

Analytic gradients in correlated Gaussian method and applications to small atoms

(Analytiske gradienter indenfor den korrelerede gaussiske metode og deres anvendelser i små atomer)

Bachelorprojekt
afleveret af:
Christian Foldager Andersen
Studienummer: 201507816



Afdeling for fysik og astronomi
Aarhus Universitet
Danmark
15 Juni 2018

Vejleder:
Dmitri Vladimir Fedorov

Contents

1. Introduction	2
2. Theoretical Considerations	3
2.1 Units	3
2.2 Systems	3
2.2.1 Hydrogen	3
2.2.2 Helium	3
2.3 The Schrödinger equation	3
2.4 The variational method	4
3. The ECG method	5
3.1 Basis-functions and the correlation matrix	5
3.2 Matrix-elements and their gradients	5
3.2.1 Overlap matrix	6
3.2.2 Potential energy	6
3.2.3 Kinetic energy	7
3.2.4 Analytic gradients with respect to the correlation matrix using non-shifted Gaussians	7
3.2.5 Coordinate transformation	8
4. Code and optimization-methods	10
Exponential distribution	10
4.1 Optimization	10
4.2 Stochastic variational approach	11
4.2 Quasi-Newton method (analytic downhill gradient optimization)	11
4.3 Hybrid optimization	12
5. Results and data analysis	13
5.1 Number of Gaussians and convergence	13
5.2 Paired basis vs. random basis	14
5.3 Step size tolerance for fminunc	14
5.4 Stochastic optimization	15
5.4.1 Stochastic variational method for Hydrogen	15
5.4.2 Stochastic variational optimization for Helium	15
5.5 Analytic downhill gradient optimization	16
5.5.1 Analytic downhill gradient optimization for Hydrogen	16
5.5.2 Analytic downhill gradient optimization for Helium	17
5.6 Hybrid optimization	18
5.6.1 Hybrid optimization for Helium	19
5.7 Global optimization	19
5.7.1 Global analytic gradient optimization	19
5.7.2 Global stochastic optimization	20
6. Discussion	21
6.1 Hydrogen	21
6.2 Helium	21
7. Conclusion	23
8. Summary	24
8.1 Summary in English	24
8.2 Resume på Dansk	25

1. Introduction

In variational calculations with correlated Gaussians the optimization of the basis is the task which takes the longest time. One popular approach is to first run stochastic optimization in order to approximately find the global minimum and then run deterministic optimization (e.g. downhill gradient method) to quickly find the precise position of the minimum. The downhill methods converge fast but need the gradients of the matrix elements for their operations. The gradients are notoriously difficult to calculate numerically. However for certain potentials (including Coulomb) the gradients can be actually calculated analytically. So the goal of this project is to calculate analytically the gradients of the matrix elements with correlated Gaussians for the Coulomb potential, apply the analytic gradients to calculations of small atoms, and estimate how efficient they are as compared to traditional methods.

In quantum mechanical few-body systems the stochastic variational method yields incredibly accurate results for the ground state energy. This powerful method opens the possibility of finding the ground state energy of systems numerically, as we cannot solve the Schrödinger equation analytically due to electron-electron interaction.

To solve these systems numerically we use the "Shifted Correlated Gaussian method" which employs building a wavefunction of a linear combination of shifted correlated Gaussians and numerically optimizing the non-linear parameters to minimize the ground state energy in accordance with the variational principle.

Many different optimization algorithms exist that exploit this principle, each with their pros and cons. The Nelder-Mead method and Quasi-Newton method (referred to here as the downhill gradient method) are both deterministic approaches (initial-value problems) and as such, they can get stuck in local minima as opposed to global minima of the parameter space. The stochastic variational method is typically employed - varying the non-linear parameters of your basis randomly and finding what values minimize the energy - however, this is quite time consuming. A better way may be to combine the two and has indeed been employed numerically ([7]). However, numerical gradients introduce a large uncertainty in the calculations. A potentially better way to do it is to use analytic gradients, which can indeed be solved for Coulombic few-body systems using correlated Gaussians.

In this thesis a few-body system constitutes a two- and three- body system, Hydrogen and Helium respectively. The atomic motion is removed using a coordinate transformation to the center-of-mass system and removing the coordinates corresponding to center-of-mass. The focus will remain on these simple systems as the aim of the project is to determine the accuracy and computational advantage of this analytic deterministic approach, not to determine ground state energies for the sake of it.

