





# Development of a PIMD algorithm

Edoardo Maria Centamori*
Scuola Normale Superiore
Università di Pisa

Supervisor
Adrien Marjollet
DESY

Supervisor
Ralph Welsch
DESY

September 4, 2019

**Abstract**

A PIMD algorithm is developed from scratch and his functionality is tested simulating a quantum harmonic oscillator. The algorithm is extended with a Langevin thermostat to enforce canonical distribution and with Rattle algorithm to impose constraints. The rattle algorithm is developed both for a generic MD simulation and for a PIMD simulation and his properties are confronted in the two cases. The algorithm is used to initialize a system of interest that will be used in future for RPMD simulations.

*edoardo.centamori@sns.it

# Contents

# 1 Theoretical introduction

## 1.a PIMD

It's well known that path integral reformulation of quantum mechanics yields to

$$K(x', t, x, 0) = \int [\mathcal{D}x(\tau)] e^{-S[x(\tau)]/\hbar}$$

where $K$ is the propagator, $S[x(\tau)]$ the action of the path $x(\tau)$ and $[\mathcal{D}x(\tau)]$ indicates the set of paths starting from $x$ at time 0 and ending in $x'$ at time $t$.
Then considering a simple Minkowski rotation, we can obtain a partition function

$$\mathcal{Z}(\beta) = \oint [\mathcal{D}x(\tau)] e^{iS[x(\tau)]/\hbar}$$

where now integration is intended over all closed path of imaginary time $\beta\hbar$.
Since we would like to compute these quantities numerically we need to discretize such equation, in particular is straightforward to show that [1]

$$\mathcal{Z}_N(\beta) := \left(\frac{mN}{2\pi\beta\hbar^2}\right)^{N/2} \int \mathrm{d}x_1 \cdots \mathrm{d}x_N \exp\left\{-\sum_{i=1}^{N} \frac{mN}{2\beta\hbar^2}(x_{i+1} - x_i)^2 + \frac{\beta}{N} V(x_i)\right\} \xrightarrow[N\to\infty]{} \mathcal{Z}(\beta)$$

where the indexes are intended to be cyclic.
What is particularly surprising is that since

$$\left(\frac{mN}{2\pi\beta\hbar^2}\right)^{N/2} = \int \frac{\mathrm{d}p_1 \cdots \mathrm{d}p_N}{\hbar^N} \exp\left\{-\beta \sum_{i=1}^{N} \frac{p_i^2}{2m}\right\}$$

the partition function reduces to

$$\mathcal{Z}_N(\beta) = \int \frac{\mathrm{d}x_1 \cdots \mathrm{d}x_N \mathrm{d}p_1 \cdots \mathrm{d}p_N}{\hbar^N} e^{-\beta_N H_n}$$

$$H_n(x_1, \ldots, x_N, p_1, \ldots, p_N) = \sum_{i=1}^{N} \frac{p_i^2}{2m} + \frac{m\omega_N^2}{2}(x_{i+1} - x_i)^2 + V(x_i)$$

$$\omega_N = \frac{N}{\beta\hbar} \text{ and } \beta_n = \frac{\beta}{N}$$

This is actually the partition function of a classical system, a set of $N$ masses cyclically connected by harmonic springs with frequency $\omega_N$ and immersed in a potential $V$ canonically distributed with inverse temperature $\beta_N$. From now on, we'll call the various masses beads or replicas while the whole chain will be referred to as necklace.

This idea provides us a method for simulating quantum systems using classical algorithms, in particular $\mathcal{Z}_N(\beta)$ could be evaluated using either a Monte Carlo sampling or a molecular dynamics simulation, we focus on the latter option. In order to achieve ergodicity, that is to distribute the system canonically over time, the presence of a thermostat is required, the most used are certainly the Nosé-Hoover [5] [6] and the Langevin thermostats [7] [8], we'll describe the second one.

## 1.b SDE and Langevin thermostating

*Langevin equations*

$$\frac{\mathrm{d}q}{\mathrm{d}t} = \frac{p}{m}$$

$$\frac{\mathrm{d}p}{\mathrm{d}t} = -\frac{\mathrm{d}V}{\mathrm{d}q} - \gamma p + \sqrt{\frac{2m\gamma}{\beta}}\xi \qquad \langle \xi(t) \rangle = 0 \ , \ \langle \xi(t)\xi(s) \rangle = \delta(t - s)$$

are a set of stochastic differential equation (SDE), $\gamma$ is a viscous coefficient and $\xi$ a random variable.

We can transform such equation into an equation concerning the density distribution of the random variable $p$, such equation is known as *Fokker-Plank equation* [2]

$$\frac{\partial \rho}{\partial t} = -p\frac{\partial \rho}{\partial q} + \frac{\partial}{\partial p}\left[\left(\frac{dV}{dq} + \gamma p\right)\rho\right] + \frac{m\gamma}{\beta}\frac{\partial^2 \rho}{\partial p^2}$$

where $\rho = \rho(q, p, t|q_0, p_0, 0)$ is a conditional distribution for the stochastic variable $(q, p)$ in the phase space. By direct substitution it's easy to check that

$$\rho = C \exp\left\{-\beta\left(\frac{p^2}{2m} + V(q)\right)\right\}$$

is the only stationary solution and $C$ is fixed by the normalization condition. We can work out analytically a solution for the SDE even in the time-dependent case, we simply need a little bit of Ito calculus [2] .
If $V$ is an harmonic potential then the Langevin equation reduce to a *Ornstein-Uhlenbeck process*, for which the general SDE (in the unidimensional case) is:

$$\mathrm{d}x = -kx\mathrm{d}t + \sqrt{D}\mathrm{d}W(t)$$

Where $dW(t)$ is the differential of the Wigner process. Considering the stochastic variable $y = xe^{kt}$ and using Ito formula yield a very simple SDE

$$\mathrm{d}y = \sqrt{D}e^{kt}\mathrm{d}W(t)$$

that can be integrated so that

$$x(t) = x(0)e^{-kt} + \sqrt{D}\int_0^t e^{-k(t-t')}\mathrm{d}W(t')$$

if $x(0)$ is either Gaussian or nonrandom, the whole process is Gaussian too and thus completely specified by mean and variance (which can be evaluated easily using Ito's formulas for averages and correlations):

$$\langle x(t)\rangle = \langle x(0)\rangle e^{-kt}$$

$$\mathrm{Var}\{x(t)\} = \left\{\mathrm{Var}\{x(0)\} - \frac{D}{2k}\right\}e^{-2kt} + \frac{D}{2k}$$

In our case $x = (q, p)$ and the Ornstein-Uhlenbeck process is multidimensional but this doesn't affect any conclusion.
The exponential decaying behavior shows that, not only the stationary solution of this process is the canonical distribution but also that any process, regardless of its initial condition, approaches the stationary solution with a time scale dictated by the viscous coefficient.
In future sections we'll search for optimal $\gamma$ coefficient in order to find efficient integrators of the motion. For the moment, the relevant concept is that such equation of motion effectively acts as a thermostat imposing canonical distribution to the paths.

## 1.c Rattle algorithm

Sometimes in these simulations is important to impose certain constraints, for example, if one is simulating the scattering of a set of particles, he might wants to ignore the energy associated to the motion of the center of mass of the system and thus he might want to remove such degree of freedom from my simulation.
For classical systems this is done by mean of the *Rattle algorithm*, evaluating the generalized forces coming from the constraints. For time-independent constraints of the form :

$$g_i(\boldsymbol{q}) = 0 \text{ for } i = 1, \ldots, m$$

such forces must be perpendicular to the surface defined by the constraint itself (because of D'Alembert principle). In the end we have

$$\begin{cases} \frac{\mathrm{d}\boldsymbol{q}}{\mathrm{d}t} = \boldsymbol{v} \\ \boldsymbol{M}\frac{\mathrm{d}\boldsymbol{v}}{\mathrm{d}t} = -\boldsymbol{\nabla}_q V(\boldsymbol{q}) - \sum_i^m \lambda_i \boldsymbol{\nabla}_q g_i(\boldsymbol{q}) \\ g_i(\boldsymbol{q}) = 0 \end{cases}$$

Where the $\lambda_i$ can be thought also as Lagrange multipliers and can be found explicitly by imposing the constraint (third equation). Rattle algorithm simply reimposes such constraints at each step as it will be shown in more

detail in subsequent sections.

# 2 Computational implementation

## 2.a Classical Stormer-Verlet integrator

I started implementing a classical integrator to simulate the motion of necklaces immersed in an external potential (basically PIMD without thermalization). The integrator chosen is the Stormer-Verlet algorithm, which is a second-order integrator meaning that at each step then the error is $o(\mathrm{d}t^3)$ ($\mathrm{d}t$ is the time step in the discretization).
Specifically the implemented algorithm at each step acts as :

**Stormer-Verlet algorithm**

1. $$\boldsymbol{p} \leftarrow \boldsymbol{p} - \boldsymbol{\nabla}_q V(\boldsymbol{q}) \frac{\mathrm{d}t}{2}$$
2. $$\boldsymbol{q} \leftarrow \boldsymbol{q} + \frac{\boldsymbol{p}}{m} \frac{\mathrm{d}t}{2}$$
3. $$\boldsymbol{p} \leftarrow \boldsymbol{p} - \boldsymbol{\nabla}_q V(\boldsymbol{q}) \frac{\mathrm{d}t}{2}$$

To check the effective functionality of such integrator, I checked for the conservation of classical energy and angular momentum of the system (considered completely classical at the moment) which are reported in Fig 1. As can be seen explicitly, energy is conserved up to one part in a million while angular momentum is conserved

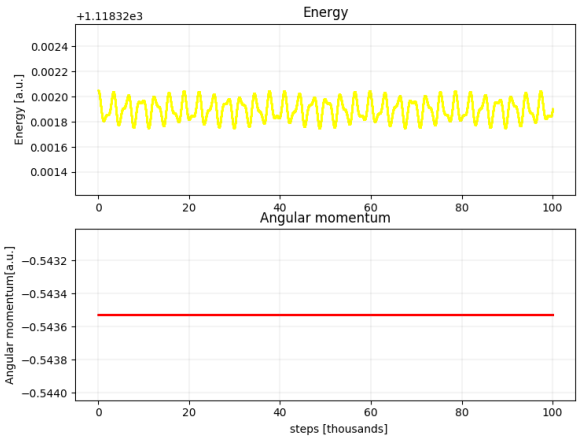



Figure 1: Energy and angular momentum conservation for the classical Stormer-Verlet algorithm.

up to floating-point precision for this specific case.
In general such fluctuations on energy are caused by the finite time step in the integration and can be reduced reducing the time step as the general fluctuation goes as [9]

$$\frac{\sqrt{(\Delta E^2)}}{E} = O(\mathrm{d}t^2).$$

What is not obvious in general (and in general not true for all the commonly used integrators) is the conservation of the mean value of energy, which is true for this specific algorithm for $\mathrm{d}t$ sufficiently small. Too big values of $\mathrm{d}t$ can bring to systematic drift which have in general an exponentially divergent behavior as shown in Fig 2.
Such conservation is not obvious at all because in general small errors at each step accumulate so that trajectories diverge exponentially from the exact ones, nevertheless such wrong trajectories still have the correct energy for this particular integrator.

