



UNIVERSITY OF TRENTO
DEPARTMENT OF PHYSICS
Master Degree in Physics

NEURAL NETWORK QUANTUM STATES IN THE
OCCUPATION NUMBER FORMALISM

Supervisors:

Francesco Pederiva
Alessandro Lovato

Graduant:

Mauro Rigo

Abstract

Quantum many-body problems are widespread in numerous physics research fields. Studying their properties, however, is highly complicated as their complexity increases exponentially with the number of particles involved. For more than a century, substantial effort has been invested in exploring new theoretical approaches and devising algorithms to find approximate solutions to many-body problems, always attempting to retain as much information as possible in the representation of quantum systems using classical resources. In this work, we discuss the theoretical foundations and the motivations behind a novel Variational Monte Carlo approach that combines the flexibility of Fock space wave functions with the expressivity of neural networks. We describe in detail the step-by-step implementation of the algorithm specifically designed for studying Fermi gases, testing and verifying every part of it. We consider both a one-body and a two-body potential, in the latter case using a more general approach that can be easily employed to study different systems with different interactions. Finally, we present a proof of concept for the application of the algorithm and discuss its current limitations, advantages and future perspectives.

Acknowledgements

I would like to thank Professor Francesco Pederiva for giving me the opportunity to work on this project and for always being available. His deep knowledge and extensive experience in this field are stimulating and highly valuable for defining the fine details of the implementation of sophisticated algorithms to tackle physics problems. Moreover, I would like to thank Dr. Alessandro Lovato for following me closely throughout the project, always being open to offer advice, support and practical help. Our discussions allowed me to better understand the different aspects of this research field as well as improve my critical thinking.

I would also like to address a word of thank to my colleagues and roommates in Trento and in particular to the well known group of “i cetriolini”, their company contributed to make the whole experience of the university lighter. The same is true for my friends in Feltre, which I’m always looking forward to spend a casual weekend with. Moreover, I would like to thank Francesco and Filippo in particular, who have been the closest to me in these months.

Finally, but very importantly nonetheless, I am extremely grateful to my family. Their constant support, concern and interest towards what I do are fundamental for me.

Contents

Abstract	i
Acknowledgements	ii
List of figures	vi
List of tables	viii
1 Introduction	1
1.1 Slater determinant basis	1
1.2 Variational Monte Carlo method	4
1.3 Neural Network Quantum States	8
2 Fock space VMC with one-body potential	13
2.1 Finite difference implementation	14
2.2 Implementation of the VMC algorithm in 1D	15
2.2.1 Metropolis algorithm	15
2.2.2 One particle case	17
2.2.3 Arbitrary number of particles	20
2.2.4 Full Configuration Interaction procedure	23
2.2.5 Variational algorithm	24
2.3 Full VMC procedure for arbitrary D and M	27
2.4 Implementation of the Neural-network Quantum States	31
2.5 Stochastic Reconfiguration	31
2.6 Spin	35
3 Fock space VMC with two-body potential	37
3.1 Practical calculation of the local energy	37
3.1.1 Faster summation over the transferred momenta	42
3.2 Ultracold interacting Fermi gases	43
4 Conclusions and perspectives	45
A Markov chains and Metropolis algorithm	47
B Exact discretization of the HO Hamiltonian	49
C Software and hardware specifications	51

List of Figures

1.1	<i>Left:</i> Schematic of a FCNN. Each light blue circle represents a neuron and the gray lines are the connections between them. This network has a depth $D = 3$, width $W = 6$, takes 5 inputs and outputs 2 values. <i>Right:</i> Common activation functions.	9
2.1	$ E_0 - E_0^{theo} $ for $S = 5, 11, 21, 33, 41$ and $L \in [2, 60]$	15
2.2	Local kinetic, potential and total energies for 10^4 out of 10^6 steps of an MC run with $S = 21$ and $L = 10$ and wave function coefficients $c_N = c_\alpha$ calculated via exact diagonalization.	18
2.3	Kinetic energy autocorrelation coefficients for a MC run of 10^6 samples with $S = 21$ and $L = 10$. <i>Left:</i> $N_{\text{void}} = 1, N_{\text{walk}} = 1$ (samples taken every MC step). <i>Right:</i> $N_{\text{void}} = 200, N_{\text{walk}} = 1000$. Note the different scales of the x axis.	19
2.4	Results of a full VMC procedure with $M = 1, S = 15$ and $L = 10$. At each minimization step we ran 10000 walkers for 20 MC steps with $N_{\text{void}} = 200$. For the RMSprop algorithm we used $\rho = 0.9$ and $\eta = 0.05$ for the first 100 steps and $\eta = 0.02$ for the remaining 100. <i>Left:</i> Behaviour of the variational energy as a function of the minimization step. <i>Right:</i> Final coefficients compared with the exact ones calculated using the FD orbitals. The two curves overlap nearly perfectly. Since $M = 1$, we indicate the wave function coefficients using the state index α ($ N\rangle = K_\alpha\rangle$).	27
2.5	Convergence of different update rules for the variational parameters to the exact energy as calculated using the FD approach. The time step (or learning rate) of SGD was 0.007, while for RMSprop we used standard parameters ($\rho = 0.9$ and $\eta = 0.001$). The parameters of adaptive time step SR were $\delta\tau_0 = 0.0007, \varepsilon = 2 \times 10^{-5}$ while those of adaptive epsilon SR were $\varepsilon_{\text{min}} = 10^{-4}, \varepsilon_{\text{min}} = 10^{-6}, \delta\tau = 0.007$ and both schemes checked 6 different values for $\delta\tau$ and ε as specified above.	34
3.1	Representation of the reachable momenta in 2D with $S = 25$. The vector associated with each square represents the momentum itself \mathbf{K}_α ($\delta k = 1$) while the gray number in the top right corner the associated momentum index α . The colored squares indicate two arbitrary α_1 and α_2 , while the black rectangle highlights the momentum states reachable by both \mathbf{K}_{β_1} and \mathbf{K}_{β_2} through valid \mathbf{Q} in Eq. (3.5).	38

A.1 *Left:* Comparison between the lowest eigenenergy of the correctly discretized Schrödinger equation and the true ground-state energy of a 1D HO for different sizes of the periodic box L and different numbers of momentum states S . The discrepancies on the left and on the right of the plot are due to the infrared and ultraviolet cutoffs respectively (note that, for constant L , the UV cutoff depends on the number of states S via $k_{max} = \pi(S - 1)/L$.) *Right:* Comparison between the correct discretization and the FD approximation of the HO potential in coordinate space. 50

List of Tables

I	Average values with estimated errors of the energies obtained from a MC run not accounting for autocorrelations and accounting for autocorrelations compared with the results of the finite difference approach. The total energy is reported with an arbitrary number of significant digits and with no error as this is only determined by machine precision (which is of order 10^{-15} in this case, thus the error may be either <code>nan</code> , 0 or of order $\sim 10^{-10}$).	19
II	Average values with estimated errors of the energies obtained from various MC runs of 10^5 steps compared with the results of the FD approach. The total energy is once again reported with an arbitrary number of significant digits and with no error as this is only determined by machine precision. In all cases we used $N_{\text{void}} = 200$ and $N_{\text{walk}} = 1000$	22
III	Example of the Fock states ordering we shall use in the $D = 1$ case, for $M = 4$ particles and $S = 6$ momentum states. The numbers in the round brackets represent the indices of the occupied momenta $\alpha_i = 0, \dots, S - 1, i = 0, \dots, M - 1$. The image on the right is a visualization of this ordering: each stack essentially represents one vector describing $ N\rangle$ and the filled and empty squares represent the occupied and unoccupied SPOs.	23
IV	Final energies obtained using the VMC procedure compared with the FD results for different combinations of M, S and L . We also indicate the number of Fock states N_F . Excluding the first line, all results were obtained with more than 1500 minimization steps, each of 2×10^5 samples in total with $N_{\text{void}} = 200$. For each optimization we decreased η from $0.05 \div 0.01$ to a final value of 0.005 for the last ~ 500 steps in order to reduce fluctuations. Note that, for each combination of parameters, we run multiple optimizations, always obtaining results larger or similar to the ones reported here. The results presented are obtained from a final VMC calculation with the optimized parameters.	27
V	Average values with estimated errors of the energies obtained from a MC run of 10^5 samples with $M = D = 2, A = 9$ and $L = 5$ compared with the results of the full CI approach. As usual, the total energy is reported with an arbitrary number of significant digits and with no error for the reasons discussed in Sec. 2.2.2. . . .	30

VI	Results of two optimizations using the VMC algorithm with NQS. In the first case, the FCNNs were comprised of 16 and 8 neurons in the first and second hidden layers and we performed 1000 optimization steps with $\delta\tau = 0.07$. In the second case, we employed two layers of 32 and 16 neurons and performed 1500 optimization steps with $\delta\tau = 0.02$	35
VII	Results of four optimizations using the VMC algorithm with NQS including spin. For the first two results, the NNs are comprised of layers of 32, 16 and 16 neurons and we performed 3000 optimization steps with a time step $\delta\tau$ decreasing from 0.003 to 0.0005, while for the other optimizations featured NNs with layers of 50, 32 and 32 neurons and 2000 optimization steps with $\delta\tau$ decreasing from 0.01 to 0.006 and from 0.02 to 0.001.	36
VIII	Results of optimizations using the VMC algorithm with NQS for a two-body potential against full CI and DMC calculations. For the results with $A = 11$ we used NNs with layers of 32, 16 and 16 neurons and we performed 2000 optimization steps with a time step $\delta\tau$ decreasing from 0.005 to 0.0007, while for the $A = 7$ optimization the NNs featured layers of 20, 12 and 12 neurons and 2000 optimization steps were performed with $\delta\tau$ from 0.007 to 0.001. The full CI procedure was not used for the $L = 10$, $A = 11$ and $M = 6$ case due to the fact that filling the Hamiltonian matrix was too expensive in terms of computational time. Note that the DMC in coordinate space is not affected by the UV cutoff and therefore it only represents a qualitative comparison in the infinite cutoff limit for testing the correctness of the implementation of the two-body potential in the VMC and in the full CI approaches.	44

Chapter 1

Introduction

Predicting the properties of quantum many-body systems is one of the most challenging tasks in physics. Similarly to classical many-body problems, no analytical solutions to a general Schrödinger equation expressed in a closed form exist for more than two interacting particles. Even finding the eigenstates of simple discrete systems, such as spin systems, which could be done by diagonalizing the Hamiltonian matrix, rapidly becomes an intractable problem due to the Hilbert space growing exponentially in size as the number of particles increases. Nonetheless, quantum many-body problems are encountered in several physical contexts, including nuclear and subnuclear physics, high-energy physics, condensed matter physics, quantum chemistry, quantum computing, and quantum information theory. For this reason, the search for numerical methods to find approximate solutions to these problems has been an active field of research for many decades, leading to the development of a wide range of progressively more sophisticated techniques. Based on the physical context and the properties of interest, different approaches are routinely adopted to study quantum many-body problems; as far as the determination of the ground-state properties of a given Hamiltonian is concerned, quantum Monte Carlo (QMC) algorithms have proven to yield the most accurate results [1]. In this Chapter, we introduce the main concepts at the basis of a novel QMC approach to study fermionic many-body systems that merges different powerful techniques from current state-of-the-art methods.

1.1 Slater determinant basis

As imposed by the Fermi-Dirac statistics, the wave function of a system of fermions $|\psi\rangle$ needs to be antisymmetric under the exchange of any two particles. In numerical techniques, one standard way to ensure this condition consists in including Slater determinants [2] in the functional expression of $|\psi\rangle$. For example, the Hartree-Fock (HF) method [3] provides an eigenfunction equation $\hat{F}\phi_\gamma(\mathbf{q}) = \varepsilon_\gamma\phi_\gamma(\mathbf{q})$ (where \mathbf{q} are the spacial and spin coordinates of a particle) whose solutions are single particle orbitals (SPOs) $\phi_\gamma(\mathbf{q})$ with orbital energies ε_γ . The M orbitals with

lowest energy are then used to construct the M -body wave function as

$$\psi(\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_M) = \frac{1}{\sqrt{M!}} \begin{vmatrix} \phi_1(\mathbf{q}_1) & \phi_2(\mathbf{q}_1) & \cdots & \phi_M(\mathbf{q}_1) \\ \phi_1(\mathbf{q}_2) & \phi_2(\mathbf{q}_2) & \cdots & \phi_M(\mathbf{q}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_1(\mathbf{q}_M) & \phi_2(\mathbf{q}_M) & \cdots & \phi_M(\mathbf{q}_M) \end{vmatrix} \quad (1.1)$$

Exchanging the coordinates of two particles translates into swapping two rows in Eq. (1.1) and, due to the properties of determinants, this leads to a change of sign. Moreover, if two fermions have the same coordinates \mathbf{q} the determinant is null, fulfilling the Fermi-Dirac statistics. The HF method provides the best possible wave function written as a single Slater determinant; however, being a simple anti-symmetrized product of SPOs, the one in Eq. (1.1) is a mean-field wave function, and as such it is not sufficient to fully capture the dynamical correlations among the particles arising from their interaction.

A Slater determinant wave function in the form of Eq. (1.1)¹ is also used in the Kohn-Sham Density Functional Theory (DFT) approach [4], though in this case correlations are explicitly taken into account through an effective potential term in the Hamiltonian in the form of a functional of the density of the system. This may in principle allow the method to reproduce the exact ground-state of the system, as the Hohenberg-Kohn theorem implies [5]; nevertheless, there is no general procedure for calculating the correlation functional, and approximate parametrizations are typically used [6].

An effective way to include the relevant dynamical correlations of a system is to expand the wave function as a linear combination of Slater determinants. One method that does so is the full Configuration Interaction (CI) one [7]: for a system of M interacting particles, given a discrete set of SPOs $|\phi_\gamma\rangle$, $\gamma = 1, \dots, S$ (for example the ones obtained using a HF procedure) and a reference determinant $|\psi_0\rangle$ containing the M SPOs up to Fermi level, the full CI wave function is given by

$$\begin{aligned} |\psi_{\text{FCI}}\rangle &= |\psi_0\rangle + \sum_{i,a} C_i^a |\psi_i^a\rangle + \sum_{i>j, a>b} C_{ij}^{ab} |\psi_{ij}^{ab}\rangle + \dots \\ &+ \sum_{\substack{i>j>k>\dots \\ a>b>c>\dots}} C_{ijk\dots}^{abc\dots} |\psi_{ijk\dots}^{abc\dots}\rangle + \dots \\ &= |\psi_0\rangle + \sum_{p=1}^M \hat{C}_p |\psi_0\rangle \end{aligned} \quad (1.2)$$

where the $|\psi_{ijk\dots}^{abc\dots}\rangle$ are p -particles- p -holes excitation determinants constructed by replacing p occupied orbitals i, j, k, \dots with p unoccupied ones a, b, c, \dots in $|\psi_0\rangle$. The coefficients $C_i^a, C_{ij}^{ab}, \dots$ can be determined via exact diagonalization of the Hamiltonian matrix written in the basis of these p -particles- p -holes Slater determinants. In general, for systems with an infinite-dimensional Hilbert space \mathcal{H} , using a finite number of SPOs and thus a finite Slater determinant basis implies restricting the possible wave functions to a subspace of \mathcal{H} and effectively modify the Hamiltonian

¹In truth, in general combinations of degenerate Slater determinants may be used to ensure that the density fulfills the symmetry requirements of the system.

of the system. For this reason, in principle employing the wave function in Eq. (1.2) precludes the possibility of finding the true ground-state of an arbitrary Hamiltonian; nevertheless, in general a finite set of SPOs is sufficient to represent a wave function that is able to reproduce the ground-state properties of a system with high accuracy (for example, full CI solutions are used as a benchmark for more approximate methods in quantum chemistry). However, the number of M -particles Slater determinants that can be constructed from S SPOs is given by

$$N_F = \binom{S}{M} = \frac{S!}{M!(S-M)!} \quad (1.3)$$

that scales factorially with M . For this reason, full CI calculations with large S are typically limited to small systems; nonetheless, being an exact method in the space spanned by the SPOs, the full CI method provides very strong reference results for testing the performance of other approximate algorithms.

One common way to mitigate the factorial scaling of the Slater determinant basis while preserving an efficient representation of dynamical correlations consists in truncating the number of terms in Eq. (1.2) to include only double excitations (CID) or single and double excitations (CISD). Another parallel approach is the Coupled Cluster (CC) one [8], which consists in defining the wave function as $|\psi_{CC}\rangle = \exp(\hat{T})|\psi_0\rangle$, where the cluster operator $\hat{T} = \hat{T}_1 + \hat{T}_2 + \dots$ is defined as

$$\hat{T}_1|\psi_0\rangle = \sum_{i,a} t_i^a |\psi_i^a\rangle, \quad \hat{T}_2|\psi_0\rangle = \sum_{i>j, a>b} t_{ij}^{ab} |\psi_{ij}^{ab}\rangle, \quad \dots \quad (1.4)$$

The advantage of using such a definition can be noticed by expanding the exponential of the cluster operator and comparing the p -particles- p -holes operators with the full CI expansion in Eq. (1.2):

$$\hat{C}_1 = \hat{T}_1, \quad \hat{C}_2 = \hat{T}_2 + \frac{1}{2}\hat{T}_1^2, \quad \hat{C}_3 = \hat{T}_3 + \hat{T}_1\hat{T}_2 + \frac{1}{6}\hat{T}_1^3, \quad \dots \quad (1.5)$$

thus truncating the cluster operator up to single (CCS), double (CCSD) or more excitation brings contributions to all \hat{C}_p , which frequently leads to more accurate results with respect to CID and CISD and guarantees a linear scaling of the energy with the number of particles [9], which is not the case for truncated CI expansions.

The full CI wave function and its truncations as well as the CC one have also been included in different Projector Monte Carlo (PMC) approaches [10, 11, 12]; on top of being highly effective at incorporating correlations, in this context these wave functions also allow to mitigate the notorious fermion sign problem. Additionally, a major advantage of working in a discrete many-body space is that it allows to also deal with non-local interactions in a much more tractable way compared to continuous-space approaches, see for instance Ref. [13]. This enables the application of QMC techniques to interactions that have never been treated in this context before.

In parallel with PMC algorithms using finite basis sets and also due to their success, different authors have proposed Variational Monte Carlo (VMC) algorithms using such bases [14]; in this work, we follow this line and expand the variational wave function in a Slater determinant basis, representing each determinant in the occupation number (or second quantization) formalism such

that a generic multi-body wave function is given by a linear combination of Fock states

$$|\psi\rangle = \sum_N |N\rangle\langle N|\psi\rangle = \sum_N c_N |N\rangle \quad (1.6)$$

where $|N\rangle = |n_1, \dots, n_S\rangle$ and each n_γ represents the occupation number of the SPOs $|\phi_\gamma\rangle$, with $\sum_\gamma n_\gamma = M$. Since the particles are fermions, clearly $n_\gamma = 0, 1$.

1.2 Variational Monte Carlo method

The Variational Monte Carlo (VMC) approach is based on the variational principle. Considering a Hilbert space \mathcal{H} and a Hamiltonian operator \hat{H} acting on it and given a state $|\psi\rangle$ defined on \mathcal{H} and the ground-state of the Hamiltonian $|\psi_0\rangle$ with energy E_0 , the variational principle states

$$E_0 = \frac{\langle\psi_0|\hat{H}|\psi_0\rangle}{\langle\psi_0|\psi_0\rangle} \leq \frac{\langle\psi|\hat{H}|\psi\rangle}{\langle\psi|\psi\rangle} = E \quad (1.7)$$

with the equal sign if and only if $|\psi\rangle = |\psi_0\rangle$. Many techniques exploit this theorem to provide approximate solutions of quantum many-body problems, including the previously discussed HF and CI methods. For the purpose of this work, we simply observe that one could propose a *variational wave function* $|\psi_V(\boldsymbol{\lambda})\rangle$ parametrized in terms of a set of *variational parameters* $\boldsymbol{\lambda}$, calculate the *variational energy* $E_V = \langle\psi_V|\hat{H}|\psi_V\rangle / \langle\psi_V|\psi_V\rangle$ and find its minimum, which corresponds to an optimal set of variational parameters. Since the variational principle ensures that the resulting minimum energy cannot be lower than the true ground-state energy of the Hamiltonian E_0 , the method provides an upper bound for E_0 and, theoretically, it could come arbitrarily close to $|\psi_0\rangle$ for a suitably parametrized $|\psi_V(\boldsymbol{\lambda})\rangle$. However, in practice this procedure is not possible for arbitrary \hat{H} and $|\psi_V(\boldsymbol{\lambda})\rangle$, since an analytical expression of E_V may not exist.

Let us now assume the wave function to be written as a linear combination of orthogonal Slater determinants using the notation in Eq.(1.6) and consider the subspace spanned by $|N\rangle$ such that $\sum_N |N\rangle\langle N| = \hat{1}$. Minimizing the variational energy E_V corresponds to imposing its gradient with respect to the variational parameters $\boldsymbol{\lambda}$ to be equal to 0:

$$\partial_{\lambda_p} E_V = G_p = 2 \left(\frac{\langle\partial_{\lambda_p} \psi_V|\hat{H}|\psi_V\rangle}{\langle\psi_V|\psi_V\rangle} - E_V \frac{\langle\partial_{\lambda_p} \psi_V|\psi_V\rangle}{\langle\psi_V|\psi_V\rangle} \right) = 0 \quad \forall p \quad (1.8)$$

where we leave the dependence of the wave function on the variational parameters implicit for ease of notation. Adding some completeness relations in the expression of G_p , one can find

$$\begin{aligned} G_p &= 2 \left(\frac{\sum_{N'N''} \langle\partial_{\lambda_p} \psi_V|N'\rangle\langle N'|\hat{H}|N''\rangle\langle N''|\psi_V\rangle}{\sum_{N'} \langle\psi_V|N'\rangle\langle N'|\psi_V\rangle} - E_V \frac{\sum_{N'} \langle\partial_{\lambda_p} \psi_V|N'\rangle\langle N'|\psi_V\rangle}{\sum_{N'} \langle\psi_V|N'\rangle\langle N'|\psi_V\rangle} \right) \\ &= 2 \left(\frac{\sum_{N'N''} \partial_{\lambda_p} \psi_V^*|_{N'} \langle N'|\hat{H}|N''\rangle c_{N''}}{\sum_{N'} |c_{N'}|^2} - E_V \frac{\sum_{N'} \partial_{\lambda_p} \psi_V^*|_{N'} c_{N'}}{\sum_{N'} |c_{N'}|^2} \right) \end{aligned} \quad (1.9)$$

For example, the full CI method essentially uses the c_N in Eq.(1.6) as variational parameters,

in which case imposing $G_p = G_{c_N} = 0$ implies

$$\sum_{N''} \langle N | \hat{H} | N'' \rangle c_{N''} = E_V c_N \quad \forall N \quad (1.10)$$

which corresponds to diagonalizing the Hamiltonian matrix $\langle N | \hat{H} | N' \rangle$. Nevertheless, as discussed in the previous Section, the number of basis states $|N\rangle$ scales factorially with M , making this diagonalization intractable except for systems comprised of few particles. Moreover, the exact solution of the equations $G_p = 0$ may not be possible for arbitrary parametrizations of the variational wave function, thus approximate minimization methods are required.

