeds analogously.

The Trotter expansion carried out to higher orders will yield higher orders integrators. For example, to obtain an integrator with an error of $\mathcal{O}(\delta t^4)$ one would start with the Trotter expansion

$$\begin{aligned} \exp(i(\hat{L}_1 + \hat{L}_2)\delta t) &\approx \exp(i\hat{L}_1\frac{\delta t}{2}) \times \exp(i\hat{L}_2\frac{\delta t}{2}) \\ &\times \sum (-i\hat{C}\frac{\delta t^3}{24}) \\ &\times \exp(i\hat{L}_2\frac{\delta t}{2}) \times \exp(i\hat{L}_1\frac{\delta t}{2}) , \end{aligned} \quad (\text{B-12})$$

where

$$-i\hat{C} = [(i\hat{L}_1 + 2i\hat{L}_2), [i\hat{L}_1, i\hat{L}_2]] . \quad (\text{B-13})$$

In Eq. (B-13), $[\dots, \dots]$ denotes Lie brackets or commutator. The disadvantage of using such a higher order integrator is that the commutator will bring in derivatives of the force with respect to positions which may be difficult to calculate.

B.2 Systems with Multiple Time Scales

Systems consisting of high frequency oscillators coupled to low frequency oscillators or slow baths, and systems consisting of massive (slow moving) particles interacting with light (fast moving) particles are systems with two or more time scales. Obviously, such systems have a so-called disparate mass problem. Methods for accelerating the integration of the EOM of such systems are based on defining a reference system for the “fast” subsystem. The “fast” subsystem is solved either analytically or numerically for a sequence of small time steps, deriving EOM for the deviations from the exact trajectory and then solving the EOM for this deviation and for the slow degrees of freedom numerically using a much larger time step. When the reference system is solved numerically the method is called RESPA (reference system propagator algorithm), whereas when it solved analytically the method is called NAPA (numerical analytical propagator algorithm).

Consider the case where the degrees of freedom of the system can be subdivided into fast and slow degrees of freedom labeled x and y , respectively. E.g., a binary mixture of heavy particles (the y subsystem) and light particles (the x subsystem). Then we can decompose the Liouvillian as

$$i\hat{L} = i\hat{L}_x + i\hat{L}_y \quad , \quad (\text{B-14})$$

where

$$i\hat{L}_x = \dot{x} \frac{\partial}{\partial x} + F_x(x, y) \frac{\partial}{\partial p_x} \quad (\text{B-15})$$

$$i\hat{L}_y = \dot{y} \frac{\partial}{\partial y} + F_y(x, y) \frac{\partial}{\partial p_y} \quad (\text{B-16})$$

The propagator can thus be factorized as

$$\hat{G}_{xyx}(\delta t) = \sum (i\hat{L}_x \frac{\delta t}{2}) \times \exp(i\hat{L}_y \delta t) \times \exp(i\hat{L}_x \frac{\delta t}{2}) \quad (\text{B-17})$$

This \hat{G}_{xyx} factorization the first and last propagators involve the fast motion whereas the middle propagator involves the slow motion. The fast propagator can be further factorized:

$$\exp(i\hat{L}_x \frac{\delta t}{2}) = \left[\exp\left(\frac{\Delta t}{2} F_x \frac{\partial}{\partial p_x}\right) \times \exp\left(\Delta t \dot{x} \frac{\partial}{\partial x}\right) \times \exp\left(\frac{\Delta t}{2} \frac{\delta t}{2} F_x \frac{\partial}{\partial p_x}\right) \right]^{n/2} \quad (\text{B-18})$$

where $\Delta t = \delta t/n$ The propagator in the middle in \hat{G}_{xyx} can also be factorized but it involves the slow motion only:

$$\exp(i\hat{L}_y \frac{\delta t}{2}) = \exp\left(\frac{\delta t}{2} F_y \frac{\partial}{\partial p_y}\right) \times \exp\left(\delta t \dot{y} \frac{\partial}{\partial y}\right) \times \exp\left(\frac{\delta t}{2} \frac{\delta t}{2} F_y \frac{\partial}{\partial p_y}\right) \quad (\text{B-19})$$

If these factorizations are used in Eq. (B-17) the fast degrees of freedom and their conjugate momenta $\{x, p_x\}$ are determined numerically using the velocity Verlet integrator iteratively for n small time steps Δt subject to whatever the initial conditions might be, whereas the slow degrees of freedom are determined using the velocity Verlet integrator for only one large time step δt . From application of \hat{G}_{xyx} the following procedure can be derived:

1. Use the velocity Verlet integrator for $n/2$ time steps $\Delta t = \delta t/n$ to generate the state at time $\delta t/2$ under the action of the propagator $\exp(i\hat{L}_x\delta t/2)$ starting from the initial state $\{x(0), y(0), p_x(0), p_y(0)\}$.
2. Use the velocity Verlet integrator for one time step δt to advance the system from the final state of step 1 using the slow propagator, Eq. (B-19)
3. Use the velocity Verlet integrator for $n/2$ time steps $\Delta t = \delta t/n$ to generate the state at time $\delta t/2$ under the action of the propagator $\exp(i\hat{L}_x\delta t/2)$ starting from the initial state of step 2.

