Path-Integral Monte Carlo Worm Algorithm for Bose Systems with Periodic Boundary Conditions

Gabriele Spada [1,*] , Stefano Giorgini [1] and Sebastiano Pilati [2,3]

1. Dipartimento di Fisica, Università di Trento, CNR-INO BEC Center, 38123 Trento, Italy; stefano.giorgini@unitn.it
2. School of Science and Technology, Physics Division, Università di Camerino, 62032 Camerino, Italy; sebastiano.pilati@unicam.it
3. INFN-Sezione di Perugia, 06123 Perugia, Italy
* Correspondence: gabriele.spada@unitn.it

Abstract: We provide a detailed description of the path-integral Monte Carlo worm algorithm used to exactly calculate the thermodynamics of Bose systems in the canonical ensemble. The algorithm is fully consistent with periodic boundary conditions, which are applied to simulate homogeneous phases of bulk systems, and it does not require any limitation in the length of the Monte Carlo moves realizing the sampling of the probability distribution function in the space of path configurations. The result is achieved by adopting a representation of the path coordinates where only the initial point of each path is inside the simulation box, the remaining ones being free to span the entire space. Detailed balance can thereby be ensured for any update of the path configurations without the ambiguity of the selection of the periodic image of the particles involved. We benchmark the algorithm using the non-interacting Bose gas model for which exact results for the partition function at finite number of particles can be derived. Convergence issues and the approach to the thermodynamic limit are also addressed for interacting systems of hard spheres in the regime of high density.

Keywords: path-integral; Monte Carlo; periodic boundary conditions; worm algorithm; ultracold Bose gases

1. Introduction

The path-integral Monte Carlo (PIMC) method is a computational approach that allows one to exactly calculate the equilibrium properties of Bose systems at finite temperature starting from the microscopic Hamiltonian. The first applications of the method addressed the study of bulk liquid and solid ^{4}He approaching the quantum degenerate regime [1,2]. Similarly to these first simulations, many later implementations of the PIMC algorithm addressed homogeneous systems featuring infinite spatial extension in the physically relevant dimensions. To mimic such configurations, the use of periodic boundary conditions in the computer simulations is crucial [3]. The size of the simulation cell enters as an important parameter of the numerical procedure, calling for a careful extrapolation of all the results to a properly defined thermodynamic limit. Another crucial aspect of PIMC simulations is the efficient sampling of Bose-particle permutations. This gets increasingly important with the level of quantum degeneracy and is essential to obtain reliable estimates of observables such as the superfluid density and the condensate fraction [4,5]. In this context, an important technical advancement emerged with the introduction of the worm algorithm, first devised for lattice models [6,7] and later extended to continuous-space simulations [8,9]. The PIMC method implementing the worm algorithm has proven to be one of the most powerful computational quantum many-body techniques. It allowed performing accurate simulations of intriguing phenomena in different condensed matter systems, such as dipolar systems [10–12], ultracold gases [13–17] and quantum fluids and solids [18–24]. However, the original implementation of the worm algorithm is not fully

compatible with periodic boundary conditions. This can lead to biased results when the de Broglie thermal wavelength starts to be comparable to the size of the fundamental periodic cell. The basic idea behind the worm algorithm is the use of both diagonal and off-diagonal configurations of the paths describing particles in the many-body system. Various moves of portions of the paths are devised to ensure the ergodic sampling of both types of configurations as well as to switch from the diagonal to the off-diagonal sector and vice-versa. Such moves, satisfying detailed balance, have a straightforward implementation in the case of a system with infinite extension or of a confined finite geometry. In periodic systems, instead, the ambiguity in the choice of periodic images might lead to biased results, unless stringent constraints are imposed on the portions of the paths involved in certain Monte Carlo moves. Such bias can be avoided by adopting periodic cells much larger than the thermal wavelength, but this becomes impractical in the zero temperature limit. Furthermore, the limitations in the updates can affect the efficiency of the Monte Carlo sampling. We present here a formalism and a detailed recipe for a PIMC algorithm for bosons in the canonical ensemble, which is rigorously compatible with periodic boundary conditions. With this algorithm, no constraint has to be imposed on the Monte Carlo updates, even for small periodic cells. We use as a benchmark the canonical non-interacting Bose gas in a periodic box for which exact results for the energy can be obtained for any number of particles in the box. Direct comparison with these results permits us to check one by one the various elementary moves of our novel implementation of the worm algorithm, verifying that they provide an unbiased ergodic sampling. Interacting systems are considered only with the hard-sphere model interaction and the pair-product approximation. In this case, no exact result is available for finite box-like geometries even for just a pair of particles. Nonetheless, we can investigate the convergence of the results for a given number of particles in terms of the length of the path steps in imaginary time as well as the approach to the thermodynamic limit when the number of particles in the simulation is increased.

The structure of the paper is as follows. In Section 2, we introduce the representation of the particle coordinates consistent with the use of periodic boundary conditions. Trying to be pedagogical, we first consider the case of a single particle in a one-dimensional periodic box and then move to the case of N identical particles. In the same section, we describe the general scheme of the worm algorithm and we introduce in details the various Monte Carlo updates. In Section 3, we benchmark the algorithm against exact results of the non-interacting Bose gas for one, two and many particles at different temperatures. Section 4 is devoted to interacting systems for which we use the hard-sphere model in the regime of high density. We address the convergence of the algorithm for a given number of particles N and for different temperatures, as well as the approach to the thermodynamic limit. Finally, in the last section, we draw our conclusions. In Appendix A, we outline the useful formulas used for the calculation of the internal energy, both the thermodynamic and virial estimator, and of the pressure.

2. Path Integral Monte Carlo

In a PIMC simulation, one aims at calculating the partition function Z_N of a Bose system of N identical particles described by the Hamiltonian H and with inverse temperature $\beta = 1/(k_B T)$, where k_B is Boltzmann's constant. In the coordinate representation, the partition function is defined as the trace over all states $|\mathbf{R}\rangle$ of the density matrix $\rho(\mathbf{R}, \mathbf{R}', \beta) = \langle \mathbf{R} | e^{-\beta H} | \mathbf{R}' \rangle$ properly symmetrized,

$$Z_N = \frac{1}{N!} \sum_P \int d\mathbf{R} \, \rho(\mathbf{R}, P\mathbf{R}, \beta) \, . \tag{1}$$

Here, $\mathbf{R} = (\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_N)$ collectively denotes the position vectors of the particles and $P\mathbf{R} = (\mathbf{r}_{p(1)}, \mathbf{r}_{p(2)}, \ldots, \mathbf{r}_{p(N)})$ corresponds to the position vectors with permuted labels. Furthermore, the sum in the above equation extends over the $N!$ permutations of the particle

labels. The partition function can be conveniently rewritten in terms of the convolution integral

$$Z_N = \frac{1}{N!} \sum_P \int d\mathbf{R} \int d\mathbf{R}_1 \cdots \int d\mathbf{R}_{M-1} \, \rho(\mathbf{R}, \mathbf{R}_1, \delta_\tau) \ldots \rho(\mathbf{R}_{M-1}, P\mathbf{R}, \delta_\tau) \,, \tag{2}$$

where $\delta_\tau = \beta/M$. Starting from the above decomposition, the calculation is mapped to a classical-like simulation of polymeric chains with a number of beads M equal to the number of terms of the convolution integral. More specifically, one makes use of suitable approximations for the density matrix $\rho(\mathbf{R}, \mathbf{R}', \delta_\tau)$ at the higher temperature $1/\delta_\tau$ and performs the multidimensional integration over $\mathbf{R}, \mathbf{R}_1, \ldots, \mathbf{R}_{M-1}$ as well as the sum over permutations P by Monte Carlo sampling [1,2]. The whole procedure can be unambiguously followed if the particle coordinates $\mathbf{r}_1, \ldots, \mathbf{r}_N$ can span the entire space, for example when the Hamiltonian includes a confining external potential. In the case of interest of a bulk system, where the simulation cell corresponds to a box periodically replicated in space, the coordinates $\mathbf{r}_i$ can either denote the position of the i-th particle in the box or of one of its periodic images in the neighboring boxes. This ambiguity has troublesome consequences in the proper sampling of configurations $\mathbf{R}$ with the correct probability distribution. In particular, a naïve implementation of the Monte Carlo updates might lead to the violation of the detailed balance condition. In the following subsection we motivate the use of a specific representation for the particle coordinates that allows one to unambiguously construct a PIMC algorithm that is consistent with periodic boundary conditions. The upshot is that one should consider coordinates as belonging to the infinite space throughout the whole simulation, and invoke periodic boundary conditions only when considering the periodicity of the trajectories in imaginary time or the interaction among different particles. First, we consider the simplest possible example, namely a single particle in one dimension, to show how this coordinate representation naturally emerges in the context of the path-integral representation of the partition function. Once the single particle case has been clarified, we extend the formalism to the N-body system with periodic boundary conditions.

2.1. Path Integral for One Particle with Periodic Boundary Conditions

We consider a particle moving on a circle $\mathcal{S}^1 \sim \mathbb{R}/\mathbb{Z}$ of length L whose momentum is quantized in units of $2\pi\hbar/L$. A position eigenstate on the circle can be represented as the Fourier series

$$|x\rangle_{\mathcal{S}^1} = \frac{1}{\sqrt{L}} \sum_{n=-\infty}^{\infty} e^{-\frac{ip_n x}{\hbar}} |p_n\rangle \,, \qquad p_n = \frac{2\pi\hbar}{L} n \,, \tag{3}$$

but it can also be expressed as the superposition of eigenstates in the space $\mathbb{R}$ as

$$|x\rangle_{\mathcal{S}^1} = \sqrt{\frac{L}{2\pi\hbar}} \sum_{n=-\infty}^{\infty} |x + nL\rangle \,. \tag{4}$$

In the above equations the coordinate x is limited to the fundamental cell $[0, L)$ and we used the notation $|\rangle_{\mathcal{S}^1}$ to represent a state on $\mathcal{S}^1$ while with the standard ket $|\rangle$, without subscripts, we represented a state on the real line $\mathbb{R}$. For later convenience, we introduce the following alternative labeling of the states on $\mathbb{R}$:

$$|x + nL\rangle \equiv |x, n\rangle \,, \tag{5}$$

which explicitly separates the coordinate in the fundamental cell from the integer identifying one of the periodic images. The momentum eigenstate $|p_n\rangle$ does not have this distinction since the plane-wave decomposition can equivalently be expressed in the two spaces as

$$|p_n\rangle = \frac{1}{\sqrt{L}} \int_0^L dx \, e^{\frac{ip_n x}{\hbar}} |x\rangle_{\mathcal{S}^1} = \frac{1}{\sqrt{2\pi\hbar}} \int_{-\infty}^{\infty} dx \, e^{\frac{ip_n x}{\hbar}} |x\rangle \,. \tag{6}$$

We can then write the partition function of a free particle of mass m in one dimension with periodic boundary conditions using the set of coordinates on the real line $\mathbb{R}$:

$$Z_1 = \int_0^L dx \sum_{n=-\infty}^{\infty} \langle x, 0| e^{-\frac{\beta p^2}{2m}} |x, n\rangle \,, \tag{7}$$

where p is the momentum operator. Notice that the above equation computes the trace over the Boltzmann factors by fixing a reference in the fundamental cell and then summing over all periodic images. The reader can easily verify that the above expression evaluates to the same result as the more familiar formula $Z_1 = \sum_n \exp[-\beta p_n^2/(2m)]$. It is also important to notice that if one limits to $n = 0$ the summation over images in Equation (7), i.e., only the particle in the simulation box is considered, the result for Z_1 would be correct only in the limit $\lambda_T \ll L$, where $\lambda_T = \sqrt{2\pi\hbar^2\beta/m}$ is the thermal wavelength. For the particular example of a single particle in a box, this issue explains the difficulty of a path-integral algorithm with periodic boundary conditions [3].