In a more general view, studying the ground-state properties of a system involves the calculation of expectation values of different operators \mathcal{O} , whether these are actually needed for computing observables or for numerical minimization techniques. For a generic basis $|\mathbf{q}\rangle$ with a completeness relation² $\int d\mathbf{q} |\mathbf{q}\rangle\langle\mathbf{q}| = \hat{1}$, the expectation value of \mathcal{O} over a variational wave function is given by

$$\langle \hat{\mathcal{O}} \rangle_V = \frac{\langle \psi_V | \hat{\mathcal{O}} | \psi_V \rangle}{\langle \psi_V | \psi_V \rangle} = \frac{\int d\mathbf{q} \langle \psi_V | \mathbf{q} \rangle \langle \mathbf{q} | \hat{\mathcal{O}} | \psi_V \rangle}{\int d\mathbf{q} \langle \psi_V | \mathbf{q} \rangle \langle \mathbf{q} | \psi_V \rangle} = \frac{\int d\mathbf{q} \psi_V^*(\mathbf{q}) \hat{\mathcal{O}} \psi_V(\mathbf{q})}{\int d\mathbf{q} |\psi_V(\mathbf{q})|^2} \quad (1.11)$$

In most cases, the integrals cannot be calculated analytically, thus one could discretize the hyperspace into n points and use approximate integration algorithms; however, in general for a system of M particles in D dimensions the number of integration variables increases linearly with MD , and the error of standard integration algorithms in MD dimensions scales as $\mathcal{O}(n^{-1/MD})$. For this reason, for systems of many particles, the number of points needed to estimate the integrals accurately may become intractable. The most efficient method for approximating high-dimensional integrals involves the usage of stochastic Monte Carlo (MC) techniques [15]. Considering again the expression in Eq. (1.11) and multiplying and dividing by $\psi_V(\mathbf{q})$ in the integral at the numerator, we obtain

$$\langle \hat{\mathcal{O}} \rangle_V = \frac{\int d\mathbf{q} |\psi_V(\mathbf{q})|^2 \frac{\hat{\mathcal{O}} \psi_V(\mathbf{q})}{\psi_V(\mathbf{q})}}{\int d\mathbf{q} |\psi_V(\mathbf{q})|^2} \quad (1.12)$$

where

$$\frac{\hat{\mathcal{O}} \psi_V(\mathbf{q})}{\psi_V(\mathbf{q})} := \mathcal{O}_L(\mathbf{q}) \quad (1.13)$$

is the *local* expectation of the operator $\hat{\mathcal{O}}$. Eq. (1.12) now has the form of an expectation value of $\mathcal{O}_L(\mathbf{q})$ over the probability distribution

$$P(\mathbf{q}) = \frac{|\psi_V(\mathbf{q})|^2}{\int d\mathbf{q} |\psi_V(\mathbf{q})|^2} \quad (1.14)$$

which is correctly normalized. Thus, if we define \mathbf{q} as a random variable, an integral of the kind of Eq. (1.12) can be evaluated stochastically exploiting the central limit theorem. In fact, given a set of N_s independent samples x_i , $i = 1, \dots, N_s$ of a stochastic variable X with probability

²Here we use a generic continuous basis to better illustrate the advantage of MC methods, though all the formalism can be straightforwardly translated for a discrete basis, as we discuss below.

distribution $P[X]$ and a function $f(x) : x \mapsto t \in \mathbb{R}$, the central limit theorem states that the average estimator

$$S_{N_s} = \frac{1}{N_s} \sum_{i=1}^{N_s} f(x_i) \quad (1.15)$$

is such that

$$\lim_{N_s \rightarrow \infty} P[S_{N_s}] = \frac{1}{\sqrt{2\pi\sigma_{N_s}^2(f)}} \exp\left\{-\frac{S_{N_s} - \langle f \rangle}{2\sigma_{N_s}^2(f)}\right\} \quad (1.16)$$

where

$$\langle f \rangle = \int dx P(x) f(x) \quad \text{and} \quad \sigma_{N_s}(f) = \sqrt{\frac{\sigma^2(f)}{N_s - 1}}, \quad \sigma^2(f) = \langle f^2 \rangle - \langle f \rangle^2 \quad (1.17)$$

that is, the probability distribution of S_{N_s} approaches a Gaussian centered in $\langle f \rangle$ with width $\sigma_{N_s}(f)$ as N_s increases. Hence, the central limit theorem provides a recipe for the calculation of the expression in Eq. (1.12), i.e.

$$\langle \hat{\mathcal{O}} \rangle_V \simeq \frac{1}{N_s} \sum_{i=1}^{N_s} \mathcal{O}_L(\mathbf{q}^{[i]}) \pm \sqrt{\frac{1}{N_s - 1} \left[\left(\frac{1}{N_s} \sum_{i=1}^{N_s} \mathcal{O}_L^2(\mathbf{q}^{[i]}) \right) - \left(\frac{1}{N_s} \sum_{i=1}^{N_s} \mathcal{O}_L(\mathbf{q}^{[i]}) \right)^2 \right]} \quad (1.18)$$

where the $\mathbf{q}^{[i]}$ are independent samples distributed with the probability $P(\mathbf{q})$ defined in Eq. (1.14). The advantages of using this integration method are clearly highlighted in Eq. (1.18): the error of the estimate has a favourable scaling with the number of samples ($\mathcal{O}(N_s^{-1/2})$) and it is independent of M and D . Another important property to notice is that the stochastic integration is not affected by the normalization of the wave function, which can thus be arbitrary.

The basis $|\mathbf{q}\rangle$ does not actually need to be continuous, since the central limit theorem is valid for discrete random variables as well. Thus, in the case of the basis defined in the previous Section in terms of Fock states $|N\rangle$, one can write the expectation of an arbitrary operator following the same passages discussed above as

$$\langle \hat{\mathcal{O}} \rangle_V = \frac{\langle \psi_V | \hat{\mathcal{O}} | \psi_V \rangle}{\langle \psi_V | \psi_V \rangle} = \sum_N \frac{|c_N|^2}{\sum_{N'} |c_{N'}|^2} \frac{\langle N | \hat{\mathcal{O}} | \psi_V \rangle}{\langle N | \psi_V \rangle} = \sum_N P(N) \mathcal{O}_L(N) \quad (1.19)$$

where

$$\frac{\langle N | \hat{\mathcal{O}} | \psi_V \rangle}{\langle N | \psi_V \rangle} := \mathcal{O}_L(N) \quad , \quad P(N) = \frac{|c_N|^2}{\sum_{N'} |c_{N'}|^2}. \quad (1.20)$$

Calculating the sum in Eq. (1.19) explicitly requires dealing with the full complexity of the basis $|N\rangle$, which in this case scales factorially both with M and S . On the contrary, using the MC method this expectation value can be calculated using Eq. (1.18) (with $\mathbf{q}^{[i]} \rightarrow N^{[i]}$), with the error being independent of M and S .

However, evaluating expectation values stochastically is only possible under the assumption of being able to sample states X with a target probability distribution $P(X)$. This can be done using the *Metropolis-Hastings algorithm* [16], a Markov Chain Monte Carlo (MCMC) method that we discuss in Appendix A. The algorithm essentially generates samples from the target

distribution $P[X]$ by performing a random walk in the space spanned by the random variable X . Starting from an arbitrary state x , the algorithm proposes a transition to a new state y via a known transition kernel $\tilde{T}(x \rightarrow y)$ and accepts it (or retains the current state x) with probability

$$P_{acc} = \min \left(1, \frac{P(y)\tilde{T}(y \rightarrow x)}{P(x)\tilde{T}(x \rightarrow y)} \right) \quad (1.21)$$

In the context of the MC summation introduced above, we may identify a current state $x = N_{curr}$ and a proposed one $y = N_{prop}$ and define the acceptance probability (for a symmetric kernel) as

$$P_{acc} = \frac{|c_{N_{prop}}|^2}{|c_{N_{curr}}|^2} \quad (1.22)$$

The Metropolis-Hastings algorithm then allows to sample configurations N distributed according to the probability in Eq. (1.20), which can then be used to estimate the expectation value in Eq. (1.19) using Eq. (1.18) with $\mathbf{q}^{[i]} \rightarrow N^{[i]}$. Practically, the way one accepts or rejects the proposed transition is by drawing a random number u from a uniform distribution between 0 and 1 and checking whether $u \leq P_{acc}$. If the condition is satisfied, the proposed state is accepted, otherwise the previous one is retained as new state of the Markov chain. We shall refer to this procedure as the “acceptance/rejection” step. Note that, once again, the knowledge of the normalization of the wave function is not required.

At this point, it is important to mention that, by construction, it is likely that two successive configurations generated using this procedure are correlated. For this reason, if one uses each of N_s consecutive samples to estimate a quantity, the estimate of its error could be flawed, since the number of independent samples may actually be smaller than N_s by a factor τ , called *correlation length*. This parameter can be determined by calculating the autocorrelation coefficient:

$$r_k = \frac{\sum_{i=1}^{N_c-k} \left(\mathcal{O}_L(N^{[i]}) - \langle \hat{\mathcal{O}} \rangle \right) \left(\mathcal{O}_L(N^{[i+k]}) - \langle \hat{\mathcal{O}} \rangle \right)}{\sum_{i=1}^{N_c} \left(\mathcal{O}_L(N^{[i]}) - \langle \hat{\mathcal{O}} \rangle \right)^2} \quad (1.23)$$

where N_c is some number of samples, $\mathcal{O}_L(N^{[i]})$ are the local expectations calculated for every sample and $\langle \hat{\mathcal{O}} \rangle$ their average. For a Markovian process, r_k generally follows the behaviour of a decreasing exponential (see for instance Fig. 2.3) starting from $r_0 = 1$ (completely correlated data) and approaching 0 as $k \rightarrow \infty$ (completely uncorrelated data). Assuming $r_k \sim e^{-k/\tau}$, it is thus possible to estimate the autocorrelation length τ by calculating the sum of all r_k from 0 to ∞ (or, practically, a large k_{max}).

In truth, to account for this correlation, it is generally much easier to calculate the local quantities once every $N_{void} \gg 1$ acceptance/rejection step. Additionally, the errors are typically estimated from averages over the \mathcal{O}_L calculated for some $N_{walk} \gg 1$ samples (always skipping N_{void} steps between each sample) rather than directly over all \mathcal{O}_L , as averages are more likely to be distributed according to the limiting Gaussian making their sample error more accurate. Finally, the algorithm needs to be initialized to some state N_0 . It is customary to choose N_0 randomly and, since it may be a bad representative of the target distribution, in general it is beneficial to discard some (say N_{eq}) initial equilibration steps before actually starting to compute

the local quantities once every N_{void} steps.

As mentioned above, in a VMC algorithm the aim is minimizing the variational energy E_V with respect to the variational parameters $\boldsymbol{\lambda}$, which can be performed in many different ways. In case an exact minimization is not feasible, one can use approximate methods to iteratively update the variational parameters in order to reach the minimum of E_V . The simplest update rule for the parameters is provided by the stochastic gradient descent (SGD) method:

$$\boldsymbol{\lambda} \rightarrow \boldsymbol{\lambda} - \eta \mathbf{G} \quad (1.24)$$

where \mathbf{G} represents the gradient of the energy E_V with respect to the variational parameters $\boldsymbol{\lambda}$ and η is the *learning rate*. Returning to Eq. (1.8) and introducing the derivative operator $\hat{\mathcal{O}}_p$ such that $\hat{\mathcal{O}}_p |\psi_V\rangle = |\partial_{\lambda_p} \psi_V\rangle$, one can find

$$\begin{aligned} G_p &= 2 \left(\sum_N \frac{\langle \psi_V | \hat{\mathcal{O}}_p^\dagger | N \rangle \langle N | \hat{H} | \psi_V \rangle}{\langle \psi_V | \psi_V \rangle} - E_V \sum_N \frac{\langle \psi_V | \hat{\mathcal{O}}_p^\dagger | N \rangle \langle N | \psi_V \rangle}{\langle \psi_V | \psi_V \rangle} \right) \\ &= 2 \left(\sum_N P(N) \frac{\langle \psi_V | \hat{\mathcal{O}}_p^\dagger | N \rangle}{\langle \psi_V | N \rangle} \frac{\langle N | \hat{H} | \psi_V \rangle}{\langle N | \psi_V \rangle} - E_V \sum_N P(N) \frac{\langle \psi_V | \hat{\mathcal{O}}_p^\dagger | N \rangle}{\langle \psi_V | N \rangle} \right) \\ &= 2 \left(\sum_N P(N) \mathcal{O}_p^*(N) E_L(N) - \sum_{N'} P(N') E_L(N') \sum_N P(N) \mathcal{O}_p^*(N) \right) \end{aligned} \quad (1.25)$$

where $E_L(N)$ is the local expectation of the energy and in this case $\mathcal{O}_p(N)$ indicates the local expectation of the derivative, which is simply given by the logarithmic derivative of ψ_V calculated for the state $|N\rangle$. Using the MC theory described above, the gradient can then be estimated as

$$G_p \simeq 2 \left(\frac{1}{N_s} \sum_{i=1}^{N_s} \mathcal{O}_p^*(N^{[i]}) E_L(N^{[i]}) \right) - 2 \left(\frac{1}{N_s} \sum_{i=1}^{N_s} E_L(N^{[i]}) \right) \left(\frac{1}{N_s} \sum_{i=1}^{N_s} \mathcal{O}_p^*(N^{[i]}) \right) \quad (1.26)$$

In the next Chapter we shall discuss different optimization techniques, though in all cases this formalism will be essential.

1.3 Neural Network Quantum States

The accuracy reachable by a VMC calculation is, of course, in a large part determined by the expression one uses for the variational wave function $|\psi_V\rangle$. Since the variational principle guarantees that the variational energy E_V cannot be lower than the true ground-state energy of the system E_0 , the more flexible the expression of the variational wave function and thus the lower E_V can reach, the tighter the upper bound for E_0 . Typical VMC calculations in coordinate space for fermions expand the wave function as a product of a mean field part, given by combinations of Slater determinants satisfying the required symmetries, and two-body terms (in the form of Jastrow factors [17]) or beyond to capture dynamical correlations. Despite reaching significant accuracy, these wave functions are nonetheless limited by the choice of specific functional forms both of the SPOs in the Slater determinants and in the multi-body correlation terms.

Recently, a novel approach for the parametrization of a variational wave function using artificial neural networks (ANNs) was proposed [18], dubbed as neural-network quantum state (NQS). ANNs constitute a subclass of machine-learning algorithms which loosely imitate the functioning of the human brain by exchanging information between multiple interconnected “neurons”, i.e. simple nodes which output a linear combination of their inputs passed through some non-linear activation function. These computing systems are essentially black boxes that output highly non-linear functions of their inputs. There exist many different ANN architectures and, in the context of QMC techniques, many authors explored different possibilities, e.g. restricted Boltzmann machines (RMBs) [18] and more elaborate architectures such as convolutional neural networks (CNNs) [19] and recurrent neural networks (RNNs) [20]; in this work, we use simple fully-connected neural networks (FCNNs). In FCNNs, the neurons are stacked in layers and the output of each neuron in a layer is fed to each neuron in the next layer, as illustrated in Fig. 1.1.

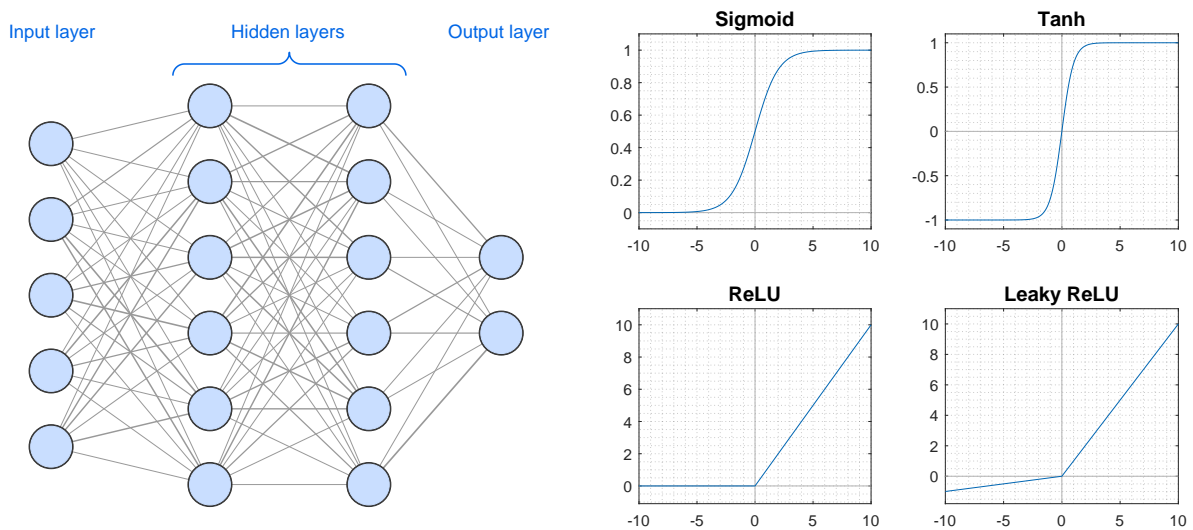


FIG 1.1: *Left*: Schematic of a FCNN. Each light blue circle represents a neuron and the gray lines are the connections between them. This network has a depth $D = 3$, width $W = 6$, takes 5 inputs and outputs 2 values. *Right*: Common activation functions.

In a FCNN, the output of a neuron i in layer j is therefore given by

$$v_i^{(j)} = f \left(\sum_{k=1}^{N_{j-1}} w_{ik}^{(j)} v_k^{(j-1)} + a_i^{(j)} \right) \quad (1.27)$$

where f is the non-linear activation function — some common examples of which are included in Fig. 1.1 — while N_{j-1} is the number of neurons in the $(j - 1)$ -th layer and the $w_{ik}^{(j)}$ and $a_i^{(j)}$ are the *weights* and *biases* for layer j . The structure of a FCNN is determined mainly by three hyperparameters³: the *depth* D (the number of hidden layers plus the output one), the *width* W (the maximum number of neurons in a layer) and the activation function.

A computing system is said to “learn” if its performance with respect to a given task improves

³Hyperparameters are parameters used to control the learning process, to be distinguished with the ones derived from the training such as the weights and biases of an ANN.

with training experience (i.e. repeatedly executing the task). ANNs learn by minimizing a *loss function* which quantifies their prediction performance. In supervised learning settings, the ANN is provided with a set of training examples x connected to some labels y via a mapping $x \mapsto y$ and its task is to model this mapping; in this case, the loss function typically represents a distance between the predictions of the NN for the inputs x and the true associated labels y and the training consists in repeatedly feeding the examples to the network and adjusting its parameters to minimize the loss function. In the case of the problem discussed here, the variational principle itself provides a function to be minimized without the need of training examples: the variational energy E_V . Therefore, one can represent a wave function $\psi_V(\mathbf{q})$ using an ANN, feeding the variables \mathbf{q} as inputs and identifying the weights and biases as the variational parameters; the network is then trained by minimizing E_V , which essentially translates into driving the ANN to reproduce the ground-state wave function in a reinforcement learning scheme. The minimization itself is performed using stochastic techniques that we shall discuss in due time.

FCNNs are universal approximators [21], therefore in principle capable of approximating any (sufficiently well-behaved) function of their inputs. Their flexibility and expressivity makes them very well-suited for representing highly non-linear multi-body functions: in fact, many authors using QMC techniques obtained results competitive with existing state-of-the-art approaches for a variety of spin [22], bosonic [23] and fermionic [24] systems using FCNN quantum states, in some cases even outperforming existing variational results [25].

One of the great challenges that using NQSS entails consists in imposing the physical symmetries that characterize the system investigated and, most notably, in particular for fermionic many-body systems, the exchange symmetry. In this context, the problem has been approached in many ways, namely using backflow transformations of Slater determinants [25], mapping fermionic degrees of freedom to spin ones [26] or directly working in first quantization [27]. As discussed above, in this work we use a different representation of fermionic many-body systems using a Slater determinant space. This is advantageous in that Fock states $|N\rangle = |n_1, \dots, n_S\rangle$ already encode the required exchange symmetry, therefore the occupation numbers n_i can be directly fed to a FCNN to compute the wave function coefficients c_N , which are needed for the VMC algorithm described above.

The reason why it is potentially beneficial to combine ANNs with wave functions in Fock space is associated with the complexity of the description of generic many-body states. In fact, the size of a Hilbert space of a many-body system increases exponentially with the number of particles, therefore, in principle, a complete description of a many-body wave function by classical means requires encoding an exponential amount of information⁴. For systems of fermions, in PMC methods this problem manifests in the form of the sign problem, which in general is NP-hard [28]. However, physically-relevant systems usually only explore small regions of the entire Hilbert space, thereby potentially one may be able to construct efficient representations of physical many-body states with classical resources. In this sense, the success of Fock space wave functions in PMC and VMC algorithms and the flexibility and the ability to efficiently sustain entanglement [29] of ANNs provide an interesting new approach for studying quantum many-body problems.

⁴As a side note, one important advantage of using quantum computers for simulations of quantum many-body systems is that, due to their inherently quantum character, they naturally encode this complexity.

On top of this, for a fixed number of SPOs S , the scaling of the representation discussed here is independent of the number of particles, and the scaling with S itself is in general favorable (typically $\mathcal{O}(S^2)$). For this reason, it is also easy to expand the basis set used, possibly allowing to obtain results independent of its size through extrapolation.

The algorithm we discuss in this work merges the three main concepts described in the previous Sections into a Variational Monte Carlo in Fock space using the occupation number formalism, with NQs parametrizing the variational wave function.

Chapter 2

Fock space VMC with one-body potential

The algorithm we develop here is designed for gases of interacting fermions in an arbitrary number of dimensions. In coordinate space, we essentially describe the gas by dividing the D -dimensional space into hypercubes of side L assuming periodic boundary conditions (PBC). We choose momentum eigenstates (plane waves) as SPOs, since the PBC conveniently force the momenta to lie on a lattice of size $\delta k = 2\pi/L$, that is:

$$\mathbf{k} = \frac{2\pi}{L} (i_1, i_2, \dots, i_D) = \delta k \mathbf{i} \quad (2.1)$$

where $i_d, d = 1, \dots, D$ are integers and D is the dimensionality of the system. Each occupation number $n_\gamma = n_{\mathbf{k}}$ in a Fock state $|N\rangle = |n_1, \dots, n_S\rangle$ then represent the number of particles occupying the state $u_{\mathbf{k}}(\mathbf{r}) \propto \exp(i\mathbf{k} \cdot \mathbf{r})$. Note that we shall refer to the momentum associated with an occupied SPO ($n_\gamma = n_{\mathbf{k}} = 1$) simply as an “occupied momentum”. To deal with a finite number S of SPOs, we impose an ultraviolet cutoff by limiting the component of the momenta to $|k^d| < k_{max} = \delta k(S - 1)/2$. For now we ignore the spin for simplicity.

In this Chapter we implement the algorithm for a one-body potential for the purpose of testing and verifying each part of the code. In fact, applying a VMC procedure in this case is not strictly necessary, because for a separable M -body Hamiltonian the ground-state is simply the antisymmetrized product of the first M eigenstates of the one-body Hamiltonian, while the ground-state energy is given by the sum of the energies associated to these orbitals. However, since the exact M -particles ground-state can be found by diagonalizing a simple matrix (see Eq. (2.6)), studying a one-body potential first allows us to check the correctness of the code at a very low computational cost.