2. Theoretical Considerations

2.1 Units

Atomic units is a system of units (like SI or cgs) that are convenient to use when dealing with atomic calculations, as one sets many constants equal to 1 to avoid writing them all the time.

Atomic units (here referred to as a.u.) come in two kinds, Rydberg and Hartree. For this thesis, the Hartree unit system has been used. Here $m_e = \hbar = e = 1$ and $c = \frac{1}{137}$, making 1 Ha=27.2 eV ([2]).

2.2 Systems

In this project we will be focusing our attention on Hydrogen and Helium, as these are fairly simple systems that the methods can be tested on.

2.2.1 Hydrogen

Hydrogen is the simplest atom and the only analytically solvable atom as demonstrated by Bohr in 1913. It consists of a proton with a single orbiting electron, making it a two-body system (problem) with the potential being determined by one relative coordinate (a one-body potential).

The ground state energy of Hydrogen is -13.6 eV which is -0.500(0) Ha ([5], p.150).

2.2.2 Helium

Helium is the second element of the periodic table. It consists of a nucleus (consisting of two protons and two neutrons) and two orbiting electrons. Due to electron-electron repulsion this system can not be solved analytically.

In Helium, one electron will always stays in the 1s state (the ground state) ([5], p.299-304) - this electron is either spin up or spin down. If the other electron is to be in the 1s state as well, Pauli's exclusion principle, which states that two fermions (such as electrons) cannot be in the same quantum state, dictates it must have opposite spin. Thus for a $L = 0$ state of Helium, $S = 0$, and one is working with Parahelium (as non-shifted Gaussians are used, see section 3).

The ground state energy of Parahelium is -2.9040 Ha ([1]).

This distinction is made, since the other configuration of Helium (Orthohelium) where the electrons have the same spin has a different ground state energy. This is due to Pauli's exclusion principle: both electrons cannot be in the 1s state with the same spin orientation - thus, the second electron is forced to be in a different configuration than 1s.

The ground state energy of Orthohelium is lower than that of Parahelium; this is due to the fact that the total wave function of the system needs to be anti-symmetric to satisfy Pauli's exclusion principle. The total wave function of a fermion is the product of its spin wave function and its spatial wave function - thus, for a symmetric spin wave function one needs an anti-symmetric spatial wave function. An anti-symmetric spatial wave function results in a larger distance between the electrons, which results in a lower potential energy from the electron-electron interaction (repulsion) and that is the case for Orthohelium.

2.3 The Schrödinger equation

In general, an eigenvalue-problem is a mathematical problem from the realm of linear algebra. It is the basis for the Schrödinger equation and will be touched upon briefly now.

For a matrix A , z is an eigenvector to A with the eigenvalue λ if the following equation is satisfied:

$$Az = \lambda z, \quad z \neq 0 \tag{1}$$

The Schrödinger equation ([5]) that we need for finding the energy of quantum-mechanical systems takes a similar form:

$$H\psi = E\psi \tag{2}$$

Where ψ denotes a wave function/eigenfunction to the Hamiltonian H and E the corresponding energy in that particular state.

2.4 The variational method

Say we do not know the wavefunction of the system we are working with. The variational principle gives us a way to approximate it. This very powerful, practical result states that any "guess" or trial-function plugged into the Schrödinger equation is guaranteed to overshoot the actual energy of the system.

In general, the Hamiltonian H can be deduced, but E and ψ are unknown. Furthermore, as mentioned under the section on Helium, the Schrödinger equation is difficult to solve. The variational principle gives us a method to find ψ - the closer one gets to the experimental ground state energy, the closer one gets to the correct wave function. Its essence is contained in Ritz Theorem, which states:

For an arbitrary function ψ of the state space the expectation value (mean value) of H in the state ψ is such that

$$E = \frac{\langle \psi | H | \psi \rangle}{\langle \psi | \psi \rangle} \geq E_1 \quad (3)$$

where the equality holds if and only if ψ is an eigenstate of H with the eigenvalue E_1 - that is, if ψ is the true wave function ([8]).

This theorem is the basis of this project and all variational methods - through optimization (minimization of the energy) of parameters in the wave function one gets closer and closer to the "true" energy and wave function.