We can then define $\delta_\tau = \beta/M$, with M being a positive integer, and introduce $M-1$ completeness relations on the real line $\mathbb{R}$,

$$1 = \int_{-\infty}^{\infty} dx \, |x\rangle \langle x| \,, \tag{8}$$

to obtain a representation of Z_1 suitable for a PIMC simulation:

$$Z_1 = \int_0^L dx \sum_{n=-\infty}^{\infty} \int_{-\infty}^{\infty} dx_1 \ldots dx_{M-1} \langle x, 0| e^{-\frac{\delta_\tau p^2}{2m}} |x_1\rangle \langle x_1| \ldots |x_{M-1}\rangle \langle x_{M-1}| e^{-\frac{\delta_\tau p^2}{2m}} |x, n\rangle \,. \tag{9}$$

Notice that the expression above involves a product of matrix elements obtained from states defined on the real line $\mathbb{R}$. The periodicity of the space just constrains the leftmost bra to span the fundamental cell, and the rightmost ket to coincide with one of the images labeled by n. Before moving to the more general case of a system of N particles in D-dimensions, let us introduce the analog of the partition function in Equation (7) for non-diagonal (or two-point) configurations,

$$G_1 = \frac{1}{L} \int_0^L dx' \int_0^L dx \sum_{n=-\infty}^{\infty} \langle x, 0| e^{-\frac{\beta p^2}{2m}} |x', n\rangle \,, \tag{10}$$

where the matrix element is between a state in the fundamental cell and a generic other state. This quantity will characterize the simulation in the off-diagonal sector, and it can be rewritten as

$$G_1 = \frac{1}{L} \int_{-\infty}^{\infty} dx' \int_0^L dx \langle x, 0| e^{-\frac{\beta p^2}{2m}} |x'\rangle \,. \tag{11}$$

The generalizations of Equations (7) and (11) to the D-dimensional N-particle system, in which the paths of the particles in imaginary time are represented on the infinite space, with only one reference coordinate in the fundamental cell, provides an unambiguous parametrization of the PIMC method and will be employed below to construct a worm algorithm that is consistent with periodic boundary conditions.

2.2. Path Integral for N Bosons with Periodic Boundary Conditions

We consider now a system composed of N identical bosons of mass m contained in a D-dimensional hypercube of volume $V = L^D$. The use of periodic boundary conditions makes the space of coordinates the D-dimensional torus $(\mathbb{R}/\mathbb{Z})^D$. Generalizing the notation of the previous subsection, we introduce the coordinates inside the box $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N)$,

with $\mathbf{x}_i \in [0, L)^D$, and a collection of integers $\mathbf{W} = (\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_N)$, with $\mathbf{w}_i \in \mathbb{Z}^D$ labeling the image of the i-th particle. An N-particle state can then be defined as

$$|\mathbf{R} = \mathbf{X} + \mathbf{W}L\rangle \equiv |\mathbf{X}, \mathbf{W}\rangle . \tag{12}$$

The partition function can then be written as

$$Z_N = \frac{1}{N!} \sum_P \sum_{\mathbf{W}} \int_V d\mathbf{X} \, \langle \mathbf{X}, \mathbf{0}| \, e^{-\beta H} \, |P\mathbf{X}, \mathbf{W}\rangle , \tag{13}$$

where $P\mathbf{X} = (\mathbf{x}_{p(1)}, \mathbf{x}_{p(2)}, \ldots, \mathbf{x}_{p(N)})$ indicates the position vector in the box V with permuted indices. By using the convolution rule and introducing $M - 1$ intermediate beads each at the inverse temperature $\delta_\tau = \beta/M$, the partition function can be written as

$$Z_N = \frac{1}{N!} \sum_P \int \prod_{j=0}^{M-1} d\mathbf{R}_j \, \rho(\mathbf{R}_j, \mathbf{R}_{j+1}, \delta_\tau), \qquad \mathbf{R}_M \equiv P\mathbf{X}_0 + \mathbf{W}L . \tag{14}$$

In the above equation, we have combined the integration over $\mathbf{X}_0$ and the sum over the integers $\mathbf{W}$ into an integration over $\mathbf{R}_0$, and implicitly required that the vector $\mathbf{R}_0$ appearing as the argument of the density matrix has to be considered as the image in the fundamental cell, i.e., the vector $\mathbf{X}_0$. Similarly to the one-dimensional case in the previous subsection, the remaining vectors $\mathbf{R}_1, \ldots, \mathbf{R}_M$, span the entire space. As previously stated, in a PIMC calculation one performs a classical-like simulation of polymers formed by M links, connecting $M + 1$ *beads*. Each polymer corresponds to one boson, that can close on itself (if $p(i) = i$) or on another particle specified by the permutation index $p(i)$. Notice that each polymer i starts at the bead 0 inside the fundamental cell, and ends at bead M, closing on an image of the first bead of the polymer $p(i)$, identified up to the periodicity of the D-dimensional torus. Denoting with X a configuration of the N particles (that will be defined more in detail later), one can see that Equation (14) naturally provides a probability density function (PDF), given by

$$\pi(X) = \frac{1}{Z_N} \frac{1}{N!} \prod_{j=0}^{M-1} \rho(\mathbf{R}_j, \mathbf{R}_{j+1}, \delta_\tau), \tag{15}$$

that can be sampled using the Metropolis–Hastings algorithm [25,26]. Diagonal thermodynamic observables can then be directly computed using the above PDF from the configurations visited by the simulation. With obvious notation, the equilibrium statistical average of a given operator O is given by

$$\langle O \rangle = \int \mathcal{D}X \, \pi(X) O(X) . \tag{16}$$

For the systems that are considered here, the Hamiltonian can be expressed as the sum $H = \mathcal{T} + \mathcal{V}$ of a quadratic kinetic part $\mathcal{T} = \sum_i p_i^2/(2m)$, with p_i being the momentum operator associated to particle i, and a potential part $\mathcal{V}$, usually diagonal in the coordinate representation. In such a case, for sufficiently large M an accurate approximation of the high-temperature density matrices is ensured by the Trotter product formula [27]

$$e^{-\beta(\mathcal{T}+\mathcal{V})} = \lim_{M\to\infty} \left[e^{-\delta_\tau \mathcal{T}} e^{-\delta_\tau \mathcal{V}} \right]^M , \tag{17}$$

allowing us to rewrite the density matrices appearing in Equation (15) as

$$\rho(\mathbf{R}_j, \mathbf{R}_{j+1}, \delta_\tau) = \rho_{\text{free}}(\mathbf{R}_j, \mathbf{R}_{j+1}, \delta_\tau) \exp\left[-U(\mathbf{R}_j, \mathbf{R}_{j+1}) \right], \tag{18}$$

where U is the potential energy term, while ρ_{free} is the free particle density matrix obtained from the kinetic operator

$$\rho_{\text{free}}(\mathbf{R}_j, \mathbf{R}_{j+1}, \delta_\tau) \equiv \prod_{i=1}^{N} \rho_{\text{free}}^{\text{sp}}(\mathbf{r}_{i,j}, \mathbf{r}_{i,j+1}, \delta_\tau) = \prod_{i=1}^{N} (4\pi\lambda\delta_\tau)^{-D/2} \exp\left[-\frac{(\mathbf{r}_{i,j} - \mathbf{r}_{i,j+1})^2}{4\lambda\delta_\tau}\right], \tag{19}$$

with $\lambda = \hbar^2/(2m)$. In the so-called *symmetrized primitive approximation*, the potential energy term is given by $U(\mathbf{R}_j, \mathbf{R}_{j+1}) = \delta_\tau(\mathcal{V}(\mathbf{R}_j) + \mathcal{V}(\mathbf{R}_{j+1}))/2$. Depending on the problem at hand, one might significantly reduce the number M of slices needed for convergence using different approximation schemes. An effective scheme in the case of dilute systems with hard-sphere interaction is the pair-product ansatz [3]. This is discussed in more detail in Section 4. Let us anticipate that the factorization of Equation (18) highlights the possibility of using efficient strategies to exactly sample the free Gaussian part ρ_{free}, leaving the acceptance/rejection stage of the Metropolis–Hastings algorithm to be determined only by the potential interaction term. Among the possible schemes, we adopt the staging algorithm [28,29], a smart collective displacement of an arbitrary number of beads.