The one-body potential we use is an approximation of a Harmonic Oscillators (HO) potential, derived from the continuous M -body Hamiltonian in momentum space

$$\hat{H} = - \sum_{i=1}^M \frac{1}{2} \hat{\nabla}_{\mathbf{k}_i}^2 + \sum_{i=1}^M \frac{1}{2} k_i^2 \quad (2.2)$$

Note that we shall always use (arbitrary) units that simplify the Hamiltonian as much as possible. To discretize this equation, we use a finite difference approximation of the Laplacian operator and consider a new system with a Schrödinger equation given by (for the correct discretization,

see Appendix B)

$$\langle \mathbf{k} | \hat{H} | \psi \rangle = - \sum_{i=1}^M \sum_{d=1}^D \frac{\psi(\mathbf{k}, k_i^d - \delta k) - 2\psi(\mathbf{k}) + \psi(\mathbf{k}, k_i^d + \delta k)}{2\delta k^2} + \frac{k^2}{2} \psi(\mathbf{k}) = E\psi(\mathbf{k}) \quad (2.3)$$

where $\psi(\mathbf{k}, k_i^d \pm \delta k)$ is the wave function calculated at a point $\mathbf{k} = (\mathbf{k}_1, \dots, \mathbf{k}_M)$ with the d -th component of the i -th particle shifted of $\pm\delta k$. In coordinate space, the corresponding potential felt by each particle along each direction is (using an inverse Fourier transform):

$$V(x) = \frac{1}{2\delta k^2} \left(2 - e^{-i\delta k x} - e^{i\delta k x} \right) = \frac{1}{\delta k^2} (1 - \cos(\delta k x)) = \frac{2}{\delta k^2} \sin^2 \left(\frac{\delta k}{2} x \right) = \frac{x^2}{2} + \mathcal{O}(\delta k^2) \quad (2.4)$$

which, in the limit $L \rightarrow \infty$, converges to the HO potential. Note that we shall begin our treatment of this potential without making use of the creation and annihilation operators to make it easier to follow.

2.1 Finite difference implementation

Since the potential is separable, the Schrödinger equation in Eq. (2.3) is very simple to solve using the finite difference (FD) method by considering the 1D, $M = 1$ case and constructing the D -dimensional M -body wave function with the required symmetry. Defining

$$U_k = \frac{1}{\delta k^2} + \frac{k^2}{2} \quad (2.5)$$

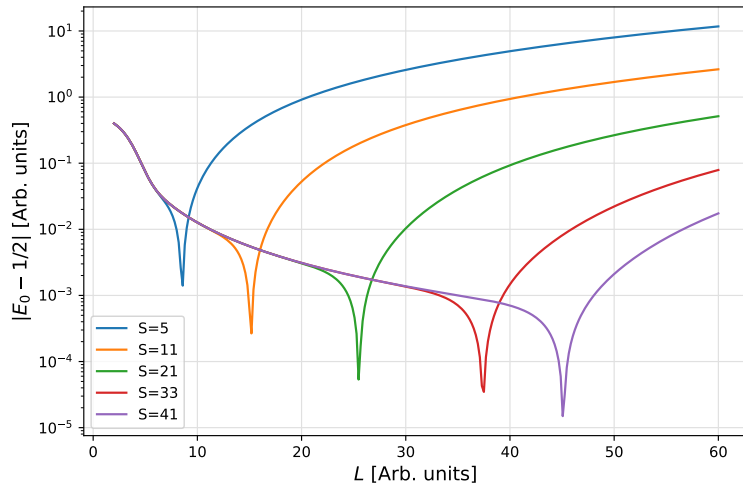
one can write Eq. (2.3) for $M = D = 1$ as

$$\mathbf{U}\phi = E\phi$$

$$\begin{pmatrix} U_{-\bar{k}} & -\frac{1}{2\delta k^2} & 0 & \cdots & 0 & 0 \\ -\frac{1}{2\delta k^2} & U_{-\bar{k}+1} & -\frac{1}{2\delta k^2} & \cdots & 0 & 0 \\ 0 & -\frac{1}{2\delta k^2} & U_{-\bar{k}+2} & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & U_{\bar{k}-1} & -\frac{1}{2\delta k^2} \\ 0 & 0 & 0 & \cdots & -\frac{1}{2\delta k^2} & U_{\bar{k}} \end{pmatrix} \begin{pmatrix} \phi_{-\bar{k}} \\ \phi_{-\bar{k}+1} \\ \phi_{-\bar{k}+2} \\ \vdots \\ \phi_{\bar{k}-1} \\ \phi_{\bar{k}} \end{pmatrix} = E \begin{pmatrix} \phi_{-\bar{k}} \\ \phi_{-\bar{k}+1} \\ \phi_{-\bar{k}+2} \\ \vdots \\ \phi_{\bar{k}-1} \\ \phi_{\bar{k}} \end{pmatrix} \quad (2.6)$$

where we use the symbol ϕ to denote the wave function in order distinguish it from the one we shall use in the Variational Monte Carlo and we assumed $\phi_k = 0$, $|k| > k_{max} := \bar{k}$ for the UV cutoff. The matrix \mathbf{U} is tridiagonal and symmetric and thus it is simple to find its eigendecomposition.

For completeness, we illustrate the goodness of the approximation introduced for Eq. (2.3) by plotting the difference between its numerical ground-state energy E_0 of Eq. (2.6) and the true ground-state energy of the unconfined HO $E_0^{theo} = 1/2$ as a function of the box size L and the number of momentum states S . Figure 2.1 portrays this difference for several values of S . The curves feature the expected L^{-2} dependence associated with the finite difference approximation,


 FIG 2.1: $|E_0 - E_0^{theo}|$ for $S = 5, 11, 21, 33, 41$ and $L \in [2, 60]$.

though they are truncated due to the ultraviolet cutoff. In fact, for fixed S , as L increases k_{max} decreases and, when it reaches a value for which the momenta do not cover the interval where the first eigenvalue of the matrix in Eq. (2.6) without cutoff is significantly larger than 0, the error starts to increase again. The shape of the curves in Fig. 2.1 is determined by the interplay between this effect and the L^{-2} dependence. The approximation is not particularly efficient; nevertheless, in what follows, to test the VMC implementation, we shall use the Hamiltonian in Eq. (2.3) anyway because of its simplicity. If one intends to include a harmonic term in the Hamiltonian to study in the end, a more accurate approximation of the potential should be chosen (for a brief discussion see Appendix B).

2.2 Implementation of the VMC algorithm in 1D

2.2.1 Metropolis algorithm

Here we briefly summarize the procedure described in Sec. 1.2 by reporting the key equations and we describe the actual implementation of the procedure. We remind that the variational wave function is written as in Eq. (1.6). As in the previous Sections, we use S and M to indicate the number of Fock states and the number of particles, respectively.

Given a Fock state $|N_{curr}\rangle$, at each Metropolis iteration we propose a transition to a new state $|N_{poss}\rangle$. Assuming the transition kernel to be symmetric, the acceptance probability is given by $\min(c_{N_{poss}}^2/c_{N_{curr}}^2, 1)$. Since the expectation value of the energy over an arbitrary state $|\psi\rangle$ can be written as

$$E = \frac{\langle \psi | \hat{H} | \psi \rangle}{\langle \psi | \psi \rangle} = \sum_N \frac{1}{\langle \psi | \psi \rangle} \langle \psi | N \rangle \langle N | \hat{H} | \psi \rangle = \sum_N \frac{|\langle \psi | N \rangle|^2}{\langle \psi | \psi \rangle} \frac{\langle N | \hat{H} | \psi \rangle}{\langle N | \psi \rangle} = \sum_N P(N) E_L(N) \quad (2.7)$$

where $P(N) = c_N^2 / \langle \psi | \psi \rangle$, $\langle \psi | \psi \rangle = \sum_N c_N^2$ and $E_L(N) = \langle N | \hat{H} | \psi \rangle / \langle N | \psi \rangle$ is the local energy,

E can be estimated using the sampled states (distributed according to $P(N)$) as

$$E \simeq \frac{1}{N_s} \sum_{i=1}^{N_s} E_L(N^{[i]}) \pm \sqrt{\frac{1}{N_s - 1} \left(\frac{1}{N_s} \sum_{i=1}^{N_s} E_L(N^{[i]})^2 \right) - \frac{1}{N_s - 1} \left(\frac{1}{N_s} \sum_{i=1}^{N_s} E_L(N^{[i]}) \right)^2} \quad (2.8)$$

where N_s is the number of samples $N^{[i]}$. As discussed in Sec. 1.2, we may actually discard N_{void} samples between each calculation of the local energy and estimate the error in Eq. (2.8) using averages rather than single samples (that is, considering the $E_L(N^{[i]})$ in Eq. (2.8) as averages over a number N_{walk} of configurations). In general, unless explicitly specified, in the next Sections we shall always use suitable N_{void} (~ 200), N_{eq} ($\sim 40 \times N_{\text{void}}$) and N_{walk} (~ 500). Finally, we shall refer to a single acceptance/rejection step as a Monte Carlo (MC) step and to a collection of MC steps as a MC run.

The Fock states can be simply implemented as arrays of zeros and ones. The initial random state N_0 can be generated by creating an array of size S , setting the first M elements to one and the others to zero and then shuffling the array elements randomly. To perform the random walk using a symmetric transition kernel while conserving the number of particles and maintaining $n_\lambda = 0, 1 \forall \lambda$, the proposed new state $|N_{\text{poss}}\rangle$ can be generated by swapping two random occupation numbers in the previous state $|N_{\text{curr}}\rangle$. This update can be implemented via a $N_{\text{pairs}} \times 2$ matrix containing the indices of all possible unordered occupation number pairs, say $\alpha^i, \beta^i = 0, \dots, S - 1$, where $i = 0, \dots, N_{\text{pairs}} - 1$ ¹; for each proposed transition one can then generate a random pair index \bar{i} and swap the corresponding occupation numbers $n_{\alpha^{\bar{i}}}, n_{\beta^{\bar{i}}}$. Notice that, in general, the total number of momentum eigenstates in D dimensions scales as $S = A^D$, where A is the number of momenta along each directions, thus the total number of unordered occupation number pairs that can be formed is $S(S - 1)/2 \sim A^{2D}$. In case the memory of the computing device is limited, an alternative to this implementation consists in finding a mapping between the pair index i (which ranges from 0 to $S(S - 1)/2 - 1$) and the indices constituting the pair α^i, β^i . Below is an example of this mapping for $S = 5$, with the pairs sharing the first index grouped together.

$$\begin{array}{l} i \quad \alpha \quad \beta \\ S - 1 \left[\begin{array}{l} 0 \rightarrow 0, 1 \\ 1 \rightarrow 0, 2 \\ 2 \rightarrow 0, 3 \\ 3 \rightarrow 0, 4 \end{array} \right. \\ S - 2 \left[\begin{array}{l} 4 \rightarrow 1, 2 \\ 5 \rightarrow 1, 3 \\ 6 \rightarrow 1, 4 \end{array} \right. \\ S - 3 \left[\begin{array}{l} 7 \rightarrow 2, 3 \\ 8 \rightarrow 2, 4 \end{array} \right. \\ S - 4 \left[\begin{array}{l} 9 \rightarrow 3, 4 \end{array} \right. \end{array} \quad (2.9)$$

¹Note that here we use zero-based numbering for convenience; this shall in general be done when discussing the practical implementation of part of the algorithm and will be clearly highlighted.

In general, one can derive (see the example in (2.9)) that a given pair index i lies in the interval $\left[\sum_{n=S-\alpha}^{S-1} n, \sum_{n=S-\alpha-1}^{S-1} n \right)$. For this reason, the expression

$$\left(\sum_{n=S-x}^{S-1} n \right) - i = \frac{x}{2} (2S - x - 1) - i \quad (2.10)$$

features a 0 in the interval $[\alpha, \alpha + 1)$ (and one between $2S - 2 - \alpha$ and $2S - 1 - \alpha$), thus α can be obtained as

$$\alpha = \lfloor x \rfloor = \left\lfloor S - \frac{1 + \sqrt{(1 - 2S)^2 - 8i}}{2} \right\rfloor \quad (2.11)$$

Once α is found, the second index of the pair β can be easily calculated by adding $\alpha + 1$ to the position of the index i within a group at constant α , which is given by $i - \sum_{n=S-\alpha}^{S-1} n$:

$$\beta = \alpha + 1 + \left(i - \sum_{n=S-\alpha}^{S-1} n \right) = \alpha + 1 + \left[i - \frac{\alpha}{2} (2S - \alpha - 1) \right] = i + 1 + \frac{\alpha}{2} (\alpha + 3 - 2S) \quad (2.12)$$

This way one simply needs to generate a random number from 0 to $S(S-1)/2 - 1$, calculate α and β using Eq.s (2.11) and (2.12) and then swap n_α and n_β for the proposed state.

2.2.2 One particle case

We now derive the expression of the local energy for Eq. (2.8). For simplicity, we consider the $M = 1$ case first. To do this, we insert a momentum completeness relation (for the finite discrete basis considered) in $\langle N | \hat{H} | \psi \rangle$ to obtain

$$\langle N | \hat{H} | \psi \rangle = \langle N | (\hat{T} + \hat{V}) | \psi \rangle = \sum_k \langle N | k \rangle \langle k | \hat{H} | \psi \rangle \quad (2.13)$$

For a single particle, $|N\rangle = |n_1, \dots, n_S\rangle$ is such that $n_\alpha = 1$ and $n_{\gamma \neq \alpha} = 0$, thus we write $|N\rangle = |K_\alpha\rangle^2$, where K_α indicates the momentum associated with n_α . Note that, in principle, the mapping between the index γ of an occupation number n_γ and the momentum of the associated SPO is arbitrary; however, in the one dimensional case it is reasonable to associate n_0 to $-K_{max}$, n_1 to $-K_{max} + \delta k$ and so on, where once again $K_{max} = \delta k (S-1)/2$ and $\delta k = 2\pi/L$. This allows to avoid a subtle issue that we shall discuss in due time. Since for one particle the momentum representation of the Fock states is simply $\langle k | N \rangle = \langle k | K_\alpha \rangle = \delta(k - K_\alpha)$, from Eq. (2.3):

$$\langle K_\alpha | \hat{H} | \psi \rangle = \langle K_\alpha | \hat{T} | \psi \rangle + \langle K_\alpha | \hat{V} | \psi \rangle = \frac{K_\alpha^2}{2} c_\alpha - \frac{c_{\alpha-1} - 2c_\alpha + c_{\alpha+1}}{2\delta k^2} \quad (2.14)$$

²From here on, we shall always use capital letters for the momenta associated to the SPOs in the Fock states, while generic momentum eigenstates shall be indicated with lowercase letters.

where c_α is the wave function coefficient for the state $|N\rangle = |K_\alpha\rangle$. Finally, going back to Eq. (2.7):

$$\begin{aligned} E_L(N) &= \frac{\langle K_\alpha | \hat{H} | \psi \rangle}{\langle K_\alpha | \psi \rangle} = \frac{1}{c_\alpha} \left(\frac{K_\alpha^2}{2} c_\alpha - \frac{c_{\alpha-1} - 2c_\alpha + c_{\alpha+1}}{2\delta k^2} \right) \\ &= \frac{K_\alpha^2}{2} + \frac{1}{\delta k^2} \left(1 - \frac{c_{\alpha+1}}{2c_\alpha} - \frac{c_{\alpha-1}}{2c_\alpha} \right) \end{aligned} \quad (2.15)$$

The orbitals obtained from the exact diagonalization of the matrix in Eq. (2.6) can be directly used in the expression above and in the Metropolis as described in Sec. 2.2.1 ($c_\alpha = \phi_{K_\alpha}$) to check the correctness of the implementation of the algorithm discussed so far. As an example, in Fig. 2.2 we plot some values of the local kinetic, potential and total energies for a MC run with $S = 21$ and $L = 10$.

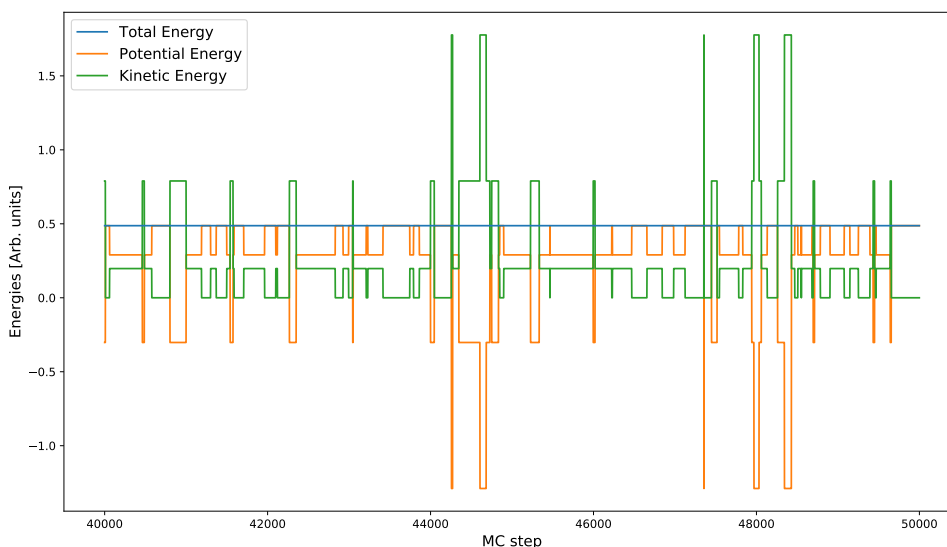


FIG 2.2: Local kinetic, potential and total energies for 10^4 out of 10^6 steps of an MC run with $S = 21$ and $L = 10$ and wave function coefficients $c_N = c_\alpha$ calculated via exact diagonalization.

As the figure shows, the local energy $E_L(N)$ is exactly constant, and the resulting average is equal to the FD one up to machine precision, see Tab. I. This confirms the correctness of the implementation of the MC algorithm: in fact, the zero variance principle states that, theoretically, the statistical error of the expectation of the total energy $\langle \psi | \hat{H} | \psi \rangle / \langle \psi | \psi \rangle$ vanishes when the wave function $|\psi\rangle$ is an actual eigenstate of \hat{H} ; for this reason, the error is entirely affected by machine precision (in the example discussed here, the error obtained was `nan`).

However, it is important to highlight the fact that the average kinetic and potential energies are not compatible with the results of the FD approach (Tab. I), which can be calculated from the ground-state eigenvector ϕ as $\langle \phi | \hat{T} | \phi \rangle / \langle \phi | \phi \rangle$ and $\langle \phi | \hat{V} | \phi \rangle / \langle \phi | \phi \rangle$ by separating these two contributions to the matrix in Eq. (2.6). This is due to the effect of autocorrelations, discussed in Sec. 1.2: since the number of independent samples is actually smaller than the number of MC steps, the error is underestimated. The autocorrelation coefficient r_k for the kinetic energy calculated in the MC run above is illustrated in Figure 2.3.

As the left plot of Fig. (2.3) shows, r_k drops to about 0 around $k = 250$ and then oscillates around it due to the finite number of samples. Instead, the plot on the right represents the same

	MC results (no N_{void})	MC results (N_{void})	FD results
Kinetic Energy	0.23188 ± 0.00033	0.23747 ± 0.00034	0.23715
Potential Energy	0.25545 ± 0.00033	0.24986 ± 0.00034	0.25018
Total Energy	0.487333	0.487333	0.487333

Table I: Average values with estimated errors of the energies obtained from a MC run not accounting for autocorrelations and accounting for autocorrelations compared with the results of the finite difference approach. The total energy is reported with an arbitrary number of significant digits and with no error as this is only determined by machine precision (which is of order 10^{-15} in this case, thus the error may be either `nan`, 0 or of order $\sim 10^{-10}$).

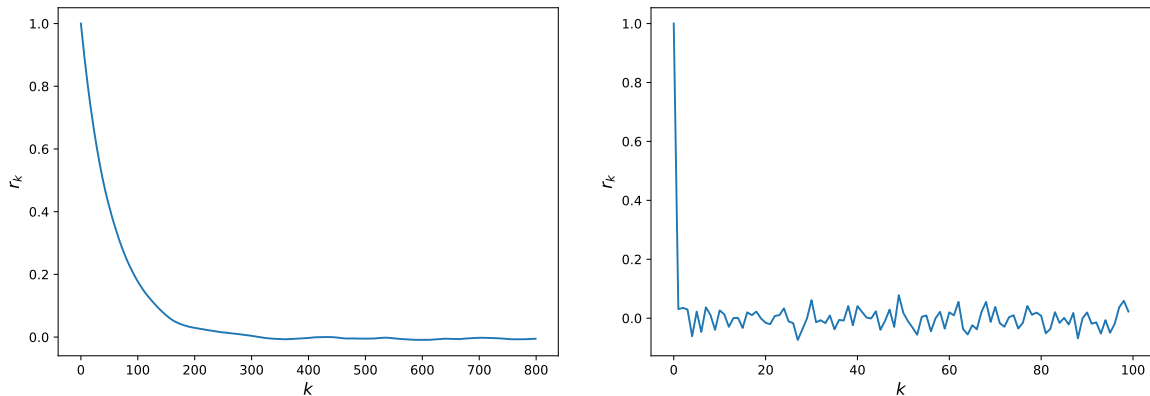


FIG 2.3: Kinetic energy autocorrelation coefficients for a MC run of 10^6 samples with $S = 21$ and $L = 10$. *Left*: $N_{\text{void}} = 1$, $N_{\text{walk}} = 1$ (samples taken every MC step). *Right*: $N_{\text{void}} = 200$, $N_{\text{walk}} = 1000$. Note the different scales of the x axis.

coefficient calculated for a new run using $N_{\text{void}} = 200$, $N_{\text{walk}} = 1000$ and $N_{\text{av}} = 1000$ (to maintain the total number of samples 10^6), in which case r_k drops to ~ 0 immediately. Contrarily to the MC run described above, in this case the average kinetic and potential energies calculated are compatible with the FD results within one standard deviation, as Table I shows.

Note that in both cases the total energy was not affected by autocorrelations. However, this is simply due to the fact that we used the exact c_α ; in the general variational procedure or for expectations of other operators rather than the Hamiltonian, accounting for autocorrelations is important in order not to underestimate the errors.

2.2.3 Arbitrary number of particles

Let us now consider the case of an arbitrary number of particles M , in which case the Fock states are $|n_1, \dots, n_S\rangle$, $n_i = 0, 1$ ($S \geq M$) with the constraint $\sum_i n_i = M$. The antisymmetry of these states is explicit once we project them onto an M -particles momentum eigenstate:

$$\langle k_1, \dots, k_M | n_1, \dots, n_S \rangle = \frac{1}{\sqrt{M!}} \begin{vmatrix} \delta(k_1 - K_{\alpha_1}) & \delta(k_2 - K_{\alpha_1}) & \cdots & \delta(k_M - K_{\alpha_1}) \\ \delta(k_1 - K_{\alpha_2}) & \delta(k_2 - K_{\alpha_2}) & \cdots & \delta(k_M - K_{\alpha_2}) \\ \vdots & \vdots & \ddots & \vdots \\ \delta(k_1 - K_{\alpha_M}) & \delta(k_2 - K_{\alpha_M}) & \cdots & \delta(k_M - K_{\alpha_M}) \end{vmatrix} \quad (2.16)$$

where $K_{\alpha_1}, K_{\alpha_2}, \dots, K_{\alpha_M}$ are the M occupied momenta for which $n_{\alpha_1} = n_{\alpha_2} = \dots = n_{\alpha_M} = 1$ with $\alpha_1 < \alpha_2 < \dots < \alpha_M$. Similarly to the previous Section, we indicate the Fock states as $|N\rangle = |K_{\alpha_1}, \dots, K_{\alpha_M}\rangle$.

