



# Quantum Monte Carlo: Basics, Approaches and Examples

Ricardo Almada Monter, Chemistry Department, University of Chicago.

## Abstract

Solving the Schrödinger equation for the many body problem is central in the development of quantum mechanics and for the study of atoms and molecules. In the past there have been different approximations addressing this problem: From Hartree Fock Theory to Configuration Interaction. Most of these methods rely on the physical approximation of the wave function to solve the Schrödinger equation. Quantum Monte Carlo uses a trial wave function that is used for modeling the integral of the energy through sampling across all the integral space. In the Variational Quantum Monte Carlo, the Schrödinger equation is modified to give an integral of a local energy and a probability of density that is then solved through the Metropolis algorithm. In the Diffusion Monte Carlo branch, the Schrödinger equation is written as a diffusion equation in an imaginary time and the solution is given by taking the asymptotic limit at high values of this time. In this work, some results obtained from different Quantum Monte Carlo Methods (from references) are compared with those obtained with other methods.

## Introduction

The many body problem is central in the development of modern quantum mechanics [1,2]. For a multielectron system with  $N$  electrons and  $M$  nuclei under the Born-Oppenheimer approximation, the Hamiltonian of the Schrödinger Equation (in atomic units) is given by [1]:

$$H = -\sum_{i=1}^N \left( \frac{1}{2} \nabla_i^2 \right) - \sum_{i=1}^N \sum_{A=1}^M \left( \frac{Z_A}{r_{iA}} \right) + \sum_{i=1}^N \sum_{j>i}^N \left( \frac{1}{r_{ij}} \right)$$

Where  $A$  stands for a certain Nuclei and  $i$  for an electron. The first approach to solve the many body Schrödinger Equation was proposed independently by Hartree and Fock. In their view, a set of orthonormal orbital functions  $\phi(r,s)$  is introduced and optimized to obtain a trial wave function that consists of wedge product of these optimized orbitals.

$$\psi(1,2,3,\dots,N) = \phi_1(1) \wedge \phi_2(2) \wedge \phi_3(3) \wedge \dots \wedge \phi_n(N)$$

This approach takes as an approximation that electrons are treated as independent and therefore ignores the correlation energy. Further approaches were later proposed to include this energy in the calculations. In the Configuration Interaction view, a trial wave function consists of a linear combination of determinants and the energy of the system is calculated by obtaining the linear coefficients that minimize the energy of the system [2]:

$$\begin{aligned} \Psi(1, 2, 3, \dots, N) = & C_0 \Psi_0 + \sum_{ar} C_a^r \Psi_a^r + \sum_{ab} C_{ab}^{rs} \Psi_{ab}^{rs} + \left(\frac{1}{2}\right)^2 \sum_{abc} C_{abc}^{rst} \Psi_{abc}^{rst} \\ & + \left(\frac{1}{6}\right)^2 \sum_{abcd} C_{abcd}^{rstu} \Psi_{abcd}^{rstu} + \dots \end{aligned}$$

Where the first term of the last equation corresponds to a Hartree-Fock optimized trial wave function and the second term corresponds to a single excitation wave function, and the third term to the double excitation and so on. Other approaches were later introduced to address problems that may come from the Configuration Interaction view.

So far, some approximations were introduced to solve the Schrödinger Equation through definitions of trial wave functions that could minimize the energy of the system. In Quantum Monte Carlo Methods, the purpose is to solve the time-dependent Schrödinger Equation  $H\Psi = E\Psi$  approximating the wave function as a Monte Carlo Random Walk (Therefore, proposing a trial function that comes from a Monte Carlo Calculation). In the next sections of these document, the general idea, different implementations, and some examples will be given of Quantum Monte Carlo [3].