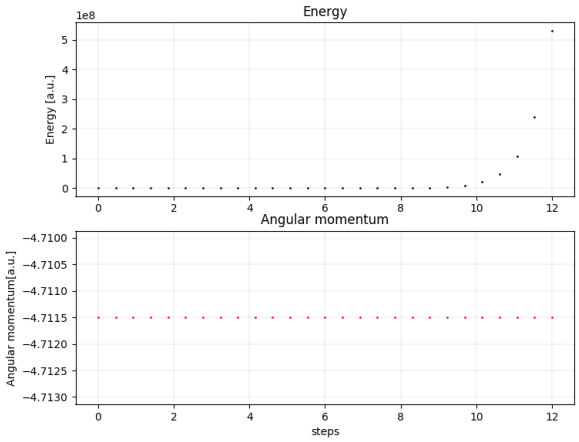



Figure 2: Divergence of energy for too big values of $\mathrm{d}t$.

To have a better visualization of what is happening I have designed an algorithm to visualize dynamically the motion of the classical system (for clear reasons the visualization is only implemented for 2-dimensional system and could be easily extended to 3-dimensional ones while the computing algorithm is fully generalized to any number of dimension). Some randomly chosen frames of the video generated by this algorithms are shown in Fig 3.

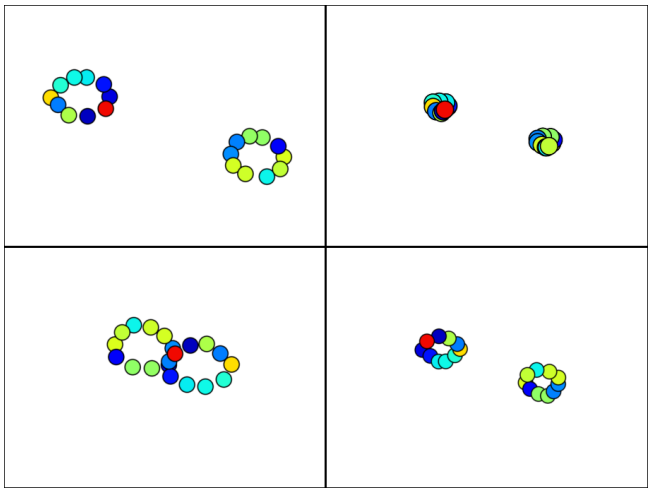


Figure 3: Some random frames of the classical evolution of 2 necklaces with 10 beads each.

## 2.b Classical Rattle implementation

As we said previously Rattle algorithm is a second-order integrator that at each step assures the preservation of constraints, as can be seen, is a direct generalization of the Stormer-Verlet algorithm [4]

$$\begin{cases} \boldsymbol{q}^{n+1} = \boldsymbol{q}^n + \mathrm{d}t\boldsymbol{v}^{n+\frac{1}{2}} \\ \boldsymbol{M}\boldsymbol{v}^{n+\frac{1}{2}} = \boldsymbol{M}\boldsymbol{v}^n - \frac{\mathrm{d}t}{2}\boldsymbol{\nabla}_{\boldsymbol{q}}V(\boldsymbol{q}^n) - \frac{\mathrm{d}t}{2}\boldsymbol{G}(\boldsymbol{q}^n)^T\boldsymbol{\lambda}_r^n \\ \boldsymbol{g}(\boldsymbol{q}^{n+1}) = \boldsymbol{0} \\ \boldsymbol{M}\boldsymbol{v}^{n+1} = \boldsymbol{M}\boldsymbol{v}^{n+\frac{1}{2}} - \frac{\mathrm{d}t}{2}\boldsymbol{\nabla}_{\boldsymbol{q}}V(\boldsymbol{q}^{n+1}) - \frac{\mathrm{d}t}{2}\boldsymbol{G}(\boldsymbol{q}^{n+1})^T\boldsymbol{\lambda}_v^{n+1} \\ \boldsymbol{G}(\boldsymbol{q}^{n+1})\boldsymbol{v}^{n+1} = \boldsymbol{0} \end{cases}$$

where $\boldsymbol{G}^T(\boldsymbol{q}) = \boldsymbol{\nabla}_{\boldsymbol{q}}\boldsymbol{g}(\boldsymbol{q})$. The two constraints are the requirement of $\boldsymbol{q}$ to point on the constrained surface and of $\boldsymbol{v}$ to lie on it.
In practice the algorithm at each step act as:

Rattle algorithm

1. Evaluate $\boldsymbol{\lambda}_r$ by solving :

$$\boldsymbol{g}\left(\boldsymbol{q}+\mathrm{d}t\boldsymbol{v}-\frac{\mathrm{d}t^2}{2}\boldsymbol{M}^{-1}\left(\boldsymbol{\nabla}_{\boldsymbol{q}}V(\boldsymbol{q})-\boldsymbol{G}(\boldsymbol{q})^T\boldsymbol{\lambda}_r\right)\right)=\boldsymbol{0}$$

2.
$$\boldsymbol{v}\leftarrow\boldsymbol{v}-\frac{\mathrm{d}t}{2}\boldsymbol{M}^{-1}\boldsymbol{\nabla}_{\boldsymbol{q}}V(\boldsymbol{q})-\frac{\mathrm{d}t}{2}\boldsymbol{M}^{-1}\boldsymbol{G}(\boldsymbol{q})^T\boldsymbol{\lambda}_r$$

3.
$$\boldsymbol{q}\leftarrow\boldsymbol{q}+\mathrm{d}t\boldsymbol{v}$$

4. Evaluate $\boldsymbol{\lambda}_v$ by solving :

$$\boldsymbol{G}(\boldsymbol{q})\left(\boldsymbol{v}-\frac{\mathrm{d}t}{2}\boldsymbol{M}^{-1}\boldsymbol{\nabla}_{\boldsymbol{q}}V(\boldsymbol{q})-\frac{\mathrm{d}t}{2}\boldsymbol{M}^{-1}\boldsymbol{G}(\boldsymbol{q})^T\boldsymbol{\lambda}_v\right)=\boldsymbol{0}$$

5.
$$\boldsymbol{v}\leftarrow\boldsymbol{v}-\frac{\mathrm{d}t}{2}\boldsymbol{M}^{-1}\boldsymbol{\nabla}_{\boldsymbol{q}}V(\boldsymbol{q})-\frac{\mathrm{d}t}{2}\boldsymbol{M}^{-1}\boldsymbol{G}(\boldsymbol{q})^T\boldsymbol{\lambda}_v$$

Solving the implicit equations to evaluate $\boldsymbol{\lambda}_r$ in the most general case would require the utilization of Newton method at each step which would probably become the most expensive part of the algorithm, fortunately for linear or quadratic constraints (and technically also cubic and quartic, but I haven't implemented them explicitly) such equation can be solved analytically thus removing this burden (notice that, since $5^{\text{th}}$ grade or more equations cannot be solved algebraically, such idea cannot be further generalized).

As an archetype of the general linear and quadratic constraint, I'll show explicitly how to evaluate $\boldsymbol{\lambda}_r$ in order to constrain a centroid position or the distance between two centroids.

### 2.b.1 Fixed centroid

Let $\boldsymbol{q}_{a,i}$ indicate the position of the $i$-th replica of the $a$-th necklace, then the requirement of the $a$-th centroid to be fixed is given by

$$\boldsymbol{g}_1(\boldsymbol{q})=\frac{\sum_i\boldsymbol{q}_{a,i}}{N}=\boldsymbol{0}$$

Let's say we're in 3 dimensions just for sake of clarity, then since that vectorial constraint can be written as 3 different constraint we're supposed to find 6 Lagrange multiplier $\lambda_{rx}$, $\lambda_{ry}$, $\lambda_{rz}$,$\lambda_{vx}$, $\lambda_{vy}$, $\lambda_{vz}$ which we collectively call $\boldsymbol{\lambda}_r$ and $\boldsymbol{\lambda}_v$.
By direct evaluation

$$\left(\boldsymbol{G}(\boldsymbol{q})^T\boldsymbol{\lambda}_r\right)_{c,i}=\begin{cases}\boldsymbol{0}, & c\neq a\\ \boldsymbol{\lambda}_r, & c=a\end{cases}$$

We can rewrite the first constraint as

$$\boldsymbol{c}^n+\boldsymbol{v}_c^n\mathrm{d}t-\frac{\mathrm{d}t^2}{2}\boldsymbol{M}^{-1}\left(\frac{\sum_i^N\boldsymbol{\nabla}_{\boldsymbol{q}}V(\boldsymbol{q}_{a,i}^n)}{N}+\boldsymbol{G}(\boldsymbol{q})^T\boldsymbol{\lambda}_r^n\right)=0$$

where we indicated with $\boldsymbol{c}$ and $\boldsymbol{v}_c$ position and velocity of the centroid, then direct evaluation of the Lagrange multiplier is straightforward (it's easier to evaluate directly $\boldsymbol{M}^{-1}\boldsymbol{G}(\boldsymbol{q})^T\boldsymbol{\lambda}_r^n$ since that's what appears in the formula for updating velocities)
The constraint on velocities is

$$\boldsymbol{0}=\boldsymbol{v}_c^{n+1}=\boldsymbol{v}_c^{n+\frac{1}{2}}-\frac{\mathrm{d}t}{2}\frac{\sum_i^N\boldsymbol{M}^{-1}\boldsymbol{\nabla}_{\boldsymbol{q}}V(\boldsymbol{q}_{a,i}^{n+1})}{N}-\frac{\mathrm{d}t}{2}\boldsymbol{M}^{-1}\boldsymbol{\lambda}_v^{n+1}$$

that is once again a trivial linear equation for $\boldsymbol{M}^{-1}\boldsymbol{\lambda}_v^{n+1}$. The correctness of the algorithm is checked both qualitatively thanks to the video visualization (some frames are shown in Fig 4) and quantitatively by controlling that the energy remains conserved and plotting explicitly the position of the fixed centroid, as shown in Fig 5a and Fig 5b.