2.3. Worm Algorithm

The configuration space detailed above, which will be called Z-sector in what follows, consists of polymers organized by the permutation vector P into a number of cycles, i.e., subsets with cyclic permutations, meaning that each polymer has links connecting it to a preceding and a following polymer. Instead of directly summing the $N!$ integrals corresponding to just as many permutations, we use a Monte Carlo integration strategy based on the worm algorithm [6–9,30], which is based on an extended configuration space where the Z-sector is augmented by a G-sector, obtained by cutting one of the cycles and leaving one sequence open-ended. This open sequence of polymers constitutes the worm, with the first polymer called the *tail* and the last one called the *head*. During the simulation, the system randomly fluctuates between the two sectors by opening or closing the worm and, when in the G-sector, the head of the worm might be swapped with another polymer, thus sampling the permutations and allowing the creation of long permutation cycles. Configurations in the G-sector are obtained from a configuration in the Z-sector complemented by a particle index i_H, indicating the head of the worm, and an extra position vector, $\mathbf{r}_{i_H,M} = \mathbf{x}_H + \mathbf{w}_{i_H} L$, representing an additional bead at the head of the worm dangling at the time slice $j = M$. We need this additional position vector because we want the PDF in the G-sector to be the analogue of the one in the Z-sector, thus requiring the same total number of links. We can now define the analogue of the partition function in the G-sector as

$$G_N = \frac{1}{VN!} \sum_P \sum_{i_H=1}^{N} \int_V d\mathbf{x}_H \int \prod_{j=0}^{M-1} d\mathbf{R}_j \, \rho(\mathbf{R}_j, \mathbf{R}_{j+1}, \delta_\tau),$$

$$\text{with } \mathbf{r}_{i,M} = \begin{cases} \mathbf{x}_{p(i),0} + \mathbf{w}_i L & \text{for } i \neq i_H, \\ \mathbf{x}_H + \mathbf{w}_{i_H} L & \text{for } i = i_H, \end{cases} \tag{20}$$

where the $1/V$ factor has been introduced for dimensional reasons and, as before, we have hidden the summation over the integers $\mathbf{W}$ inside the integration over $\mathbf{R}_0$. Notice that, with respect to the partition function in Equation (14), the above equation contains the additional integration over the vector $\mathbf{x}_H$ and the sum over the head index i_H, representing the N possible choices for cutting the permutation cycles. We can then combine Z_N and G_N into a generalized partition function

$$Z_N^{\text{worm}} = Z_N + CG_N, \tag{21}$$

where C is an arbitrary constant that controls the relative simulation time spent in the two sectors. Denoting with N_G and N_Z the number of times the simulation is found in the G and Z-sector, respectively, the proportionality

$$\frac{N_G}{N_Z} = C \frac{G_N}{Z_N},$$ (22)

must be satisfied, implying that the parameter C can be tuned to optimize the simulation. Indeed, the statistical autocorrelation of observables is minimized if $N_G \sim N_Z$. Using Equations (14) and (20) we can rewrite Equation (21) as

$$Z_N^{\text{worm}} = \sum_{S=Z,G} \sum_P \sum_{i_H=1}^N \int d\mathbf{x}_H \prod_{j=0}^{M-1} d\mathbf{R}_j \times$$
$$\times \frac{1}{VN!} \left(\prod_{j=0}^{M-2} \rho(\mathbf{R}_j, \mathbf{R}_{j+1}, \delta_\tau) \right) \left\{ \frac{\delta_{S,Z}}{N} \rho(\mathbf{R}_{M-1}, \mathbf{R}_M, \delta_\tau) + C\delta_{S,G} \rho(\mathbf{R}_{M-1}, \mathbf{R}_M^H, \delta_\tau) \right\}, \quad (23)$$

where we have introduced an explicit summation over the two sectors as well as the Kronecker delta functions to select the proper integrand for each sector. With the notation $\mathbf{R}_M^H$, we represent the vector with components $\mathbf{r}_{i,M} = \mathbf{x}_{p(i),0} + \mathbf{w}_i L$ for $i \neq i_H$ and with component $\mathbf{r}_{i_H,M} = \mathbf{x}_H + \mathbf{w}_{i_H} L$ for $i = i_H$. A generic configuration X for the worm algorithm is specified as $X = (S, P, i_H, \mathbf{x}_H, \{\mathbf{R}_j\})$ with S the sector, P the permutation vector, i_H the head index, $\mathbf{x}_H$ the extra head vector and $\{\mathbf{R}_j\}$ the set of NM position vectors of the polymers. The PDF for the worm algorithm is readily obtained from Equation (23) and reads

$$\pi_W(X) = \frac{1}{Z_N^{\text{worm}}} \frac{1}{VN!} \left(\prod_{j=0}^{M-2} \rho(\mathbf{R}_j, \mathbf{R}_{j+1}, \delta_\tau) \right) \times$$
$$\times \left\{ \frac{\delta_{S,Z}}{N} \rho(\mathbf{R}_{M-1}, \mathbf{R}_M, \delta_\tau) + C\delta_{S,G} \rho(\mathbf{R}_{M-1}, \mathbf{R}_M^H, \delta_\tau) \right\}. \quad (24)$$

For sufficiently large number of beads M, the density matrices are then further factorized into a free particle term and an interaction term as in Equation (18). Before turning to the implementation of the Monte Carlo algorithm, we point out that the worm algorithm offers one additional advantage besides the efficient sampling of permutations: it allows accessing the G-sector configurations (also called off-diagonal configurations) from which one can extract other useful observables such as a properly normalized one-body density matrix [9].

2.4. Monte Carlo Updates

The Monte Carlo procedure is based on a random walk and is obtained by means of semi-local updates $X \to X'$ in the configuration space. The transition matrix $P(X, X')$, i.e., the probability to go from the state X to X', is chosen such that it satisfies the detailed balance condition $\pi_W(X)P(X, X') = \pi_W(X')P(X', X)$. Together with the ergodicity condition, this ensures that, after equilibration, the random walk samples points with the probability $\pi_W(X)$. According to the Metropolis–Hastings criterion, the transition probability for $X \neq X'$ can be factorized into an a priori sampling distribution $T(X, X')$ and an acceptance probability $A(X, X')$ as $P(X, X') = T(X, X')A(X, X')$. The trial moves are then accepted (or rejected) according to the probability:

$$A(X, X') = \min\left[1, \frac{T(X', X)\pi_W(X')}{T(X, X')\pi_W(X)}\right]. \quad (25)$$

We provide below the details for an efficient set of moves (translate, redraw, open/close, move head, move tail) that allow the sampling of the probability distribution π_W. Before

we start, let us stress once again that in order to develop a code compatible with the periodic boundary conditions, one must consistently use the coordinates as defined on $\mathbb{R}^D$ throughout the simulation. The only instance in which one needs to use the representation of the periodic images is to "recenter" a polymer when the initial bead is moved outside of the fundamental cell: in that case, one rigidly translates the whole polymer by the factor $\Delta \mathbf{w} L$, with $\Delta \mathbf{w}$ chosen to make the polymer start in the fundamental cell. We note that in the simulation, one can either choose to use the integers $\mathbf{W}$ or to use the additional vector $\mathbf{R}_M$, making sure that $\mathbf{R}_M = P\mathbf{X}_0 + \mathbf{W}L$ in the Z-sector and $\mathbf{R}_M = \mathbf{R}_M^H$ in the G-sector. We find the second choice more convenient because it provides a uniform representation of the polymers, which better suits a computer program. A separate discussion is needed in the case of the computation for the interaction term. For systems with pairwise interactions, one needs to compute the distance, in the compactified space, between beads belonging to different polymers. Different approaches might be used in this case, but for interactions that decay sufficiently fast with the distance, one can effectively use the *nearest image* convention, in which the distance along each direction is computed modulo L. In the computation of the energy terms for a given link, say, from time slice j to $j + 1$, we identify the closest periodic image of a bead at j, and then find the corresponding image of the subsequent bead at $j + 1$.

Before describing the various Monte Carlo updates, we should emphasize that standard implementations of the worm algorithm had to put constraints on the length of the path segments, namely the number of beads, involved in the open/close and swap moves, which had to be much smaller than the size L of the box to avoid biased sampling. Apart from adding an inconvenient issue related to the fine tuning of the parameters of the simulation, such constraints reduce in general the efficiency of the Monte Carlo sampling.

2.4.1. Translate

This update translates all polymers belonging to a permutation cycle as a rigid body. We select a particle index from a uniform random distribution and construct a list of all the particles in the same permutation cycle. We then select a displacement vector $\Delta \mathbf{r}$ by sampling D random variables uniformly between 0 and a maximum displacement $r_{\max}$ and we perform the shift $\mathbf{r}'_{i,j} = \mathbf{r}_{i,j} + \Delta \mathbf{r}$ for all the beads $j = 0, \ldots, M$ and all the particles i belonging to the permutation cycle. This shift keeps the internal links of all polymers fixed, hence the probability to accept this update is given by

$$A_T = \min\left\{1, \exp\left[\sum_{j=0}^{M-1}\left(U(\mathbf{R}_j, \mathbf{R}_{j+1}) - U(\mathbf{R}'_j, \mathbf{R}'_{j+1})\right)\right]\right\}, \quad (26)$$

where $\mathbf{R}'_j$ collectively denotes the new coordinates at the time slice j. We note that the newly proposed coordinates have the same permutation vector P of the old ones. The parameter $r_{\max} \leq L/2$ can be tuned to optimize the sampling efficiency. If the initial bead of one of the polymers is translated outside of the fundamental cell, we recenter it as discussed above.

2.4.2. Redraw

With this update, we redraw the part of a polymer between two fixed beads. As we have already mentioned, various algorithms with similar efficiencies can be used for this task, but we find the staging algorithm to be convenient because it allows us to redraw a segment with an arbitrary number of beads, as opposed, for example, to the bisection method [1,3] that fixes this number to powers of 2. We select a particle index i_0, an initial bead j_0, and the number of beads $\Delta j \in [2, j_{\max}]$ involved in the update, sampling them from uniform random distributions. The final bead is determined by the bead index $j_1 = j_0 + \Delta j$. To avoid clutter in the presentation we restrict ourselves to present the details for the case $j_1 \leq M$, noting that the generalization to $j_1 > M$ is quite straightforward: one needs to follow the path to the next particle index $p(i_0)$ and use $(j_1 \mod M)$ as the final bead. Moreover, in this case one should follow the path to the next polymer preserving the length

of the links, i.e., momentarily translating the next polymer to the same cell of the bead $\mathbf{r}_{i_0,M}$, and translating it back at the end of the staging procedure. In what follows, we also omit the particle index in the position vectors. The next step is to propose new coordinates for the $\Delta j - 1$ beads between j_0 and j_1 using the so-called Lévy construction [31]. This allows us to directly sample the product of the free particle propagators by rewriting them as

$$
\prod_{j=j_0}^{j_1-1} \rho_{\text{free}}^{\text{sp}}(\mathbf{r}_j, \mathbf{r}_{j+1}, \delta_\tau) = \frac{1}{(4\pi\lambda\Delta j\,\delta_\tau)^{D/2}} \exp\left[-\frac{(\mathbf{r}_{j_0} - \mathbf{r}_{j_1})^2}{4\lambda\Delta j\,\delta_\tau}\right] \times
$$
$$
\times \prod_{j=j_0+1}^{j_1-1} \frac{1}{(4\pi\lambda a_j\delta_\tau)^{D/2}} \exp\left[-\frac{(\mathbf{r}_j - \mathbf{r}_j^*)^2}{4\pi\lambda a_j\delta_\tau}\right], \quad (27)
$$

where we have defined

$$
\mathbf{r}_j^* = \frac{\mathbf{r}_{j_1} + (j_1 - j)\mathbf{r}_{j-1}}{j_1 - j + 1}, \quad a_j = \frac{j_1 - j}{j_1 - j + 1}. \quad (28)
$$

The above equation shows that we can use the product of gaussians in Equation (27) as the a priori sampling distribution for the redraw move and sequentially sample the beads from $j_0 + 1$ to $j_1 - 1$ according to the conditional (free) probability for picking the point j based on the previous bead $j - 1$ and the final bead j_1. The acceptance probability is computed as

$$
A_R = \min\left\{1, \exp\left[\sum_{j=j_0}^{j_1-1}\left(U(\mathbf{R}_j, \mathbf{R}_{j+1}) - U(\mathbf{R}_j', \mathbf{R}_{j+1}')\right)\right]\right\}, \quad (29)
$$

The permutation vector P is left unchanged by this update, and the simulation parameter $j_{\max}$ can be tuned to maximize the sampling efficiency. It is worth pointing out that, if $j_1 > M$, the first bead is involved in the displacement and, if moved outside of the fundamental cell, we shall recenter it, as discussed above.