## Theory/Methodology

### Variational Quantum Monte Carlo

As the name indicates the Variational Quantum Monte Carlo uses the fact that a variational wave function  $\phi$  can be optimized to obtain a quota of the energy [5]:

$$E[\alpha_i] = \frac{\langle \phi | H | \phi \rangle}{\langle \phi | \phi \rangle} \geq E_0$$

The essence of this method is to compute the expectation of the energy as a statistical average over configurations sampled according to the square of the wave function[6]. As such, the last expression is given by:

$$\begin{aligned} E[\alpha_i] &= \frac{\langle \phi | H | \phi \rangle}{\langle \phi | \phi \rangle} = \frac{\int \phi^*(r) H \phi(r) \, dr}{\int \phi^*(r) \phi(r) \, dr} = \frac{\int \phi(r)^* \phi(r) (\phi^{-1}(r) H \phi(r)) \, dr}{\int \phi^*(r) \phi(r) \, dr} \\ E[\alpha_i] &= \frac{\int |\phi(r)|^2 (\phi^{-1}(r) H \phi(r)) \, dr}{\int |\phi(r)|^2 \, dr} = \int E_{Local}(r) W(r) \, dr = \frac{1}{M} \sum_l^n E_{Local}(r_l) \end{aligned}$$

$$\text{where } : E_{Local}(r) = \phi^{-1}(r) H \phi(r)$$

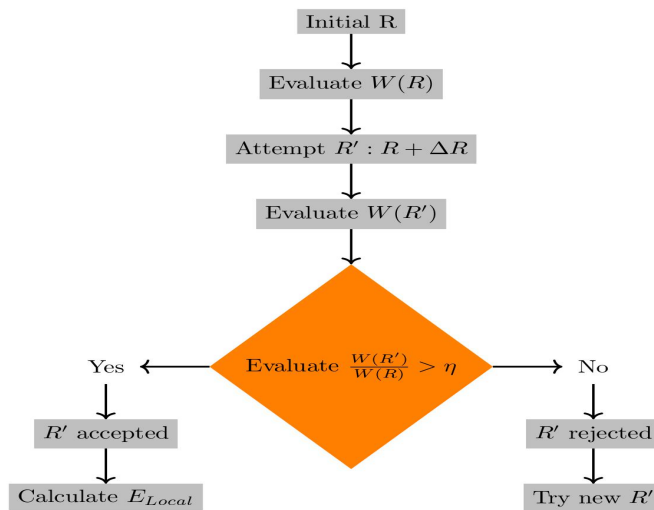
$$W(r) = \frac{|\phi(r)|^2}{\int |\phi(r)|^2 dr}$$

From the last equation is clear that the energy can be calculated as an integral of the energy weighted by the probability density. The above integral can be model by sampling the function point wise within the whole space [2]. In this part, the model depends in the building the best trial wave function possible. The most common Monte Carlo Algorithm to address the calculation of this integral is the Metropolis Algorithm. This consists in sampling the configuration of the positions in the wave function such that the move from one position (R) to another position (R') is accepted if the ratio of the transition rates satisfies:

$$\frac{T(R \rightarrow R')}{T(R' \rightarrow R)} = \frac{W(R')}{W(R)} > \eta \text{ where } \eta \in [1, 0]$$

In simple words, the metropolis algorithm creates a set of R positions. To evaluate the integral, we first randomly select a configuration R and then evaluate W(R). Next the new configuration R' is defined as R' = R + ΔR where ΔR is a vector that usually depends on a value h that is determined from the acceptance rate of the transition or walker. As such, higher h values result in smaller acceptance probability. Then, the ratio of the distribution function at the new configuration W(R') and the old configuration are computed W(R), if the ratio is higher than η the new configuration is accepted, otherwise the old configuration is the new configuration. This method is performed iterative at M "partitions" until the desired result [5]. A descriptive summary of the Metropolis Algorithm and its application in Quantum Monte Carlo is shown in Figure 1. With Variational Quantum Monte Carlo, other matrix elements values from an operator A can be computed in a similar way by [6]:

$$\langle \phi_i | A | \phi_j \rangle = \frac{1}{M} \sum_l^M \frac{\phi_i(r_l) \phi_j(r_l)}{P(r_l)} \frac{A \phi_j(r_l)}{\phi_j(r_l)}$$