For instance, if ψ is dependent on just one parameter α , one would calculate the mean value of the Hamiltonian in this state $E(\alpha)$ and minimize it with respect to α . The minimal value obtained this way would be an approximate solution to the ground state of the system.

3. The ECG method

The explicitly correlated Gaussian (ECG method) is a common variational method to solve quantum-mechanical few-body problems. The variational principle needs trial wavefunctions - in the ECG-method, these trial functions are correlated Gaussians, meaning we build a basis of correlated Gaussians.

An advantage of using Gaussian functions for the basis is, that their matrix elements as well as their gradients are easy to calculate, and in this project, specifically the gradients are shown to be fully analytically calculable for a Coulombic potential ([3]).

This is particularly important as analytic gradients can be employed in the Quasi-Newton (downhill gradient) method, as this method usually fails due to the large uncertainty introduced in the computation of numerical gradients ([6]).

In general, the Gaussians can be shifted or non-shifted. In this thesis only $L = 0$ states are examined, making shifted Gaussians obsolete. Thus, only non-shifted Gaussians will be used.

3.1 Basis-functions and the correlation matrix

A Gaussian function $|g\rangle$ in an N -particle system with coordinates $\mathbf{r}_i|_{i=1\dots N}$ is of the form:

$$\langle \mathbf{r} | g \rangle = \exp \left(- \sum_{i,j=1}^N A_{ij} \mathbf{r}_i \mathbf{r}_j \right) = e^{-\mathbf{r}^T A \mathbf{r}} \quad (4)$$

Where \mathbf{r} is a N -size vector with the particle coordinates $\vec{\mathbf{r}}_i$, $A = A_{ij}$ is a N -size square symmetric positive-definite matrix referred to here as the correlation matrix. A is defined as ([3], [8]):

$$A = \sum_{i>j=1}^N \alpha_{ij} w_{ij} w_{ij}^T \quad (5)$$

Where w_{ij} is a N -size column of numbers all equal to 0 except for $w_i = 1$ for a one-body potential (one relevant relative coordinate, such as Hydrogen) and $w_i = -w_j = 1$ for a two-body potential (two relevant relative coordinates, such as Helium).

Having built a basis of Gaussians, the wave function is represented as a linear combination:

$$|\psi\rangle = \sum_{i=1}^n c_i |g_i\rangle \quad (6)$$

Where c_i are linear coefficients, g_i are basis-Gaussian functions and n is the number of Gaussians in the basis.

Inserting this wavefunction into the Schrödinger equation and taking the inner product on the left with $\langle g_j |$ one obtains:

$$\langle g_i | \hat{H} | g_j \rangle c = E \langle g_i | g_j \rangle c \quad (7)$$

Where c is the n -size column of linear parameters, $\langle g_i | \hat{H} | g_j \rangle$ is the Hamiltonian on the i 'th and j 'th Gaussian (effectively the energy interaction between the i 'th and j 'th element) and $\langle g_i | g_j \rangle$ is the overlap between the i 'th and j 'th element. In matrix form:

$$\mathcal{H}c = E\mathcal{N}c \quad (8)$$

Where \mathcal{H} is the matrix containing all matrix elements of the Hamiltonian and \mathcal{N} is the matrix containing all matrix elements of the overlap.

Equation 8 is the equation used for optimization in this thesis.

The linear parameters are determined by solving the generalized eigenvalue equation using methods from linear algebra. The non-linear parameters (the correlation matrix elements) are solved using an optimization method, such as the stochastic variational approach (as described in section 4).

3.2 Matrix-elements and their gradients

As have already been mentioned, what makes the ECG method so powerful is how simple the matrix-elements of the Schrödinger equation and the gradients of these become. In this section, it will be described how to calculate the overlap matrix and how to build the correlation matrix for each Gaussian-function and how to transform it using relative (Jacobian) coordinates to get rid of the energy associated with the center-of-mass. The Hamiltonian will also be described using the kinetic- and potential energy matrices, as well as deriving the gradients of these.