The exact wave function coefficients $c_N = \langle N | \psi \rangle$ can be written as

$$\begin{aligned} c_N &= \sum_{k_1, \dots, k_M} \langle N | k_1, \dots, k_M \rangle \langle k_1, \dots, k_M | \psi \rangle \\ &= \sum_P \frac{(-)^P}{\sqrt{M!}} P \sum_{k_1, \dots, k_M} \delta(k_1 - K_{\alpha_1}) \dots \delta(k_M - K_{\alpha_M}) \langle k_1, \dots, k_M | \psi \rangle \\ &= \sum_P \frac{(-)^P}{\sqrt{M!}} P \psi(K_{\alpha_1}, K_{\alpha_2}, \dots, K_{\alpha_M}) \\ &= \sum_P \frac{(-)^P}{\sqrt{M!}} (-)^P \psi(K_{\alpha_1}, K_{\alpha_2}, \dots, K_{\alpha_M}) \\ &= \sqrt{M!} \psi(K_{\alpha_1}, K_{\alpha_2}, \dots, K_{\alpha_M}) \end{aligned} \quad (2.17)$$

where we used Eq. (2.16), P indicates the permutation operator of the momenta $K_{\alpha_1}, \dots, K_{\alpha_M}$ and $(-)^P$ its sign and we used the antisymmetry of the wave function to write $P\psi = (-)^P\psi$. For example, for $M = 2$, one can easily see that the coefficients are

$$\langle 0, \dots, 0, 1_\alpha, 0, \dots, 0, 1_\beta, 0, \dots | \psi \rangle = \phi_0(K_\alpha)\phi_1(K_\beta) - \phi_0(K_\beta)\phi_1(K_\alpha) \quad (2.18)$$

where $\phi_0(k)$ and $\phi_1(k)$ are the first and second eigenstates of the Hamiltonian in Eq. (2.3) for $M = D = 1$. It is important to highlight the fact that the exact c_N are antisymmetric in the momenta $K_{\alpha_1}, \dots, K_{\alpha_M}$; for this reason, the way the occupation numbers in a Fock state are associated with the momentum eigenstates is very important in the determination of the overall sign of these coefficients. Note that, for instance, if we were to associate the first 1 in the state in Eq. (2.18) to K_β and the second one to K_α , the expression on the RHS would have an overall minus sign. This point will be of great importance in multiple dimensions.

The generalization of Eq. (2.15) to a arbitrary M is rather straightforward. We start from

$$\langle N | \hat{H} | \psi \rangle = \sum_{k_1, \dots, k_M} \langle N | k_1, \dots, k_M \rangle \langle k_1, \dots, k_M | \hat{H} | \psi \rangle \quad (2.19)$$

where $|N\rangle = |K_{\alpha_1}, \dots, K_{\alpha_M}\rangle$ and the last matrix element $\langle k_1, \dots, k_M | \hat{H} | \psi \rangle$ can be derived from Eq. (2.3). For the kinetic energy we have

$$\begin{aligned}
 \langle N | \hat{T} | \psi \rangle &= \sum_{k_1, \dots, k_M} \langle K_{\alpha_1}, \dots, K_{\alpha_M} | k_1, \dots, k_M \rangle \langle k_1, \dots, k_M | \hat{T} | \psi \rangle \\
 &= \sum_P \frac{(-)^P}{\sqrt{M!}} P \left(\frac{K_{\alpha_1}^2}{2} + \dots + \frac{K_{\alpha_M}^2}{2} \right) \psi(K_{\alpha_1}, \dots, K_{\alpha_M}) \\
 &= \left(\frac{K_{\alpha_1}^2}{2} + \dots + \frac{K_{\alpha_M}^2}{2} \right) \sum_P \frac{(-)^P}{\sqrt{M!}} P \psi(K_{\alpha_1}, \dots, K_{\alpha_M}) \\
 &= \left(\frac{K_{\alpha_1}^2}{2} + \dots + \frac{K_{\alpha_M}^2}{2} \right) \langle N | \psi \rangle = \left(\frac{K_{\alpha_1}^2}{2} + \dots + \frac{K_{\alpha_M}^2}{2} \right) c_N
 \end{aligned} \tag{2.20}$$

since the sum in the round brackets remains unaltered under any permutation of the K_{α_i} and the remaining sum over the permutations is by definition the coefficient c_N (see Eq. (2.17)). The kinetic contribution to the local energy is then

$$\frac{\langle N | \hat{T} | \psi \rangle}{\langle N | \psi \rangle} = \frac{K_{\alpha_1}^2}{2} + \dots + \frac{K_{\alpha_M}^2}{2} \tag{2.21}$$

If one chooses the mapping between the the occupation numbers and the momentum eigenstates mentioned in the previous Section $n_i \rightarrow 2\pi(i - (S - 1)/2)/L$, $i = 0, \dots, S - 1$, this term can be easily computed by finding the indices of the non-zero elements of the array which defines $|N\rangle$, say α_i with $i = 0, \dots, M - 1$, and computing

$$\frac{\langle N | \hat{T} | \psi \rangle}{\langle N | \psi \rangle} = \frac{2\pi^2}{L^2} \sum_{i=0}^{M-1} \left(\alpha_i - \frac{S-1}{2} \right)^2 \tag{2.22}$$

The potential term is slightly more cumbersome:

$$\begin{aligned}
 \langle N | \hat{V} | \psi \rangle &= \sum_{k_1, \dots, k_M} \langle K_{\alpha_1}, \dots, K_{\alpha_M} | k_1, \dots, k_M \rangle \langle k_1, \dots, k_M | V | \psi \rangle \\
 &= \sum_{k_1, \dots, k_M} \langle K_{\alpha_1}, \dots, K_{\alpha_M} | k_1, \dots, k_M \rangle \frac{1}{\delta k^2} \sum_{i=1}^M \left(\psi(\mathbf{k}) - \frac{\psi(\mathbf{k}, k_i + \delta k)}{2} - \frac{\psi(\mathbf{k}, k_i - \delta k)}{2} \right) \\
 &= \frac{1}{\delta k^2} \sum_P \frac{(-)^P}{\sqrt{M!}} P \sum_{i=1}^M \left(\psi(\mathbf{K}) - \frac{\psi(\mathbf{K}, K_{\alpha_i} + \delta k)}{2} - \frac{\psi(\mathbf{K}, K_{\alpha_i} - \delta k)}{2} \right) \\
 &= \frac{1}{\delta k^2} \sum_{i=1}^M \left(c_N - \frac{c_{N, \alpha_i+1}}{2} - \frac{c_{N, \alpha_i-1}}{2} \right) = \frac{M}{\delta k^2} c_N - \frac{1}{2\delta k^2} \sum_{i=1}^M (c_{N, \alpha_i+1} + c_{N, \alpha_i-1})
 \end{aligned} \tag{2.23}$$

where $\psi(\mathbf{k}, k_i \pm \delta k)$ indicates the total wave function calculated at k_1, \dots, k_M though with $k_i \pm \delta k$ and similarly for $\psi(\mathbf{K}, K_{\alpha_i} \pm \delta k)$, while, with a slight abuse of notation, with $c_{N, \alpha_i \pm 1}$ we indicate the coefficient c_N , though calculated at $K_{\alpha_i} \pm \delta k$. It is important to notice that, due to the antisymmetry of the wave function, the $c_{N, \alpha_i \pm 1}$ are null if the momentum $K_{\alpha_i} \pm \delta k$ is already

occupied in $|N\rangle$. This is automatically verified if one uses the exact wave function as in Eq. (2.18); however, for a generic expression of the c_N (for instance the one we shall use for the full VMC procedure) it may not be the case, therefore this condition has to be imposed in the calculations, otherwise the ground-state energy may be wrong. Finally, the potential contribution to the local energy is

$$\frac{\langle N | \hat{V} | \psi \rangle}{\langle N | \psi \rangle} = \frac{M}{\delta k^2} - \frac{1}{2\delta k^2} \sum_{i=1}^M \left(\frac{c_{N,\alpha_i+1}}{c_N} + \frac{c_{N,\alpha_i-1}}{c_N} \right) \quad (2.24)$$

The calculation of the terms in the sum can be implemented by defining again the array α_i , $i = 0, \dots, M-1$ of non-zero indices in $|N\rangle$, then looping over i and calculating the wave function coefficients for new states $|N'\rangle = |N\rangle$ with $n'_{\alpha_i} = 0$ and $n'_{\alpha_i \pm 1} = n_{\alpha_i \pm 1} + 1$ (again assuming the ordering of the momenta defined above). Computing the new occupation numbers this way also allows to implement the antisymmetry condition discussed above by multiplying each term in the sum by a factor $\theta(2 - n'_{\alpha_i \pm 1})$, where $\theta(n)$ is the Heaviside step function. This way, if the SPO associated with $K_{\alpha_i} \pm \delta k$ is already occupied then $n'_{\alpha_i \pm 1} = n_{\alpha_i \pm 1} + 1 = 2$ and the Heaviside factor sets the term to zero. In order to test the expressions of the local energies derived above, one can once again exploit the exact wave function coefficients calculated using the results of the FD approach in Sec. 2.1 and Eq. (2.17). Practically, the c_N are simply computed as Slater determinants with the first M orbitals $\phi(k)$ obtained through the diagonalization of the matrix in Eq. (2.6) calculated at $K_{\alpha_0}, K_{\alpha_1}, \dots, K_{\alpha_{M-1}}$, ignoring any irrelevant overall normalization factors.

We tested the algorithm for many different combinations of S , L and M and in all cases the total energy was correct and the kinetic and potential energies were compatible with the FD results within 2σ . Table II reports 4 examples. The FD total energies are calculated by summing those of the first M orbitals $\phi(k)$.

		MC result	FD result
$M = 2, S = 21, L = 10$	Kinetic Energy	0.9213 ± 0.0022	0.9205
	Potential Energy	1.0012 ± 0.0022	1.0020
	Total Energy	1.922496	1.922496
$M = 4, S = 11, L = 5$	Kinetic Energy	4.8073 ± 0.0014	4.8071
	Potential Energy	2.3947 ± 0.0014	2.3950
	Total Energy	7.202060	7.202060
$M = 6, S = 33, L = 15$	Kinetic Energy	8.1212 ± 0.0063	8.1250
	Potential Energy	9.0315 ± 0.0063	9.0276
	Total Energy	17.152680	17.152680
$M = 10, S = 15, L = 8$	Kinetic Energy	26.6522 ± 0.0036	26.6473
	Potential Energy	15.3446 ± 0.0036	15.3495
	Total Energy	41.996862	41.996862

Table II: Average values with estimated errors of the energies obtained from various MC runs of 10^5 steps compared with the results of the FD approach. The total energy is once again reported with an arbitrary number of significant digits and with no error as this is only determined by machine precision. In all cases we used $N_{\text{void}} = 200$ and $N_{\text{walk}} = 1000$.

2.2.4 Full Configuration Interaction procedure

As an additional check, it may be useful to also implement the full CI algorithm, which practically consists in diagonalizing the Hamiltonian in the Fock basis $|N\rangle$, see Sec. 1.2. As already mentioned in Sec. 1.1, the number of Fock states is given by $N_F = S!/(M!(S-M)!)$ and thus it scales factorially with the number of particles. For this reason, a numerical diagonalization quickly becomes unfeasible as M increases; however, being an exact method, it provides a reliable benchmark to validate the VMC algorithm. As discussed in the beginning of this Chapter, this step is not strictly necessary in the case of the Hamiltonian of Eq. (2.3), though it can be very useful when a two-body potential is present, thereby we include it for completeness. The matrix elements $\langle N|H|N'\rangle$ can be easily obtained from Eq.s (2.20) and (2.23) by introducing an identity to the right of the Hamiltonian operator:

$$\begin{aligned}\langle N|\hat{H}|\psi\rangle &= \sum_{N'} \langle N|\hat{H}|N'\rangle \langle N'|\psi\rangle \\ &= \sum_{N'} \langle N|\hat{T}|N'\rangle c_{N'} + \sum_{N'} \langle N|\hat{V}|N'\rangle c_{N'}\end{aligned}\quad (2.25)$$

and so one can easily see that

$$\begin{aligned}\langle N|\hat{T}|N'\rangle &= \delta_{N,N'} \sum_{\alpha_i} \frac{K_{\alpha_i}^2}{2} \\ \langle N|\hat{V}|N'\rangle &= \begin{cases} \frac{M}{\delta k^2} & \text{if } N = N' \\ -\frac{1}{2\delta k^2} & \text{if } n'_\beta = 1 \mid K_\beta = K_\alpha \pm \delta k \wedge n'_{\gamma \neq \alpha} = n_{\gamma \neq \alpha} \\ 0 & \text{otherwise} \end{cases}\end{aligned}\quad (2.26)$$

where $|N\rangle = |K_{\alpha_1}, \dots, K_{\alpha_M}\rangle$ as customary. Note that, in order to implement a diagonalization on this basis, an ordering for the Fock states needs to be defined. A rather convenient one may be given iteratively in terms of the indices of the occupied momenta $\alpha_i = 0, \dots, S-1$ with $i = 0, \dots, M-1$ in a given state $|N\rangle$. Essentially, the ordering is constructed by always incrementing α_{M-1} and carrying a +1 to α_{i-1} each time α_i reaches $S-M+i$, in which case the $\alpha_{j \geq i}$ are reset to $\alpha_{j-1} + 1$. An example for $S = 6$, $M = 4$ is given below.

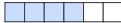
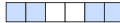
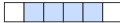












0: (0, 1, 2, 3)	5: (0, 1, 4, 5)	10: (1, 2, 3, 4)			
1: (0, 1, 2, 4)	6: (0, 2, 3, 4)	11: (1, 2, 3, 5)			
2: (0, 1, 2, 5)	7: (0, 2, 3, 5)	12: (1, 2, 4, 5)			
3: (0, 1, 3, 4)	8: (0, 2, 4, 5)	13: (1, 3, 4, 5)			
4: (0, 1, 3, 5)	9: (0, 3, 4, 5)	14: (2, 3, 4, 5)			

Table III: Example of the Fock states ordering we shall use in the $D = 1$ case, for $M = 4$ particles and $S = 6$ momentum states. The numbers in the round brackets represent the indices of the occupied momenta $\alpha_i = 0, \dots, S-1$, $i = 0, \dots, M-1$. The image on the right is a visualization of this ordering: each stack essentially represents one vector describing $|N\rangle$ and the filled and empty squares represent the occupied and unoccupied SPOs.

For the practical implementation of the calculation of the matrix elements in Eq. (2.26) it may be useful to define a $N_F \times M$ matrix $\mathbf{\Gamma}$ containing the indices of the occupied momenta for each Fock state according to the ordering defined above. This could be done straightforwardly through M nested loops; however, to make the diagonalization valid for a generic M , one could either use a recursive algorithm or the following procedure, which uses only two nested loops:

1. Define $\mathbf{\Gamma}$ as a $N_F \times (M + 2)$ matrix;
2. Set $\Gamma_{0,j} = (0, 0, 1, 2, \dots, M - 1, S)$;
3. For $i = 1$ to N_F do:
 - Set $\Gamma_{i,0} = 0$ and $\Gamma_{i,M+1} = S$;
 - For $j = 1$ to M do:
 - Define $c = \lfloor \Gamma_{i-1,j+1} / (S - M + j) \rfloor$;
 - Set $\Gamma_{i,j} = \max[(\Gamma_{i-1,j} + c) \bmod (S - M + j), c(\Gamma_{i,j-1} + 1)]$;
4. Delete the columns 0 and $M + 1$ of $\mathbf{\Gamma}$.

Given a state i and an occupied momentum j , the parameter c checks whether the next occupied momentum in the previous Fock state $\Gamma_{i-1,j+1}$ has reached its maximum index $(S - M + j)$, in which case, as shown in the example above, $\Gamma_{i,j} = \Gamma_{i-1,j} + 1$. This increase happens in the last line of the inner loop; the max ensures that, if $\Gamma_{i-1,j} + 1$ exceeds the maximum index $(S - M + j - 1)$ for occupied momentum j , $\Gamma_{i,j}$ is reset to $\Gamma_{i,j-1} + 1$. The first and last columns of $\mathbf{\Gamma}$ are purely functional and they are deleted at the end.

We tested this diagonalization for different L , S and M . In all cases the kinetic, potential and total energies were consistent with the ones obtained using the FD approach up to machine precision. Note that, since the matrix elements $\langle N | \hat{T} | N' \rangle$ and $\langle N | \hat{V} | N' \rangle$ are known, the kinetic and potential energies can be calculated straightforwardly from the c_N obtained using the diagonalization.

2.2.5 Variational algorithm

As discussed in Sec. 1.2, in a generic variational approach we propose an expression for a variational wave function $|\psi_V(\boldsymbol{\lambda})\rangle$ in terms of some parameters λ_p , then the parameters are optimized in order to minimize the variational energy

$$E_V = \frac{\langle \psi_V | \hat{H} | \psi_V \rangle}{\langle \psi_V | \psi_V \rangle} \quad (2.27)$$

The minimization is performed using stochastic methods and, in general, the gradient of the variational energy is required; in a Monte Carlo context, this can be expressed in the following covariance form (see Sec. 1.2)

$$G_p \simeq 2 \left(\frac{1}{N_s} \sum_{i=1}^{N_s} \mathcal{O}_p^*(N^{[i]}) E_L(N^{[i]}) \right) - 2 \left(\frac{1}{N_s} \sum_{i=1}^{N_s} E_L(N^{[i]}) \right) \left(\frac{1}{N_s} \sum_{i=1}^{N_s} \mathcal{O}_p^*(N^{[i]}) \right) \quad (2.28)$$

For this specific type of problem, the SGD algorithm briefly mentioned in Sec. 1.2 (see Eq. (1.24)) may be badly suboptimal. In our full-scale calculations, we shall use the so-called Stochastic Reconfiguration (SR) algorithm, which is very well suited for QMC problems, although for now we simply implement the RMSprop algorithm, which adjusts the learning rate based on the steepness of the gradient along each direction as

$$\begin{aligned} r_p &\rightarrow \rho r_p + (1 - \rho) G_p^2 \\ \lambda_p &\rightarrow \lambda_p - \frac{\eta}{\sqrt{r_p}} G_p \end{aligned} \quad (2.29)$$

Purely for the sake of testing the optimization of the parameters, a first full VMC procedure can be performed using the wave function

$$|\psi_V\rangle = \sum_N \lambda_N |N\rangle \quad (2.30)$$

that is, simply using the wave function coefficients as variational parameters. As mentioned above, the number of these coefficients scales factorially with the number of particles, thus in general this parametrization is not convenient.

With this parametrization, Eq.s (2.21) and (2.24) can be directly used to compute the local energy. However, for the optimization one generally stores the λ_N in an array $\boldsymbol{\lambda}$, assuming some ordering of the states $|N\rangle$ (such as the one defined in Sec. 2.2.4, which we shall use here); since the Metropolis algorithm produces samples $|N\rangle = |n_0, \dots, n_{S-1}\rangle$ and requires the calculation of the associated wave function coefficients λ_N , it is necessary to find a way to recover the index k_N of λ_N in the array $\boldsymbol{\lambda}$ from the occupation numbers n_i , or rather from the occupied momentum indices $\boldsymbol{\alpha} = (\alpha_0, \dots, \alpha_{M-1})$ which are such that $n_{\alpha_0} = \dots = n_{\alpha_{M-1}} = 1$, $\alpha_i \in [0, S-1]$. To find this mapping, we can notice that, given some i and the associated α_i , the number of possible configurations $C_i^{\alpha_i}$ for the remaining $\alpha_{j>i}$ is given by the possible ways in which one can arrange the available momentum indices $\alpha_i + 1, \dots, S-1$ into the remaining $M-1-i$ indices $\alpha_{j>i}$ in an unordered way (see for instance Tab. III), that is

$$C_i^{\alpha_i} = \binom{S-1-\alpha_i}{M-1-i} \quad (2.31)$$

Therefore, we may think of finding the mapping $\boldsymbol{\alpha} \rightarrow k_N$ by simply counting the number of arrangements of the occupied momentum indices $\boldsymbol{\beta}$ before the one of state $|N\rangle$. To do so, we can essentially loop over i and count the number of arrangements $\boldsymbol{\beta}$ having $\beta_j = \alpha_j$ for $j < i$ and $\alpha_{i-1} = \beta_{i-1} < \beta_i < \alpha_i$ (since there are no arrangements with $\beta_{i-1} \geq \beta_i$) with $\alpha_{-1} = -1$ and then sum these counts. In formula:

$$k_N = \sum_{i=0}^{M-1} \sum_{\gamma=\alpha_{i-1}+1}^{\alpha_i-1} C_i^{\gamma} = \sum_{i=0}^{M-1} \sum_{\gamma=\alpha_{i-1}+1}^{\alpha_i-1} \binom{S-1-\gamma}{M-1-i} \quad (2.32)$$

For a concrete interpretation, for $i=0$ the inner sum counts the number of arrangements of kind $(0 \leq \beta_0 < \alpha_0, \beta_1, \beta_2, \dots)$, for $i=1$ those of kind $(\alpha_0, \alpha_0 < \beta_1 < \alpha_1, \beta_2, \dots)$ and so on. Recalling the example given in Tab. III one can convince himself that this expression is indeed correct.

Now, the double sum is not actually necessary: in fact, by defining $z = S - 1 - \gamma$ we can write

$$\begin{aligned}
 k_N &= \sum_{i=0}^{M-1} \sum_{z=S-\alpha_i}^{S-\alpha_{i-1}-2} \binom{z}{M-1-i} \\
 &= \sum_{i=0}^{M-1} \left[\sum_{z=0}^{S-\alpha_{i-1}-2} \binom{z}{M-1-i} - \sum_{z=0}^{S-\alpha_i-1} \binom{z}{M-1-i} \right] \\
 &= \sum_{i=0}^{M-1} \left[\binom{S-\alpha_{i-1}-1}{M-i} - \binom{S-\alpha_i}{M-i} \right]
 \end{aligned} \tag{2.33}$$

where we used

$$\sum_{i=0}^n \binom{i}{m} = \binom{n+1}{m+1} \tag{2.34}$$

To give an example, the α associated with the last state $|N\rangle$ in the ordering defined in the previous Section is always such that $\alpha_i = \alpha_{i-1} + 1$, $i > 0$ and $\alpha_0 = S - M$, thus in the sum above only the term $i = 0$ remains and the result is

$$k_N = \binom{S - (-1) - 1}{M} - \binom{S - (S - M)}{M} = \binom{S}{M} - 1 \tag{2.35}$$

which is the correct index for the last Fock state. Notice that, to be consistent with the FD results, when any $\alpha_i = 0$ or $\alpha_i = S - 1$, the corresponding λ_{N,α_i-1} or λ_{N,α_i+1} in the calculation of the potential energy (c_{N,α_i-1} or c_{N,α_i+1} in Eq. (2.24)) need to be set to 0 respectively.

As far as the calculation of the gradient in Eq. (2.28) is concerned, the derivation of a theoretical expression for $\mathcal{O}_p(N^{[i]})$ is straightforward for a variational wave function as in Eq. (2.30); however, since we plan on using NQS, it would be much simpler to leave the task of computing the derivative of the variational wave function with respect to the variational parameters to the program itself, which can be conveniently done using the `grad()` function from the Python library we use (see Appendix C). To make the code slightly more efficient, for each MC run after the first one we recycle the last N_{walk} Fock states from the previous run and use $N_{\text{eq}} \sim 10 \times N_{\text{void}}$.

Figure 2.4 features the behaviour of the variational energy as a function of the minimization step and the final shape of the variational coefficients for a full VMC procedure of 200 minimization steps with $M = 1$, $S = 15$ and $L = 10$. The final energy, along with other values obtained for different configurations, is reported in Table IV.