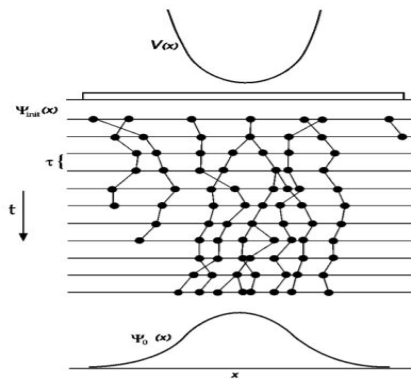


**Figure 1.** Diagram for a Variational Quantum Monte Carlo (Adapted from [5], [9])

[>

## Diffusion Monte Carlo

As stated before, in the Variational Quantum Monte Carlo our predictions of the many body system will always depend on the trial function that we initially settle for the ground state. In general, we do not always have at hand approximations or information about the form of the wave function at ground state. Therefore, VQMC is limited to this problem. The general idea of Diffusion Monte Carlo is to map the time dependent Schrödinger equation into an imaginary time diffusion equation [4,5,6,7] to improve the initial trial function that we propose (a brief schematic of this idea is given in Figure 2).



**Figure 2.** Visual Example of the Walkers in DQMC (taken from [8]) .

The time dependent Schrödinger equation is given by:

$$i\hbar \frac{d\Psi(r, t)}{d t} = H\Psi(r, t)$$

Considering a general Hamiltonian, the equation can also be written into a diffusion equation given by:

$$\frac{d\Psi(r, t)}{d t} = - \left( -\frac{1}{2} \nabla^2 + V(R) - E_R \right) \Psi(r, t)$$

Where  $E_R$  stands for a reference energy, usually the zero point of the system energy and  $\tau$  stands for an imaginary time  $\tau = \frac{it}{\hbar}$ . The equation is solved numerically by a kinetic branching term (given by the potential energy of the system) and a diffusion term (given by the kinetic energy of the system). The first refers to the branching of electrons positions relating to their multiplication and the second refers to the diffusion of electrons in the imaginary time. A common choice for the wave function at imaginary time 0 is  $\phi(r)$  that comes from VQM. Rewriting the diffusion equation in terms in its integral form.

$$\Psi(R, t + \tau) = \int G(R \rightarrow R', \tau) \Psi(R, t) dR'$$

Where:  $G(R \rightarrow R', \tau)$  is a Green function that also follows the diffusion equation

$$G(R \rightarrow R', \tau) = \langle R | e^{-\tau(H - E_R)} | R' \rangle$$

The Green function can be expressed as:

$$G(R \rightarrow R', \tau) = \sum_i \Psi_i(R) e^{-\tau(E_i - E_R)} \Psi_i^*(R')$$

Therefore, the time dependent exact wave function can be expanded with power series of the eigenfunctions of the Hamiltonian [4]:

$$\phi(r, t) = \sum_{i=0}^{\infty} \phi_i(r) e^{-(E_i - E_0)\tau}$$

This equation will decay to the ground state function with times that tend to infinity.

Both VMQ and DQM were originally developed for bosonic systems, the Fixed Node DMC is an alternative to deal with fermionic systems. A trial wave function is a function of  $3N$  variables and the trial node surface is  $3N-1$  dimensional surface on which it changes sign, This method ensures that the ground state wave function produce will follow the antisymmetric of a fermionic system and allows different nodes of the wave function to be determined separately by Monte Carlo simulations [8]. This means that in the many body problem, if nodes are known exactly absorbing barriers of the walkers may be placed everywhere on the nodal surface, dividing up the configuration space into disjoint nodal pockets that can carry parallel Monte Carlo simulations.

[>

## Trial Wave Functions for QMC

None of the methods of Quantum Monte Carlo generates a wave function of its own. But the accuracy and efficiency of QMC depends on the trial function used because the repeated evaluation of the trial function is also the most demanding part of the calculation. Although there are no restrictions on the type of trial wave function to be used, it must follow two basic conditions, that the trial wave function and its gradient must be continuous. In most codes typically use the Slater-Jastrow wave function:

$$\Psi(R) = e^{J(R)} \Psi_s(R)$$

Where  $\Psi_s(R)$  (Slater part) comes from a Slater determinants of single particles orbitals for  $\alpha$  and  $\beta$  electrons and the exponential part is a Jastrow correlation factor that is a representation of the correlation of the system.