3.2.1 Overlap matrix

The overlap between a non-shifted Gaussian g with parameter A and a non-shifted Gaussian g' with parameters A' is given as:

$$\langle g'|g \rangle = \int d^3\mathbf{r}_1 \dots d^3\mathbf{r}_N e^{-\mathbf{r}^T A' \mathbf{r}} e^{-\mathbf{r}^T A \mathbf{r}} = \int d^3\mathbf{r}_1 \dots d^3\mathbf{r}_N e^{-\mathbf{r}^T B \mathbf{r}} \quad (9)$$

Where $B = A' + A$. To evaluate the integral, consider an orthogonal coordinate transformation $\mathbf{r} = Q\mathbf{x}$ to the basis where B is diagonal: $B = QDQ^T$ where $Q^T Q = QQ^T = 1$ and D is a diagonal matrix. In doing so, one obtains:

$$\langle g'|g \rangle = \int d^3\mathbf{x}_1 \dots d^3\mathbf{x}_N e^{-\sum_{i=1}^N \mathbf{x}_i \cdot D \mathbf{x}_i} = \prod_{i=1}^N \int d^3\mathbf{x}_i e^{-\mathbf{x}_i \cdot D_{ii} \mathbf{x}_i} = \prod_{i=1}^N \left(\frac{\pi}{D_{ii}} \right)^{\frac{3}{2}} = \left(\frac{\pi^N}{\det(B)} \right)^{\frac{3}{2}} = M \quad (10)$$

3.2.2 Potential energy

There are many potentials one can work with. For atoms as simple as Hydrogen and Helium where screening effects are minimal and $L = 0$ it seems most reasonable to work with simple, central Coulombic potentials. For two charges at different coordinates the Coulombic potential is given as:

$$V_{Coulomb} = \frac{q_i \cdot q_j}{4\pi\epsilon_0} \cdot \frac{1}{|\mathbf{r}_j - \mathbf{r}_i|} \quad (11)$$

Where $q_i, q_j, \mathbf{r}_i, \mathbf{r}_j$ is the charge and coordinate of the i 'th and j 'th particle, respectively.

For central potentials V can be represented in terms of $w^T \mathbf{r}$ where w is a size- N column vector with all entries equal to 0 except $w_i = 1$ for one-body central potentials (such as Hydrogen) and $w_i = -w_j = 1$ for two-body central potentials (such as Helium) ([3], p. 3).

For a spherical potential with a general form-factor $V_\alpha f(|(w^T \mathbf{r})|)$ with shifted Gaussians ([3] p. 4):

$$\langle g'|f(|(w^T \mathbf{r})|)|g \rangle = M \left(\frac{\beta}{\pi} \right)^{\frac{3}{2}} 2\pi \frac{e^{-\beta q^2}}{\beta q} \int_0^\infty r dr f(r) e^{-\beta r^2} \sinh(2\beta qr) = MJ[f] \quad (12)$$

Where $\beta = (w^T B^{-1} w)^{-1}$, $\mathbf{q} = w^T \mathbf{u} = w^T \frac{1}{2} B^{-1} \mathbf{1} \mathbf{v}$ where $\mathbf{v} = s' + s$ and s', s are the shifts for the g' and g Gaussians respectively.

For a Coulomb form factor $1/r$, the integral evaluates to ([3], p. 4):

$$\langle g'|f(|(w^T \mathbf{r})|)|g \rangle = MJ \left[\frac{1}{r} \right] = M \cdot \frac{\text{erf}(\sqrt{\beta} q)}{q} \quad (13)$$

For the systems in this thesis using non-shifted Gaussians, $s = s' = 0$ and thus $q = 0$. Thus one needs to take the limit of equation 13 as q approaches 0:

$$\lim_{q \rightarrow 0} J \left[\frac{1}{r} \right] = \lim_{q \rightarrow 0} \left(\frac{\text{erf}(\sqrt{\beta} q)}{q} \right) = \lim_{q \rightarrow 0} \left(\frac{2}{\sqrt{\pi}} \frac{\sqrt{\beta} q}{q} \right) = 2\sqrt{\frac{\beta}{\pi}} \quad (14)$$

In the third step a first-order Taylor-approximation for the error-function $\text{erf}(x) \approx \frac{2}{\sqrt{\pi}} x$ has been used. Thus the matrix elements are of the form:

$$\langle g'|f(|(w^T \mathbf{r})|)|g \rangle = M \cdot 2\sqrt{\frac{\beta}{\pi}} \quad (15)$$