The results show that, in general, the larger the number of coefficients λ_N , the worse the convergence. This may be due to the fact that the variational parameters determine whether a state is sampled or not: if, for any reason, one λ_N is updated incorrectly to a value much smaller than the other ones during the optimization, the corresponding state $|N\rangle$ in the Metropolis is always rejected and λ_N is never updated again. This problem is more likely to occur the larger the number of coefficients λ_N , possibly explaining the behaviour of the results. Additionally, the RMSprop algorithm as well as the choice of its parameters may not be optimal for this problem. However, this is not an issue, as we only use this expression of the variational wave function (Eq. (2.30)) in order to test the correctness of the implementation of the variational algorithm.

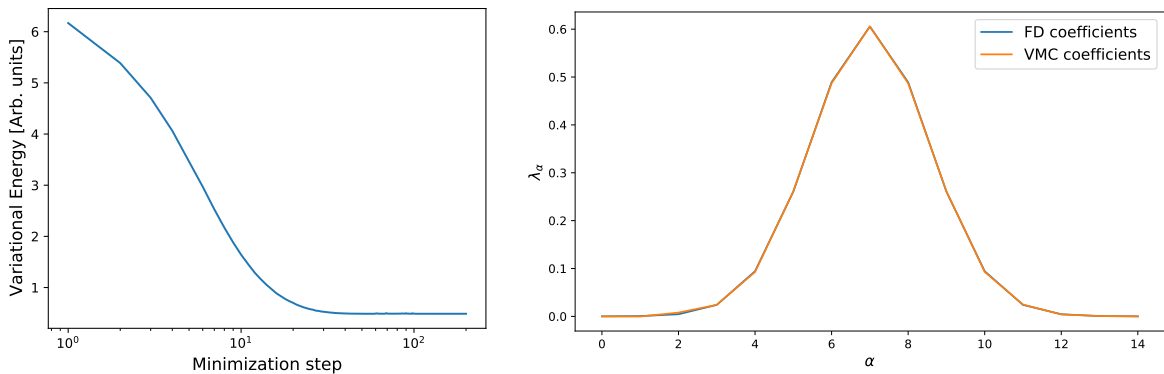


FIG 2.4: Results of a full VMC procedure with $M = 1$, $S = 15$ and $L = 10$. At each minimization step we ran 10000 walkers for 20 MC steps with $N_{void} = 200$. For the RMSprop algorithm we used $\rho = 0.9$ and $\eta = 0.05$ for the first 100 steps and $\eta = 0.02$ for the remaining 100. *Left*: Behaviour of the variational energy as a function of the minimization step. *Right*: Final coefficients compared with the exact ones calculated using the FD orbitals. The two curves overlap nearly perfectly. Since $M = 1$, we indicate the wave function coefficients using the state index α ($|N\rangle = |K_\alpha\rangle$).

Parameters	N_F	VMC result	FD result	Difference
$M = 1, S = 15, L = 10$	15	$0.487404 \pm 5.0 \times 10^{-5}$	0.487333	$(7.1 \pm 5.0) \times 10^{-5}$
$M = 6, S = 9, L = 9$	84	$15.469760 \pm 8.3 \times 10^{-5}$	15.468900	$8.60 \times 10^{-4} \pm 8.3 \times 10^{-5}$
$M = 12, S = 15, L = 13$	455	$63.26760 \pm 1.3 \times 10^{-4}$	63.25627	$1.143 \times 10^{-2} \pm 1.3 \times 10^{-4}$
$M = 4, S = 17, L = 12$	2380	$7.77546 \pm 1.6 \times 10^{-4}$	7.59939	$0.17607 \pm 1.6 \times 10^{-4}$

Table IV: Final energies obtained using the VMC procedure compared with the FD results for different combinations of M , S and L . We also indicate the number of Fock states N_F . Excluding the first line, all results were obtained with more than 1500 minimization steps, each of 2×10^5 samples in total with $N_{void} = 200$. For each optimization we decreased η from $0.05 \div 0.01$ to a final value of 0.005 for the last ~ 500 steps in order to reduce fluctuations. Note that, for each combination of parameters, we run multiple optimizations, always obtaining results larger or similar to the ones reported here. The results presented are obtained from a final VMC calculation with the optimized parameters.

We figure the results show that the code functions correctly.

2.3 Full VMC procedure for arbitrary D and M

Having discussed the problem step-by-step, the general case of arbitrary D and M is now rather simple to treat. The expression of the kinetic energy contribution remains the one in Eq. (2.21), while the potential energy formula in Eq. (2.24) continues to be valid with the addition of a sum over the dimensions (and so, for instance, the first term becomes $MD/\delta k^2$). However, now each n_α in a Fock state $|N\rangle$ refers to a vector $\mathbf{K}_\alpha = (K_\alpha^0, \dots, K_\alpha^{D-1})$ of occupied components of the momentum with $K_\alpha^d \in [-K_{max}, -K_{max} + \delta k, \dots, K_{max}]$, where $K_{max} = (A - 1)\delta k/2$ and A is the number of momenta along each direction. It is therefore not straightforward to define a relation between the n_α , $\alpha = 0, \dots, S - 1$ and the possible \mathbf{K}_α . One way to do this consists in starting from $\mathbf{K}_0 = (-K_{max}, \dots, -K_{max})$, increasing K^0 by δk and, each time a K^d exceeds K_{max} , resetting it to $-K_{max}$ and increasing K^{d+1} by δk . Note that this is similar to the ordering

of the Fock states discussed above in the context the exact diagonalization, though in this case there is no constraint between the components of the momenta and, for this reason, the total number of Fock states is A^D . A simple example for $D = 2$, $A = 3$ ($K^d = -\delta k, 0, \delta k$) is shown below.

$$\begin{aligned}
 n_0 &\rightarrow (-\delta k, -\delta k) & n_3 &\rightarrow (-\delta k, 0) & n_6 &\rightarrow (-\delta k, \delta k) \\
 n_1 &\rightarrow (0, -\delta k) & n_4 &\rightarrow (0, 0) & n_7 &\rightarrow (0, \delta k) \\
 n_2 &\rightarrow (\delta k, -\delta k) & n_5 &\rightarrow (\delta k, 0) & n_8 &\rightarrow (\delta k, \delta k)
 \end{aligned} \tag{2.36}$$

This ordering allows to find a simple relation between the index α of the occupation number n_α and the momentum along each direction. In particular, one can see that

$$K_\alpha^d = \left[I_\alpha^d - \frac{A-1}{2} \right] \delta k = \left[\left(\left\lfloor \frac{\alpha}{A^d} \right\rfloor \bmod A \right) - \frac{A-1}{2} \right] \delta k \tag{2.37}$$

where $d = 0, \dots, D-1$.

Using this relation, one can easily pre-compute the $S = A^D$ norms of each available momentum \mathbf{K}_α ; then the local kinetic energy can be calculated straightforwardly by summing the norms of the occupied momenta in the current state N (and dividing by two).

For the potential energy, if an occupation number n_α is 1, one needs to calculate, for each direction \bar{d} , the wave function coefficients for a new state with $n_\alpha = 0$ and $n_\beta = 1$ such that $K_\beta^{\bar{d}} = K_\alpha^{\bar{d}} \pm \delta k$ and $K_\beta^d = K_\alpha^d$ for $d \neq \bar{d}$. With the ordering described above, given the index α , it is straightforward to find the index β as $\alpha \pm A^{\bar{d}}$. This is actually valid in general, that is, given \mathbf{K}_α , it is possible to find the index of $\mathbf{K}_\beta = \mathbf{K}_\alpha + (q_0, q_1, \dots, q_D)\delta k$ as

$$\beta = \alpha + \sum_d q_d A^d \tag{2.38}$$

provided that \mathbf{K}_β is an available momentum (that is, its components do not exceed K_{max} in modulus). Of course, in the calculation of the potential energy one also needs to pay attention to the extreme cases ($K_\alpha^{\bar{d}} = \pm K_{max}$), where

$$\begin{aligned}
 I_\alpha^{\bar{d}} &= 0 & \text{for } K_\alpha^{\bar{d}} &= -K_{max}, \text{ or} \\
 I_\alpha^{\bar{d}} &= A-1 & \text{for } K_\alpha^{\bar{d}} &= K_{max}
 \end{aligned} \tag{2.39}$$

In both cases, one would need to compute a new wave function coefficient for a state with one of the momenta having a component $|K_{max} + \delta k|$ (see Eq. (2.23) and the discussion below), which is assumed to be null due to the UV cutoff and therefore the corresponding term in the potential energy needs to be set to 0.

As briefly introduced above, in the case $D > 1$ the calculation of the potential energy presents a rather subtle problem. For the sake of clarity, we shall discuss this point making use of the fermion creation \hat{c}_α^\dagger and annihilation \hat{c}_α operators, which create and annihilate a particle in the α SPO in a Fock state, and satisfy

$$\left\{ \hat{c}_\alpha^\dagger, \hat{c}_\beta^\dagger \right\} = \left\{ \hat{c}_\alpha, \hat{c}_\beta \right\} = 0 \quad , \quad \left\{ \hat{c}_\alpha, \hat{c}_\beta^\dagger \right\} = \delta_{\alpha\beta} \tag{2.40}$$

Since we are dealing with fermions, it is important to keep track of phase terms which are accumulated at each application of these operators. In fact, swapping two creation (or annihilation) operators carries a $-$ sign and this reflects on their action on an arbitrary Fock state $|N\rangle = |n_1, n_2, \dots, n_S\rangle$, which is given by

$$\begin{aligned}\hat{c}_\alpha^\dagger |n_1, n_2, \dots, n_\alpha, \dots, n_S\rangle &= (-)^{\sum_{\beta < \alpha} n_\beta} \sqrt{1 - n_\alpha} |n_1, n_2, \dots, 1 - n_\alpha, \dots, n_S\rangle \\ \hat{c}_\alpha |n_1, n_2, \dots, n_\alpha, \dots, n_S\rangle &= (-)^{\sum_{\beta < \alpha} n_\beta} \sqrt{n_\alpha} |n_1, n_2, \dots, 1 - n_\alpha, \dots, n_S\rangle\end{aligned}\quad (2.41)$$

Consider now an off-diagonal term of the local potential and suppose one needs to add δk to some component of the occupied momentum \mathbf{K}_α in a state $|N\rangle$, bringing \mathbf{K}_α to \mathbf{K}_β . Then the corresponding term in Eq. (2.23) can be written as

$$\langle N | \hat{v} | \psi \rangle = -\frac{1}{2\delta k^2} \langle N | \hat{c}_\alpha^\dagger \hat{c}_\beta \left(\sum_{N'} c_{N'} |N'\rangle \right) \rangle \quad (2.42)$$

The matrix element $\langle N | \hat{c}_\alpha^\dagger \hat{c}_\beta |N'\rangle$ essentially selects the N' with the same occupied momenta as N , though with \mathbf{K}_β instead of \mathbf{K}_α . However, remembering Eq. (2.41), on top of selecting the correct $c_{N'}$ this matrix element also brings the phase factor

$$(-)^{\sum_{\gamma < \beta} n_\gamma + \sum_{\gamma < \alpha} n_\gamma} \quad (2.43)$$

Note that the ordering of the operators is such that the simple 1D case has the correct sign. This phase factor only depends on the way the indices α in the n_α are mapped onto the available momenta.

To verify the correctness of Eq. (2.42) and the associated phase factor with a practical example, suppose we have a state $|N\rangle$ of M particles with occupied momenta \mathbf{K}_{α_i} , $i = 1, \dots, M$, $\alpha_i < \alpha_{i+1}$ ($n_{\alpha_1} = \dots = n_{\alpha_M} = 1$). The first term of the exact expression of the wave function coefficient c_N (Eq. (2.17)) is

$$\phi_1(\mathbf{K}_{\alpha_1}) \phi_2(\mathbf{K}_{\alpha_2}) \dots \phi_M(\mathbf{K}_{\alpha_M}) \quad (2.44)$$

where the $\phi(\mathbf{K})$ are the eigenstates of Eq. (2.3) in one dimensions. Suppose that, for the calculation of the potential energy, we need $\mathbf{K}_{\alpha_j} \rightarrow \mathbf{K}_\beta$. The first term of this new Slater determinant is

$$\phi_1(\mathbf{K}_{\alpha_1}) \dots \phi_{j-1}(\mathbf{K}_{\alpha_{j-1}}) \phi_j(\mathbf{K}_\beta) \phi_{j+1}(\mathbf{K}_{\alpha_{j+1}}) \dots \phi_M(\mathbf{K}_{\alpha_M}) \quad (2.45)$$

Now, suppose that $\alpha_{j+1} < \beta < \alpha_{j+2}$. Then, by definition, the first term of the coefficient associated with the state with occupied momenta \mathbf{K}_{α_i} , $i \neq j$ and \mathbf{K}_β is (again from Eq. (2.17))

$$\phi_1(\mathbf{K}_{\alpha_1}) \dots \phi_{j-1}(\mathbf{K}_{\alpha_{j-1}}) \phi_j(\mathbf{K}_{\alpha_{j+1}}) \phi_{j+1}(\mathbf{K}_\beta) \phi_{j+2}(\mathbf{K}_{\alpha_{j+2}}) \dots \phi_M(\mathbf{K}_{\alpha_M}) \quad (2.46)$$

to preserve the correct ascending order of the indices α_i , β . It is simple to see that this Slater determinant has opposite sign with respect to the one whose first term is in Eq. (2.45), since Eq. (2.45) and Eq. (2.46) have \mathbf{K}_β and $\mathbf{K}_{\alpha_{j+1}}$ swapped. Therefore, in this case the off-diagonal term of the potential requiring $\mathbf{K}_{\alpha_j} \rightarrow \mathbf{K}_\beta$ actually has a -1 in front. Indeed, in the exponent of Eq. (2.43), the first sum contains j ones while the second only $j - 1$, thus the phase factor is

-1 as expected.

The reason why this issue was not addressed in the unidimensional case is that in 1D we order the states such that if n_α is associated to the occupied momentum \tilde{K} , then $n_{\alpha+1}$ is associated with $\tilde{K} + \delta k$. For this reason, when adding or subtracting δk to some occupied momentum, the indices α and β in Eq. (2.43) are always such that $|\alpha - \beta| = 1$ and, unless n_β is already occupied in the state $|N\rangle$ (in which case the whole term is null), the two sums at the exponent are equal and therefore the phase is always +1. On the contrary, in the $D > 1$ case there exist no ordering of the states which ensures that there never is an occupied state between α and β .

Notice that, in Eq. (2.43), only the occupation numbers *between* α and β actually determine the phase, since the ones before both α and β are counted twice while the ones after them are not counted. For this reason, a possible way of calculating the exponent simply consists in summing over the occupation numbers between indices α and β in the state $|N\rangle$ (a quick check shows that the values of n_α or n_β do not influence the result, thus also the state $|N'\rangle$ can be used for this calculation).

This issue can be spotted by performing full CI in multiple dimensions. Due to the unfavorable scaling of the size of the basis, we only implement it for the case $M = D = 2$. In this case, the energy obtained from the diagonalization can be compared with the prediction given by the diagonalization in 1D, since the energy of the ground-state and the first excited state in 2D are given by $E_0^{2D} = 2E_0^{1D}$ and $E_1^{2D} = E_0^{1D} + E_1^{1D}$. To give an example, for $L = 5$ and $A = 9$ the 1D diagonalization yields $E_0^{1D} = 0.4259089$ and $E_1^{1D} = 1.380962$, predicting a value for the energy for $M = D = 2$ of $E = E_0^{2D} + E_1^{2D} = 3E_0^{1D} + E_1^{1D} = 2.658689$; on the contrary, the result of the 2D diagonalization without accounting for the sign factor in Eq. (2.43) gives $\tilde{E} = 2.411557$, whereas the correct value is obtained when the factor is implemented.

The wave function coefficients resulting from the full CI diagonalization can be used to test the implementation of the MC algorithm for $M = D = 2$. Table V reports the comparison between the results of an MC run using the c_N obtained from full CI and the full CI results themselves.

	MC result	FCI result
Kinetic Energy	1.2604 ± 0.0020	1.2631
Potential Energy	1.3983 ± 0.0020	1.3956
Total Energy	2.658689	2.658689

Table V: Average values with estimated errors of the energies obtained from a MC run of 10^5 samples with $M = D = 2$, $A = 9$ and $L = 5$ compared with the results of the full CI approach. As usual, the total energy is reported with an arbitrary number of significant digits and with no error for the reasons discussed in Sec. 2.2.2.

The results show that the MC code is correct and, since it is implemented with no specific references to the number of particles and the dimensionality, we assume it is also valid for generic M and D . We shall actually verify this statement in more detail when implementing the NQSs with the Stochastic Reconfiguration algorithm.

2.4 Implementation of the Neural-network Quantum States

As mentioned in Sec. 1.3, a fundamental aspect of this Thesis work is the use of NQSs. The wave function coefficients are given by

$$c_N = e^{U(N)} \tanh(\mathcal{V}(N)) \quad (2.47)$$

where $U(N)$ and $\mathcal{V}(N)$ are calculated by two different FCNNs, both with a number of inputs equal to the total number of SPOs S . We tested many different choices of hyperparameters (always with tanh activation function), including the width and the depth of the NN and the Stochastic Reconfiguration parameters (see the next Section). However, we did not employ an actual model selection, which may be actually useful for future usage of the code developed, since we observed that the convergence is usually heavily affected by these parameters. In general, we found that using 2 or 3 hidden layers is beneficial, with the first layer being at least as wide as the input layer (giving the $\mathcal{O}(S^2)$ scaling of the number of variational parameters). For deeper architectures, at least in the cases tested here, a very small time step $\delta\tau$ for the SR algorithm (with respect to other architectures) was in general required in order for the convergence not to stop early, making the optimization very slow. We feed the FCNNs the occupation numbers n_i of an input state $|N\rangle$ as $2n_i - 1$, since in general this representation eases the convergence. We stress the fact that the hyperparameters chosen for the results reported in the next Sections may not be the optimal ones, and further attention to this point should be paid for future applications of this algorithm and to assess the efficiency of this representation of the wave function.

2.5 Stochastic Reconfiguration

In the context of machine learning, many different rules for updating the parameters of a Neural Network have been proposed [30], from the well-known Stochastic Gradient Descent (SGD) to more sophisticated versions including “Momentum” [31], “AdaGrad” [32], “RMSprop” [33] and the widely used “Adam” [34]; however, for VMC applications, the Stochastic Reconfiguration (SR) algorithm [35, 36] has proven to be significantly more efficient than the standard gradient-based algorithms, which provide an exceedingly slow convergence (see for instance Fig. 2.5). The algorithm is essentially equivalent to performing an imaginary-time evolution of the variational wave function: in fact, given a generic Schrödinger equation with a time-independent Hamiltonian \hat{H} :

$$i\frac{\partial}{\partial t} |\psi\rangle = \hat{H} |\psi\rangle \quad \rightarrow \quad -\frac{\partial}{\partial \tau} |\psi\rangle = \hat{H} |\psi\rangle \quad , \quad (2.48)$$

for sufficiently small $\delta\tau$ one can write

$$|\psi(\tau + \delta\tau)\rangle = (1 - \delta\tau\hat{H}) |\psi(\tau)\rangle \quad (2.49)$$

Now, notice that successively applying the operator in the brackets to an arbitrary state³ in a VMC context, which can in general be written as the linear combination of the eigenstates of the Hamiltonian \hat{H} , would converge to the ground-state of the system, while the excited states

³The state cannot be completely arbitrary, as it needs to have non-zero overlap with the ground-state.

are projected out. Therefore, in Eq. (2.49) we identify $|\psi(\tau)\rangle$ with the variational wave function $|\psi(\boldsymbol{\lambda})\rangle$ for some set of variational parameters $\boldsymbol{\lambda}$ and we approximate $|\psi(\tau + \delta\tau)\rangle$ with a linear combination of the wave function $|\psi(\tau)\rangle = |\psi(\boldsymbol{\lambda})\rangle$ and its first derivatives with respect to the variational parameters, assuming the variation of the parameters to be small:

$$a_0 |\psi(\boldsymbol{\lambda})\rangle + \sum_q \delta\lambda_q |\partial_{\lambda_q} \psi|_{\boldsymbol{\lambda}}\rangle = (1 - \delta\tau \hat{H}) |\psi(\boldsymbol{\lambda})\rangle \quad (2.50)$$

Now, introducing again the operator \hat{O}_q such that $\hat{O}_q |\psi(\boldsymbol{\lambda})\rangle = |\partial_{\lambda_q} \psi|_{\boldsymbol{\lambda}}\rangle$ and multiplying both sides by $\langle\psi(\boldsymbol{\lambda})|$ and $\langle\psi(\boldsymbol{\lambda})| \hat{O}_p^\dagger$ we obtain

$$\begin{aligned} a_0 \langle\psi(\boldsymbol{\lambda})|\psi(\boldsymbol{\lambda})\rangle + \sum_q \delta\lambda_q \langle\psi(\boldsymbol{\lambda})|\hat{O}_q|\psi(\boldsymbol{\lambda})\rangle &= \langle\psi(\boldsymbol{\lambda})|(1 - \delta\tau \hat{H})|\psi(\boldsymbol{\lambda})\rangle \\ a_0 \langle\psi(\boldsymbol{\lambda})|\hat{O}_p^\dagger|\psi(\boldsymbol{\lambda})\rangle + \sum_q \delta\lambda_q \langle\psi(\boldsymbol{\lambda})|\hat{O}_p^\dagger\hat{O}_q|\psi(\boldsymbol{\lambda})\rangle &= \langle\psi(\boldsymbol{\lambda})|\hat{O}_p^\dagger(1 - \delta\tau \hat{H})|\psi(\boldsymbol{\lambda})\rangle \end{aligned} \quad (2.51)$$

Defining $\langle\hat{A}\rangle = \langle\psi(\boldsymbol{\lambda})|\hat{A}|\psi(\boldsymbol{\lambda})\rangle / \langle\psi(\boldsymbol{\lambda})|\psi(\boldsymbol{\lambda})\rangle$ and dividing on both sides of both equations by $\langle\psi(\boldsymbol{\lambda})|\psi(\boldsymbol{\lambda})\rangle$, we can write

$$\begin{aligned} a_0 + \sum_q \delta\lambda_q \langle\hat{O}_q\rangle &= \langle 1 - \delta\tau \hat{H} \rangle \\ a_0 \langle\hat{O}_p^\dagger\rangle + \sum_q \delta\lambda_q \langle\hat{O}_p^\dagger\hat{O}_q\rangle &= \langle\hat{O}_p^\dagger(1 - \delta\tau \hat{H})\rangle \end{aligned} \quad (2.52)$$

At this point, we can isolate a_0 in the first equation and substitute it in the second:

$$\sum_q \left[\langle\hat{O}_p^\dagger\hat{O}_q\rangle - \langle\hat{O}_p^\dagger\rangle\langle\hat{O}_q\rangle \right] \delta\lambda_q = -\delta\tau \left[\langle\hat{O}_p^\dagger\hat{H}\rangle - \langle\hat{O}_p^\dagger\rangle\langle\hat{H}\rangle \right] \quad (2.53)$$

so that, defining the overlap matrix and the gradient

$$\begin{aligned} S_{pq} &= \langle\hat{O}_p^\dagger\hat{O}_q\rangle - \langle\hat{O}_p^\dagger\rangle\langle\hat{O}_q\rangle \\ G_p &= \langle\hat{O}_p^\dagger\hat{H}\rangle - \langle\hat{O}_p^\dagger\rangle\langle\hat{H}\rangle \end{aligned} \quad (2.54)$$

we can write

$$\sum_q S_{pq} \delta\lambda_q = -\delta\tau G_p \quad (2.55)$$

The solutions to this linear system $\delta\lambda_p$ are used to update the parameters of the wave function $|\psi(\boldsymbol{\lambda})\rangle$. Note that there is actually no need to compute the coefficient a_0 and update the wave function parameters as $\lambda_p \rightarrow \lambda_p + \delta\lambda_p/a_0$, since the expansion in Eq. (2.50) is an approximation (up to order $\delta\lambda^2$) anyway. The gradient G_p can be estimated as discussed in Section 2.2.5, while the matrix S_{pq} is expressed in the following covariance form:

$$S_{pq} \simeq \left(\frac{1}{N_s} \sum_{i=1}^{N_s} \mathcal{O}_p^*(N^{[i]}) \mathcal{O}_q(N^{[i]}) \right) - \left(\frac{1}{N_s} \sum_{i=1}^{N_s} \mathcal{O}_p^*(N^{[i]}) \right) \left(\frac{1}{N_s} \sum_{i=1}^{N_s} \mathcal{O}_q(N^{[i]}) \right) \quad (2.56)$$

Since the matrix S_{pq} is positive definite, the system in Eq. (2.55) can be solved using Cholesky decomposition. However, this solver requires storing the full matrix S_{pq} , which may become exceedingly expensive in terms of memory if the number of variational parameters is large, as may be the case for large numbers of dimensions D and SPOs S . At the time of writing, we are also implementing a conjugate gradient solver in order to deal with large sets of variational parameters, as in this case only a function that calculates $\sum_q S_{pq}x_q$ for an arbitrary \mathbf{x} is needed. To make the algorithm more stable, we actually solve a slightly different system, substituting S_{pq} with $S_{pq} + \varepsilon[\mathbb{1} + \text{diag}(S_{pq})]$, $\varepsilon \ll 1$.