[>  
=>

# Applications

In this section a set of molecules and atoms will be calculated with other methods like Quantum Monte Carlo and later will be compared with those reported in the literature for Quantum Monte Carlo. For this purpose, the coordinates of equilibrium were taken from the Computational Chemistry Comparison Benchmark Data Base [8]. The atoms to be calculated are: He, Li, Be, B, C, N, O, F, Ne, Na, Mg, Al, Si, P, S, Cl, Ar (Z<19).

```
> Digits := 15;
                                     Digits := 15                                (4.1)
```

```
> with(QuantumChemistry);
[AOLabels, ActiveSpaceCI, ActiveSpaceSCF, AtomicData, BondAngles, BondDistances, Charges, (4.2)
ChargesPlot, ContractedSchrodinger, CorrelationEnergy, CoupledCluster,
DensityFunctional, DensityPlot3D, Dipole, DipolePlot, Energy, ExcitationEnergies,
ExcitationSpectra, ExcitationSpectraPlot, ExcitedStateEnergies, ExcitedStateSpins, FullCI,
GeometryOptimization, HartreeFock, Interactive, Isotopes, MOCoefficients, MODiagram,
MOEnergies, MOIntegrals, MOOccupations, MOOccupationsPlot, MOSymmetries, MP2,
MolecularData, MolecularDictionary, MolecularGeometry, NuclearEnergy,
NuclearGradient, OscillatorStrengths, Parametric2RDM, PlotMolecule, Populations,
Purify2RDM, RDM1, RDM2, RTM1, ReadXYZ, Restore, Save, SaveXYZ, SearchBasisSets,
SearchFunctionals, SkeletalStructure, Thermodynamics, TransitionDipolePlot,
TransitionDipoles, TransitionOrbitalPlot, TransitionOrbitals, Variational2RDM,
VibrationalModeAnimation, VibrationalModes, Video]
```

```
> atoms := [[["He", 0, 0, 0]], [["Li", 0, 0, 0]], [["Be", 0, 0, 0]], [["B", 0, 0, 0]], [["C", 0, 0,
0]], [["N", 0, 0, 0]], [["O", 0, 0, 0]], [["F", 0, 0, 0]], [["Ne", 0, 0, 0]], [["Na", 0, 0, 0]],
[["Mg", 0, 0, 0]], [["Al", 0, 0, 0]], [["Si", 0, 0, 0]], [["P", 0, 0, 0]], [["S", 0, 0, 0]],
[["Cl", 0, 0, 0]], [["Ar", 0, 0, 0]];
atoms := [[["He", 0, 0, 0]], [["Li", 0, 0, 0]], [["Be", 0, 0, 0]], [["B", 0, 0, 0]], [["C", 0, 0, (4.3)
0]], [["N", 0, 0, 0]], [["O", 0, 0, 0]], [["F", 0, 0, 0]], [["Ne", 0, 0, 0]], [["Na", 0, 0, 0]],
[["Mg", 0, 0, 0]], [["Al", 0, 0, 0]], [["Si", 0, 0, 0]], [["P", 0, 0, 0]], [["S", 0, 0, 0]],
[["Cl", 0, 0, 0]], [["Ar", 0, 0, 0]]]
```

```
> Energies_atoms_hf := [seq(Energy(atom, method = HartreeFock, basis = "def2-qzvp"), atom
in atoms)];
Energies_atoms_hf := [-2.86162484, -7.43271513, -14.57300093, -24.52908527, (4.4)
-37.60468659, -54.26386531, -74.69098682, -99.41105404, -128.54684910,
-161.85875239, -199.61448789, -241.87670379, -288.79779412, -340.62797459,
-397.42860895, -459.48362091, -526.81719490]
```

```
> Energies_atoms_dft := [seq(DensityFunctional(atom, basis = "def2-qzvp", xc
="B3LYP")[e_tot], atom in atoms)];
Energies_atoms_dft := [-2.90801957, -7.48253216, -14.65935693, -24.64647904, (4.5)
```