The forces of the slow coordinates are recalculated only once every time step, $\delta t = n\Delta t$. If the dimensionality of the fast subsystem is small compared to the dimensionality of the entire system, the forces on the slow coordinates will thus be calculated much less frequently than would be required in the standard methods.

B.3 Higher Order Integrators

Higher orders integrators generated from low-order integrators due to the Yoshida-Suzuki integration [7, 16] are particularly useful for integrating the EOM of the Nose-Hoover and generalized Gaussian moment type of thermostats. Suppose the Liouville operator has the general form

$$i\hat{L} = \sum_{k=1}^M i\hat{L}_k \quad . \quad (\text{B-20})$$

The symmetric Trotter factorization of the corresponding propagator is

$$\begin{aligned} \hat{U}_1(\delta t) &= \exp(i\hat{L}_1\frac{\delta t}{2}) \times \cdots \times \exp(i\hat{L}_{M-1}\frac{\delta t}{2}) \\ &\times \exp(i\hat{L}_M\delta t) \\ &\times \exp(i\hat{L}_{M-1}\frac{\delta t}{2}) \times \cdots \times \exp(i\hat{L}_1\frac{\delta t}{2}) \quad . \end{aligned} \quad (\text{B-21})$$

It has been shown that all higher order integrators of the form $\hat{U}_{2m}(\delta t)$ satisfy the recursion relation

$$\begin{aligned} \hat{U}_{2m}(\delta t) &= \hat{U}_{2m-1}(\delta t) \\ &= \left[\hat{U}_{2m-3}(w_m\delta t) \right]^2 \hat{U}_{2m-3}((1-4w_m)\delta t) \times \left[\hat{U}_{2m-3}(w_m\delta t) \right]^2 \quad , \end{aligned} \quad (\text{B-22})$$

where

$$w_m = \frac{1}{4 - 4^{1/(2m-1)}} \quad . \quad (\text{B-23})$$

Thus, starting with Eq. (B-21), Eq. (B-22) can be used to generate higher order factorizations corresponding to higher order integrators. Eq. (B-22) reveal that even and odd order integrators are equivalent and in particular $\hat{U}_2(\delta t) = \hat{U}_1(\delta t)$ which simply expresses the fact that Eq. (B-21) is accurate to $\mathcal{O}(\delta t^3)$. Usually the thermostat propagation is done using the fifth (=sixth)-order integration algorithm, i.e., $m = 3$ in Eq. (B-22) [7, 16].

References

- [1] Allen, M. P. and Tildesley, D. J. *Computer Simulations of Liquids*. Clarendon Press, Oxford, (1987).
- [2] Gibson, D. A. and Carter, E. A. *Mol. Phys.* **89**(5), 1265 (1996).
- [3] Evans, D. J. and Holian, B. L. *J. Chem. Phys.* **83**(8), 4069 (1985).
- [4] Evans, D. J., Hoover, W. G., Failor, B. H., Moran, B., and Ladd, A. J. C. *Phys. Rev. B* **28**(2), 1016 (1983).
- [5] Edberg, R., Evans, D. J., and Moriss, G. P. *J. Chem. Phys.* **84**(12), 6933 (1986).
- [6] Hoover, W. G. *Phys. Rev. A* **31**(3), 1695 (1985).
- [7] Tuckerman, M. E. and Parrinello, M. *J. Chem. Phys.* **101**(2), 1302 (1994).
- [8] Tuckerman, M. E. private communication.
- [9] Bond, S. D., Leimkuhler, B. J., and Laird, B. B. *J. Comp. Phys.* **151**, 114 (1999).
- [10] Martyna, G. J., Tobias, D. J., and Klein, M. L. *J. Chem. Phys.* **101**(5), 4177 (1994).
- [11] Frenkel, D. and Smit, B. *Understanding Molecular Simulations – From Algorithm to Application*. Academic Press, San Diego, (1996).
- [12] Tuckerman, M. E., Berne, B. J., and Martyna, G. J. *J. Chem. Phys.* **97**(3), 1990 (1992).
- [13] Ferrari, M., Fiorino, A., and Ciccotti, G. *Physica* **240**, 268 (1997).
- [14] Martyna, G. J., Klein, M. L., and Tuckerman, M. *J. Chem. Phys.* **97**(4), 2635 (1992).

- [15] Tobias, D. J., Martyna, G. J., and Klein, M. L. *J. Phys. Chem.* **97**(49), 12959 (1993).
- [16] Martyna, G. J., Tuckerman, M. E., Tobias, D. J., and Klein, M. L. *Mol. Phys.* **87**(5), 1117 (1996).
- [17] Liu, Y. and Tuckerman, M. E. *J. Chem. Phys.* **112**(4), 1685 (2000).
- [18] Jang, S. and Voth, G. A. *J. Chem. Phys.* **107**(22), 9514 (1997).
- [19] Barth, E., Kuczera, K., Leimkuhler, B., and Skeel, R. D. *J. Comp. Chem.* **16**(10), 1192 (1995).
- [20] Press, W. H., Flannery, B. P., Teukolsky, S. A., and Vetterling, W. T. *Numerical Recipes: The Art of Scientific Computing (Fortran version)*. Cambridge University Press, Cambridge, 2nd edition, (1990).
- [21] Trotter, H. F. *Proc. Am. Math. Soc.* **10**, 545 (1959).