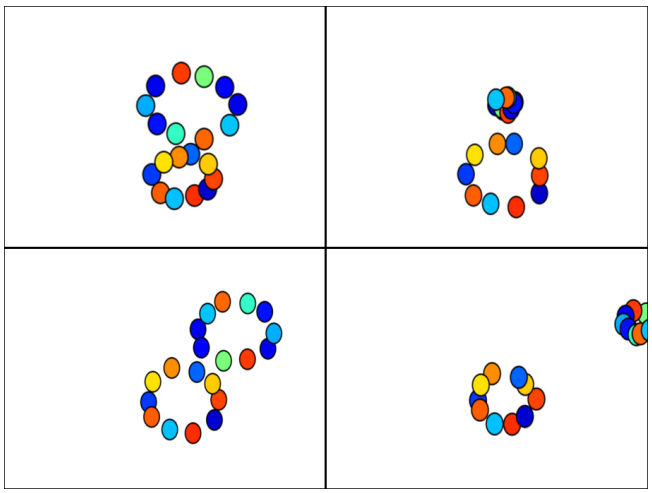
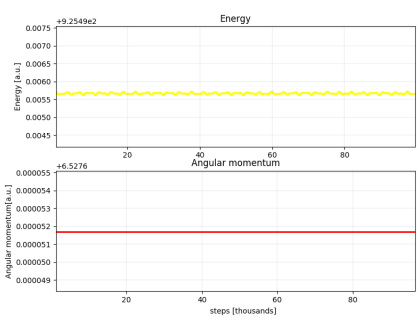
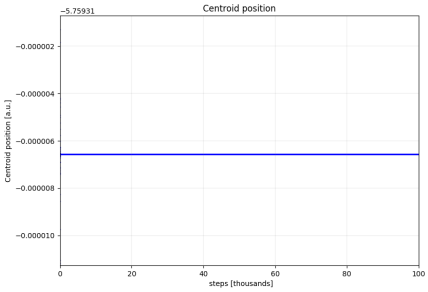


Figure 4: Some random frames of the classical evolution of 2 necklaces, one of which has his centroid constrained.




(a) Energy and angular momentum.




(b) Centroid position conservation (along x-axis).

Figure 5: Rattle algorithm for fixed centroid results.

Fixing the position in a place different than $\mathbf{0}$ is straightforward and simply account for changing $\boldsymbol{\lambda}_r$ formula by $\boldsymbol{c}^n \to \boldsymbol{c}^n - \boldsymbol{c}_0$.

### 2.b.2 Fixed length between centroids

By the same token, in the quadratic case

$$g_1(\boldsymbol{q}) = \frac{1}{2}\left(\left|\boldsymbol{c}_a^{n+1} - \boldsymbol{c}_b^{n+1}\right|^2 - l^2\right) = 0$$

since it's only one constraint we'll have only two Lagrange multiplier $\lambda_r$ and $\lambda_v$. By direct derivation it's clear that

$$(\boldsymbol{G}(\boldsymbol{q})\boldsymbol{\lambda}_r)_{c,i} = \begin{cases} \mathbf{0}, & c \neq a \wedge c \neq a \\ (\boldsymbol{c}_a - \boldsymbol{c}_b)\boldsymbol{\lambda}_r, & c = a \\ (\boldsymbol{c}_b - \boldsymbol{c}_a)\boldsymbol{\lambda}_r, & c = b \end{cases}$$

and the first constraint is rewritten as

$$\left|(\boldsymbol{c}_a^n - \boldsymbol{c}_b^n) + \mathrm{d}t\,(\boldsymbol{v}_{ca}^n - \boldsymbol{v}_{cb}^n) - \frac{\mathrm{d}t^2}{2}\boldsymbol{M}^{-1}\sum_i^N\left(\frac{\boldsymbol{\nabla}_{\boldsymbol{q}}V(\boldsymbol{q}_{a,i}^n) - \boldsymbol{\nabla}_{\boldsymbol{q}}V(\boldsymbol{q}_{b,i}^n)}{N}\right) - \frac{\mathrm{d}t^2}{2}\boldsymbol{M}^{-1}\frac{(\boldsymbol{c}_a^n - \boldsymbol{c}_b^n)}{N}\lambda_r\right|^2 - l^2 = 0$$

Thus defining

$$\boldsymbol{A} = (\boldsymbol{c}_a^n - \boldsymbol{c}_b^n) + \mathrm{d}t\,(\boldsymbol{v}_{ca}^n - \boldsymbol{v}_{cb}^n) - \frac{\mathrm{d}t^2}{2}\boldsymbol{M}^{-1}\sum_i^N\left(\frac{\boldsymbol{\nabla}_{\boldsymbol{q}}V(\boldsymbol{q}_{a,i}^n) - \boldsymbol{\nabla}_{\boldsymbol{q}}V(\boldsymbol{q}_{b,i}^n)}{N}\right)$$

$$\boldsymbol{B} = \frac{\mathrm{d}t^2}{2}\boldsymbol{M}^{-1}\frac{(\boldsymbol{c}_a^n - \boldsymbol{c}_b^n)}{N}$$

one get

$$|\boldsymbol{A} + \boldsymbol{B}\lambda_r|^2 - l^2 = \boldsymbol{A}^2 - \boldsymbol{B}^2\lambda_r^2 + 2\boldsymbol{A}\cdot\boldsymbol{B}\lambda_r - l^2 = 0$$

Which is just a second-order equation for $\lambda_r$ with solutions

$$\lambda_r = \frac{-\boldsymbol{A}\cdot\boldsymbol{B} \pm \sqrt{(\boldsymbol{A}\cdot\boldsymbol{B})^2 - \boldsymbol{B}^2(\boldsymbol{A}^2 - l^2)}}{\boldsymbol{B}^2}$$

For $\mathrm{d}t = 0$ there is no constraining force, assuming $\boldsymbol{\lambda}_r(\mathrm{d}t)$ as an analytical function of $\mathrm{d}t$, for small enough values od $\mathrm{d}t$ such multipliers must be small too, thus the appropriate solution is the one with smaller absolute value. The second constrain is

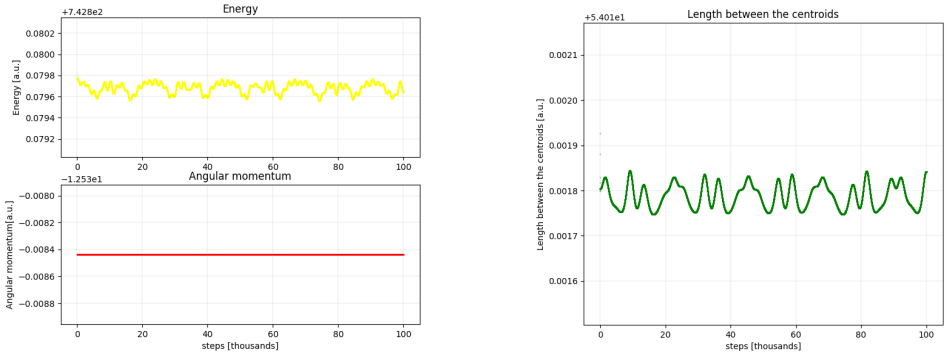
$$\left(\boldsymbol{c}_a^{n+1} - \boldsymbol{c}_b^{n+1}\right)\cdot\left(\boldsymbol{v}_{ca}^{n+1} - \boldsymbol{v}_{cb}^{n+1}\right) = \boldsymbol{0}$$

$$\left(\boldsymbol{c}_a^{n+1} - \boldsymbol{c}_b^{n+1}\right)\cdot\left[\boldsymbol{v}_{ca}^{n+\frac{1}{2}} - \boldsymbol{v}_{cb}^{n+\frac{1}{2}} - \frac{\mathrm{d}t}{2}\boldsymbol{M}^{-1}\left(\frac{\sum_i \boldsymbol{\nabla}_{\boldsymbol{q}}V(\boldsymbol{q}_{a,i}^n) - \boldsymbol{\nabla}_{\boldsymbol{q}}V(\boldsymbol{q}_{b,i}^n)}{N}\right) + \left(\boldsymbol{c}_a^{n+1} - \boldsymbol{c}_b^{n+1}\right)\lambda_v\right] = \boldsymbol{0}$$

Which is simply a linear equation and can be easily solved yielding

$$\lambda_v = \frac{\left(\boldsymbol{c}_a^{n+1} - \boldsymbol{c}_b^{n+1}\right)}{\left|\boldsymbol{c}_a^{n+1} - \boldsymbol{c}_b^{n+1}\right|^2}\cdot\left[\boldsymbol{v}_{ca}^{n+\frac{1}{2}} - \boldsymbol{v}_{cb}^{n+\frac{1}{2}} - \frac{\mathrm{d}t}{2}\boldsymbol{M}^{-1}\left(\frac{\sum_i \boldsymbol{\nabla}_{\boldsymbol{q}}V(\boldsymbol{q}_{a,i}^n) - \boldsymbol{\nabla}_{\boldsymbol{q}}V(\boldsymbol{q}_{b,i}^n)}{N}\right)\right]$$

The algorithm has been implemented and results concerning the conservation of the proper quantities are shown in Fig 6a and Fig 6b, while some random frames in order to visualize the motion of the classical system can be seen in Fig 7.




(a) Energy and angular momentum.

(b) Length between centroid conservation.

Figure 6: Rattle algorithm for fixed-length between centroids results.

### 2.b.3 Combining constrains

I would like to stress out that combining constrains is actually straightforward. Since the Lagrange multipliers play the role of the constraining forces, they can be simply added numerically all together thanks to the superposition principle, this means that we don't need to implement more complex algorithm for multiple constraints, we simply need to evaluate the formulas for $\lambda_r$ and $\lambda_v$ for every constraint and then add all of them together.