$-37.77468159, -54.47874929, -74.96923957, -99.73981911, -128.94187530,$   
 $-162.25712523, -200.05451613, -242.34468765, -289.30872805, -341.17354233,$   
 $-398.02341940, -460.11487969, -527.49721427]$

$\triangleright$   $Energies\_mp2\_atoms := [seq(MP2(atom, basis = "def2-qzvp")[e\_tot], atom \text{ in } atoms)];$   
 $Energies\_mp2\_atoms := [-2.89704532, -7.49693694, -14.63496000, -24.64139412,$  **(4.6)**  
 $-37.71671134, -54.46904373, -74.89487571, -99.72895451, -128.86037547,$   
 $-162.04339118, -199.80781182, -242.14231896, -289.07310408, -340.99135173,$   
 $-397.78920106, -459.93278007, -527.26351909]$

$\triangleright$   
 Th summary of the above calculations is presented in the following table (The exact values come from the reference [9], the Quantum Monte Carlo values are from reference [10]).

Atom	Exact Energy	VMC Energy	FN-DMC	HF	DFT (B3LYP)	MP2
He	-2.90372	-2.90364	-2.90371	-2.86162484	-2.90801957	-2.89704532
Li	-7.47806	-7.47732	-7.47802	-7.43271513	-7.48253216	-7.49693694
Be	-14.66736	-14.64320	-14.65685	-14.57300093	-14.65935693	-14.63496000
B	-24.65391	-24.63440	-24.63999	-24.52908527	-24.64647901	-24.64139412
C	-37.8450	-37.8126	-37.8296	-37.60468659	-37.77468163	-37.71671134
N	-54.5892	-54.5612	-54.5761	-54.26386531	-54.47874940	-54.46904373
O	-75.0673	-75.0288	-75.0528	-74.69098682	-74.96923971	-74.89487571
F	-99.7339	-99.6963	-99.7172	-99.41105404	-99.73981922	-99.72895451
Ne	-128.9376	-128.9002	-128.9232	-128.5468491 0	-128.9418753 0	-128.8603754 7
Na	-162.2546	-162.2190	-162.2393	-161.8587523 9	-162.2571252 3	-162.0433911 8
Mg	-200.053	-199.988	-200.033	-199.6144878 9	-200.0545161 3	-199.8078118 2
Al	-242.346	-242.270	-242.322	-241.8767037 9	-242.3446876 5	-242.1423189 6
Si	-289.359	-289.273	-289.329	-288.7977941 2	-289.3087280 8	-289.0731040 8
P	-341.259	-341.146	-341.226	-340.6279745 9	-341.1735423 3	-340.9913517 3
S	-398.110	-398.012	-398.070	-397.4286089 5	-398.0234194 0	-397.7892010 6
Cl	-460.148	-460.028	-460.099	-459.4836209 1	-460.1148796 9	-459.9327800 7
Ar	-527.540	-527.405	-527.485	-526.8171949 0	-527.4972142 7	-527.2635190 9

```

[
[
> Energies_atoms_exact :=
-2.90372
-7.47806
-14.66736
-24.65391
-37.8450
-54.5892
-75.0673
-99.7339
-128.9376 ;
-162.2546
-200.053
-242.346
-289.359
-341.259
-398.110
-460.148
-527.540
]
]

```

†

*Energies\_atoms\_exact* :=

```

[
-2.90372000
-7.47806000
-14.66736000
-24.65391000
-37.84500000
-54.58920000
⋮
]

```

17 element Vector[column]

(4.7)



> *Energies\_atoms\_VMC* :=

-2.90364
-7.47732
-14.64320
-24.63440
-37.8126
-54.5612
-75.0288
-99.6963
-128.9002
-162.2190
-199.988
-242.270
-289.273
-341.146
-398.012
-460.028
-527.405