2.4.3. Open/Close

The open and close moves are sector-changing updates and are particularly delicate. With the open move, we cut a link, creating two loose extremities and taking the system from the Z-sector of closed paths to the G-sector with one worm. With the close move, we bind together the two extremities of the worm, returning to the Z-sector. Since one move is the opposite of the other, we need to carefully weight the transition probability in order to achieve the detailed balance. In our implementation, we propose to open or close the worm at the same rate, independently of the sector the simulation is in, and, only afterwards, we abort the move if either the open move is called within the G-sector or the close move is called within the Z-sector. The updates that we present below differ from the ones originally introduced in Refs. [6–9,30] and are constructed to be compatible with the periodic boundary conditions, yielding exact results even for systems where the thermal wavelength is of the order of the system size, regardless of how many beads are involved in the updates. In the rest of this subsection, we consistently use the notation $X = (Z, P, i_H, \mathbf{x}_H, \{\mathbf{R}_j\})$ to refer to the configuration in the Z-sector and the notation $\widehat{X} = (G, P, \widehat{i}_H, \widehat{\mathbf{x}}_H, \{\widehat{\mathbf{R}}_j\})$ to refer to the configuration in the G-sector. Notice that X and $\widehat{X}$ share the same permutation vector P and that, while the probability distribution in the Z-sector doesn't depend on the head index i_H and the extra bead $\mathbf{x}_H$, they differ from the ones of the state $\widehat{X}$. In the updates presented below, we embed the coordinate of the extra bead $\widehat{\mathbf{x}}_H$ into its representation on $\mathbb{R}^D$ as $\widehat{\mathbf{r}}_{\widehat{i}_H,M} = \widehat{\mathbf{x}}_H + \widehat{\mathbf{w}}_{\widehat{i}_H} L$ and sample the new configuration based on $\widehat{\mathbf{r}}_{\widehat{i}_H,M}$.

The open move consists of the following steps:

- Select the particle index $\widehat{i}_H$ from a uniform random distribution.

- Select a time slice $j_0 \leq j_{\max}^{\text{open}}$ from a uniform random distribution, with the positive integer $j_{\max}^{\text{open}} < M$ a tunable parameter.
- Propose a new value for the position of the new head $\widehat{\mathbf{r}}_{i_H,M}$ by displacing the point $\mathbf{r}_{\widehat{i}_H,M} \equiv \mathbf{x}_{p(\widehat{i}_H),0} + \mathbf{w}_{\widehat{i}_H}L$ by a quantity $\Delta \mathbf{r}$ uniformly sampled in the space $[-\Delta, \Delta]^D$, with $\Delta < L/2$ an adjustable parameter.
- Redraw the portion of the polymer $\widehat{i}_H$ going from the bead $j_0 + 1$ to the bead $M - 1$ by constructing a free particle path starting at $\mathbf{r}_{\widehat{i}_H,j_0}$ and ending after $\Delta j = M - j_0$ steps at $\widehat{\mathbf{x}}_H$ with the staging algorithm described above.
- Accept the update with probability

$$A_O = \min\left\{1, \frac{CN(2\Delta)^D}{V}\exp\left[\sum_{j=j_0}^{M-1}\left(U(\mathbf{R}_j, \mathbf{R}_{j+1}) - U(\widehat{\mathbf{R}}_j, \widehat{\mathbf{R}}_{j+1})\right)\right] \times \right.$$
$$\left. \times \frac{\rho_{\text{free}}^{\text{sp}}(\widehat{\mathbf{r}}_{\widehat{i}_H,j_0}, \widehat{\mathbf{r}}_{\widehat{i}_H,M}, \Delta j\, \delta_\tau)}{\rho_{\text{free}}^{\text{sp}}(\mathbf{r}_{\widehat{i}_H,j_0}, \mathbf{r}_{\widehat{i}_H,M}, \Delta j\, \delta_\tau)}\right\}. \quad (30)$$

The close move provides the detailed balanced complement to the open move. In explaining the steps for the close move, we remind the reader that as per the open move detailed above, the position vectors $\{\mathbf{R}_j\}$ and $\{\widehat{\mathbf{R}}_j\}$ only differ at particle $\widehat{i}_H$ for the bead indices going from $j_0 + 1$ to M. The close move consists of the following steps:

- Identify the particle indices $\widehat{i}_H$ and $\widehat{i}_T = p(\widehat{i}_H)$ corresponding to the head and the tail of the worm, respectively.
- Select a time slice $j_0 \leq j_{\max}^{\text{open}}$ from a uniform random distribution.
- Find the periodic image $\widehat{\mathbf{r}}_T = \widehat{\mathbf{x}}_{\widehat{i}_T,0} + \mathbf{w}_{\widehat{i}_H}L$ of the first bead of the tail $\widehat{\mathbf{x}}_{\widehat{i}_T,0}$ that is the nearest to the head bead $\widehat{\mathbf{r}}_{\widehat{i}_H,M}$ and check whether their difference $\widehat{\mathbf{r}}_{\widehat{i}_H,M} - \widehat{\mathbf{r}}_T$ is within $[-\Delta, \Delta]$ in every direction. If that is the case set $\mathbf{r}_{\widehat{i}_H,M} = \widehat{\mathbf{r}}_T$, otherwise abort the update.
- Redraw the portion of the polymer $\widehat{i}_H$ going from the bead $j_0 + 1$ to the bead $M - 1$ by constructing a free particle path starting at $\mathbf{r}_{\widehat{i}_H,j_0}$ and ending after $\Delta j = M - j_0$ steps at $\mathbf{r}_{\widehat{i}_H,M}$ with the staging algorithm.
- Accept the update with probability

$$A_C = \min\left\{1, \frac{V}{CN(2\Delta)^D}\exp\left[\sum_{j=j_0}^{M-1}\left(U(\widehat{\mathbf{R}}_j, \widehat{\mathbf{R}}_{j+1}) - U(\mathbf{R}_j, \mathbf{R}_{j+1})\right)\right] \times \right.$$
$$\left. \times \frac{\rho_{\text{free}}^{\text{sp}}(\mathbf{r}_{\widehat{i}_H,j_0}, \mathbf{r}_{\widehat{i}_H,M}, \Delta j\, \delta_\tau)}{\rho_{\text{free}}^{\text{sp}}(\widehat{\mathbf{r}}_{\widehat{i}_H,j_0}, \widehat{\mathbf{r}}_{\widehat{i}_H,M}, \Delta j\, \delta_\tau)}\right\}. \quad (31)$$

As a last step, one should randomly select the particle index i_H and the value of $\mathbf{x}_H$ inside the fundamental cell, but, since the PDF in the Z-sector does not depend on them, one can actually forget to update their value in the simulation.

Given the delicate nature of the present moves it is instructive to explicitly derive the acceptance probability in Equations (30) and (31) to highlight the different elements that make the sector-changing moves possible. We first write the ratio of the PDFs,

$$\frac{\pi_W(\widehat{X})}{\pi_W(X)} = CN\frac{\prod_{j=j_0}^{M-1}\rho_{\text{free}}^{\text{sp}}(\widehat{\mathbf{r}}_{\widehat{i}_H,j}, \widehat{\mathbf{r}}_{\widehat{i}_H,j+1}, \delta_\tau)}{\prod_{j=j_0}^{M-1}\rho_{\text{free}}^{\text{sp}}(\mathbf{r}_{\widehat{i}_H,j}, \mathbf{r}_{\widehat{i}_H,j+1}, \delta_\tau)}\exp\left[\sum_{j=j_0}^{M-1}\left(U(\mathbf{R}_j, \mathbf{R}_{j+1}) - U(\widehat{\mathbf{R}}_j, \widehat{\mathbf{R}}_{j+1})\right)\right], \quad (32)$$

and then we evaluate the transition probabilities. Since the choice of the particle indices i_H and $\widehat{i}_H$ and the choice of the time slice j_0 are based on uniform distributions and proceed identically for the open and close moves, we don't report the corresponding factors below

since they simplify in the ratio and are not relevant for the discussion. For the open move, the procedure explained above gives

$$T(X, \widehat{X}) = \frac{1}{(2\Delta)^D} \prod_{j=j_0+1}^{M-1} \frac{1}{(4\pi\lambda a_j \delta_\tau)^{D/2}} \exp\left[-\frac{(\widehat{\mathbf{r}}_{\widehat{i}_H,j} - \widehat{\mathbf{r}}^*_{\widehat{i}_H,j})^2}{4\pi\lambda a_j \delta_\tau} \right], \tag{33}$$

with $\widehat{\mathbf{r}}^*_{\widehat{i}_H,j}$ and a_j obtained with the Levy construction as in Equation (27). For the close move, we have instead

$$T(\widehat{X}, X) = \frac{1}{V} \prod_{j=j_0+1}^{M-1} \frac{1}{(4\pi\lambda a_j \delta_\tau)^{D/2}} \exp\left[-\frac{(\mathbf{r}_{\widehat{i}_H,j} - \mathbf{r}^*_{\widehat{i}_H,j})^2}{4\pi\lambda a_j \delta_\tau} \right], \tag{34}$$

with $\mathbf{r}^*_{\widehat{i}_H,j}$ obtained with the Levy construction and with the factor $1/V$ coming from uniformly sampling the vector $\mathbf{x}_H$. Putting these equations together and using the identity (27), one gets the acceptance probabilities in Equations (30) and (31). As stated at the beginning of the subsection, the open/close updates introduced here are compatible with the periodic boundary conditions and the parameter $j_{\max}^{\mathrm{open}}$ can be freely chosen to maximize the sampling efficiency. In contrast to the original algorithm, where one faces the ambiguity in the choice of the image of the tail $\mathbf{r}_T$ when closing the polymer—resulting in a violation of the detailed balance condition when the thermal wavelength is of the order of the size of the system—the updates presented here are always perfectly balanced. This is achieved by a different sampling choice for $\widehat{\mathbf{r}}_{\widehat{i}_H,M}$, with the cutoff $\Delta < L/2$ removing said ambiguity. A convenient choice for the parameter Δ is to make it dependent on the choice of j_0 as

$$\Delta = \min\left(\sqrt{2\lambda(M-j_0)\,\delta_\tau}, L/2 \right). \tag{35}$$