## 2.c PIMD implementation

Notice that in this PIMD evolution, the Hamiltonian can be decomposed as

$$H_n(x_1,\ldots,x_N,p_1,\ldots,p_N) = \sum_{i=1}^{N}\frac{p_i^2}{2m} + \frac{m\omega_N^2}{2}(x_{i+1} - x_i)^2 + V(x_i) = H_n^0 + V_n$$

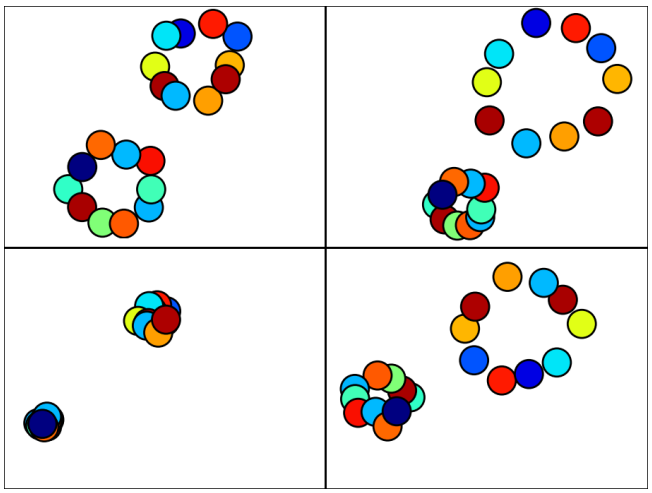


Figure 7: Some random frames of the classical evolution of 2 necklaces with fixed distance between centroids.

where $H_n^0$ is the Hamiltonian associated with a free necklace evolution and $V_n = \sum_i V(x_i)$. Then the following decomposition is called Trotter splitting [3]

$$e^{\mathcal{L}t} = \lim_{P\to\infty} \left( e^{\mathcal{L}^0 \frac{t}{P}} e^{\mathcal{L}_V \frac{t}{P}} \right)^P$$

where $\mathcal{L} = \mathcal{L}^0 + \mathcal{L}_V$ are the Liouvillian associated to $H_n = H_n^0 + V_n$. The symmetric version is called symmetric trotter splitting

$$e^{\mathcal{L}t} = \lim_{P\to\infty} \left( e^{\mathcal{L}^0 \frac{t}{2P}} e^{\mathcal{L}_V \frac{t}{P}} e^{\mathcal{L}^0 \frac{t}{2P}} \right)^P$$

and in case of multiple splitting (which is going to happen later) yield to improved precision. So we can evolve separately the free necklace before and then the free replicas (ignoring the harmonic springs connecting them) inside the potential $V_n$ at each step.

Notice that Trotter splitting introduces an error in the propagation since we cannot reach $P = \infty$ (which would be the same as imposing $\mathrm{d}t = 0$ in our algorithm), but the error introduced vanishes as $\mathrm{d}t \to 0$. The advantage of such splitting is that the motion of the free necklace is that of a set of harmonic oscillators so we can analytically solve it.
In the free case the equation of motion for the replicas are

$$\frac{\mathrm{d}^2 x_i}{\mathrm{d}t^2} = -\omega_N^2 (2x_i - x_{i+1} - x_{i-1})$$

and we can guess the form of normal modes searching for discrete waves propagation.

$$\tilde{x}_k = \sum_j e^{i\alpha_k j} x_j$$

$$x_k = \frac{1}{N} \sum_j e^{-i\alpha_k j} \tilde{x}_j$$

$$\tilde{x}_k = \tilde{x}_k^0 e^{-i\omega_k t}$$

direct substitution in the equation of motion yield

$$\omega_k = 2\omega_N \left| \sin\left(\frac{\alpha_k}{2}\right) \right|$$

while the requirement of cyclicity $x_N = x_0$ imposes

$$\alpha_k = \frac{2\pi k}{N}$$

Transformation to normal modes is not really expensive since it can be done with a simple fast Fourier transform.

Anyway it should be noted that the eigenfrequencies are degenerate since $\omega_k = \omega_{N-k}$, so in general to avoid the usage of complex numbers in the algorithm linear combination of the degenerate $\tilde{x}$ are chosen so that (assume $N$ even for simplicity)

$$C_{jk} = \begin{cases} \sqrt{\frac{1}{N}}, & k = 0 \\ \sqrt{\frac{2}{N}}\cos(2\pi jk/N), & 1 \le k \le \frac{N}{2} - 1 \\ \sqrt{\frac{1}{N}}(-1)^j, & k = \frac{N}{2} \\ \sqrt{\frac{2}{N}}\sin(2\pi jk/N), & \frac{N}{2} + 1 \le k \le N \end{cases}$$

$$\tilde{\boldsymbol{q}}_k = \sum_j C_{jk}\boldsymbol{q}_j$$

$$\tilde{\boldsymbol{p}}_k = \sum_j C_{jk}\boldsymbol{p}_j$$

$$\boldsymbol{q}_j = \sum_j C_{jk}\tilde{\boldsymbol{q}}_k$$

$$\boldsymbol{p}_j = \sum_j C_{jk}\tilde{\boldsymbol{p}}_k$$

Since now the normal modes represent independent harmonic oscillator we can solve exactly their motion, thanks to this consideration our improved version of the Stormer-Verlet algorithm is

**PIMD algorithm**

1. $$\boldsymbol{p}_k \leftarrow \boldsymbol{p}_k - \boldsymbol{\nabla}_q V_n(\boldsymbol{q}_k)\frac{\mathrm{d}t}{2}$$

2. $$\begin{pmatrix} \tilde{\boldsymbol{p}}_k \\ \tilde{\boldsymbol{q}}_k \end{pmatrix} \leftarrow \begin{pmatrix} \cos(\omega_k dt) & -m\omega_k \sin(\omega_k dt) \\ \frac{1}{m\omega_k}\sin(\omega_k dt) & \cos(\omega_k) \end{pmatrix} \begin{pmatrix} \tilde{\boldsymbol{p}}_k \\ \tilde{\boldsymbol{q}}_k \end{pmatrix} \tag{1}$$

3. $$\boldsymbol{p}_k \leftarrow \boldsymbol{p}_k - \boldsymbol{\nabla}_q V_n(\boldsymbol{q}_k)\frac{\mathrm{d}t}{2}$$

This idea is particularly useful if the frequency $\omega_N$ is the biggest in the system (which is always the case since $\omega_N \propto N$) since this evolution is exact, no error is introduced in the evolution of the free necklace and bigger time steps can be used without affecting the results significantly (in common MD algorithms it's usually chosen $\mathrm{d}t \sim 0.05\frac{1}{\omega_{\max}}$ but in our case we can use $\mathrm{d}t \sim 0.05\frac{1}{\omega_{\mathrm{ext}}}$ improving considerably the efficiency of the algorithm).

## 2.d Rattle combined with PIMD

Applying the rattle to the PIMD algorithm is not entirely trivial since analytical equation is messy, for example it can be shown that

$$\boldsymbol{q}_i^{n+1} = \sum_k \sum_j C_{ki}C_{kj}\left[\cos(\omega_k \mathrm{d}t)\boldsymbol{q}_j + \frac{1}{\omega_k}\sin(\omega_k \mathrm{d}t)\left(\boldsymbol{v}_j^n - \frac{\mathrm{d}t}{2}\boldsymbol{M}^{-1}\boldsymbol{\nabla}_{\boldsymbol{q}}V_n(\boldsymbol{q}_j) - \frac{\mathrm{d}t}{2}\boldsymbol{M}^{-1}\boldsymbol{G}(\boldsymbol{q})^T\boldsymbol{\lambda}_r\right)\right]$$

So that in theory one could solve the implicit equation $\boldsymbol{g}(\boldsymbol{q}^{n+1}) = \boldsymbol{0}$ analytically for the constraint and find the corresponding Lagrange multipliers. This is done explicitly for general linear (appendix A) and quadratic (appendix B) constraints.

It might be useful to know that this might not be always necessary and need to be implemented only if high efficiency is required, in fact since Stormer-Verlet algorithm is an approximation of the version with the normal mode trick, and it's true that in the limit $\mathrm{d}t \to 0$ the two algorithms become identical, we could just use the expression of $\boldsymbol{\lambda}_r$ and $\boldsymbol{\lambda}_v$ found before, this work because the small error introduced does not propagate in time since the constrain are reimposed at each step.

The effect of using this simplified version is that the conservation of the constrained quantity will have slightly higher fluctuation around the preserved value but that can be made smaller by diminishing the value of $dt$.

## 2.e Thermostatting and PIMD implementation

Let's ignore for the moment the Rattle implementation and focus on the Stormer-Verlet algorithm, if we want to use it to have a PIMD working algorithm we must add a thermostat to guarantee canonical distribution. As we saw, for white noise we have the Fokker-Plank equation

$$\frac{\partial \rho}{\partial t} = -p\frac{\partial \rho}{\partial q} + \frac{\partial}{\partial p}\left[\left(\frac{\mathrm{d}V}{\mathrm{d}q} + \gamma p\right)\rho\right] + \frac{m\gamma}{\beta}\frac{\partial^2 \rho}{\partial p^2} = -\mathcal{L}\rho$$

Where $\mathcal{L}$ is the Liouvillian operator, the fundamental idea is to introduce another Trotter splitting, $\mathcal{L} = \mathcal{L}_\gamma + \mathcal{L}_0$ with

$$\mathcal{L}_\gamma = -\gamma\left(\frac{\partial}{\partial p}p + \frac{m}{\beta}\frac{\partial^2}{\partial p^2}\right)$$

$$\mathcal{L}_0 = \frac{p}{m}\frac{\partial}{\partial q} - \frac{\mathrm{d}V(q)}{\mathrm{d}q}\frac{\partial}{\partial p}$$

$\mathcal{L}_0$ is associated with the motion without a thermostat and we already know how to treat it. $\mathcal{L}_\gamma$ can be integrated explicitly too (in the sense of stochastic integration, by solving the associated Ornstein-Uhlenbeck process) and finally the algorithm adds up to

**PIMD with Langevin thermostat algorithm**

1. $$\tilde{\boldsymbol{p}}_k \leftarrow c_{1k}\tilde{\boldsymbol{p}}_k + \sqrt{\frac{m}{\beta}}c_{2k}\boldsymbol{\xi}_k$$

2. $$\boldsymbol{p}_k \leftarrow \boldsymbol{p}_k - \boldsymbol{\nabla}_q V_n(\boldsymbol{q}_k)\frac{\mathrm{d}t}{2}$$

3. $$\begin{pmatrix}\tilde{\boldsymbol{p}}_k \\ \tilde{\boldsymbol{q}}_k\end{pmatrix} \leftarrow \begin{pmatrix}\cos(\omega_k \mathrm{d}t) & -m\omega_k \sin(\omega_k \mathrm{d}t) \\ \frac{1}{m\omega_k}\sin(\omega_k \mathrm{d}t) & \cos(\omega_k \mathrm{d}t)\end{pmatrix}\begin{pmatrix}\tilde{\boldsymbol{p}}_k \\ \tilde{\boldsymbol{q}}_k\end{pmatrix} \tag{2}$$

4. $$\boldsymbol{p}_k \leftarrow \boldsymbol{p}_k - \boldsymbol{\nabla}_q V_n(\boldsymbol{q}_k)\frac{\mathrm{d}t}{2}$$