*Energies\_atoms\_VMC* :=

-2.90364000
-7.47732000
-14.64320000
-24.63440000
-37.81260000
-54.56120000
⋮

17 element Vector[column]

(4.8)

```

> Energies_atoms_fn :=

```

$$\begin{bmatrix} -2.90371 \\ -7.47802 \\ -14.65685 \\ -24.63999 \\ -37.8296 \\ -54.5761 \\ -75.0528 \\ -99.7172 \\ -128.9232 \\ -162.2393 \\ -200.033 \\ -242.322 \\ -289.329 \\ -341.226 \\ -398.070 \\ -460.099 \\ -527.485 \end{bmatrix};$$

$$Energies\_atoms\_fn := \begin{bmatrix} -2.90371000 \\ -7.47802000 \\ -14.65685000 \\ -24.63999000 \\ -37.82960000 \\ -54.57610000 \\ \vdots \end{bmatrix} \quad (4.9)$$

17 element Vector[column]

```

> labels_ := Vector([2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18]);

```

$$labels\_ := \begin{bmatrix} 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ \vdots \end{bmatrix} \quad (4.10)$$

17 element Vector[column]

```

> ea_d_hf := <<(labels_) | (abs(Energies_atoms_exact - Vector(Energies_atoms_hf)) ) >>;

```

$$ea\_d\_hf := \begin{bmatrix} 2 & 0.04209516 \\ 3 & 0.04534487 \\ 4 & 0.09435907 \\ 5 & 0.12482473 \\ 6 & 0.24031341 \\ 7 & 0.32533469 \\ \vdots & \vdots \end{bmatrix} \quad (4.11)$$

17 × 2 Matrix

>  $ea\_d\_dft := \langle \langle labels\_ \rangle | \langle abs(Energies\_atoms\_exact - Vector(Energies\_atoms\_dft)) \rangle \rangle;$

$$ea\_d\_dft := \begin{bmatrix} 2 & 0.00429957 \\ 3 & 0.00447216 \\ 4 & 0.00800307 \\ 5 & 0.00743096 \\ 6 & 0.07031841 \\ 7 & 0.11045071 \\ \vdots & \vdots \end{bmatrix} \quad (4.12)$$

17 × 2 Matrix

>  $ea\_d\_mp2 := \langle \langle labels\_ \rangle | \langle abs(Energies\_atoms\_exact - Vector(Energies\_mp2\_atoms)) \rangle \rangle;$

$$ea\_d\_mp2 := \begin{bmatrix} 2 & 0.00667468 \\ 3 & 0.01887694 \\ 4 & 0.03240000 \\ 5 & 0.01251588 \\ 6 & 0.12828866 \\ 7 & 0.12015627 \\ \vdots & \vdots \end{bmatrix} \quad (4.13)$$

17 × 2 Matrix

>  $ea\_d\_vmc := \langle \langle labels\_ \rangle | \langle abs(Energies\_atoms\_exact - Vector(Energies\_atoms\_VMC)) \rangle \rangle;$

(4.14)

$$ea\_d\_vmc := \begin{bmatrix} 2 & 0.00008000 \\ 3 & 0.00074000 \\ 4 & 0.02416000 \\ 5 & 0.01951000 \\ 6 & 0.03240000 \\ 7 & 0.02800000 \\ \vdots & \vdots \end{bmatrix} \quad (4.14)$$

17 × 2 Matrix

>  $ea\_d\_fn := \langle \langle labels\_ \rangle | \langle abs(Energies\_atoms\_exact - Vector(Energies\_atoms\_fn)) \rangle \rangle;$

$$ea\_d\_fn := \begin{bmatrix} 2 & 0.00001000 \\ 3 & 0.00004000 \\ 4 & 0.01051000 \\ 5 & 0.01392000 \\ 6 & 0.01540000 \\ 7 & 0.01310000 \\ \vdots & \vdots \end{bmatrix} \quad (4.15)$$