2.4.4. Swap

The swap update allows one to sample the permutations in a very efficient way by connecting the worm head to a near polymer. It requires the worm to be present, hence it is performed only in the *G*-sector, meaning that it must be aborted if proposed in the *Z*-sector. We denote with i_H the particle index associated with the worm's head. We first select from a uniform random distribution a time slice $j_P \leq j_{\max}^{\mathrm{swap}}$ with $j_{\max}^{\mathrm{swap}} < M$, which will act as pivot. We first compute the free propagators

$$\Pi_P(i) = \rho_{\mathrm{free}}^{\mathrm{sp}}(\mathbf{x}_H, \mathbf{r}_{i,j_P}, j_P\delta_\tau), \tag{36}$$

for every particle i, and we then use tower sampling to select a particle index i_0 with probability $\Pi_P(i)/\Sigma_P$, where the normalization factor is given by

$$\Sigma_P = \sum_{i=1}^N \Pi_P(i). \tag{37}$$

Before proceeding further, we need to test the particle i_0. If it corresponds to the tail of the worm i_T, we abort the move and reject the update. This is necessary to prevent the worm from closing on itself, and hence disappearing. If that is not the case, we proceed by computing the normalization factor for the inverse process

$$\Sigma_0 = \sum_{i=1}^N \rho_{\mathrm{free}}^{\mathrm{sp}}(\mathbf{x}_{i_0,0}, \mathbf{r}_{i,j_P}, j_P\delta_\tau), \tag{38}$$

which is necessary to ensure the detailed balance. Notice that the free propagators in Equations (36) and (38) are computed using the simulation coordinates, without invoking periodic boundary conditions. We then cut the polymer i_0 and we bind its first bead

to the head of the worm by setting $\mathbf{x}'_{i_0,0} = \mathbf{x}_H$. We finally construct a free particle path $\mathbf{r}'_{i_0,1}, \dots, \mathbf{r}'_{i_0,j_P-1}$ with the staging algorithm. The acceptance probability is computed as

$$A_{SW} = \min\left\{1, \frac{\Sigma_P}{\Sigma_0} \exp\left[\sum_{j=0}^{j_P-1}\left(U(\mathbf{R}_j, \mathbf{R}_{j+1}) - U(\mathbf{R}'_j, \mathbf{R}'_{j+1})\right)\right]\right\}, \qquad (39)$$

Notice that this update changes the permutation P to P' such that $p'(i_H) = i_0$, and the new worm's head i'_H is instead identified as the particle that was in permutation with i_0 before the update, i.e., we set $i'_H = i^*$ where i^* is such that $P(i^*) = i_0$. The tail index remains unchanged.

2.4.5. Move Head

This and the next move are performed only in the *G*-sector, when a worm is present. They must be aborted if proposed in the *Z*-sector. With *move head* we redraw the last beads of the worm. We first select a starting time slice j_0 and then sample the new worm's head bead $\mathbf{r}'_{i_H,M}$, from the free particle propagator

$$\rho_{\text{free}}^{\text{sp}}(\mathbf{r}_{i_H,j_0}, \mathbf{r}'_{i_H,M}, \Delta j\,\delta_\tau) = (4\pi\lambda\Delta j\,\delta_\tau)^{-D/2} \exp\left[-\frac{(\mathbf{r}_{i_H,j_0} - \mathbf{r}'_{i_H,M})^2}{4\lambda\Delta j\,\delta_\tau}\right], \qquad (40)$$

where i_H is the particle index of the worm's head and $\Delta j = M - j_0$. We then construct a free path $\mathbf{r}'_{i_H,j_0+1}, \dots, \mathbf{r}'_{i_H,M-1}$ with the staging algorithm. We accept the update with probability

$$A_H = \min\left\{1, \exp\left[\sum_{j=j_0}^{M-1}\left(U(\mathbf{R}_j, \mathbf{R}_{j+1}) - U(\mathbf{R}'_j, \mathbf{R}'_{j+1})\right)\right]\right\}. \qquad (41)$$

2.4.6. Move Tail

Similarly, with *move tail* we redraw the first beads of the worm by selecting a time slice j_0 and generating a proposal $\mathbf{r}'_T$ for the new tail bead of the worm $\mathbf{x}_{i_T,0}$ according to the distribution

$$\rho_{\text{free}}^{\text{sp}}(\mathbf{r}'_T, \mathbf{r}_{i_T,j_0}, \Delta j\,\delta_\tau) = (4\pi\lambda\Delta j\,\delta_\tau)^{-D/2} \exp\left[-\frac{(\mathbf{r}'_T - \mathbf{r}_{i_T,j_0})^2}{4\lambda\Delta j\,\delta_\tau}\right], \qquad (42)$$

where i_T is the particle index of the worm's tail and $\Delta j = M - j_0$. We set $\mathbf{x}_{i_T,0} = \mathbf{r}'_T$ and then construct a free path $\mathbf{r}'_{i_T,1}, \dots, \mathbf{r}'_{i_T,j_0-1}$ with the staging algorithm. We accept the update with probability

$$A_T = \min\left\{1, \exp\left[\sum_{j=j_0}^{M-1}\left(U(\mathbf{R}_j, \mathbf{R}_{j+1}) - U(\mathbf{R}'_j, \mathbf{R}'_{j+1})\right)\right]\right\}. \qquad (43)$$

If the coordinate $\mathbf{r}'_T$ falls outside of the fundamental cell, we recenter the tail polymer as discussed above.

3. Non-Interacting Bose Gas

When developing a code, one should have precise benchmarks aimed at validating the various aspects of the algorithm. In this section, we start with tests for non-interacting systems, where exact results are known, and we will consider interacting systems in the next section. From now on, we specialize to the $D = 3$ case. Textbook treatments of the non-interacting Bose gas model usually consider the system in the grand-canonical ensemble and in the thermodynamic limit. However, exact solutions are available also for

a fixed number of particles in a box with periodic boundary conditions. In particular, we are interested in the calculation of the internal energy

$$E = -\frac{1}{Z_N}\frac{\partial Z_N}{\partial \beta} \ . \tag{44}$$

The partition function $Z_N(\beta)$ for N particles at inverse temperature β can be calculated using the recursion formula [32,33]

$$Z_N(\beta) = \frac{1}{N}\sum_{k=1}^{N} z(k\beta)Z_{N-k}(\beta) \ , \tag{45}$$

involving the partition functions of $N - k$ particles at the same inverse temperature β with the starting value $Z_0(\beta) = 1$ and the partition function of a single particle $z(k\beta)$ at the multiple inverse temperature $k\beta$. The latter partition function for our system with periodic boundary conditions is defined as

$$z(\beta) = \sum_{n_x,n_y,n_z} e^{-\beta\varepsilon(n_x,n_y,n_z)} \ , \tag{46}$$

where $\varepsilon(n_x, n_y, n_z)$ are the single-particle energies labeled by the integers $n_{x,y,z} = 0, \pm 1, \pm 2, \dots$

$$\varepsilon(n_x, n_y, n_z) = \lambda\left(\frac{2\pi}{L}\right)^2 (n_x^2 + n_y^2 + n_z^2) \ . \tag{47}$$

The calculation of the internal energy E in Equation (44) can be carried out recursively from the derivatives with respect to β using result (45).

In the following we perform PIMC calculations of the internal energy of N non-interacting bosons at different temperatures and we directly compare the results with the exact value obtained from Equation (44). First, we compare for the case $N = 1$, where no swap moves are involved in the PIMC simulation but it provides a stringent benchmark for all the other moves. Then, we move to $N = 2$ in order to test the correct implementation also of the swap moves. Finally we consider the case $N \gg 1$ and the approach to the thermodynamic limit.

Benchmarks

We carry out a first non-trivial check on the worm algorithm by considering a single particle in a regime where the thermal wavelength λ_T is of the order of the side of the box L, thus testing the compatibility of the updates (except swap) with the periodic boundary conditions. In Figure 1, we present the results for the internal energy E for different values of the ratio λ_T/L and compare them with the exact results. We performed the test varying the total number of beads M while keeping unconstrained the simulation parameters controlling the portion of the polymers involved in the moves, namely setting j_{max} and j_{max}^{open} to the maximum value M. Since, in the absence of the interaction, the algorithm is exact for any number of beads, we recover the exact result even for $M = 1$.