5. $$\tilde{\boldsymbol{p}}_k \leftarrow c_{1k}\tilde{\boldsymbol{p}}_k + \sqrt{\frac{m}{\beta}}c_{2k}\boldsymbol{\xi}'_k$$

Where [7]

$$c_{1k} = e^{-\gamma_k \frac{\mathrm{d}t}{2}}\ ,\ c_{2k} = \sqrt{1 - c_{1k}^2}$$

Technically $\gamma_k$ could be chosen arbitrarily and the canonical distribution would come up anyway but this could lead to inefficient sampling.
For harmonic oscillator with frequency $\omega_k$ thermalized with Langevin thermostat it can be shown [10] that the autocorrelation time for the Hamiltonian H is given by

$$\tau_H = \frac{2}{\gamma} + \frac{\gamma}{2\omega_k^2}$$

so that the we can define an adimensional *efficiency*

$$k_H = \frac{1}{\omega_k \tau_H} = 2\left(\frac{2\omega_k}{\gamma} + \frac{\gamma}{2\omega_k}\right)^{-1}$$

whose maximization yield to the optimal sampling for

$$\gamma_k = 2\omega_k$$

Since for $k = 0$ it's $\omega_0 = 0$ we need another value of $\gamma_0$, I used

$$\gamma_0 = \frac{1}{\omega_{\text{ext}}}$$

and noticed that it leads to a good efficiency, but the choice is arbitrary.
The functionality of the algorithm has been tested carefully checking explicitly that for each normal mode, canonical distribution $dP \propto e^{-\beta_N H} \mathrm{d}q^d \mathrm{d}p^d$ (with $d$ number of dimensions of the classical system) is actually achieved. In the case of harmonic external potential $V$ the modes are independent harmonic oscillators, and Boltzmann distribution in terms of energies becomes

$$D(E) = \int \delta(E - E(\boldsymbol{q}, \boldsymbol{p})) \frac{\mathrm{d}q^d \mathrm{d}p^d}{h^d} \propto E^{d-1}$$

$$P(E) = D(E) e^{-\beta E} \propto E^{d-1} e^{-\beta E}$$

that such distribution arises has actually been verified mode for mode by numerically fitting the canonical distribution (with $\beta_N$ only parameter to find in the fit).
Fig 8 refer to the parameter $\beta = 4.a.u.$ $N = 40$ $\beta_N = 0.1a.u.$. $\omega_{\text{ext}}$ is the frequency of the external harmonic oscillator.

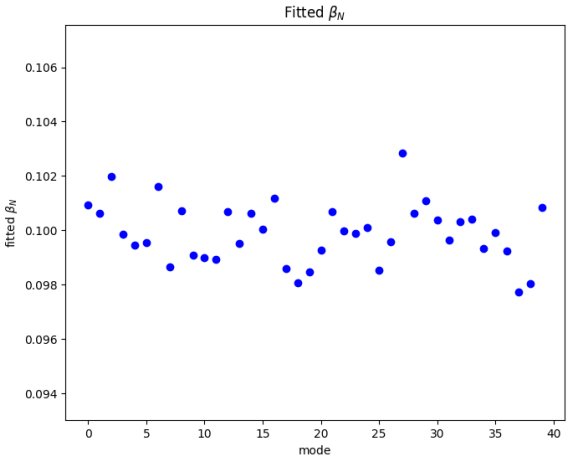
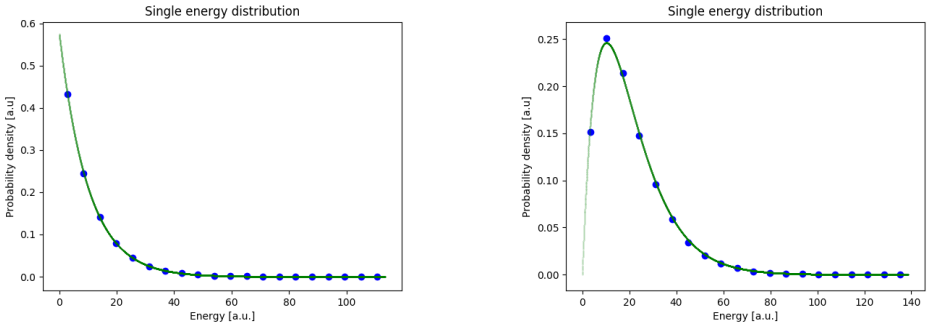



Figure 8: Fitted values of $\beta_N$ for every single mode.

Fig 9a and Fig 9b refer to a specific mode fit (the $5^{\text{th}}$, chosen arbitrarily) and show explicitly the energies distribution in the case of $d = 1, 2$ (obtained binning all the energies achieved during a long-time evolution in 20 bins), the green curve represents the fitted function.




(a) Case $d = 1$.

(b) Case $d = 2$.

Figure 9: Energies distribution.

Since the external potential is the one of an harmonic oscillator we expect to find as the energy of the system,

fixed the inverse temperature $\beta$

$$\mathcal{Z}(\beta) = \sum_{n=0}^{+\infty} e^{-\beta\hbar\omega_{\text{ext}}(n+\frac{1}{2})} = \frac{e^{-\beta\frac{\hbar\omega_{\text{ext}}}{2}}}{1-e^{-\beta\hbar\omega_{\text{ext}}}}$$

$$E(\beta) = -\frac{\mathrm{d}}{\mathrm{d}\beta}\ln(\mathcal{Z}(\beta)) = \frac{\hbar\omega_{\text{ext}}}{2}\coth\left(\frac{\beta\hbar\omega_{\text{ext}}}{2}\right)$$

We also need an estimator for the quantum energy of the system given his PIMD simulation. By computing

$$E_N = -\frac{\mathrm{d}}{\mathrm{d}\beta}\ln(\mathcal{Z}_N(\beta)) = \frac{\sum_{a,i}\left(\frac{1}{2}m_a v_{a,i}^2 - \frac{1}{2}m_i\omega_N^2(q_{a,i}-q_{a,i-1})^2 + V_{\text{ext}}(q_{a,i})\right)}{N}$$

we find the *primitive estimator*, we could also use the simplified version with

$$\frac{\frac{1}{2}\sum_{a,i} m_a v_{a,i}^2}{N} \to \frac{Nd}{2\beta}$$

Both of them are used to provide an estimate of the energy of the harmonic oscillator with $\omega_{\text{ext}} = 1a.u.$ ($\hbar = 1a.u.$) and $\beta = 4a.u.$ in Fig 10.

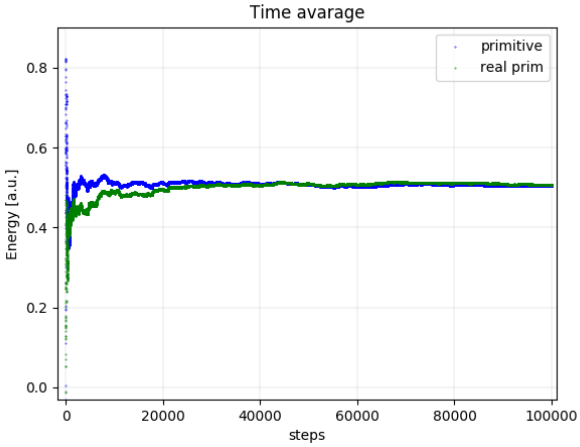



Figure 10: Primitive estimators for the energy of a quantum harmonic oscillator. The blue curve represent the simplified version.

The process is repeated for several values of $\omega_{ext}$ and confronted with the expected values obtained by the previous theoretical calculation ($\beta$ is fixed and has value $\beta = 4$), the results are shown in Fig 11.

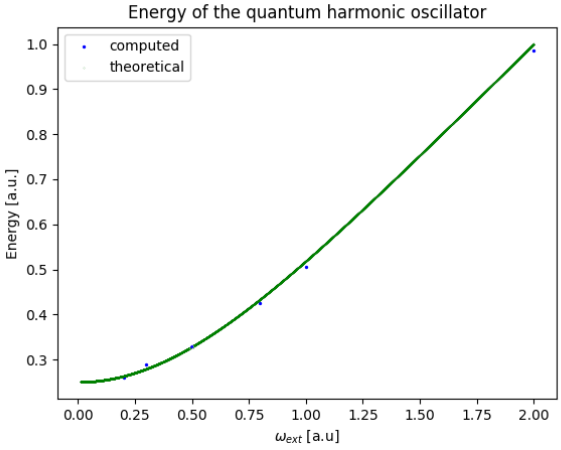



Figure 11: Theoretical and computed values for the energy of a quantum harmonic oscillator.

## 2.f White noise thermostat and Rattle combined

Combining Rattle algorithm and white noise Langevin thermalization yield to a new step

**PIMD with Langevin thermostat and Rattle constraints algorithm**

1.
$$\tilde{\boldsymbol{p}}_k \leftarrow c_{1k}\tilde{\boldsymbol{p}}_k + \sqrt{\frac{m}{\beta}}c_{2k}\boldsymbol{\xi}_k$$

2.
$$\tilde{\boldsymbol{p}}_k \leftarrow c_{1k}\tilde{\boldsymbol{p}}_k + \sqrt{\frac{m}{\beta}}c_{2k}\boldsymbol{\xi}'_k$$

3. Evaluate $\boldsymbol{\lambda}_r$ by solving :
$$\boldsymbol{g}(\boldsymbol{q} + \mathrm{d}t\boldsymbol{v} - \frac{\mathrm{d}t^2}{2}\boldsymbol{M}^{-1}(\boldsymbol{\nabla}_{\boldsymbol{q}}V_n(\boldsymbol{q}) - \boldsymbol{G}(\boldsymbol{q})^T\boldsymbol{\lambda}_r)) = \boldsymbol{0}$$

4.
$$\boldsymbol{p}_k \leftarrow \boldsymbol{p}_k - \boldsymbol{\nabla}_{\boldsymbol{q}}V_n(\boldsymbol{q})\frac{\mathrm{d}t}{2} - (\boldsymbol{G}(\boldsymbol{q})^T\boldsymbol{\lambda}_r)_k\frac{\mathrm{d}t}{2}$$

5.
$$\begin{pmatrix}\tilde{\boldsymbol{p}}_k\\ \tilde{\boldsymbol{q}}_k\end{pmatrix} \leftarrow \begin{pmatrix}\cos(\omega_k\mathrm{d}t) & -m\omega_k\sin(\omega_k\mathrm{d}t)\\ \frac{1}{m\omega_k}\sin(\omega_k\mathrm{d}t) & \cos(\omega_k\mathrm{d}t)\end{pmatrix}\begin{pmatrix}\tilde{\boldsymbol{p}}_k\\ \tilde{\boldsymbol{q}}_k\end{pmatrix} \tag{3}$$