Furthermore, the potential scales with the product of the charges, so the total potential energy term between the i 'th and j 'th particle is:

$$U = q_i q_j M \cdot 2\sqrt{\frac{\beta}{\pi}} \quad (16)$$

3.2.3 Kinetic energy

For a non-relativistic system, the kinetic energy of a N -body system is the sum of the kinetic energy of its constituents, meaning our kinetic energy operator is:

$$\hat{K} = - \sum_{i=1}^N \frac{\hbar^2}{2m_i} \frac{\partial^2}{\partial \mathbf{r}_i^2} \quad (17)$$

For completeness consider a more general form of the kinetic energy operator ([3], p. 3):

$$\hat{K} = - \sum_{i,j=1} \frac{\partial}{\partial \mathbf{r}_i} \Lambda_{ij} \frac{\partial}{\partial \mathbf{r}_j} = - \frac{\partial}{\partial \mathbf{r}} \Lambda \frac{\partial}{\partial \mathbf{r}^T} \quad (18)$$

Where Λ is a positive-definite symmetric $[N-1] \times [N-1]$ matrix given by:

$$\Lambda = \frac{\hbar^2}{2\mu} \cdot I \quad (19)$$

Where μ is the reduced mass of the system and I is the $[N-1] \times [N-1]$ identity matrix.

The matrix elements of this operator are needed. It will be solved for the general case (with shifted Gaussians) and afterwards simply set $s' = s = 0$.

For this, two integrals need to be evaluated, the first one being:

$$\langle g' | \mathbf{r} | g \rangle = \left(\frac{\partial}{\partial \mathbf{v}^T} \right) e^{\frac{1}{4} \mathbf{v}^T B^{-1} \mathbf{v}} M = \mathbf{u} e^{\frac{1}{4} \mathbf{v}^T B^{-1} \mathbf{v}} M \quad (20)$$

Where $\mathbf{u} = \frac{1}{2} B^{-1} \mathbf{v}$. The second integral:

$$\langle g' | \mathbf{r}^T F \mathbf{r} | g \rangle = \left(\frac{\partial}{\partial \mathbf{v}} F \frac{\partial}{\partial \mathbf{v}^T} \right) e^{\frac{1}{4} \mathbf{v}^T B^{-1} \mathbf{v}} M = \left(\frac{3}{2} \text{trace}(F B^{-1}) + \mathbf{u}^T F \mathbf{u} \right) e^{\frac{1}{4} \mathbf{v}^T B^{-1} \mathbf{v}} M \quad (21)$$

From which one obtains:

$$\langle g' | - \frac{\partial}{\partial \mathbf{r}} \Lambda \frac{\partial}{\partial \mathbf{r}^T} | g \rangle = (6 \cdot \text{trace}(A' \Lambda A B^{-1}) + (\mathbf{s}' - 2A' \mathbf{u})^T \Lambda (\mathbf{s} - 2A \mathbf{u})) M \quad (22)$$

Setting $s' = s = 0$ (and thus $u = 0$) one obtains the needed result for non-shifted Gaussians:

$$\langle g' | - \frac{\partial}{\partial \mathbf{r}} \Lambda \frac{\partial}{\partial \mathbf{r}^T} | g \rangle = 6 \cdot \text{trace}(A' \Lambda A B^{-1}) M \quad (23)$$

Both the matrix elements of the potential- and kinetic energy are now in simple analytic form.

3.2.4 Analytic gradients with respect to the correlation matrix using non-shifted Gaussians

To minimize the energy with respect to the non-linear parameters, we need to calculate the gradient of the energy with respect to α where α is a non-linear parameter. Isolating the energy in the Schrödinger equation in a given Gaussian basis, we have:

$$E = \frac{\langle g_i | \hat{H} | g_j \rangle}{\langle g_i | g_j \rangle} \quad (24)$$

For the downhill gradient algorithm we need the gradient of the energy with respect to a given non-linear parameter of a Gaussian, α :

$$\frac{\partial E}{\partial \alpha} = \frac{\partial \mathcal{H}}{\partial \alpha} - E \frac{\partial \mathcal{N}}{\partial \alpha} \quad (25)$$

Seeing as $\hat{H} = \hat{K} + \hat{U}$, we need to evaluate the gradients of the potential- and kinetic energy.