We then turn a closer look to the open/close update, verifying the proportionality between the ratio N_G/N_Z and the parameter C, as expressed in Equation (22). In the left panel of Figure 2, we report the ratios obtained at different values of C together with one-parameter fits. In the right panel, the fitted coefficient is in turn compared with the exact value of G_1/Z_1 given by

$$\frac{G_1}{Z_1} = \left[\vartheta_3\left(0, e^{-\pi\frac{\lambda_T^2}{L^2}}\right)\right]^{-3} , \tag{48}$$

where $\vartheta_3(z, q)$ is the theta function with rational characteristic 3 (see, e.g., chapter 16 of ref. [34]). As one can see from the figure, there is perfect agreement between the PIMC points and the expected results, with the inset showing the difference between them.
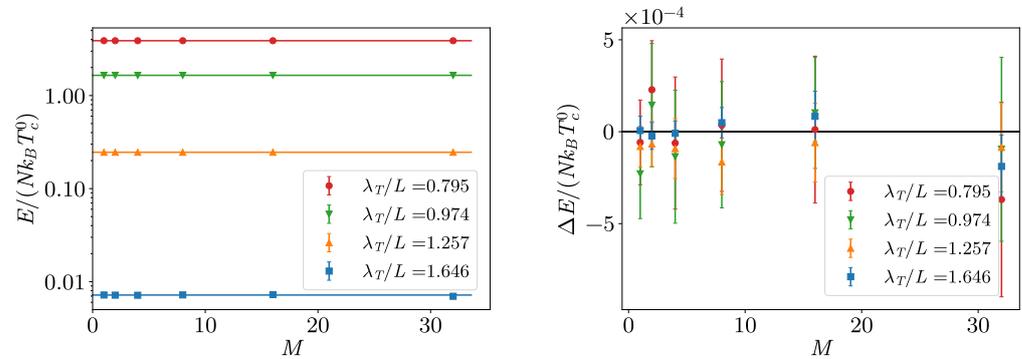
Figure 1. Results for the $N = 1$ system at different values of λ_T/L. (**Left**) Internal energy E, in units of $Nk_BT_c^0$, with T_c^0 being the critical temperature defined in Equation (49). The values obtained at different number of beads M are compared with the exact results (horizontal lines). (**Right**) Differences with the exact values.
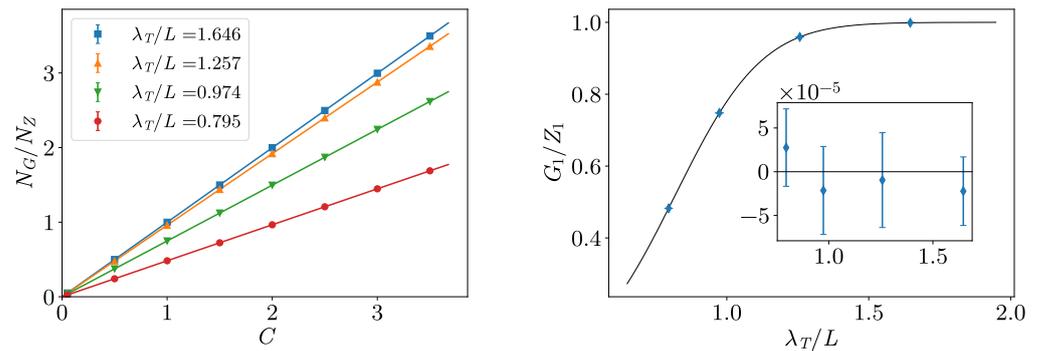
Figure 2. Results for the $N = 1$ system at different values of λ_T/L. (**Left**) check of the proportionality in Equation (22), the lines are one-parameter fits. (**Right**) the proportionality coefficients extracted from the fits (diamonds) are compared with the exact value (solid line). Inset: difference between the coefficients and the exact value.

We now move to the $N = 2$ system where the swap update makes its first appearance and we check the results for different values of λ_T/L and various values of M. As before, we used unconstrained parameters $j_{\max} = j_{\max}^{\text{open}} = j_{\max}^{\text{swap}} = M$, allowing updates involving a whole polymer. We report the results in Figure 3, where we show that we recover the exact result for any number of beads used, meaning that the implementation of the swap is fully compatible with periodic boundary conditions.
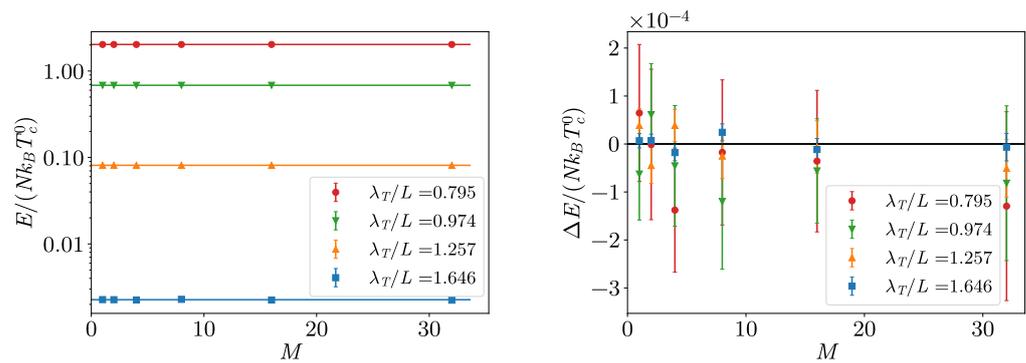
Figure 3. Results for the $N = 2$ system at different values of λ_T/L. (**Left**) Internal energy E computed with different number of beads M and compared with the exact results (horizontal lines). As in Figure 1 the energy scale is defined in Equation (49). (**Right**) Differences with the exact values.

The one and two-particle systems provide clean—and to some extent independent—tests for the whole set of updates in the regime where the effect of periodic boundary conditions is the largest. We now show that the exactness of the algorithm is preserved when taking larger and larger numbers of particles. For this purpose, we vary N while keeping the temperature T fixed. In Figure 4, we show the results for the internal energy for three values of the temperature, expressed in units of the critical temperature T_c^0 defined as

$$k_B T_c^0 = 4\pi\lambda \left(\frac{n}{\zeta(3/2)} \right)^{2/3}, \tag{49}$$

where $n = N/V$ is the number density. For every value of N, the worm algorithm gives results that are in agreement with the exact values computed from Equation (45) and shown in the left panel of Figure 4 with the dot-dashed lines. Notice that the error bars are of the order of 10^{-4} and are completely hidden by the symbols. The horizontal dotted lines represent the value of the internal energy in the thermodynamic limit. In the right panel of Figure 4, we show a linear fit in $1/N$ that extrapolates to the thermodynamic limit the three largest system sizes at $T = 0.5 T_c^0$. The gray band covers the 1-sigma region of the linear fit and perfectly recovers the $N \to \infty$ result shown by the horizontal dotted line.
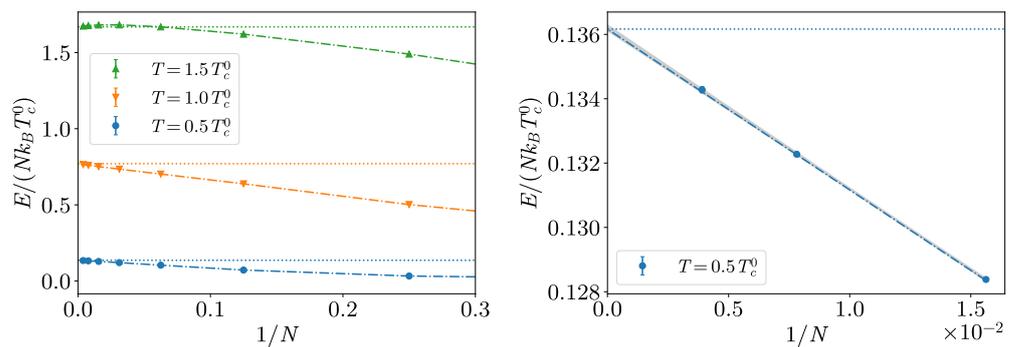
Figure 4. Internal energy E for N-particle systems at a fixed temperature. (**Left**) Worm algorithm results compared with the exact values at fixed number of particles (dot-dashed lines) and with the exact thermodynamic limit (horizontal dotted lines). Error bars, being of the order of 10^{-4}, are not visible on this scale and are perfectly compatible with the exact results. (**Right**) Zoom on the $T = 0.5 T_c^0$ data, with a linear fit extrapolation to the thermodynamic limit. The gray band covers the one-sigma region of the fit.

4. Hard-Spheres Bose Gas

In this section, we consider interacting systems described by the following microscopic Hamiltonian with two-body interactions:

$$H = -\lambda \sum_{i=1}^{N} \nabla_i^2 + \sum_{i<k} v(|\mathbf{r}_i - \mathbf{r}_k|), \tag{50}$$

where m is the mass of the N identical bosons and $\mathbf{r}_i$ indicates the particle position vector. For pedagogical reasons, we use a simple interatomic potential corresponding to a purely repulsive hard core, without any attractive tail to avoid the occurrence of possible cluster bound states. More precisely, we use the hard-sphere (HS) model defined as

$$v(r) = \begin{cases} +\infty & (r < a), \\ 0 & (r > a), \end{cases} \tag{51}$$

in terms of the HS diameter a. Besides the degeneracy parameter $n\lambda_T^3$, only one extra parameter, the gas parameter na^3, is needed to fully characterize the many-body physics of the model. Equilibrium states of the system correspond either to a gas or to a solid, the latter requiring that the gas parameter be large enough and the temperature small enough [35,36]. Furthermore, in the limit of a small gas parameter, the HS model fully captures the universal behavior of dilute gases in terms of the s-wave scattering length, which coincides with the HS diameter a. A careful study of the thermodynamics of the HS model in the dilute regime has been carried out in Ref. [37]. Here, for illustrative reasons, we consider the HS gas at a much higher density $na^3 = 0.1$, which is not so far from the corresponding density of liquid ^{4}He and where the issues of convergence with the number of beads are more relevant.

A convenient approximation scheme for the high temperature density matrix entering the PIMC algorithm is the pair-product ansatz [3]

$$\rho(\mathbf{R}, \mathbf{R}', \delta_\tau) = \prod_{i=1}^{N} \rho_{\text{free}}^{\text{sp}}(\mathbf{r}_i, \mathbf{r}_i', \delta_\tau) \prod_{i<k} \frac{\rho_{rel}(\mathbf{r}_{ik}, \mathbf{r}_{ik}', \delta_\tau)}{\rho_{rel}^0(\mathbf{r}_{ik}, \mathbf{r}_{ik}', \delta_\tau)}. \tag{52}$$

In the above equation $\rho_{\text{free}}^{\text{sp}}$ is the single-particle ideal-gas density matrix defined in Equation (19) and ρ_{rel} is the two-body density matrix of the interacting system, which depends on the relative coordinates $\mathbf{r}_{ik} = \mathbf{r}_i - \mathbf{r}_k$ and $\mathbf{r}_{ik}' = \mathbf{r}_i' - \mathbf{r}_k'$, divided by the corresponding ideal-gas term

$$\rho_{rel}^0(\mathbf{r}_{ik}, \mathbf{r}_{ik}', \delta_\tau) = (8\pi\lambda\delta_\tau)^{-3/2} \exp\left[-\frac{(\mathbf{r}_{ik} - \mathbf{r}_{ik}')^2}{8\lambda\delta_\tau}\right]. \tag{53}$$