6. Evaluate $\boldsymbol{\lambda}_v$ by solving :
$$\boldsymbol{G}(\boldsymbol{q})(\boldsymbol{v} - \frac{\mathrm{d}t}{2}\boldsymbol{M}^{-1}\boldsymbol{\nabla}_{\boldsymbol{q}}V_n(\boldsymbol{q}) - \frac{\mathrm{d}t}{2}\boldsymbol{M}^{-1}\boldsymbol{G}(\boldsymbol{q})^T\boldsymbol{\lambda}_v) = \boldsymbol{0}$$

7.
$$\boldsymbol{p}_k \leftarrow \boldsymbol{p}_k - \boldsymbol{\nabla}_{\boldsymbol{q}}V_n(\boldsymbol{q})\frac{\mathrm{d}t}{2} - (\boldsymbol{G}(\boldsymbol{q})^T\boldsymbol{\lambda}_v)_k\frac{\mathrm{d}t}{2}$$

I want to comment on this algorithm:

- First of all, the expressions previously found for $\boldsymbol{\lambda}_r$ and $\boldsymbol{\lambda}_v$ still work in this algorithm by simply replacing the previous $\boldsymbol{q}$ and $\boldsymbol{p}$ with the ones obtained after the thermalization since their utility is to reimpose the constrain at every step.
- The two thermalization steps coming from the Trotter splitting have been put together, this doesn't change anything since the step is repeated cyclically, but now when we consider the final values of $\boldsymbol{q}$ and $\boldsymbol{p}$ at the end of each cycle, they obey the constraints since no noise acted after they imposition. Failure to do so doesn't change the motion but introduce a small noise at each step (that doesn't cumulate in subsequent steps though).

The functionality of the algorithm has been checked by evaluating the effective constrained quantity over time as shown in Fig 12 for a couple of necklaces thermalized at $\beta = 4$ and with the length between their centroid constrained (In this case the version of Rattle algorithm for Stormer-Verlet and not the one for PIMD has been applied).
Some frames from the video visualization are shown in Fig 13, there are some qualitative aspects to point out :

- The motion of the replicas inside the necklace appears completely unordered, that's because while in the previous example things have been initialized in a way to visualize more easily the necklace shape, in this case the shape is rapidly lost due the ergodicity of the system.
- One of the necklaces seems always more compact than the other, that's just because in that specific simulation they have different masses ($m_1 = 4$ *a.u.*,$m_2 = 1$ *a.u.*) and by simple statistical mechanics calculation you expect their linear dimension to be proportional to the de Broglie wavelength of the particle that is
$$\Lambda(m, T) = \frac{h}{\sqrt{2\pi m k_B T}}$$

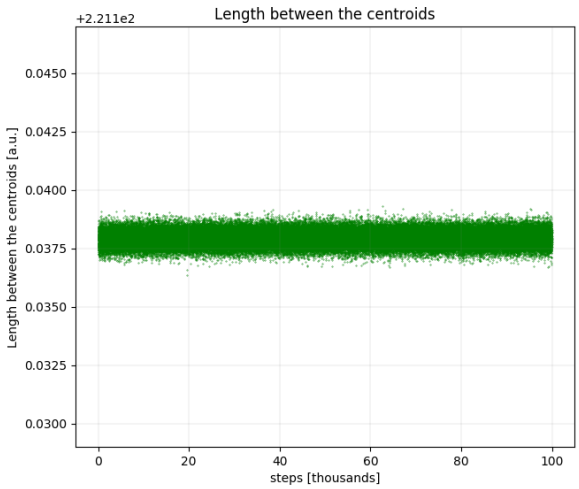



Figure 12: Constrained length of the two centroids

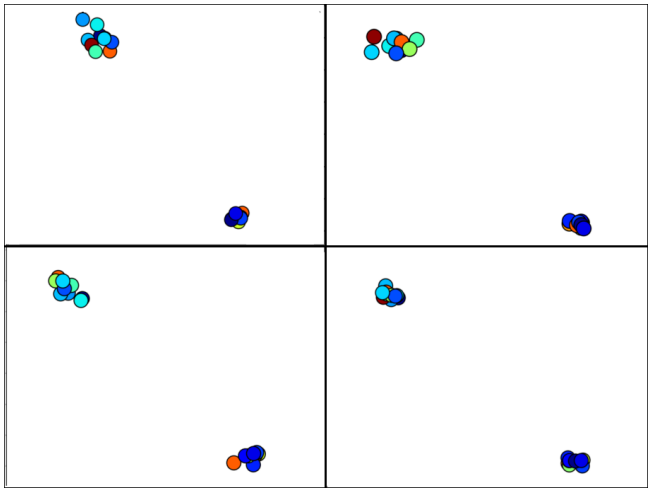


Figure 13: Some frames of the thermalization with rattle algorithm

Notice that since the constraint applies only to the motion of the centroids, the other normal modes of the single centroids are unconstrained and must thus follow Boltzmann distribution as shown in Fig 14 for a specific mode (the $5^{\text{th}}$ in this case) for $d = 2$ .

## 2.g A case of interest

My research team wants to use all this formalism in a ring polymer molecular dynamic (RPMD) simulation. The motion of the necklace, which in PIMD was considered only a mathematical tool to evaluate averages for quantum operators, now is considered a motion in real-time with identification of centroids and quantum particles, some justifications of this idea can be found in [12].
A full investigation of this problem is far beyond the scope of my internship, but at a basic level we would like to be able to simulate molecules using the instruments developed up to now, Rattle algorithm in particular, will be used to constrain translational and rotational degree of freedom of such molecules that otherwise carry some energies that add up to the relevant results.
We'll work with the reaction

$$Mu + H_2 \rightarrow MuH + H$$

Where $Mu$ is muonium, a hydrogen atom in which proton has been replaced by a muon $\mu^+$.
This reaction is particularly relevant because due to lightness of muonium the quantum effects are particularly enhanced and we can thus compare easily the differences with respect to a classical MD simulation. Furthermore, the system is really small and calculation can be done using exact quantum mechanical scattering theory (which

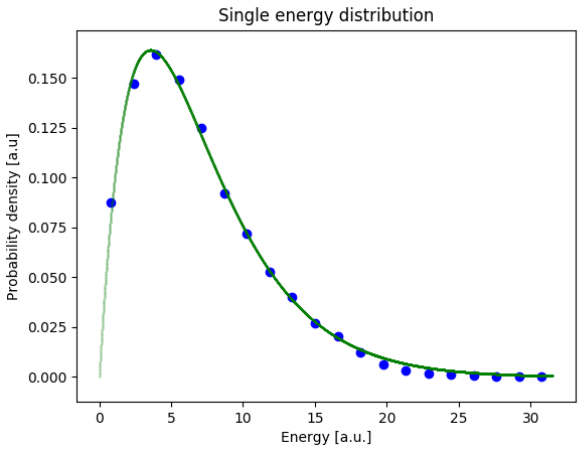



Figure 14: Boltzmann distribution for the energies of a specific mode in the constrained PIMD algorithm.

at the state of art cannot be performed computationally for systems with more than 6 effective degrees of freedom) and thus we can also check the quality of our results. Even though we are working with this particular reaction the algorithm developed is fully functional for generic molecules.
Our approach will be to initialize the two system $Mu$ and $H_2$ so that they sample Boltzmann distribution for their potential evaluated ab initio. Since the potential is really complex we cannot solve the problem analytically as we would do for an harmonic potential. The idea is the following:

- In the beginning, approximate the potential connecting different atoms of the same molecules with an harmonic one (the one obtained by second-order Taylor expansion of the real one) and ignore intermolecular forces by simply initializing different molecules far away.
- Use the thermostats seen previously to initialize the molecules in their ground state (that is high $\beta$).
- Change the potential slowly allowing the system to evolve naturally with the changing Hamiltonian so that for the adiabatic theorem, in the end, the system will be initialized in the ground state of the real potential (the technique is called adiabatic switching)[11].
- Now that the system is correctly initialized a scattering process can be simulated by kicking the muonium in the hydrogen direction.

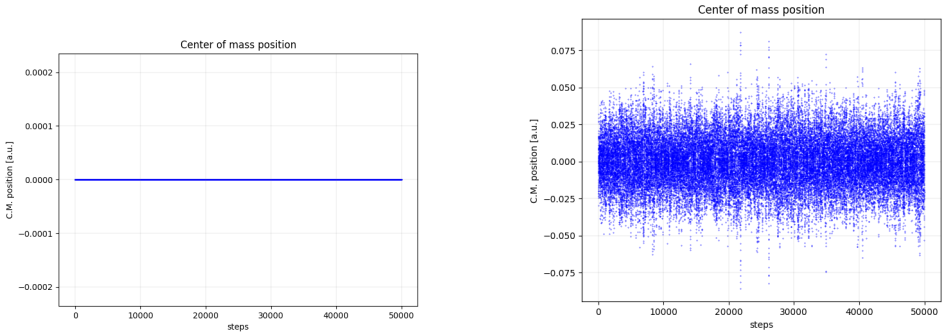
I'll take care of the first two point of the list leaving the rest to my research team.
With respect to what we've seen previously the external potential must be replaced with a many-body one (harmonic in the beginning) so that the initial quantum system is described by

$$H = \sum_{I=1}^{P} \frac{p_I^2}{2m_I} + V(q_1, \dots, q_P)$$

In this case, the PIMD previously seen for monodimensional system generalize to the effective Hamiltonian

$$H_n = \sum_{I=1}^{P}\sum_{i=1}^{N} \frac{(p_I^{(i)})^2}{2m_I} + \sum_{I=1}^{P}\sum_{i=1}^{N} \frac{m_i\omega_N^2}{2}(q_I^{(i)} - q_I^{(i+1)})^2 + \sum_{i}^{N} V(q_1^{(i)}, \dots, q_P^{(i)})$$

So that in general, considered two different atoms, only replicas with the same index $(i)$ interact with each other.
As the number of replicas $N$ increases, the difference between the two versions of the Rattle algorithm emerges, we're forced to use the more complicated one for PIMD to constrain the position of the center of mass with high precision. The difference between the two algorithms emerges for $N \to \infty$ because in that case $\omega_N \to \infty$ and the step $dt \sim 0.05\frac{1}{\omega_{\text{ext}}}$ becomes too big to have proper conservation of the constraint with an approximated algorithm. This is shown in Fig 15a and Fig 15b (A detailed analysis of linear constraints is considered in appendix A for sake of completeness).




(a) Center of mass position for the 'exact' algorithm for PIMD.

(b) Center of mass position for the 'approximate' algorithm for Stormer-Verlet.

Figure 15: Differences between the two Rattle algorithm for fixed center of mass.