As mentioned above, the procedure described up to now is valid if $\delta\tau$ and the corresponding $\delta\lambda_p$ are sufficiently small: in fact, ensuring that this condition holds is important to guarantee the convergence of the algorithm. Quantitatively, the condition can be checked by comparing two expressions of the distance between the old wave function $|\psi(\boldsymbol{\lambda})\rangle$ and the one with updated parameters $|\psi(\boldsymbol{\lambda} + \delta\boldsymbol{\lambda})\rangle$. Since their normalizations are irrelevant, the Fubini-Study distance can be shown to be the natural (Riemannian) metric [37] (in the following expressions we shall denote $|\psi(\boldsymbol{\lambda})\rangle$ by $|\psi_0\rangle$ and $|\psi(\boldsymbol{\lambda} + \delta\boldsymbol{\lambda})\rangle$ by $|\psi_1\rangle$ for ease of notation):

$$d_{FS}[\psi(\boldsymbol{\lambda}), \psi(\boldsymbol{\lambda} + \delta\boldsymbol{\lambda})] = d_{FS}[\psi_0, \psi_1] = \arccos \sqrt{\frac{|\langle\psi_0|\psi_1\rangle|^2}{\langle\psi_0|\psi_0\rangle\langle\psi_1|\psi_1\rangle}} \quad (2.57)$$

The term in the square root can be computed using the re-weighting technique:

$$\begin{aligned} \frac{|\langle\psi_0|\psi_1\rangle|^2}{\langle\psi_0|\psi_0\rangle\langle\psi_1|\psi_1\rangle} &= \frac{\langle\psi_0|\psi_1\rangle\langle\psi_1|\psi_0\rangle}{\langle\psi_0|\psi_0\rangle\langle\psi_1|\psi_1\rangle} \\ &= \left(\sum_N P_0(N) \frac{\langle N|\psi_1\rangle}{\langle N|\psi_0\rangle} \right) \left(\sum_N P_0(N) \frac{\langle\psi_1|N\rangle}{\langle\psi_0|N\rangle} \right) \left(\sum_N P_0(N) \frac{|\langle N|\psi_1\rangle|^2}{|\langle N|\psi_0\rangle|^2} \right)^{-1} \end{aligned} \quad (2.58)$$

where $P_0(N) = |\langle N|\psi_0\rangle|^2 / \langle\psi_0|\psi_0\rangle$ is the distribution sampled using the Metropolis algorithm during the Monte Carlo, thus we only need to calculate the factors in the sums and average them over a given number of sampled Fock states to calculate the distance in Eq. (2.57). Additionally, for small $\delta\boldsymbol{\lambda}$, the distance can also be expressed in terms of the overlap matrix S_{pq} [37]:

$$d_{FS2}[\psi(\boldsymbol{\lambda}), \psi(\boldsymbol{\lambda} + \delta\boldsymbol{\lambda})]^2 = \sum_{pq} S_{pq} \delta\lambda_p \delta\lambda_q \quad (2.59)$$

Imposing a threshold to the difference between these two distances allows to ensure that the variation of the parameters is sufficiently small, improving the convergence of the algorithm; in particular, we check whether

$$\frac{|d_{FS}[\psi(\boldsymbol{\lambda}), \psi(\boldsymbol{\lambda} + \delta\boldsymbol{\lambda})]^2 - d_{FS2}[\psi(\boldsymbol{\lambda}), \psi(\boldsymbol{\lambda} + \delta\boldsymbol{\lambda})]^2|}{|d_{FS}[\psi(\boldsymbol{\lambda}), \psi(\boldsymbol{\lambda} + \delta\boldsymbol{\lambda})]^2 + d_{FS2}[\psi(\boldsymbol{\lambda}), \psi(\boldsymbol{\lambda} + \delta\boldsymbol{\lambda})]^2|} < 0.8 \quad (2.60)$$

For further stabilization, we actually check this condition for two different sets of sampled Fock states, one being a subset of those used to evaluate S_{pq} and the other being a separate one, in order to test whether the estimated S_{pq} is a good representative of the actual overlap matrix.

We also add a small constant to the denominator of Eq. (2.60) to avoid issues in the case of very small distances. Finally, we also constrain the largest $\delta\lambda$ to be smaller than a given threshold.

Up to now, we assumed $\delta\tau$ and ε to be fixed hyperparameters; however, they can also be adjusted during the optimization in order to speed up the convergence. We do this by testing different values, each time calculating the new $\delta\lambda$ and estimating the associated energy and its error using the re-weighting technique (similarly to Eq. (2.58)) from a given set of sampled Fock states. We then keep the updated parameters corresponding to the smallest new energy also imposing a maximum threshold to its error for further stabilization. We consider two possible implementations: *adaptive time step* and *adaptive epsilon*. In the former case, we keep ε constant and test different $\delta\tau$ defined as $2^n\delta\tau_0$, where $\delta\tau_0$ is a starting time step (typically ~ 0.0002) and n an integer ranging from 0 to ~ 6 . In the latter case, we keep $\delta\tau$ constant and set $\log(\varepsilon_{min})$ and $\log(\varepsilon_{max})$, then we repeat the following steps for ~ 6 times: we define $\log(\varepsilon) = (\log(\varepsilon_{min}) + \log(\varepsilon_{max}))/2$, test the parameters $\delta\tau$ and ε and, if the conditions above are met, we save the proposed variation of the parameters and set $\varepsilon_{max} = \varepsilon$, otherwise we set $\varepsilon_{min} = \varepsilon$. For both algorithms, in case no choice of $\delta\tau$ and ε satisfies the conditions above, we do not update the parameters. Fig. 2.5 compares the performance of different update rules of the variational parameters for $D = 1$, $S = 11$, $M = 3$ and $L = 10$ using NQs.

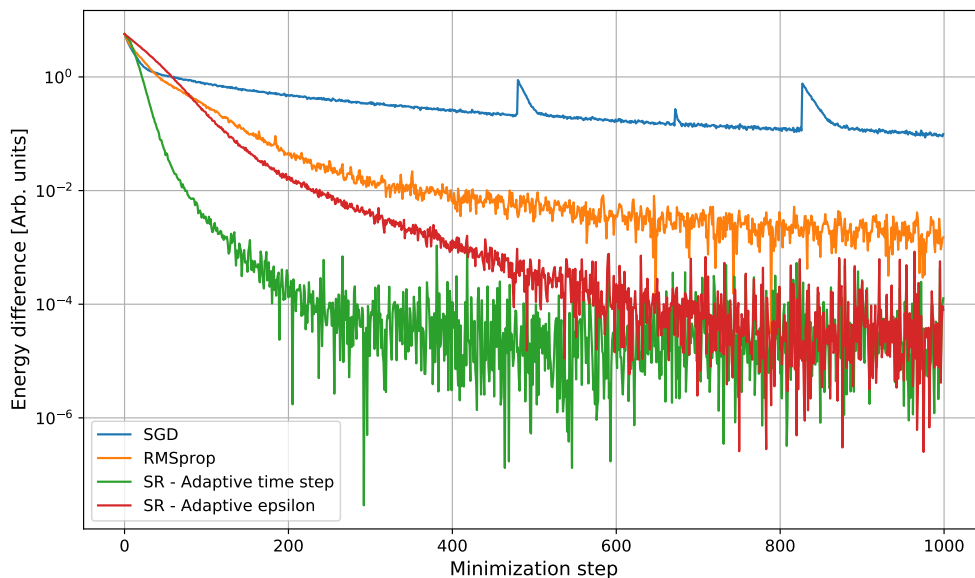


FIG 2.5: Convergence of different update rules for the variational parameters to the exact energy as calculated using the FD approach. The time step (or learning rate) of SGD was 0.007, while for RMSprop we used standard parameters ($\rho = 0.9$ and $\eta = 0.001$). The parameters of adaptive time step SR were $\delta\tau_0 = 0.0007$, $\varepsilon = 2 \times 10^{-5}$ while those of adaptive epsilon SR were $\varepsilon_{min} = 10^{-4}$, $\varepsilon_{max} = 10^{-6}$, $\delta\tau = 0.007$ and both schemes checked 6 different values for $\delta\tau$ and ε as specified above.

The advantage of using SR is evident. In general, the adaptive time step scheme converges much more quickly, although it actually remains stuck in local minima more often than adaptive epsilon; for this reason, for the results reported here and in the next Sections we always used the latter.

Tab. VI reports the results of two optimizations for different box sizes L , dimensions D , number states along each direction A and number of particles M using simple NQS and the SR algorithm. We also include the exact ground-state energies calculated using the FD approach discussed in Sec. 2.1. The values reported in the table were obtained from a final MC run using the optimized variational parameters; the same is true for all results presented in the next Sections.

L	D	A	M	NQS VMC	FD
10	1	11	3	$4.249885 \pm 7.3 \times 10^{-5}$	4.249792
5	2	5	5	$8.49972 \pm 5.9 \times 10^{-4}$	8.49298

Table VI: Results of two optimizations using the VMC algorithm with NQS. In the first case, the FCNNs were comprised of 16 and 8 neurons in the first and second hidden layers and we performed 1000 optimization steps with $\delta\tau = 0.07$. In the second case, we employed two layers of 32 and 16 neurons and performed 1500 optimization steps with $\delta\tau = 0.02$.

On top of verifying the correct implementation of both the SR algorithm and the NQS, the results show that, for small inputs, even shallow NNs can produce significantly accurate results for different L , D , A and M in relatively few optimization steps. However, it is worth mentioning that storing the full overlap matrix S_{pq} defined in Eq. (2.54) does actually represent a bottleneck for these calculations: in fact, the number of variational parameters typically grows quadratically with the number of SPOs so that the memory requirements of the present version of the SR method scale as A^{4D} . This is the reason why we did not test the code for $D = 3$ and for $A > 5$ in the $D = 2$ case for the results in the next Sections. Nevertheless, as mentioned above, we are currently implementing the conjugate gradient solver for the system in Eq. (2.55), which requires storing at most a $N_{\text{walk}} \times N_{\text{params}}$ matrix, where N_{params} is the number of parameters of the FCNNs.

2.6 Spin

In this Section we discuss the introduction of spin. We shall only discuss the $s = 1/2$ case, although a generalization to a generic s is straightforward. We implement spin by concatenating two arrays $\mathbf{n} = (n_0, n_1, \dots, n_{S-1})$ of momentum occupation numbers, associating the first indices $i = 0, \dots, S - 1$ to one spin projection (say up) and the next $j = S, \dots, 2S - 1$ to the other (say down). A generic index $\Lambda = 0, \dots, 2S - 1$ of an occupation number in the new Fock state is then associated with the momentum state index $\alpha = \Lambda \bmod S$ and the spin projection index $\sigma = \lfloor \Lambda/S \rfloor$, with $\sigma = 0$ indicating a spin up particle and $\sigma = 1$ a spin down one. Correspondingly, the Fock state index for a particle with spin projection σ and momentum \mathbf{K}_α is given by $\alpha + S\sigma$. With this new definition of the states $|N\rangle$ we let the NNs learn all the correlations between same and opposite spin projections, which are not necessarily the same. In fact, in the case of the Coulomb interaction, it is well known that the cusp condition for same-spin particles is different from the one used for opposite spins.

If the Hamiltonian does not include spin-dependent interactions, the spin projection is preserved and thereby the transition kernel in the Metropolis must not mix occupation numbers

representing different σ . This can be done simply by generating two different pair indices $k_{\sigma_1}, k_{\sigma_2} = 0, \dots, S - 1$ and use the first to exchange two occupation numbers with index $\Lambda < S$ and the second to exchange two different occupation numbers with index $S \leq \Lambda < 2S$.

The new definition of the Fock states also allow to calculate the local kinetic and potential energies in the same exact way as described in the previous Sections, provided that Eq. (2.37) is modified as

$$K_{\Lambda}^d = \left[I_{\Lambda}^d - \frac{A-1}{2} \right] \delta k = \left[\left(\left\lfloor \frac{\Lambda \bmod S}{A^d} \right\rfloor \bmod A \right) - \frac{A-1}{2} \right] \delta k \quad (2.61)$$

In Tab. VIII we list the results of four optimizations with different L, D, A , and M (as in Tab. VI), though for different polarizations of the systems (we indicate the number of spin up particles with M_{up} and consider only either fully polarized or unpolarized systems for simplicity).

L	D	A	M	M_{up}	NQS VMC	FD
10	1	9	3	3	4.26890 ± 0.00023	4.26813
10	1	9	6	3	8.53689 ± 0.00024	8.53626
5	2	5	5	5	8.49760 ± 0.00078	8.49298
5	2	5	10	5	17.0324 ± 0.0012	16.9860

Table VII: Results of four optimizations using the VMC algorithm with NQS including spin. For the first two results, the NNs are comprised of layers of 32, 16 and 16 neurons and we performed 3000 optimization steps with a time step $\delta\tau$ decreasing from 0.003 to 0.0005, while for the other optimizations featured NNs with layers of 50, 32 and 32 neurons and 2000 optimization steps with $\delta\tau$ decreasing from 0.01 to 0.006 and from 0.02 to 0.001.

Here it is worth mentioning the fact that the number of Fock states for the $D = 2, M = 10$ optimization is $\sim 3 \times 10^9$. Storing and diagonalizing a matrix of this size for a full CI procedure is extremely expensive from a computational point of view; on the other hand, this algorithm reaches significant accuracy using relatively few parameters ($\sim 10^4$) and requiring the storing of a $\sim 10^4 \times 10^4$ matrix for the Cholesky decomposition (or a $\sim 10^4 \times 10^3$ one for the conjugate gradient solver). On top of that, it is also interesting to notice that the results for equal L, D and A are not heavily affected by variations in the number of particles M , which constitutes an important advantage of this representation of the many-body wave function.

Chapter 3

Fock space VMC with two-body potential

In this Chapter we describe the implementation of a two-body potential which only depends on the modulus of the relative coordinate between two particles. Making use of the algorithm discussed in the previous Sections, we test this implementation using a simple Gaussian interaction. Note that the actual calculation of the local potential energy can be generalized to arbitrary second quantized expressions with little effort.

3.1 Practical calculation of the local energy

A generic second quantized two-body potential acting on a system of spin 1/2 fermions with plane wave SPOs has the expression

$$\hat{V} = \frac{1}{2} \sum_{i,j=1}^M \hat{v}_{ij} = \frac{1}{2} \sum_{\sigma \sigma'} \sum_{\alpha_1 \alpha_2 \beta_1 \beta_2} \langle \mathbf{K}_{\alpha_1}, \mathbf{K}_{\alpha_2} | v | \mathbf{K}_{\beta_1}, \mathbf{K}_{\beta_2} \rangle \hat{c}_{\alpha_1 \sigma}^\dagger \hat{c}_{\alpha_2 \sigma'}^\dagger \hat{c}_{\beta_2 \sigma'} \hat{c}_{\beta_1 \sigma} \quad (3.1)$$

where $\hat{c}_{\alpha \sigma}^\dagger$ ($\hat{c}_{\alpha \sigma}$) creates (annihilates) a particle with momentum \mathbf{K}_α and spin projection σ and the matrix element is given by

$$\begin{aligned} \langle \mathbf{p}_1, \mathbf{p}_2 | v | \mathbf{k}_1, \mathbf{k}_2 \rangle &= \frac{1}{L^{2D}} \int_R d\mathbf{r}_1 d\mathbf{r}_2 d\mathbf{r}'_1 d\mathbf{r}'_2 e^{-i\mathbf{p}_1 \cdot \mathbf{r}_1} e^{-i\mathbf{p}_2 \cdot \mathbf{r}_2} \langle \mathbf{r}_1, \mathbf{r}_2 | v | \mathbf{r}'_1, \mathbf{r}'_2 \rangle e^{i\mathbf{k}_1 \cdot \mathbf{r}'_1} e^{i\mathbf{k}_2 \cdot \mathbf{r}'_2} \\ &= \frac{1}{L^{2D}} \int_R d\mathbf{r}_1 d\mathbf{r}_2 e^{-i\mathbf{r}_1 \cdot (\mathbf{p}_1 - \mathbf{k}_1)} e^{-i\mathbf{r}_2 \cdot (\mathbf{p}_2 - \mathbf{k}_2)} v(\mathbf{r}_1, \mathbf{r}_2) \end{aligned} \quad (3.2)$$

where the integration is performed over a hypercube of size L and we assumed the potential to be local thus diagonal in the coordinates. Now, suppose that $v(\mathbf{r}_1, \mathbf{r}_2) = v(\mathbf{r}_1 - \mathbf{r}_2)$; then, in the limit $L \rightarrow \infty$ one can introduce the center of mass coordinate $\mathbf{R} = (\mathbf{r}_1 + \mathbf{r}_2)/2$ and the relative coordinate $\mathbf{r} = \mathbf{r}_1 - \mathbf{r}_2$, so that Eq. (3.2) becomes

$$\begin{aligned} \langle \mathbf{p}_1, \mathbf{p}_2 | v | \mathbf{k}_1, \mathbf{k}_2 \rangle &= \frac{1}{L^{2D}} \int d\mathbf{r}_1 d\mathbf{r}_2 e^{-i\mathbf{r}_1 \cdot (\mathbf{p}_1 - \mathbf{k}_1)} e^{-i\mathbf{r}_2 \cdot (\mathbf{p}_2 - \mathbf{k}_2)} v(\mathbf{r}_1, \mathbf{r}_2) \\ &= \frac{1}{L^{2D}} \int d\mathbf{R} d\mathbf{r} e^{-i(\mathbf{R} + \mathbf{r}/2) \cdot (\mathbf{p}_1 - \mathbf{k}_1)} e^{-i(\mathbf{R} - \mathbf{r}/2) \cdot (\mathbf{p}_2 - \mathbf{k}_2)} v(\mathbf{r}) \\ &= \frac{1}{L^{2D}} \int d\mathbf{R} e^{-i\mathbf{R} \cdot (\mathbf{p}_1 + \mathbf{p}_2 - \mathbf{k}_1 - \mathbf{k}_2)} \int d\mathbf{r} e^{-i\mathbf{r}/2 \cdot (\mathbf{p}_1 - \mathbf{p}_2 - \mathbf{k}_1 + \mathbf{k}_2)} v(\mathbf{r}) \end{aligned} \quad (3.3)$$

and so, defining the transferred momentum $2\mathbf{q} = \mathbf{p}_1 - \mathbf{p}_2 - (\mathbf{k}_1 - \mathbf{k}_2)$

$$\langle \mathbf{p}_1, \mathbf{p}_2 | v | \mathbf{k}_1, \mathbf{k}_2 \rangle = \delta(\mathbf{p}_1 + \mathbf{p}_2 - \mathbf{k}_1 - \mathbf{k}_2) \frac{1}{L^D} \int d\mathbf{r} e^{-i\mathbf{r} \cdot \mathbf{q}} v(\mathbf{r}) = \delta_{\mathbf{p}_1 + \mathbf{p}_2, \mathbf{k}_1 + \mathbf{k}_2} \tilde{v}(\mathbf{q}) \quad (3.4)$$

Note that the Kronecker delta simply ensures the conservation of the total momentum. Returning to Eq. (3.1), by adding and subtracting the conservation of total momentum $0 = \mathbf{K}_{\alpha_1} + \mathbf{K}_{\alpha_2} - \mathbf{K}_{\beta_1} - \mathbf{K}_{\beta_2}$ to the expression of the transferred momentum $2\mathbf{Q} = \mathbf{K}_{\alpha_1} - \mathbf{K}_{\alpha_2} - (\mathbf{K}_{\beta_1} - \mathbf{K}_{\beta_2})$, one finds

$$\begin{aligned} \mathbf{K}_{\beta_1} &= \mathbf{K}_{\alpha_1} - \mathbf{Q} \\ \mathbf{K}_{\beta_2} &= \mathbf{K}_{\alpha_2} + \mathbf{Q} \end{aligned} \quad (3.5)$$

and so it is possible to write

$$\hat{V} = \frac{1}{2} \sum_{\sigma \sigma'} \sum_{\alpha_1 \alpha_2, \mathbf{Q}} \tilde{v}(\mathbf{Q}) \hat{c}_{\alpha_1 \sigma}^\dagger \hat{c}_{\alpha_2 \sigma'}^\dagger \hat{c}_{\beta_2 \sigma'} \hat{c}_{\beta_1 \sigma} \quad (3.6)$$

where β_1 and β_2 are such that the relations in Eq. (3.5) are satisfied. Note that, for a finite basis, given some α_1 and \mathbf{Q} such that $\mathbf{K}_{\alpha_1} - \mathbf{Q}$ lies within the momentum limits, the same may not be true for $\mathbf{K}_{\alpha_2} + \mathbf{Q}$ ¹. In what follows we shall refer to a transferred momentum as “valid” if both $\mathbf{K}_{\alpha_1} - \mathbf{Q}$ and $\mathbf{K}_{\alpha_2} + \mathbf{Q}$ lie within the limits, which amounts to requiring the components Q^d , $d = 1, \dots, D$ of \mathbf{Q} to satisfy

$$\begin{aligned} |K_{\alpha_1}^d - Q^d| &\leq (A - 1)\delta k / 2 \\ |K_{\alpha_2}^d + Q^d| &\leq (A - 1)\delta k / 2 \end{aligned} \quad (3.7)$$

where A is the number of states along each direction. Figure 3.1 illustrates an example of the reachable momenta \mathbf{K}_{β_1} , \mathbf{K}_{β_2} given some α_1 and α_2 for $S = 25$ in 2D.

20 (-2, 2)	21 (-1, 2)	22 (0, 2)	23 (1, 2)	24 (2, 2)
15 (-2, 1)	16 (-1, 1)	17 (0, 1)	18 (1, 1)	19 (2, 1)
10 (-2, 0)	11 (-1, 0)	12 (0, 0)	13 (1, 0)	14 (2, 0)
5 (-2, -1)	6 (-1, -1)	7 (0, -1)	8 (1, -1)	9 (2, -1)
0 (-2, -2)	1 (-1, -2)	2 (0, -2)	3 (1, -2)	4 (2, -2)

FIG 3.1: Representation of the reachable momenta in 2D with $S = 25$. The vector associated with each square represents the momentum itself \mathbf{K}_α ($\delta k = 1$) while the gray number in the top right corner the associated momentum index α . The colored squares indicate two arbitrary α_1 and α_2 , while the black rectangle highlights the momentum states reachable by both \mathbf{K}_{β_1} and \mathbf{K}_{β_2} through valid \mathbf{Q} in Eq. (3.5).