17 × 2 Matrix

>  $plot([ea\_d\_hf, ea\_d\_dft, ea\_d\_mp2, ea\_d\_vmc, ea\_d\_fn], color = ["red", "black", "yellow", "orange", "blue"], legend = ["HF", "B3LYP", "MP2", "VMC", "FN-DQMC"]);$

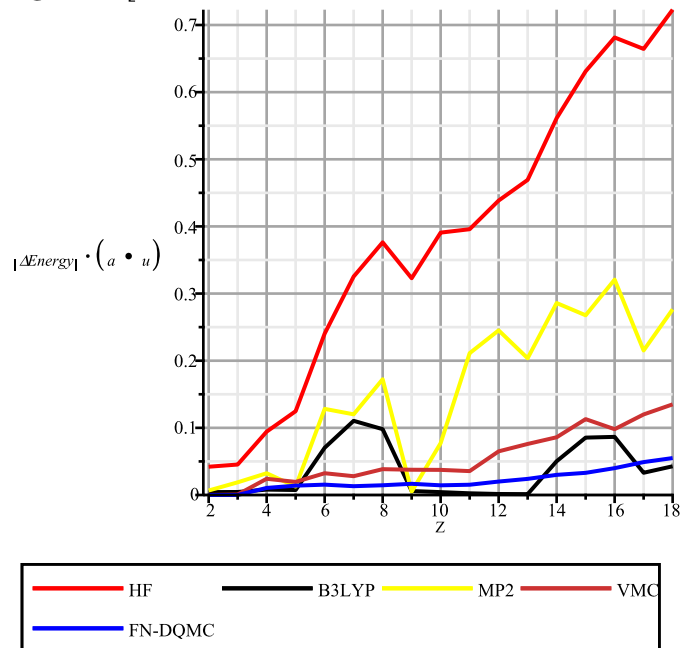


Fig 3. Difference Between the Exact energy and the Energy calculated with different meth

In the above graph, the two Quantum Monte Carlo Methods have a relative low mistake in comparisons with the methods in this work. In

## Conclusions

From the calculations above, it can be concluded that Quantum Monte Carlo Methods is able to replicate with a acceptable values the ground state energy of the many body solution to the Schrödinger Equation of simple atoms that go from  $2 < Z < 19$ , and given that the method is easy to make in parallel.

## References

1. A. Szabo and N. S. Ostlund, *Modern Quantum Chemistry: Introduction to Advanced Electronic Structure Theory* (Dover Books, New York, 1996).
2. F. Jensen, *Introduction to Computational Chemistry* 3rd Edition (John Wiley & Sons, New York, 2017).
3. W.A Lester and B.L Hammond, [Annu. Rev. Phys. Chem. 41, 283-311 \(1990\)](#). "Quantum Monte Carlo for the Electronic Structure of Atoms and Molecules"
4. S.M Rothstein, E. Ospadov and C. Bruzzese, *Introduction to fixed node Quantum Monte Carlo, Mathematical Physics in Theoretical Chemistry* (Elsevier, Massachusetts, 2019)
5. T, Pang, *An Introduction to Quantum Monte Carlo Methods*, (Morgan & Claypool Publication, California, 2016)
6. P.H Acioli, [M. Theo.Chem 394, 75-85 \(1997\)](#). "Review of quantum Monte Carlo methods and their applications"
7. B.M Austin, D.Y Zubarev and W.A Lester, [Chem.Rev. 2012, 112, 263-288](#). "Quantum Monte Carlo and Related Approaches"
8. W.M.C Foulkes, L. Mitas, R.J Needs and G. Rajagopal, [Rev. Mod. Phys. 73, 33 \(2001\)](#). "Quantum Monte Carlo simulations of solids"
9. S.J Chakravorty, S.R Gwaltney, E.R Davidson, F.A Parpia, C. F Fischer, [Phys. Rev. A 47, 3649 \(1993\)](#), "Ground-state correlation energies for atomic ions with 3 to 18 electrons"
10. V. Konkov, R. Peverati, [SoftwareX 9, 7-14 \(2019\)](#). "QMC-SW: A simple workflow for quantum Monte Carlo calculations in chemistry"