It's interesting to work out analytically the math behind this problem to understand what kind of results we should expect from this problem and to understand better the effect of the constraint of the center of mass in the evaluation of the final energy.
The Hamiltonian of the classical system is given by

$$H = \sum_{a=0}^{1}\sum_{k=0}^{N-1}\frac{P_{a,k}^2}{2m} + \sum_{a=0}^{1}\sum_{k=0}^{N-1}\frac{1}{2}m_a\omega_N^2(q_{a,k+1}-q_{a,k})^2 + \sum_{k=0}^{N-1}\frac{1}{2}\mu\omega_{\text{ext}}^2(q_{1,k}-q_{0,k})^2$$

where $\mu$ is the reduced mass of the system.
Considered the normal coordinate of the single free necklace given by $q_j = \sum_j C_{jk}\tilde{q}_k$ we get

$$H = \sum_{a=0}^{1}\sum_{k=0}^{N-1}\frac{\tilde{P}_{a,k}^2}{2m} + \sum_{a=0}^{1}\sum_{k=0}^{N-1}\frac{1}{2}m_a\omega_k^2\tilde{q}_{a,k}^2 + \sum_{k=0}^{N-1}\frac{1}{2}\mu\omega_{\text{ext}}^2(\tilde{q}_{1,k}-\tilde{q}_{0,k})^2$$

Which is a separable problem since the various term with different $k$ doesn't mix with each other. For each $k$ the Hamiltonian is a two-body problem and can be separated again considering

$$\tilde{q}_{c,k} = \frac{m_0\tilde{q}_{0,k} + m_1\tilde{q}_{1,k}}{M}$$

$$\tilde{q}_{d,k} = \tilde{q}_{0,k} - \tilde{q}_{1,k}$$

$$\tilde{p}_{c,k} = \tilde{p}_{0,k} + \tilde{p}_{1,k}$$

$$\tilde{p}_{d,k} = \frac{m_1\tilde{p}_{0,k} - m_0\tilde{p}_{1,k}}{M}$$

$(M = m_0 + m_1)$ yielding

$$H_{c,k} = \frac{\tilde{p}_{c,k}^2}{2M} + \frac{1}{2}M\omega_k^2\tilde{q}_{c,k}^2$$

$$H_{d,k} = \frac{\tilde{p}_{d,k}^2}{2\mu} + \frac{1}{2}\mu\omega_k^2\tilde{q}_{d,k}^2 + \frac{1}{2}\mu\omega_{\text{ext}}^2\tilde{q}_{d,k}^2$$

And since these are the equation of motion for harmonic oscillator we can evaluate properties such as the expectation value of generic quantities for a canonical distribution at inverse temperature $\beta_N = \frac{\beta}{N}$ in particular

$$\langle\tilde{p}_{d,k}^2\rangle = \frac{\mu N}{\beta}$$

$$\langle\tilde{q}_{d,k}^2\rangle = \frac{N}{\beta\mu(\omega_k^2+\omega_{\text{ext}}^2)} = \frac{N}{\beta\mu(4\omega_N^2\sin^2(\frac{k\pi}{N})+\omega_{\text{ext}}^2)}$$

and using the analytic formula

$$\sum_{k=0}^{N} \frac{1}{1+\frac{4N^2}{x^2}\sin^2(\frac{k\pi}{N})} \xrightarrow[N\to\infty]{} \frac{x}{2}\coth\left(\frac{x}{2}\right)$$

It's easy to check that

$$\langle E_{N,d}\rangle = \frac{1}{N}\left\langle \sum_{k=0}^{N-1}\frac{\tilde{P}_{d,k}^2}{2m} - \sum_{k=0}^{N-1}\frac{1}{2}m_a\omega_k^2\tilde{q}_{d,k}^2 + \sum_{k=0}^{N-1}\frac{1}{2}\mu\omega_{\text{ext}}^2\tilde{q}_{d,k}^2 \right\rangle \xrightarrow[N\to\infty]{} \frac{\hbar\omega_{\text{ext}}}{2}\coth\left(\frac{\beta\hbar\omega_{\text{ext}}}{2}\right)$$

And the $E_{N,c}$ estimator can be obtained by $E_{N,d}$ in the limit $\omega_{\text{ext}} \to 0$

$$\langle E_{N,c}\rangle = \frac{1}{N}\left\langle \sum_{k=0}^{N-1}\frac{\tilde{P}_{c,k}^2}{2m} - \sum_{k=0}^{N-1}\frac{1}{2}m_a\omega_k^2\tilde{q}_{c,k}^2 \right\rangle \xrightarrow[N\to\infty]{} \frac{1}{\beta}$$

So that in the end

$$\langle E_N\rangle = \left\langle \frac{1}{N}\sum_{k=0}^{N-1}\left(\sum_{a=0}^{1}\frac{P_{a,k}^2}{2m} - \sum_{a=0}^{1}\frac{1}{2}m_a\omega_N^2(q_{a,k+1}-q_{a,k})^2 + \frac{1}{2}\mu\omega_{\text{ext}}^2(q_{1,k}-q_{0,k})^2\right)\right\rangle \xrightarrow[N\to\infty]{} \frac{1}{\beta}+\frac{\hbar\omega_{\text{ext}}}{2}\coth\left(\frac{\beta\hbar\omega_{\text{ext}}}{2}\right)$$

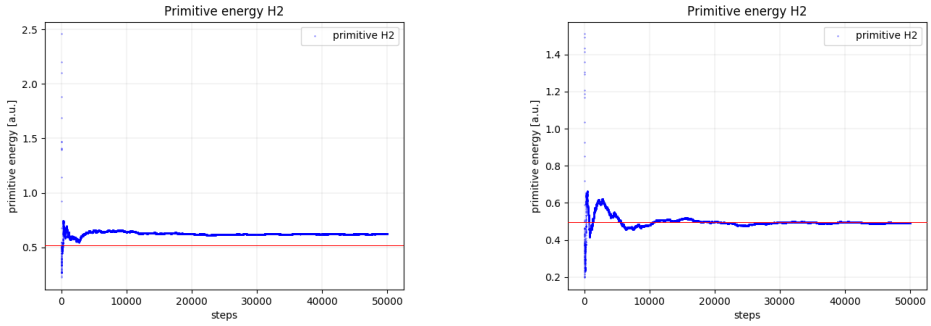
Constraining with rattle the center of mass position is equivalent to saying that $\tilde{p}_{c,0} = 0$, one might suspects that in order to remove the contribute of $E_{N,c}$ one should constrain also the modes with $k \neq 0$ that appear in the formula, but that's not the case, since for $k \neq 0$ it's $\omega_k \neq 0$ and the energy of the harmonic oscillator is split in half between the potential and kinetic energy terms due to the virial theorem so that

$$\langle E_{N,c,k}\rangle = \frac{1}{N}\left\langle \frac{\tilde{P}_{c,k}^2}{2m} - \frac{1}{2}m_a\omega_k^2\tilde{q}_{c,k}^2 \right\rangle \xrightarrow[N\to\infty]{} 0 \text{ for } (k\neq 0)$$

and for the constrained motion

$$\langle E_N\rangle \xrightarrow[N\to\infty]{} \frac{\hbar\omega_{\text{ext}}}{2}\coth\left(\frac{\beta\hbar\omega_{\text{ext}}}{2}\right)$$

As explicitly shown in Fig 16a and Fig 16b for $\beta = 4a.u.$, $\omega_{\text{ext}} = 1a.u.$ where the energy is converging to the expected values evaluated before in the two cases of constrained and unconstrained center of mass.




(a) Unconstrained center of mass.

(b) Constrained center of mass.

Figure 16: Differences in the estimator of energy for the constrained and unconstrained systems. The red line represent the value $\frac{\hbar\omega_{\text{ext}}}{2}\coth\left(\frac{\beta\hbar\omega_{\text{ext}}}{2}\right)$

We would like to be able to constrain also the angular momentum of molecules, this is not possible in general for the rattle algorithm since it is a constraint that involve both positions and velocities, for the particular case of a diatomic molecule like $H_2$ we can use a trick and constrain the angle that the axes of the molecule forms with another fixed axis.
In order to do so, we consider in appendices a generic algorithm to fix distances between points obtained by linear functions of the coordinate.

This can fix the angular momentum in our case. In fact let's say that $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$ are the position of the centroids, then considered

$$\boldsymbol{V}_1 = \frac{\boldsymbol{c}_1 - \boldsymbol{c}_2}{|\boldsymbol{c}_1 - \boldsymbol{c}_2|}$$

$$\boldsymbol{V}_2 = (\boldsymbol{V}_1 \cdot \boldsymbol{\eta})\boldsymbol{\eta} \text{ with } |\boldsymbol{\eta}| = 1$$

Fixing the distance between $\boldsymbol{V}_1$ and $\boldsymbol{V}_2$ amounts to fixing the angle between $\boldsymbol{c}_1 - \boldsymbol{c}_2$ and $\boldsymbol{\eta}$. By imposing such a constraint for two linearly independent axes $\boldsymbol{\eta}_1$, $\boldsymbol{\eta}_2$ we are fixing a unique direction for the line connecting the two centroids and thus eliminating rotations.
It must be noticed that technically the definition of $\boldsymbol{V}_1$ and $\boldsymbol{V}_2$ are not linear on the coordinate $\boldsymbol{q}$ since also the linear coefficient depend on them, this is not a problem since we can simply apply at every step the algorithm described in appendix B and then evaluate the new linear coefficients for the next step.

## 3 Conclusion

A functional PIMD algorithm has been developed from scratches and it has been shown how to use it in combination with a Langevin thermostat, it has been shown that this lead to improved ergodicity of the system that can thus sample properly the canonical distribution along time. Rattle algorithm has been also implemented in combination with the previous two in order to impose constraints and it has been shown the functionality of the final algorithm created on a system of interest. Several types of constraints have been tested to check the effect of the Rattle algorithm.

## Acknowledgment

I would like to thank my supervisors Adrien and Ralph for their support during the various phases of my work and for their helpful advice.