¹For a simple example, consider the case $D = 1$, $S = 5$, $\alpha_1 = 3 \rightarrow K_{\alpha_1} = \delta k$, $\alpha_2 = 4 \rightarrow K_{\alpha_2} = 2\delta k$ and $Q = \delta k$.

Notice that, if \mathbf{Q} satisfies these requirements and we choose again the ordering of the momentum states discussed in Sec. 2.3, $\alpha_1 - \beta_1 = \beta_2 - \alpha_2 := x$, as one can derive from Eq. (2.38) and Eq. (3.5); the requirements over \mathbf{Q} ensure that x is not degenerate. In the next expressions, we shall always assume the sum over \mathbf{Q} , which now depends on α_1 and α_2 , to only span valid \mathbf{Q} (we indicate this constraint with a primed sum). The expression of the potential then becomes

$$\hat{V} = \frac{1}{2} \sum_{\sigma \sigma'} \sum_{\alpha_1 \alpha_2} \sum_{\mathbf{Q}}' \tilde{v}(\mathbf{Q}) \hat{c}_{\alpha_1 \sigma}^\dagger \hat{c}_{\alpha_2 \sigma'}^\dagger \hat{c}_{\alpha_2+x \sigma'} \hat{c}_{\alpha_1-x \sigma} \quad (3.8)$$

Using the notation introduced in Sec. 2.6, we can rewrite the sums over $\sigma, \sigma', \alpha_1$ and α_2 as a sum over Λ_1 and Λ_2 :

$$\hat{V} = \frac{1}{2} \sum_{\Lambda_1, \Lambda_2} \sum_{\mathbf{Q}}' \tilde{v}(\mathbf{Q}) \hat{c}_{\alpha_1 \sigma_1}^\dagger \hat{c}_{\alpha_2 \sigma_2}^\dagger \hat{c}_{\alpha_2+x \sigma_2} \hat{c}_{\alpha_1-x \sigma_1} \quad (3.9)$$

Since the contribution $\Lambda_1 = \Lambda_2$ is null (due to the two creation or annihilation operators), the sum over the Fock state indices can be split into two parts:

$$\begin{aligned} \hat{V} &= \frac{1}{2} \left(\sum_{\Lambda_1 < \Lambda_2} \sum_{\mathbf{Q}}' \tilde{v}(\mathbf{Q}) \hat{c}_{\alpha_1 \sigma_1}^\dagger \hat{c}_{\alpha_2 \sigma_2}^\dagger \hat{c}_{\alpha_2+x \sigma_2} \hat{c}_{\alpha_1-x \sigma_1} + \right. \\ &\quad \left. + \sum_{\Lambda_2 < \Lambda_1} \sum_{\mathbf{Q}}' \tilde{v}(\mathbf{Q}) \hat{c}_{\alpha_1 \sigma_1}^\dagger \hat{c}_{\alpha_2 \sigma_2}^\dagger \hat{c}_{\alpha_2+x \sigma_2} \hat{c}_{\alpha_1-x \sigma_1} \right) \\ &= \frac{1}{2} \left(\dots + \sum_{\Lambda_2 < \Lambda_1} \sum_{\mathbf{Q}}' \tilde{v}(\mathbf{Q}) \hat{c}_{\alpha_2 \sigma_2}^\dagger \hat{c}_{\alpha_1 \sigma_1}^\dagger \hat{c}_{\alpha_1-x \sigma_1} \hat{c}_{\alpha_2+x \sigma_2} \right) \end{aligned} \quad (3.10)$$

where in the last line we swapped two creation and annihilation operators twice, resulting in an overall $+$ sign. Since given α_1 and α_2 the allowed momenta \mathbf{Q} are such that their components Q^d satisfy Eq. (3.7), if we were to swap the indices, the allowed momenta and their corresponding x would remain the same in modulus but flip their signs. To make the next steps more clear, let us now indicate the dependence on the indices as $\mathbf{Q}(\alpha_1, \alpha_2)$ and $x(\alpha_1, \alpha_2)$ rather than a prime. The reasoning above then implies that, for each valid $\mathbf{Q}(\alpha_1, \alpha_2)$, there exist an valid $\mathbf{Q}(\alpha_2, \alpha_1) = -\mathbf{Q}(\alpha_1, \alpha_2)$ with $x(\alpha_2, \alpha_1) = -x(\alpha_1, \alpha_2)$. With this in mind, we consider only the second sum in Eq. (3.10) and change the names of Λ_1 and Λ_2 :

$$\begin{aligned} S_2 &= \sum_{\Lambda_2 < \Lambda_1} \sum_{\mathbf{Q}(\alpha_1, \alpha_2)} \tilde{v}(\mathbf{Q}(\alpha_1, \alpha_2)) \hat{c}_{\alpha_2 \sigma_2}^\dagger \hat{c}_{\alpha_1 \sigma_1}^\dagger \hat{c}_{\alpha_1-x(\alpha_1, \alpha_2) \sigma_1} \hat{c}_{\alpha_2+x(\alpha_1, \alpha_2) \sigma_2} \\ &= \sum_{\Lambda_4 < \Lambda_3} \sum_{\mathbf{Q}(\alpha_3, \alpha_4)} \tilde{v}(\mathbf{Q}(\alpha_3, \alpha_4)) \hat{c}_{\alpha_4 \sigma_4}^\dagger \hat{c}_{\alpha_3 \sigma_3}^\dagger \hat{c}_{\alpha_3-x(\alpha_3, \alpha_4) \sigma_3} \hat{c}_{\alpha_4+x(\alpha_3, \alpha_4) \sigma_4} \\ &= \sum_{\Lambda_4 < \Lambda_3} \sum_{\mathbf{Q}(\alpha_4, \alpha_3)} \tilde{v}(-\mathbf{Q}(\alpha_4, \alpha_3)) \hat{c}_{\alpha_4 \sigma_4}^\dagger \hat{c}_{\alpha_3 \sigma_3}^\dagger \hat{c}_{\alpha_3+x(\alpha_4, \alpha_3) \sigma_3} \hat{c}_{\alpha_4-x(\alpha_4, \alpha_3) \sigma_4} \end{aligned} \quad (3.11)$$

If we identify $\Lambda_4 \rightarrow \Lambda_1$ and $\Lambda_3 \rightarrow \Lambda_2$, excluding the term with $\tilde{v}(\mathbf{q})$ this last expression is

identical to the first sum in Eq. (3.10):

$$S_1 = \sum_{\Lambda_1 < \Lambda_2} \sum_{\mathbf{Q}(\alpha_1, \alpha_2)} \tilde{v}(\mathbf{Q}(\alpha_1, \alpha_2)) \hat{c}_{\alpha_1 \sigma_1}^\dagger \hat{c}_{\alpha_2 \sigma_2}^\dagger \hat{c}_{\alpha_2+x(\alpha_1, \alpha_2) \sigma_2} \hat{c}_{\alpha_1-x(\alpha_1, \alpha_2) \sigma_1} \quad (3.12)$$

For this reason, indicating the constrained sum over \mathbf{Q} again with a prime, the potential reads

$$\hat{V} = \frac{1}{2} \sum_{\Lambda_1 < \Lambda_2} \sum'_{\mathbf{Q}} [\tilde{v}(\mathbf{Q}) + \tilde{v}(-\mathbf{Q})] \hat{c}_{\alpha_1 \sigma_1}^\dagger \hat{c}_{\alpha_2 \sigma_2}^\dagger \hat{c}_{\alpha_2+x \sigma_2} \hat{c}_{\alpha_1-x \sigma_1}$$

or, in case \tilde{v} only depends on the modulus of the transferred momentum:

$$\hat{V} = \sum_{\Lambda_1 < \Lambda_2} \sum'_{\mathbf{Q}} \tilde{v}(Q) \hat{c}_{\alpha_1 \sigma_1}^\dagger \hat{c}_{\alpha_2 \sigma_2}^\dagger \hat{c}_{\alpha_2+x \sigma_2} \hat{c}_{\alpha_1-x \sigma_1}$$

Going back to Eq. (2.7), the contribution to the local energy of the two-body potential is then

$$\begin{aligned} \frac{1}{c_N} \langle N | \hat{V} | \psi \rangle &= \frac{1}{c_N} \sum_{N'} \langle N | \hat{V} | N' \rangle \langle N' | \psi \rangle \\ &= \frac{1}{c_N} \sum_{N'} \langle N | \sum_{\Lambda_1 < \Lambda_2} \sum'_{\mathbf{Q}} \tilde{v}(Q) \hat{c}_{\alpha_1 \sigma_1}^\dagger \hat{c}_{\alpha_2 \sigma_2}^\dagger \hat{c}_{\alpha_2+x \sigma_2} \hat{c}_{\alpha_1-x \sigma_1} | N' \rangle c_{N'} \\ &= \frac{1}{c_N} \sum_{\Lambda_1 < \Lambda_2} \sum'_{\mathbf{Q}} \tilde{v}(Q) \sum_{N'} c_{N'} \langle N | \hat{c}_{\alpha_1 \sigma_1}^\dagger \hat{c}_{\alpha_2 \sigma_2}^\dagger \hat{c}_{\alpha_2+x \sigma_2} \hat{c}_{\alpha_1-x \sigma_1} | N' \rangle \end{aligned} \quad (3.13)$$

The calculation of this expression can be simplified by noticing that the sum over $\Lambda_1 < \Lambda_2$ simply translates into a sum over the unordered pairs of Fock state indices in $|N\rangle$ associated with occupied SPOs, since the other contributions are null due to the term $\langle N | \hat{c}_{\alpha_1 \sigma_1}^\dagger \hat{c}_{\alpha_2 \sigma_2}^\dagger$.

The sum over the valid \mathbf{Q} is not trivial. Of course, it is possible to simply sum over all $\beta_1 \in [0, S-1]$, define $\mathbf{Q} = \mathbf{K}_{\alpha_1} - \mathbf{K}_{\beta_1}$ (and $x = \alpha_1 - \beta_1$) and check if $\mathbf{K}_{\beta_2} = \mathbf{K}_{\alpha_2} + \mathbf{Q}$ lies within the momentum limits; however, in some cases only few \mathbf{Q} may be valid and this method could be inefficient, thus we also consider a different approach, which we shall mention in the next Section in order not to break this discussion.

At this point, for fixed α_1, α_2 and x , the matrix element $\langle N | \hat{c}_{\alpha_1 \sigma_1}^\dagger \hat{c}_{\alpha_2 \sigma_2}^\dagger \hat{c}_{\alpha_2+x \sigma_2} \hat{c}_{\alpha_1-x \sigma_1} | N' \rangle$ is non-zero only for at most one $|N'\rangle$ that can be found by applying the creation and annihilation operators to the left on $\langle N |$, which simply translates into subtracting 1 from $n_{\alpha_1+S\sigma_1}$ and $n_{\alpha_2+S\sigma_2}$ and then adding 1 to $n_{\alpha_1-x+S\sigma_1}$ and $n_{\alpha_2+x+S\sigma_2}$. If any of the new occupation numbers after each application of a creation (annihilation) operator is not either 0 or 1, the matrix element is null. To implement this practically, one can define a method that takes as input \mathbf{n} (the array of occupation numbers defining the state N), α, σ and a parameter d equal to +1 for creation operators or -1 for annihilation operators. The method adds d to $n_{\alpha+S\sigma}$ and returns the new array \mathbf{n}' and a boolean z that checks whether $n_{\alpha+S\sigma} \in [0, 1]$ or not. Additionally, recall that applying a creation (annihilation) operator carries a phase factor as in Eq. (2.41); once again, this can easily be calculated within the creation/annihilation operator method and returned together with \mathbf{n}' and z .

Finally, for fixed α_1, α_2 and x , one also needs to compute $\tilde{v}(Q)$ and the wave function

coefficient $c_{N'}$. The calculation of the former is simple once \mathbf{Q} is known; as for the latter, implementing the creation/annihilation operator method as described above naturally returns the array \mathbf{n}' for which the coefficient needs to be computed. However, the application of $\hat{c}_{\alpha_2+x\sigma_2}$ and $\hat{c}_{\alpha_1-x\sigma_1}$ can lead to forbidden states if $n_{\alpha_1-x+S\sigma_1}$ or $n_{\alpha_2+x+S\sigma_2}$ are already occupied; here, the boolean output z of the creation/annihilation operator method allows to implement a conditional branch and calculate $\tilde{v}(\mathbf{Q})$ and $c_{N'}$ only for valid states N' , otherwise return 0, which avoids unnecessary calculation.

To summarize, the local two-body potential can be computed as follows (using the simple way of implementing the conditioned sum over \mathbf{Q}):

1. Given the array \mathbf{n} describing the state $|N\rangle = |n_0, \dots, n_{2S-1}\rangle$, find the arrays v and s of momentum and spin projection indices of the non-zero n_Λ ;
2. Set $V = 0$;
3. For $i = 0$ to $M - 1$, where M is the number of particles, do:
 - Set $\alpha_1 = v[i]$ and $\sigma_1 = s[i]$;
 - Apply the creation/annihilation operator method with α_1 and σ_1 to \mathbf{n} , storing the output phase p_1 and the new state \mathbf{n}_1 ;
 - Set a cumulative variable $C_1 = 0$;
 - For $j = i + 1$ to $M - 1$ do:
 - Set $\alpha_2 = v[j]$ and $\sigma_2 = s[j]$;
 - Apply the creation/annihilation operator method with α_2 and σ_2 to \mathbf{n}_1 , storing the output phase p_2 and the new state \mathbf{n}_2 ;
 - Set a cumulative variable $C_2 = 0$;
 - For $\beta_1 = 0$ to $S - 1$ do:
 - * Define $\mathbf{Q} = \mathbf{K}_{\alpha_1} - \mathbf{K}_{\beta_1}$ and calculate $\mathbf{K}_{\beta_2} = \mathbf{K}_{\alpha_2} + \mathbf{Q}$ using Eq. (2.37);
 - * Define $z_0 = \text{True}$ if Eq. (3.7) is verified and **False** otherwise;
 - * Define $x = \alpha_1 - \beta_1$;
 - * Apply the last two creation operators with $\alpha_2 + x, \sigma_2$ and $\alpha_1 - x, \sigma_1$ to \mathbf{n}_2 , storing the boolean outputs z_3, z_4 , the phases p_3, p_4 and the array \mathbf{n}' describing $|N'\rangle$;
 - * If z_0, z_1 and z_2 are **True**, compute $c_{N'}$ and $\tilde{v}(\mathbf{Q})$ and return their product F , otherwise set $F = 0$;
 - * Add p_3p_4F to the variable C_2 ;
 - Add p_2C_2 to C_1 ;
 - Add p_1C_1 to V ;
4. Compute c_N and return V/c_N .

Here we specifically considered a two-body potential with plane waves as SPOs (Eq. (3.1)) and a local interaction in coordinate space for simplicity; however, in general a second quantized fermionic CI two-body potential reads

$$\hat{V} = \frac{1}{2} \sum_{abij} V_{ij}^{ab} \hat{c}_a^\dagger \hat{c}_b^\dagger \hat{c}_j \hat{c}_i \quad (3.14)$$

A similar discussion to the one above could also be made for arbitrary matrix elements V_{ij}^{ab} , independently of the locality of the interaction and of the SPOs used. Thereby, this algorithm can in general be used to study a wide range of many-body systems and interactions and the generalization of the procedure described here (also to arbitrary one-body terms in the Hamiltonian) is rather simple. It shall be noted that one could also deal with the creation and annihilation operators and the related phases with operators acting on single occupation numbers using the Jordan-Wigner representation [38].

3.1.1 Faster summation over the transferred momenta

Here we discuss a different implementation of the summation over the valid transferred momenta \mathbf{Q} described in the previous Section. Given α_1 and α_2 , one can see that the constraints that define a valid \mathbf{Q} , that is

$$\begin{aligned} |K_{\alpha_1}^d - Q^d| &\leq (A-1)\delta k/2 \\ |K_{\alpha_2}^d + Q^d| &\leq (A-1)\delta k/2 \end{aligned} \quad (3.15)$$

are fulfilled if Q^d is bound from below by $Q_{min}^d = \delta k \max(-I_{\alpha_2}^d, I_{\alpha_1}^d - A + 1)$ and from above by $Q_{max}^d = \delta k \min(I_{\alpha_1}^d, A - 1 - I_{\alpha_2}^d)$, where $I_{\alpha}^d = (\lfloor \alpha/A^d \rfloor \bmod A)$ as defined in Eq. (2.37). As an example, consider the lower bound for the first condition (which becomes an upper bound due to the minus sign):

$$\begin{aligned} K_{\alpha_1}^d - Q_{min}^d &\leq \frac{A-1}{2} \delta k \\ &\rightarrow K_{\alpha_1}^d - \delta k \max(-I_{\alpha_2}^d, I_{\alpha_1}^d - A + 1) \leq \frac{A-1}{2} \delta k \\ &\rightarrow \left(I_{\alpha_1}^d - \frac{A-1}{2} \right) \delta k - \delta k \max(-I_{\alpha_2}^d, I_{\alpha_1}^d - A + 1) \leq \frac{A-1}{2} \delta k \\ &\rightarrow \left(I_{\alpha_1}^d - A + 1 \right) - \max(-I_{\alpha_2}^d, I_{\alpha_1}^d - A + 1) \leq 0 \end{aligned} \quad (3.16)$$

which is always verified. For this reason, $\ell^d = (Q_{max}^d - Q_{min}^d)/\delta k + 1$ represents the number of valid components of the transferred momentum Q^d , and $S_x = \prod_d \ell^d$ is the total number of valid \mathbf{Q} . We can now loop from 0 to $S_x - 1$, defining the valid x as

$$\begin{aligned} x &= [Q_{min}^0/\delta k + (\lfloor i/1 \rfloor \bmod \ell^0)] A^0 + [Q_{min}^1/\delta k + (\lfloor i/\ell^0 \rfloor \bmod \ell^1)] A^1 + \\ &+ [Q_{min}^2/\delta k + (\lfloor i/\ell^0 \ell^1 \rfloor \bmod \ell^2)] A^2 + \dots \end{aligned} \quad (3.17)$$

where i is the loop index. To understand the motivation behind Eq.(3.17), notice that the terms in the square brackets represent exactly the indices of the valid \mathbf{Q} components along each direction and they are combined into a transferred momentum index x using Eq.(2.38). The term in the round brackets are similar to the definition of the I_α^d : in fact, if the I_α^d represent the indices of the components d of momentum \mathbf{K}_α in a box $A \times A \times \dots$, these expression represent the same indices but for a box of size $\ell^0 \times \ell^1 \times \dots$, which is the one where the valid \mathbf{Q} is defined. This allows to directly sum over the valid x rather than all β_1 as described in the previous Section, typically bringing a decent speedup in the calculations.

3.2 Ultracold interacting Fermi gases

In this last Section we briefly discuss a possible application of the VMC algorithm in Fock space with NQSs to ultracold interacting Fermi gases. At present, the limitations of the algorithm are rather strict, thus we shall only present the results as a proof of concept for future developments and to test the implementation of the two-body potential.

Ultracold interacting Fermi gases constitute an important and interesting topic of research. On top of allowing to observe fascinating and unique quantum mechanical properties, these systems are also deeply connected with other research fields such as nuclear astrophysics [39]. Recently there has also been a rising interest towards low-dimensional Fermi gases. In general, quantum systems in one or two dimensions can manifest new behaviours that are not encountered in the 3D case [40] and, for example, studying 2D Fermi gases can also provide useful insights about condensed matter systems such as graphene [41], unconventional superconductors [42] and topological insulators [43].

The code developed in this Thesis work is applicable in an arbitrary number of dimensions, possibly enabling its applications also to 1D, 2D and 3D interacting Fermi gases, with a favorable scaling of the number of variational parameters with the number of momentum states in each direction A ($\mathcal{O}(A^{2D})$) and with the number of particles M (potentially $\mathcal{O}(1)$). For a simple proof of concept, here we test the algorithm for a repulsive Gaussian two-body potential:

$$V(\mathbf{x}_1, \mathbf{x}_2) = V(|\mathbf{x}_1 - \mathbf{x}_2|) = V(r) = 2e^{-r^2}$$

The energies obtained using the SR optimizations are listed in Tab. VIII, where they are compared against those calculated using full CI and a Diffusion Monte Carlo (DMC) algorithm in coordinate space², which is independent of the UV cutoff. As mentioned at the beginning of this Section, the tests that can be performed with the present version of the code are effectively limited by the use of the Cholesky decomposition in the SR algorithm. On top of that, the full CI procedure quickly becomes exceedingly expensive, preventing a direct comparison with the exact energies for large A^D . For these reasons, the results presented here are limited to $D = 1$ and relatively small systems (even though in one case the system is too large to be accessible in a full CI approach). We also qualitatively check the convergence towards the UV cutoff-independent energy by considering two different values of A in the $M = 6$ and $M = 5$ cases.

²From private communications.

L	D	A	M	M_{up}	NQS VMC	FCI	DMC
10	1	11	3	3	0.54094 ± 0.00023	0.54019	0.54017(1)
10	1	7	5	5	3.11468 ± 0.00013	3.11451	3.0617(2)
10	1	11	5	5	3.06615 ± 0.00015	3.06469	3.0617(2)
10	1	7	6	3	4.0022 ± 0.0071	3.9753	3.880(2)
10	1	11	6	3	3.9661 ± 0.0010	-	3.880(2)
10	1	11	7	7	8.90604 ± 0.00015	8.90389	8.8848(2)

Table VIII: Results of optimizations using the VMC algorithm with NQS for a two-body potential against full CI and DMC calculations. For the results with $A = 11$ we used NNs with layers of 32, 16 and 16 neurons and we performed 2000 optimization steps with a time step $\delta\tau$ decreasing from 0.005 to 0.0007, while for the $A = 7$ optimization the NNs featured layers of 20, 12 and 12 neurons and 2000 optimization steps were performed with $\delta\tau$ from 0.007 to 0.001. The full CI procedure was not used for the $L = 10$, $A = 11$ and $M = 6$ case due to the fact that filling the Hamiltonian matrix was too expensive in terms of computational time. Note that the DMC in coordinate space is not affected by the UV cutoff and therefore it only represents a qualitative comparison in the infinite cutoff limit for testing the correctness of the implementation of the two-body potential in the VMC and in the full CI approaches.

Even with the limitations imposed by the current implementation of the SR method, we figure the results show the correctness of the implementation of the two-body potential at least in 1D, also in regimes that are not accessible by full CI. Once the conjugate gradient solver is implemented, we will be able to test and use the code with many different interactions and in many more regimes. Finally, it is worth noting that, with the class of potentials considered here, the system features translational invariance, therefore the resulting wave function coefficients c_N are null for states $|N\rangle$ with occupied momenta \mathbf{K}_{α_i} such that $\sum_i \mathbf{K}_{\alpha_i} \neq \mathbf{0}$. Hence, for future applications of this algorithm, it may be beneficial to facilitate the learning task by only sampling such states (for instance by rejecting proposed states with $\sum_i \mathbf{K}_{\alpha_i} \neq \mathbf{0}$ a priori in the Metropolis).