The advantage of the decomposition in Equation (52) is that the two-body density matrix at the inverse temperature δ_τ, $\rho_{rel}(\mathbf{r}, \mathbf{r}', \delta_\tau)$, can be calculated exactly for a given potential $V(r)$, thereby solving by construction the two-body problem. This is the most effective strategy when the system is diluted, but can also be pursued at higher density. For the HS potential, a simple and remarkably accurate analytical approximation of the high-energy two-body density matrix is due to Cao and Berne [38]. For $r > a$ and $r' > a$, the result is given by

$$\frac{\rho_{rel}(\mathbf{r}, \mathbf{r}', \delta_\tau)}{\rho_{rel}^0(\mathbf{r}, \mathbf{r}', \delta_\tau)} = 1 - \frac{a(r+r') - a^2}{rr'} e^{-[rr'+a^2-a(r+r')](1+\cos\theta)/(4\lambda\delta_\tau)}, \tag{54}$$

where θ is the angle between the directions of $\mathbf{r}$ and $\mathbf{r}'$, while it vanishes when either r or r' are smaller than a. An important remark concerning the Cao–Berne approximation is that it correctly describes the scattering of hard spheres at high energy (small δ_τ), and it

exactly accounts for the *s*-wave term in the partial wave expansion, which is the dominant contribution at low energy. Within the Cao–Berne approximation, the interaction energy of hard spheres evaluated at two subsequent beads j and $j + 1$ can finally be written as (setting $\mathbf{R} = \mathbf{R}_j$ and $\mathbf{R}' = \mathbf{R}_{j+1}$)

$$U(\mathbf{R}, \mathbf{R}') = - \sum_{i<k} \log \left(\frac{\rho_{rel}(\mathbf{r}_{ik}, \mathbf{r}'_{ik}, \delta_\tau)}{\rho^0_{rel}(\mathbf{r}_{ik}, \mathbf{r}'_{ik}, \delta_\tau)} \right). \tag{55}$$

In the following subsections, we investigate the convergence with the number of beads for different number of particles and different temperatures at the density $na^3 = 0.1$. Finally, we investigate the approach to the thermodynamic limit for increasing values of N at a specific temperature.

Benchmarks

The presence of the hard-sphere potential, approximated by the product of Cao–Berne density matrices, brings a dependence on the number of beads that is detectable when the gas parameter is large enough. In Figure 5, we show such dependence in a system at density $na^3 = 0.1$ for two values of the temperature, $T = T_c^0$ and $T = 0.5T_c^0$. As one can see, the value of the energy saturates at large number of beads, regardless of the system size controlled by the total number of particles N. This can be interpreted recalling that the Cao–Berne density matrix computes exactly the scattering at high energy, requiring the thermal wavelength associated with the imaginary time step δ_τ to be small compared to the interaction range a, i.e.,

$$\sqrt{\frac{4\pi\lambda\beta}{M}} \ll a. \tag{56}$$
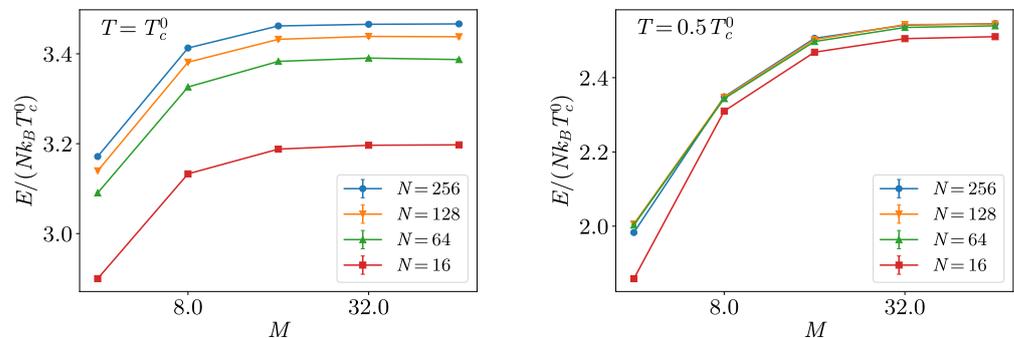
Figure 5. Internal energy E for a system of hard-spheres bosons at the density $na^3 = 0.1$, shown as a function of the number of beads used in the simulation. The energy is computed using the virial estimator reported in Appendix A.

On the contrary, in the limit of dilute systems the Cao–Berne approximation becomes exact and even a small number of beads (even 8 or 16) is sufficient to get precise results [37]. Finally, in Figure 6 we show how the points at $M = 64$ from Figure 5 extrapolate to the thermodynamic limit for the two temperatures. In the left and right panels of Figure 6, the dotted lines represent the linear fit to the data, with the gray bands covering the one-sigma region. As expected, the system at the critical temperature T_c^0 suffers from much larger finite size effects, compared with the system at temperature $T = 0.5T_c^0$.
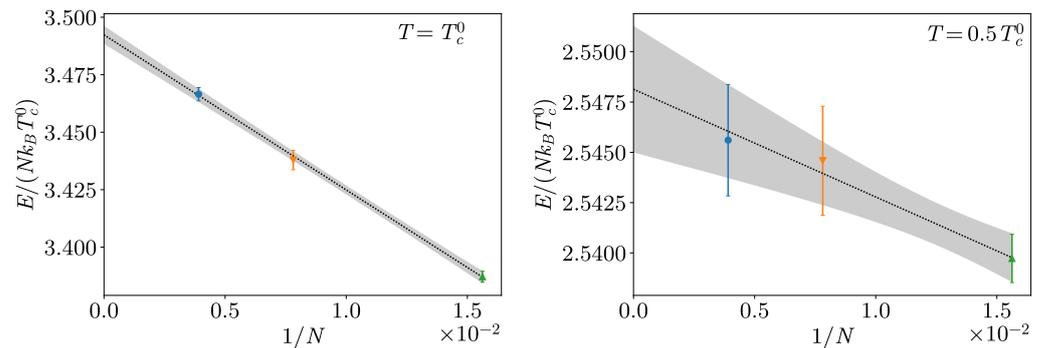
Figure 6. Extrapolation of the internal energy E to the thermodynamic limit using the points at $M = 64$ from Figure 5. We have omitted the system size $N = 16$ because it is too small to be used in a linear extrapolation in $1/N$. Symbols follow the legend in Figure 5.

5. Conclusions

We described an algorithm to perform unbiased PIMC simulations for Bose systems in the canonical ensemble with periodic boundary conditions. The formalism for path-integrals suitable for periodic systems has been developed, and the technical details required for a correct implementation of the PIMC algorithm have been discussed in a pedagogical manner. The formalism can be similarly applied to the simulation of systems in the grand-canonical ensemble. Benchmark results have been presented for non-interacting Bose gases, for which the energy can be exactly computed for any particle number. Many-body systems with hard-sphere interactions have been addressed as well, and the convergence to the continuous imaginary-time limit and to the infinite system-size limit have been analyzed. The PIMC algorithm we presented provides unbiased results for non-interacting Bose gases with periodic boundary conditions for any number of imaginary-time slices, meaning that, e.g., non-interacting particles can be exactly simulated even setting $M = 1$. Furthermore, all Monte Carlo updates can be performed without limitations on the length of the paths that are involved in the update, even in regimes where the thermal wavelength is comparable to the size of the fundamental cell. This is in contrast to previous implementations of the worm algorithm [8], for which, even for non-interacting systems, unbiased results are obtained only for a sufficiently large number of imaginary-time slices. Furthermore, those implementations become ambiguous when the thermal wavelength starts to be comparable to the size of the fundamental cell, possibly leading to biased results unless the length of the paths that are involved in some updates is constrained. For interacting systems, also the PIMC algorithm presented here requires a sufficient number of imaginary-time slices, so that the high-temperature approximation for the density matrix (in our case, the pair product approximation) becomes essentially exact. However, there is no constraint on the length of the paths involved in any updates. It is worth emphasizing that the original implementation of the worm algorithm provides unbiased results, even without constraints in the Monte Carlo updates, if the size of the fundamental periodic cell is much larger than the thermal wavelength. While such a system size is feasible for high and for moderately low temperatures, it becomes impractical close to the zero temperature limit. When the thermal wavelength starts to be comparable to the size of the fundamental periodic cell, stringent constraints in some Monte Carlo updates have to be imposed in order to avoid ambiguities due to the choice of periodic images. These constraints strongly reduce the acceptance rates, leading to excessively long autocorrelation times and, therefore, to inefficient simulations. For these reasons, we expect the algorithm presented here to allow extending the scope of the PIMC simulations, providing a better access to the intriguing quantum phenomena occurring in the low temperature limit. Furthermore, the possibility to perform exact benchmarks for small system sizes will help novel practitioners in correctly implementing unbiased PIMC codes.

Author Contributions: Conceptualization, S.G., S.P. and G.S.; software, G.S. and S.P.; formal analysis, G.S.; writing—original draft preparation, G.S., S.P. and S.G.; funding acquisition, S.G. and S.P. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Italian Ministry of University and Research under the PRIN2017 project CEnTraL 20172H2SC4. S.P. acknowledges PRACE for awarding access to the Fenix Infrastructure resources at Cineca, which are partially funded from the European Union's Horizon 2020 research and innovation programme through the ICEI project under the grant agreement No. 800858. S.P. also acknowledges the CINECA award under the ISCRA initiative, for the availability of high performance computing resources and support.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data is contained within the article.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

In this appendix, we report the expressions needed to compute the internal energy E and pressure P in a PIMC simulation, presenting two Monte Carlo estimators for each of them.