# Appendices

## A Generic linear constraint for PIMD

As we said previously

$$\boldsymbol{q}_i^{n+1} = \sum_k \sum_j C_{ki} C_{kj} \left[ \cos(\omega_k \mathrm{d}t)\boldsymbol{q}_j + \frac{1}{\omega_k} \sin(\omega_k \mathrm{d}t) \left( \boldsymbol{v}_j^n - \frac{\mathrm{d}t}{2} \boldsymbol{M}^{-1} \boldsymbol{\nabla}_{\boldsymbol{q}} V_n(\boldsymbol{q}_j) - \frac{\mathrm{d}t}{2} \boldsymbol{M}^{-1} \boldsymbol{G}(\boldsymbol{q})^T \boldsymbol{\lambda}_r \right) \right]$$

so we define the matrices

$$\bar{C}_{ij} = \sum_k C_{ki} C_{kj} \cos(\omega_k \mathrm{d}t)$$

$$\bar{S}_{ij} = \sum_k C_{ki} C_{kj} \frac{\sin(\omega_k \mathrm{d}t)}{\omega_k}$$

So that a generic linear constraint

$$\sum_i a_i \boldsymbol{q}_i^{n+1} = \boldsymbol{k}$$

rewrites as

$$\boldsymbol{k} = \sum_{i,j} a_i \bar{C}_{ij} \boldsymbol{q}_j + \sum_{i,j} a_i \bar{S}_{ij} \left( \boldsymbol{v}_j - \frac{\mathrm{d}t}{2} \boldsymbol{M}^{-1} \boldsymbol{\nabla}_{\boldsymbol{q}} V_n(\boldsymbol{q}_j) \right) - \sum_{i,j} a_i \bar{S}_{ij} \left( \boldsymbol{M}^{-1} \boldsymbol{G}(\boldsymbol{q})^T \boldsymbol{\lambda}_r \right)_j \frac{\mathrm{d}t}{2}$$

and considered that in this case

$$\left( \boldsymbol{M}^{-1} \boldsymbol{G}(\boldsymbol{q})^T \boldsymbol{\lambda}_r \right)_j = \frac{a_j}{m_j} \boldsymbol{\lambda}_r$$

We can solve explicitly for $\boldsymbol{\lambda}_r$

$$\boldsymbol{\lambda}_r = \frac{2}{dt \sum_{i,j} a_i \bar{S}_{ij} a_j / m_j} \left( \sum_{i,j} a_i \bar{C}_{ij} \boldsymbol{q}_j + \sum_{i,j} a_i \bar{S}_{ij} \left( \boldsymbol{v}_j - \frac{\mathrm{d}t}{2} \boldsymbol{M}^{-1} \boldsymbol{\nabla}_{\boldsymbol{q}} V_n(\boldsymbol{q}_j) \right) - \boldsymbol{k} \right)$$

Evaluation of $\boldsymbol{\lambda}_v$ is much simpler since it's the same as in the simplified version of the algorithm. In particular it's

$$\boldsymbol{v}^{n+1} = \boldsymbol{v}^{n+\frac{1}{2}} - \frac{\mathrm{d}t}{2}\boldsymbol{M}^{-1}\boldsymbol{\nabla}_{\boldsymbol{q}}V_n\left(\boldsymbol{q}^{n+1}\right) - \frac{\mathrm{d}t}{2}\boldsymbol{M}^{-1}\boldsymbol{G}\left(\boldsymbol{q}^{n+1}\right)\boldsymbol{\lambda}_{\boldsymbol{v}}$$

and since the constraint is

$$\boldsymbol{G}\left(\boldsymbol{q}^{n+1}\right)\boldsymbol{v}^{n+1} = 0$$

We get

$$\boldsymbol{0} = \sum_i a_i\left(\boldsymbol{v}_i^{n+\frac{1}{2}} - \frac{\mathrm{d}t}{2}\boldsymbol{M}^{-1}\boldsymbol{\nabla}_{\boldsymbol{q}}V_n\left(\boldsymbol{q}_i^{n+1}\right)\right) - \frac{\mathrm{d}t}{2}\sum_i \frac{a_i^2}{m_i}\boldsymbol{\lambda}_v$$

And finally

$$\boldsymbol{\lambda}_v = \frac{2}{\mathrm{d}t\sum_i \frac{a_i^2}{m_i}}\sum_i a_i\left(\boldsymbol{v}_i^{n+\frac{1}{2}} - \frac{\mathrm{d}t}{2}\boldsymbol{M}^{-1}\boldsymbol{\nabla}_{\boldsymbol{q}}V_n\left(\boldsymbol{q}_i^{n+1}\right)\right)$$

# B Generic distance constraint for PIMD

Let's consider the two points

$$\boldsymbol{c}_1 = \sum_i a_i\boldsymbol{q}_i + \boldsymbol{k}_1$$

$$\boldsymbol{c}_2 = \sum_i b_i\boldsymbol{q}_i + \boldsymbol{k}_2$$

define $d_i = a_i - b_i$ and $\boldsymbol{k} = \boldsymbol{k}_1 - \boldsymbol{k}_2$ and the constrain for their distances

$$g_1(\boldsymbol{q}) = \frac{1}{2}\left(|\boldsymbol{c}_1 - \boldsymbol{c}_2|^2 - l^2\right) = \frac{1}{2}\left(\left|\sum_i d_i\boldsymbol{q}_i + \boldsymbol{k}\right|^2 - l^2\right) = 0$$

$$\left(\boldsymbol{G}(\boldsymbol{q})^T\boldsymbol{\lambda}_r\right)_i = \left(\sum_k d_k\boldsymbol{q}_k + \boldsymbol{k}\right)d_i\lambda_r$$

We need a compact notation since formulas are going to become really complicated in a while, we thus define

$$\boldsymbol{\alpha}_i = \sum_j \bar{C}_{ij}\boldsymbol{q}_j + \sum_j \bar{S}_{ij}\left(\boldsymbol{v}_j - \frac{\mathrm{d}t}{2}\boldsymbol{M}^{-1}\boldsymbol{\nabla}_{\boldsymbol{q}}V_n(\boldsymbol{q}_i)\right)$$

$$\boldsymbol{\delta}_i = \sum_j \bar{S}_{ij}\left(\sum_k d_k\boldsymbol{q}_k + \boldsymbol{k}\right)\frac{d_j}{m_j}$$

$$\boldsymbol{q}_i^{n+1} = \boldsymbol{\alpha}_i^n - \boldsymbol{\delta}_i^n\lambda_r$$

So that applying the constrain now yield to a simple expression

$$0 = 2g_1(\boldsymbol{q}^{n+1}) = \left|\sum_i d_i\left(\boldsymbol{\alpha}_i^n - \boldsymbol{\delta}_i^n\lambda_r\right) + \boldsymbol{k}\right|^2 - l^2$$

Thus as we've done in the past for the simpler classical problem we define the two quantities

$$\boldsymbol{A} = \sum_i d_i\boldsymbol{\alpha}_i^n + \boldsymbol{k}$$

$$\boldsymbol{B} = \sum_i d_i\boldsymbol{\delta}_i^n$$

So that in the end

$$\lambda_r = \frac{-\boldsymbol{A}\cdot\boldsymbol{B} \pm \sqrt{(\boldsymbol{A}\cdot\boldsymbol{B})^2 - \boldsymbol{B}^2(\boldsymbol{A}^2 - l^2)}}{\boldsymbol{B}^2}$$

And as explained previously, for small enough $dt$ the correct solution is the smaller of the two. For the evaluation of $\lambda_v$ we have the second constraint equation

$$\boldsymbol{G}(\boldsymbol{q}^{n+1})\boldsymbol{v}^{n+1} = 0$$

That simply is

$$\left(\sum_k d_k \boldsymbol{q}_k^{n+1} + \boldsymbol{k}\right) \cdot \left(\sum_i d_i \left(\boldsymbol{v}_i^{n+\frac{1}{2}} - \frac{\mathrm{d}t}{2}\boldsymbol{M}^{-1}\boldsymbol{\nabla}_{\boldsymbol{q}} V_n(\boldsymbol{q}^{n+1}) - \frac{\mathrm{d}t}{2}\boldsymbol{M}^{-1}\boldsymbol{G}(\boldsymbol{q}^{n+1})\lambda_v\right)\right) = 0$$

and by direct resolution

$$\lambda_v = \frac{2}{dt}\frac{\left(\sum_k d_k \boldsymbol{q}_k^{n+1} + \boldsymbol{k}\right) \cdot \left(\sum_i d_i \left(\boldsymbol{v}_i^{n+\frac{1}{2}} - \frac{\mathrm{d}t}{2}\boldsymbol{M}^{-1}\boldsymbol{\nabla}_{\boldsymbol{q}} V_n(\boldsymbol{q}^{n+1})\right)\right)}{\left(\sum_k d_k \boldsymbol{q}_k^{n+1} + \boldsymbol{k}\right) \cdot \left(\boldsymbol{M}^{-1}\boldsymbol{G}(\boldsymbol{q}^{n+1})\right)}$$

# References


[1] 'Path Integration via Molecular Dynamics', *Quantum Simulations of Complex Many-Body Systems: From Theory to Algorithms, Lecture Notes*, vol. 10, pp. 269-298, 2002 *M.E. Tuckerman*

[2] 'Handbook of stochastic methods for physics, chemistry and Natural Science', 1985, *C.W. Gardiner*

[3] 'On the product of semi-groups of operators', *Proceedings of the American Mathematical Society*, vol. 10, No. 4, 1959, pp. 545-551, *H.F. Trotter*

[4] 'Geometric Numerical Methods for Hamiltonian Mechanics', pp. 185-210, 2003 *B. Leimkuhler, S. Reich*

[5] 'A unified formulation of the constant temperature molecular dynamics methods',*J. Chem. Phys.*, vol 81, 1984 *S. Nosé*

[6] 'Nosé–Hoover chains: The canonical ensemble via continuous dynamics', *J. Chem. Phys.*, vol 97, 1992 *G.J. Martyna, M. L. Klein, and M.E. Tuckerman*

[7] 'Accurate sampling using Langevin dynamics', *Phys. Rev.* vol. 75, 2007 *G. Bussi, M. Parrinello*

[8] 'Efficient stochastic thermostatting of path integral molecular dynamics', *J. Chem. Phys* vol. 133, 2010 *M. Ceriotti, M. Parrinello, T. Markland, D. E. Manolopoulos*

[9] 'Common Molecular Dynamics Algorithms Revisited: Accuracy and Optimal Time Steps of Stormer-Leapfrog Integrators', *J. Comp. Phys.* vol. 136, 1997 *A. K. Mazur*

[10] 'Statistical methods in atomistic computer simulations' *M. Ceriotti*

[11] 'Revisiting Adiabatic Switching for Initial Conditions in Quasi- Classical Trajectory Calculations: Application to CH4', *J. Chem. Phys.* 2016 *C. Qu, J. M. Bowman*

[12] 'Boltzmann-conserving classical dynamics in quantum time-correlation functions: "Matsubara dynamics"', *J. Chem. Phys.* vol. 142, 2015 *T. J. H. Hele, M. J. Willatt, A. Muolo and S. C. Althorpeb*