Chapter 4

Conclusions and perspectives

Quantum many-body problems are ubiquitous in the physical sciences and, despite having received significant attention for many decades, they continue to manifest surprising properties. Many computational methods have been proposed for studying these systems, employing different approaches and exploiting various theoretical concepts to obtain increasingly accurate results and explore new systems and regimes. In this Thesis we discussed the development of a novel VMC algorithm which leverages the effectiveness at incorporating dynamical correlations of wave functions in Fock spaces and the flexibility of NQSs. We developed the algorithm step-by-step, describing and testing each part of the code separately. We verified the correct functioning of the VMC in different dimensions and for different numbers of particles, both with an approximate one-body HO potential and a two-body Gaussian interaction.

Combining NQSs and a wave function in Fock space allows to efficiently and effectively describe dynamical correlations and encode the complexity of many-body wave functions, with a much milder scaling of the number of variational parameters with the number of particles M and SPOs S with respect to the full CI approach, while recovering well above 99% of the energy in the cases tested. Nevertheless, in order to accurately assess the effectiveness of this representation of the wave function, further investigation on the optimal hyperparameters for the learning in different regimes is required.

A great advantage of this algorithm is represented by the fact that it can be easily generalized to different SPOs and the general expression of the two-body potential allows to deal with a great variety of interactions, including non-local effective potentials. For these reasons, the possible applications of this algorithm are several. In particular, we briefly discussed the case of ultracold Fermi gases; however, using different SPOs, this procedure may be particularly effective for quantum chemistry applications, where full CI calculations are considered as practically exact even with a finite basis set, though being limited due to the factorial scaling of the number of basis states. In this context, this algorithm may provide comparable accuracy for larger systems with respect to the ones treatable now. Additionally, combining this algorithm with SPOs such as Bloch functions or Wannier functions may be beneficial for studying solid-state systems, and in general the wave functions calculated using this technique could also be used as a variational ansatz for other QMC algorithms, such as Diffusion Monte Carlo, to further improve the accuracy of the results.

At present, the downside of this approach is that it may be rather slow and memory-expensive for large numbers of SPOs, which limits its applications considerably. However, implementing the conjugate gradient solver for the Stochastic Reconfiguration algorithm could potentially allow calculations with tens of billions of variational parameters, for instance enabling calculations for 3D Fermi gases using several momentum states in each direction.

As a final remark, it is worth mentioning that the code is highly optimized to capitalize on GPU-based architectures — we observe a factor 100 speedup compared to CPUs. Nevertheless, the present version of the code only runs on a single GPU, although work is in progress to distribute the workload across multiple GPUs.

A Markov chains and Metropolis-Hastings algorithm

In Sec. 1.2 we discuss the Variational Monte Carlo method, which requires the ability to sample configurations x following some arbitrary target probability distribution $P[X]$. This can be achieved by means of Markov chains. A Markov chain is a sequence of random variables X_1, X_2, X_3, \dots where the probability of moving from a state $X_n = x_n$ to the next one $X_{n+1} = x_{n+1}$ depends only on the present state via a transition kernel:

$$T_n(X_{n+1} = x_{n+1} | X_n = x_n, X_{n-1} = x_{n-1}, \dots, X_0 = x_0) = T_n(x_n \rightarrow x_{n+1}) \quad (\text{A.1})$$

and so the probability of finding the state x_{n+1} at step $n + 1$ is

$$P_{n+1}[x_{n+1}] = \sum_{x_n} P_n[x_n] T_n(x_n \rightarrow x_{n+1}), \quad P_{n+1}[X_{n+1}] = \mathbf{T}_n P_n[X_n] \quad (\text{A.1})$$

If the transition kernel is independent of n , the probability at each step n is given by

$$P_n[X_n] = \mathbf{T}^n P_0[X_0] \quad (\text{A.2})$$

Now, notice that if the process converges¹ to a $P_\infty[X]$, then this stationary distribution must be an eigenvector of \mathbf{T} :

$$\mathbf{T} P_\infty[X] = \lim_{n \rightarrow \infty} \mathbf{T} P_n[X_n] = \lim_{n \rightarrow \infty} P_{n+1}[X_{n+1}] = P_\infty[X] \quad (\text{A.3})$$

Therefore, if one chooses a \mathbf{T} such that some target distribution $P[X]$ is its eigenfunction with unitary eigenvalue, the Markov chain can be exploited to sample $P[X]$ starting from any probability density $P_0[X_0]$. In order to find the appropriate transition kernel for a given target distribution, we can consider the process at equilibrium, i.e. once the limiting probability distribution has been reached. In this regime, we expect that the rate at which the chain moves away from a state y is balanced by that of moving to y from any other state:

$$\sum_x P(y) T(y \rightarrow x) = \sum_x P(x) T(x \rightarrow y) \quad (\text{A.4})$$

This equation corresponds to the so-called Master Equation in the steady state. The condition in Eq. (A.4) can be enforced by imposing the more stringent *detailed balance* condition, which requires the summands to be equal:

$$P(y) T(y \rightarrow x) = P(x) T(x \rightarrow y) \quad (\text{A.5})$$

Now, we rewrite the transition kernel in a more convenient way by recasting the requirement of

¹It can be proved that a Markov chain converges to a unique probability density with eigenvalue 1 if it is irreducible (any final state can be reached from any starting state with an arbitrary number of steps) and acyclic (there are no cycles of states).

Eq. (A.5) into a requisite for an *acceptance probability* A :

$$\begin{aligned}
P(y)\tilde{T}(y \rightarrow x)A(y \rightarrow x) &= P(x)\tilde{T}(x \rightarrow y)A(x \rightarrow y) \\
\Rightarrow \frac{A(x \rightarrow y)}{A(y \rightarrow x)} &= \frac{P(y)\tilde{T}(y \rightarrow x)}{P(x)\tilde{T}(x \rightarrow y)}
\end{aligned}
\tag{A.6}$$

where \tilde{T} is a transition kernel which we are able to sample and A can be interpreted as the probability to accept a proposed transition $x \rightarrow y$. Now, if we define

$$A(x \rightarrow y) = \min \left(1, \frac{P(y)\tilde{T}(y \rightarrow x)}{P(x)\tilde{T}(x \rightarrow y)} \right)
\tag{A.7}$$

then Eq. (A.6) is satisfied; we can thus devise an algorithm that, given a state x , proposes a transition $x \rightarrow y$ via a known transition kernel \tilde{T} and accepts or rejects it with the probability in Eq. (A.7), which can be calculated since the target probability P and the kernel \tilde{T} are known. This eventually allows to generate samples from the target distribution P starting from an arbitrary state x_0 . Note that Eq. (A.7) is further simplified if the transition kernel is symmetric ($\tilde{T}(x \rightarrow y) = \tilde{T}(y \rightarrow x)$). This algorithm is known as the *Metropolis-Hastings algorithm* (the original formulation by Metropolis and coworkers was developed assuming a symmetric transition kernel; the generalization to an arbitrary transition kernel using Markov chains is due to Hastings).

B Exact discretization of the HO Hamiltonian

In this appendix we derive the correctly discretized momentum representation of the Schrödinger equation for a periodic HO potential in coordinate space for 1 particle. We start from

$$\langle \mathbf{k} | \hat{H} | \psi \rangle = \langle \mathbf{k} | (\hat{T} + \hat{V}) | \psi \rangle \quad (\text{B.1})$$

For the kinetic energy term, we can insert a completeness relation

$$\mathbb{1} = \sum_{\mathbf{k}} |\mathbf{k}\rangle \langle \mathbf{k}| \quad , \quad (\text{B.2})$$

where the sum runs over all possible (discrete) values of \mathbf{k} , to obtain

$$\sum_{\mathbf{k}'} \langle \mathbf{k} | \hat{T} | \mathbf{k}' \rangle \langle \mathbf{k}' | \psi \rangle = \sum_{\mathbf{k}'} \frac{k^2}{2} \delta(\mathbf{k} - \mathbf{k}') \psi_{\mathbf{k}'} = \frac{k^2}{2} \psi_{\mathbf{k}} \quad (\text{B.3})$$

As for the potential, since it is periodic with period L , one can write its Fourier decomposition:

$$\begin{aligned} V(\mathbf{r}) &= \sum_{\mathbf{k}} c_{\mathbf{k}} \exp(i\mathbf{k} \cdot \mathbf{r}) \\ c_{\mathbf{k}} &= \frac{1}{L^D} \int_R V(\mathbf{r}) \exp(-i\mathbf{k} \cdot \mathbf{r}) d\mathbf{r} \end{aligned} \quad (\text{B.4})$$

where $V(\mathbf{r}) = r^2/2$ with periodicity L along each direction, $\mathbf{k} = i\delta k$ as in Eq. (2.1) and R is a hypercube of size L centered in $\mathbf{0}$. Substituting $V(\mathbf{r}) = r^2/2$ in the expression above, one obtains:

$$c_{\mathbf{k}} = \begin{cases} \sum_i \frac{(-)^{k_i/\delta k}}{k_i^2} \prod_{j \neq i} \delta_{k_j, 0} & \text{if } \mathbf{k} \neq \mathbf{0} \\ \frac{DL^2}{24} & \text{otherwise} \end{cases} \quad (\text{B.5})$$

Now, in Eq. (B.1), one can add a completeness relation in the position space and one in momentum space as

$$\langle \mathbf{k} | \hat{V} | \psi \rangle = \sum_{\mathbf{k}'} \int_R d^D r \langle \mathbf{k} | \mathbf{r} \rangle \langle \mathbf{r} | \hat{V} | \mathbf{r} \rangle \langle \mathbf{r} | \mathbf{k}' \rangle \langle \mathbf{k}' | \psi \rangle \quad (\text{B.6})$$

Finally, substituting Eq. (B.4) and remembering $\langle \mathbf{r} | \mathbf{k} \rangle = L^{-D/2} \exp(i\mathbf{k} \cdot \mathbf{r})$, one obtains

$$\begin{aligned} \langle \mathbf{k} | \hat{V} | \psi \rangle &= \sum_{\mathbf{k}', \mathbf{k}''} \frac{1}{L^D} \int_R d\mathbf{r} e^{i(\mathbf{k}' + \mathbf{k}'' - \mathbf{k}) \cdot \mathbf{r}} c_{\mathbf{k}''} \psi_{\mathbf{k}'} \\ &= \sum_{\mathbf{k}', \mathbf{k}''} \delta(\mathbf{k}' + \mathbf{k}'' - \mathbf{k}) c_{\mathbf{k}''} \psi_{\mathbf{k}'} \\ &= \sum_{\mathbf{k}'} c_{\mathbf{k} - \mathbf{k}'} \psi_{\mathbf{k}'} \end{aligned} \quad (\text{B.7})$$

where we have used

$$\int_{-L/2}^{L/2} e^{ikx} dx = L \operatorname{sinc} \left(\frac{kL}{2} \right) = L \operatorname{sinc}(\pi n) = L \delta_{n,0} \quad (\text{B.8})$$

The Schrödinger equation in momentum space then reads:

$$\frac{k^2}{2}\psi_{\mathbf{k}} + \sum_{\mathbf{k}'} c_{\mathbf{k}-\mathbf{k}'} \psi_{\mathbf{k}'} = E\psi_{\mathbf{k}} \quad \text{where} \quad c_{\mathbf{q}} = \begin{cases} \sum_i \frac{(-)^{q_i/\delta k}}{q_i^2} \prod_{j \neq i} \delta_{q_j, 0} & \text{if } \mathbf{q} \neq \mathbf{0} \\ \frac{DL^2}{24} & \text{otherwise} \end{cases} \quad (\text{B.9})$$

The presence in the non-local kinetic term makes this problem rather complicated for a simple starting implementation, thus we use a coarser approximation of the kinetic term using finite differences. Nevertheless, the implementation of this potential in the context of the algorithm discussed in this work is not particularly complicated and, at least theoretically, it can yield accurate results for a HO potential in coordinate space also using few momentum states as shown in Fig. A.1.

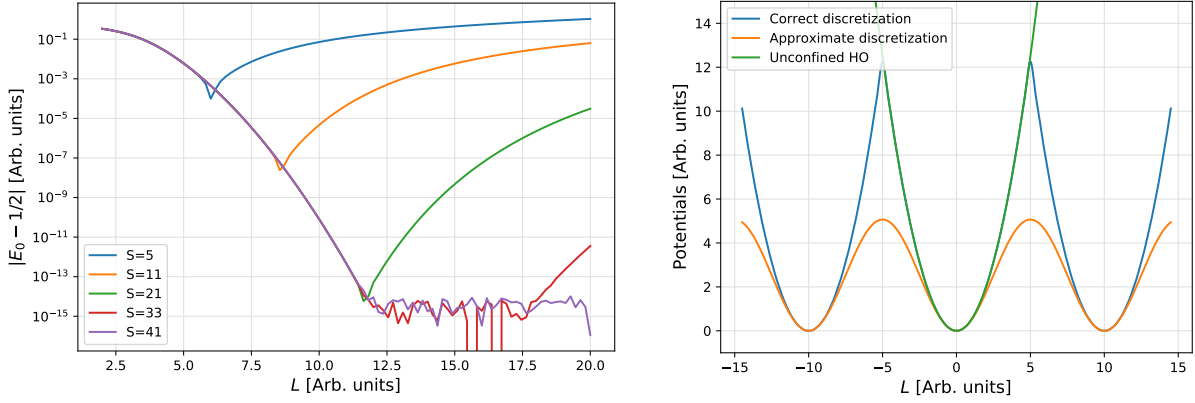


FIG A.1: *Left:* Comparison between the lowest eigenenergy of the correctly discretized Schrödinger equation and the true ground-state energy of a 1D HO for different sizes of the periodic box L and different numbers of momentum states S . The discrepancies on the left and on the right of the plot are due to the infrared and ultraviolet cutoffs respectively (note that, for constant L , the UV cutoff depends on the number of states S via $k_{max} = \pi(S - 1)/L$.) *Right:* Comparison between the correct discretization and the FD approximation of the HO potential in coordinate space.

Appendix C: Software and hardware specifications

The code was developed in Python using the JAX library (<https://github.com/google/jax>), which implements a GPU-backed version of most of the commonly used Python libraries such as NumPy and SciPy, allowing for efficient parallelization of the programs on multiple GPU cores. Below we list the advantages of using such a library:

- It provides just-in-time compilation of functions using XLA (Accelerated Linear Algebra) (<https://www.tensorflow.org/xla>).
- It allows for automatic differentiation of native Python and NumPy code via Autograd (<https://github.com/hips/autograd>); this is particularly useful for the calculation of the gradient in Eq. (2.28), especially when using complex parametrizations of the wave function such as the NQs.
- It implements a vectorizing map, `vmap()`, which allows to deal with batched inputs efficiently. This is of great advantage in many cases, such as the parallel application of the Metropolis algorithm to multiple walkers and in general the calculation of the wave function and the local quantities for batched Fock states $|N\rangle$.
- It contains multiple packages to easily deal with machine learning tasks as well as handily manage complex variable structures.

For most of the optimizations we used the Marconi100 and DGX clusters at SCAI (Super-Computing Applications and Innovation), the High Performance Computing (HPC) department of CINECA. The hardware and performance specifications of the clusters can be found at <https://www.hpc.cineca.it/content/hardware>.

Bibliography

- [1] See for instance M. H. Kalos and P. A. Whitlock, *Monte Carlo methods* (Wiley-VCH, 2009); J. B. Anderson, *Quantum Monte Carlo: origins, development, applications*, Rev. Comp. Chem. 13, 133 (2007); W. M. C. Foulkes, L. Mitas, R. J. Needs, and G. Rajagopal, *Quantum Monte Carlo simulations of solids*, Rev. Mod. Phys. 73, 33 (2001)
- [2] J. C. Slater, *The Theory of Complex Spectra*, Phys. Rev. 34 (2): 1293-1322, (1929)
- [3] D. R. Hartree, *The Wave Mechanics of an Atom with a Non-Coulomb Central Field. Part I. Theory and Methods*, Proc. Cambridge Philos. Soc. 24, 89, (1928); V. Fock, *Näherungsmethode zur Lösung des quantenmechanischen Mehrkörperproblems*, Z. Phys. 61, 126 (1930); J. C. Slater, *A simplification of the Hartree-Fock method*, Phys. Rev. 81, 385 (1951)
- [4] W. Kohn and L. J. Sham, *Self-Consistent Equations Including Exchange and Correlation Effects* Phys. Rev. 140, A1133 (1965)
- [5] P. Hohenberg and W. Kohn, *Inhomogeneous Electron Gas*, Phys. Rev. 136, 3864 (1964)
- [6] J. P. Perdew and A. Zunger, *Self-interaction correction to density-functional approximations for many-electron systems* Phys. Rev. 8 23, 5048 (1981)
- [7] See for instance C. D. Sherill and H. F. Schaefer III, *The Configuration Interaction Method: Advances in Highly Correlated Approaches*, Adv. Quant. Chem. 34 143 (1999)
- [8] F. Coester, *Bound states of a many-particle system*, Nucl. Phys. 1, 421 (1958); F. Coester and H. Kümmel, *Short-range correlations in nuclear wave functions*, Nucl. Phys. 17, 477 (1960)
- [9] R. J. Bartlett and M. Musial, *Coupled-cluster theory in quantum chemistry*, Rev. Mod. Phys. 79 291-352 (2007)
- [10] G. H. Booth, A. J. A. Thom, and A. Alavi, *Fermion Monte Carlo without fixed nodes: A game of life, death, and annihilation in Slater determinant space*, J. Chem. Phys. 131, 054106 (2009)
- [11] A. Mukherjee, Y. Alhassid, *Configuration-interaction Monte Carlo method and its application to the trapped unitary Fermi gas*, Phys. Rev. A 88, 053622 (2013)
- [12] A. Roggero, A. Mukherjee, F. Pederiva, *Quantum Monte Carlo with coupled-cluster wave functions*, Phys. Rev. B 88, 115138 (2013)

- [13] A. Roggero, A. Mukherjee, F. Pederiva, *Quantum Monte Carlo calculations of neutron matter with nonlocal chiral interactions*, Phys. Rev. Lett. 112, 221103 (2014)
- [14] See for instance E. Neuscamman, *The Jastrow antisymmetric geminal power in Hilbert space: Theory, benchmarking, and application to a novel transition state*. J. Chem. Phys. 139, 194105 (2013); M. Kurita, Y. Yamaji, S. Morita, M. and Imada, *Variational Monte Carlo method in the presence of spin-orbit interaction and its application to Kitaev and Kitaev-Heisenberg models*, Phys. Rev. B 92, 035122 (2015); I. Sabzevari and S. Sharma, *Faster and lower scaling orbital-space variational monte carlo*, arXiv preprint arXiv:1807.10633 (2018)
- [15] N. Metropolis and S. Ulam, *The Monte Carlo Method*, Journal of American Statistical Association, 44, 335-342 (1949)
- [16] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, *Equation of state calculations by fast computing machines*, J. Chem. Phys., 21, 1087 (1953); W. K. Hastings, *Monte Carlo sampling methods using Markov chains and their applications*, Biometrika, 57, 97-109 (1970)
- [17] R. J. Jastrow, *Many-body problem with strong forces*, Phys. Rev. 98, 1479 (1955)
- [18] G. Carleo and M. Troyer, *Solving the quantum many-body problem with artificial neural networks*, Science 355, 602-606 (2017)
- [19] X. Liang, W.-Y. Liu, P.-Z. Lin, G.-C. Guo, Y.-S. Zhang, and L. He, *Solving frustrated quantum many-particle models with convolutional neural networks*, Physical Review B 98 (10), 104426 (2018)
- [20] M. Hibat-Allah, M. Ganahl, L. E. Hayward, R. G. Melko, and J. Carrasquilla, *Recurrent neural network wave functions*, Phys. Rev. Res. 2, 023358 (2020)
- [21] K. Hornik, *Approximation capabilities of multilayer feedforward networks*, Neural Networks 4 251-257 (1991)
- [22] K. Choo, G. Carleo, N. Regnault, and T. Neupert, *Symmetries and many-body excitations with neural-network quantum states*, Phys. Rev. Lett. 121 (16), 167204 (2018)
- [23] H. Saito, *Method to Solve Quantum Few-Body Problems with Artificial Neural Networks*, Journal of the Physical Society of Japan 87 (7), 074002 (2018)
- [24] J. Han, L. Zhang, and W. E, *Solving many-electron Schrödinger equation using deep neural networks*, J. Comput. Phys. 399, 108929 11 (2019)
- [25] D. Luo and B. K. Clark, *Backflow transformations via neural networks for quantum many-body wave functions*, Phys. Rev. Lett. 122, 226401 (2019)
- [26] K. Choo, A. Mezzacapo, and G. Carleo, *Fermionic neural-network states for ab-initio electronic structure*, Nat. Commun., 11:2368 (2020)

- [27] C. Adams, G. Carleo, A. Lovato, and N. Rocco, *Variational Monte Carlo calculations of $A \leq 4$ nuclei with an artificial neural-network correlator ansatz*, Phys. Rev. Lett. 127(2), 022502 (2021)
- [28] M. Troyer and U.-J. Wiese, *Computational complexity and fundamental limitations to Fermionic quantum Monte Carlo simulations*, Phys. Rev. Lett. 94, 170201 (2005)
- [29] D.-L. Deng, X. Li, and S. Das Sarma, *Quantum entanglement in neural network states*, Phys. Rev. X 7 (2), 021021 (2017)
- [30] S. Ruder, *An overview of gradient descent optimization algorithms*, arXiv preprint arXiv:1609.04747, (2016)
- [31] N. Qian, *On the momentum term in gradient descent learning algorithms*, Neural Networks 12 (1) 145-151 (1999)
- [32] J. Duchi, E. Hazan, and Y. Singer, *Adaptive subgradient methods for online learning and stochastic optimization*, Journal of Machine Learning Research, 12:2121-2159, (2011)
- [33] T. Tieleman and G. Hinton, “Lecture 6.5—RmsProp: Divide the gradient by a running average of its recent magnitude”, COURSERA: Neural Networks for Machine Learning (2012)
- [34] D. P. Kingma and J. L. Ba, *Adam: a Method for Stochastic Optimization*, International Conference on Learning Representations, pages 1–13, (2015)
- [35] S. Sorella, *Green Function Monte Carlo with Stochastic Reconfiguration*, Phys. Rev. Lett. 80, 4558 (1998)
- [36] S. Sorella, *Wave function optimization in the variational Monte Carlo method*, Phys. Rev. B 71, 241103 (2005)
- [37] J. Stokes, J. Izaac, N. Killoran, and G. Carleo, *Quantum natural gradient*, Quantum 4, 269 (2020)
- [38] P. Jordan and E. Wigner, *About the Pauli exclusion principle*, Z. Phys. 47, 631 (1928)
- [39] A. Gezerlis and J. Carlson, *Strongly paired fermions: Cold atoms and neutron matter*, Phys. Rev. C 77 032801 (2008)
- [40] See for instance N. D. Mermin and H. Wagner, *Absence of ferromagnetism or antiferromagnetism in one-or two-dimensional isotropic Heisenberg models*, Phys. Rev. Lett. 17, 1133 (1966); P. C. Hohenberg, *Existence of long-range order in one and two dimensions*, Phys. Rev. 158, 383 (1966)
- [41] A. K. Geim and K. S. Novoselov, *The rise of graphene*, Nature Mat. 6, 183 (2007)
- [42] E. Dagotto, *Correlated electrons in high-temperature superconductors*, Rev. Mod. Phys. 66, 763 (1994)
- [43] M. Z. Hasan and C. L. Kane, *Colloquium: topological insulators*, Rev. Mod. Phys. 82, 3045 (2010)