Appendix A.1. Energy

The internal energy E can be computed in the diagonal sector as

$$E = -\frac{1}{Z_N}\frac{dZ_N}{d\beta} \ . \tag{A1}$$

Using the PIMC representation of Z_N in Equation (14) and working out the derivative, we obtain the thermodynamic estimator

$$\frac{E_{\text{th}}}{N} = \left\langle \frac{D}{2\delta_\tau} - \frac{1}{4\lambda\delta_\tau^2 NM}\sum_{j=0}^{M-1}\left(\mathbf{R}_j - \mathbf{R}_{j+1}\right)^2 + \frac{1}{NM}\sum_{j=0}^{M-1}\frac{\partial U\left(\mathbf{R}_j,\mathbf{R}_{j+1}\right)}{\partial\delta_\tau} \right\rangle, \tag{A2}$$

where the average is taken on the configurations sampled in the Z-sectors. In addition, the above quantity can be calculated in PIMC simulations using the so called virial estimator [3], which usually suffers from smaller statistical fluctuations. Different expressions can be derived, and we report here the version we have implemented in our code

$$\frac{E_{\text{vir}}}{N} = \left\langle \frac{D}{2\beta} + \frac{(\mathbf{R}_{M-1} - \mathbf{R}_M)\cdot(\mathbf{R}_M - \mathbf{R}_0)}{4\lambda\delta_\tau^2 NM} + \frac{1}{NM}\sum_{j=0}^{M-1}\frac{\partial U\left(\mathbf{R}_j,\mathbf{R}_{j+1}\right)}{\partial\tau} \right.$$
$$\left. + \frac{1}{2\beta N}\sum_{j=1}^{M-1}\left(\mathbf{R}_j - \mathbf{R}_0\right)\cdot\frac{\partial}{\partial\mathbf{R}_j}\left[U(\mathbf{R}_{j-1},\mathbf{R}_j) + U(\mathbf{R}_j,\mathbf{R}_{j+1})\right] \right\rangle. \tag{A3}$$

Appendix A.2. Pressure

Although we have not presented any results for the pressure, we report for completeness the expression for its two estimators. The pressure is defined as

$$P = \frac{1}{\beta Z_N}\frac{dZ_N}{dV} \ . \tag{A4}$$

To calculate the derivative of the partition function with respect to the volume V, we use the identity $V d\mathbf{R}_j / dV = \mathbf{R}_j / D$ and apply the chain rule for every j. The thermodynamic estimator reads

$$\frac{P_{\text{th}}}{n} = \left\langle \frac{1}{\delta_\tau} - \frac{1}{2\lambda\delta_\tau^2 NMD} \sum_{j=0}^{M-1} \left(\mathbf{R}_j - \mathbf{R}_{j+1}\right)^2 - \frac{V}{\beta N} \sum_{j=0}^{M-1} \frac{\partial U(\mathbf{R}_j, \mathbf{R}_{j+1})}{\partial V} \right\rangle. \quad \text{(A5)}$$

Notice that we have not used the chain rule on the last term because, depending on the interatomic potential and on the adopted approximation for the density matrix, it is sometimes easier to directly compute the derivative of the interaction term (e.g., the expression in Equation (55)) with respect to the volume, obtaining an expression without ambiguities due to periodic boundary conditions. The virial estimator can instead be computed as

$$\frac{P_{\text{vir}}}{n} = \left\langle \frac{1}{\beta} + \frac{(\mathbf{R}_{M-1} - \mathbf{R}_M) \cdot (\mathbf{R}_M - \mathbf{R}_0)}{2\lambda\delta_\tau^2 NMD} - \frac{V}{\beta N} \sum_{j=0}^{M-1} \frac{\partial U(\mathbf{R}_j, \mathbf{R}_{j+1})}{\partial V} \right.$$
$$\left. + \frac{1}{\beta ND} \sum_{j=1}^{M-1} \left(\mathbf{R}_j - \mathbf{R}_0\right) \cdot \frac{\partial}{\partial \mathbf{R}_j} \left[U(\mathbf{R}_{j-1}, \mathbf{R}_j) + U(\mathbf{R}_j, \mathbf{R}_{j+1}) \right] \right\rangle. \quad \text{(A6)}$$

References

1. Pollock, E.L.; Ceperley, D.M. Simulation of quantum many-body systems by path-integral methods. *Phys. Rev. B* **1984**, *30*, 2555. [CrossRef]
2. Ceperley, D.M.; Pollock, E.L. Path-integral computation of the low-temperature properties of liquid ^{4}He. *Phys. Rev. Lett.* **1986**, *56*, 351–354. [CrossRef]
3. Ceperley, D.M. Path integrals in the theory of condensed helium. *Rev. Mod. Phys.* **1995**, *67*, 279–355. [CrossRef]
4. Pollock, E.L.; Ceperley, D.M. Path-integral computation of superfluid densities. *Phys. Rev. B* **1987**, *36*, 8343–8352. [CrossRef] [PubMed]
5. Boninsegni, M. Permutation Sampling in Path Integral Monte Carlo. *J. Low Temp. Phys.* **2005**, *141*, 27–46. [CrossRef]
6. Prokof'ev, N.; Svistunov, B.; Tupitsyn, I.S. "Worm" algorithm in quantum Monte Carlo simulations. *Phys. Lett. A* **1998**, *238*, 253–257. [CrossRef]
7. Prokof'ev, N.; Svistunov, B. Worm algorithms for classical statistical models. *Phys. Rev. Lett.* **2001**, *87*, 160601. [CrossRef]
8. Boninsegni, M.; Prokof'ev, N.; Svistunov, B. Worm algorithm and diagrammatic Monte Carlo: A new approach to continuous-space path integral Monte Carlo simulations. *Phys. Rev. E* **2006**, *74*, 036701. [CrossRef]
9. Boninsegni, M.; Prokof'ev, N.; Svistunov, B. Worm algorithm for continuous-space path integral Monte Carlo simulations. *Phys. Rev. Lett.* **2006**, *96*, 070601. [CrossRef]
10. Cinti, F.; Cappellaro, A.; Salasnich, L.; Macrì, T. Superfluid Filaments of Dipolar Bosons in Free Space. *Phys. Rev. Lett.* **2017**, *119*, 215302. [CrossRef]
11. Cinti, F. and Wang, D.-W. and Boninsegni, M.. Phases of dipolar bosons in a bilayer geometry. *Phys. Rev. A* **2017**, *95*. [CrossRef]
12. Bombín, R.; Mazzanti, F.; Boronat, J. Berezinskii-Kosterlitz-Thouless transition in two-dimensional dipolar stripes. *Phys. Rev. A* **2019**, *100*, 063614. [CrossRef]
13. Cinti, F.; Jain, P.; Boninsegni, M.; Micheli, A.; Zoller, P.; Pupillo, G. Supersolid Droplet Crystal in a Dipole-Blockaded Gas. *Phys. Rev. Lett.* **2010**, *105*, 135301. [CrossRef] [PubMed]
14. Saccani, S.; Moroni, S.; Boninsegni, M. Phase diagram of soft-core bosons in two dimensions. *Phys. Rev. B* **2011**, *83*, 092506. [CrossRef]
15. Carleo, G.; Boéris, G.; Holzmann, M.; Sanchez-Palencia, L. Universal Superfluid Transition and Transport Properties of Two-Dimensional Dirty Bosons. *Phys. Rev. Lett.* **2013**, *111*, 050406. [CrossRef]
16. Cinti, F.; Macrì, T.; Lechner, W.; Pupillo, G.; Pohl, T. Defect-induced supersolidity with soft-core bosons. *Nat. Commun.* **2014**, *5*, 3235. [CrossRef] [PubMed]
17. Pascual, G.; Boronat, J. Quasiparticle Nature of the Bose Polaron at Finite Temperature. *Phys. Rev. Lett.* **2021**, *127*, 205301. [CrossRef]
18. Boninsegni, M.; Prokof'ev, N.; Svistunov, B. Superglass Phase of ^{4}He. *Phys. Rev. Lett.* **2006**, *96*, 105301. [CrossRef]
19. Boninsegni, M.; Kuklov, A.B.; Pollet, L.; Prokof'ev, N.V.; Svistunov, B.V.; Troyer, M. Fate of Vacancy-Induced Supersolidity in ^{4}He. *Phys. Rev. Lett.* **2006**, *97*, 080401. [CrossRef]

20. Corboz, P.; Boninsegni, M.; Pollet, L.; Troyer, M. Phase diagram of ^{4}He adsorbed on graphite. *Phys. Rev. B* **2008**, *78*, 245414. [CrossRef]
21. Boninsegni, M. Quantum statistics and the momentum distribution of liquid parahydrogen. *Phys. Rev. B* **2009**, *79*, 174203. [CrossRef]
22. Rota, R.; Boronat, J. Path Integral Monte Carlo Calculation of Momentum Distribution in Solid ^{4}He. *J. Low Temp. Phys.* **2011**, *162*, 146–153. [CrossRef]
23. Osychenko, O.N.; Rota, R.; Boronat, J. Superfluidity of metastable glassy bulk para-hydrogen at low temperature. *Phys. Rev. B* **2012**, *85*, 224513. [CrossRef]
24. Ferré, G.; Boronat, J. Dynamic structure factor of liquid ^{4}He across the normal-superfluid transition. *Phys. Rev. B* **2016**, *93*, 104510. [CrossRef]
25. Metropolis, N.; Rosenbluth, A.W.; Rosenbluth, M.N.; Teller, A.H.; Teller, E. Equation of state calculations by fast computing machines. *J. Chem. Phys.* **1953**, *21*, 1087–1092. [CrossRef]
26. Hastings, W.K. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika* **1970**, *57*, 97–109. [CrossRef]
27. Trotter, H.F. On the product of semi-groups of operators. *Proc. Am. Math. Soc.* **1959**, *10*, 545–551. [CrossRef]
28. Sprik, M.; Klein, M.L.; Chandler, D. Staging: A sampling technique for the Monte Carlo evaluation of path integrals. *Phys. Rev. B* **1985**, *31*, 4234–4244. [CrossRef]
29. Sakkos, K.; Casulleras, J.; Boronat, J. High order Chin actions in path integral Monte Carlo. *J. Chem. Phys.* **2009**, *130*, 204109. [CrossRef]
30. Prokof'ev, N.V.; Svistunov, B.V.; Tupitsyn, I.S. Exact, complete, and universal continuous-time worldline Monte Carlo approach to the statistics of discrete quantum systems. *J. Exp. Theor. Phys.* **1998**, *87*, 310–321. [CrossRef]
31. Lévy, P. Sur certains processus stochastiques homogènes. *Compos. Math.* **1940**, *7*, 283–339.
32. Borrmann, P.; Franke, G. Recursion formulas for quantum statistical partition functions. *J. Chem. Phys.* **1993**, *98*, 2484–2485. [CrossRef]
33. Krauth, W. *Statistical Mechanics Algorithms and Computations*; Oxford University Press: Oxford, UK, 2006.
34. Abramowitz, M.; Stegun, I.A. (Eds.) *Handbook of Mathematical Functions*; Dover Books on Mathematics; Dover Publications: Mineola, NY, USA, 1965.
35. Hansen, J.P.; Levesque, D.; Schiff, D. Fluid-Solid Phase Transition of a Hard-Sphere Bose System. *Phys. Rev. A* **1971**, *3*, 776–780. [CrossRef]
36. Kalos, M.H.; Levesque, D.; Verlet, L. Helium at zero temperature with hard-sphere and other forces. *Phys. Rev. A* **1974**, *9*, 2178–2195. [CrossRef]
37. Spada, G.; Pilati, S.; Giorgini, S. Thermodynamics of a dilute Bose gas: A path-integral Monte Carlo study. *Phys. Rev. A* **2022**, *105*, 013325. [CrossRef]
38. Cao, J.; Berne, B.J. A new quantum propagator for hard sphere and cavity systems. *J. Chem. Phys.* **1992**, *97*, 2382–2385. [CrossRef]