For reference, as the following equations make the gradients easier to evaluate:

$$\frac{\partial \det(B)}{\partial \alpha} = \det(B) \text{trace} \left(B^{-1} \frac{\partial B}{\partial \alpha} \right) \quad (26)$$

$$\frac{\partial B^{-1}}{\partial \alpha} = -B^{-1} \frac{\partial B}{\partial \alpha} B^{-1} \quad (27)$$

Furthermore, in our representation of the correlation matrix $\frac{\partial B}{\partial \alpha} = ww^T$ ([3], p. 12), where w is the column vector introduced in the section 3.2.2.

Starting with the gradient of the potential energy:

$$\frac{\partial}{\partial \alpha} \langle g' | f(|(w^T \mathbf{r})|) |g\rangle = \frac{\partial}{\partial \alpha} \left(M \cdot 2\sqrt{\frac{\beta}{\pi}} \right) \quad (28)$$

The derivatives can be calculated using the product rule.

$$\frac{\partial}{\partial \alpha} \left(M \cdot 2\sqrt{\frac{\beta}{\pi}} \right) = \frac{2}{\sqrt{\pi}} \left(\frac{\partial M}{\partial \alpha} \cdot \beta^{\frac{1}{2}} + M \cdot \frac{\partial \beta^{\frac{1}{2}}}{\partial \alpha} \right) \quad (29)$$

The derivatives in equation 29 evaluate to:

$$\frac{\partial M}{\partial \alpha} = -\frac{3}{2} \frac{1}{\det(B)} \frac{\partial \det(B)}{\partial \alpha} M \quad (30)$$

$$\frac{\partial \beta^{\frac{1}{2}}}{\partial \alpha} = \frac{\partial \beta^{\frac{1}{2}}}{\partial \beta} \cdot \frac{\partial \beta}{\partial \alpha} = -\frac{1}{2} \beta^{-5/2} \cdot w^T \frac{\partial B^{-1}}{\partial \alpha} w \quad (31)$$

In total using equation 26 and 27 the gradient of potential energy simplifies to:

$$\frac{\partial}{\partial \alpha} \langle g' | f(|(w^T \mathbf{r})|) |g\rangle = M \left(\frac{1}{\sqrt{\pi \beta^5}} \cdot w^T B^{-1} \frac{\partial B}{\partial \alpha} B^{-1} w - 3\sqrt{\frac{\beta}{\pi}} \text{trace} \left(B^{-1} \frac{\partial B}{\partial \alpha} \right) \right) \quad (32)$$

The gradient of the kinetic energy is ([3], p. 10):

$$\frac{\partial}{\partial \alpha} \langle g' | -\frac{\partial}{\partial \mathbf{r}} \Lambda \frac{\partial}{\partial \mathbf{r}^T} |g\rangle = 6 \cdot \text{trace} \left(-A' \Lambda A B^{-1} \frac{\partial B}{\partial \alpha} B^{-1} \right) M \quad (33)$$

The gradient of the Hamiltonian is the sum of the two energy gradients:

$$\frac{\partial \mathcal{H}}{\partial \alpha} = \frac{\partial}{\partial \alpha} \langle g' | f(|(w^T \mathbf{r})|) |g\rangle + \frac{\partial}{\partial \alpha} \langle g' | -\frac{\partial}{\partial \mathbf{r}} \Lambda \frac{\partial}{\partial \mathbf{r}^T} |g\rangle \quad (34)$$

The gradient of the overlap matrix is simply the matrix containing the gradient of each matrix-element which is given by equation 30.

3.2.5 Coordinate transformation

As one is interested in the intrinsic energy of the system, one has to separate the center-of-mass motion (that is, the motion of the atom as a whole). To do this, a transformation to Jacobian coordinates is performed and a notice that the last row and column in our Hamiltonian correspond solely to the center of mass motion is made, which are then simply removed.

Furthermore, getting an exact result to match experimental values is not the goal. For this thesis, the assumption that the nucleus is infinitely heavy has been made (a fair assumption given that the nucleus is about 1860 times heavier than the electron for Hydrogen, and about 7440 times heavier for Helium). One needs to separate the center-of-mass motion (energy) to get the correct energies. The Hamiltonian then reads:

$$H = \sum_{i=1}^N \frac{\mathbf{p}_i^2}{2m_i} - T_{cm} + \sum_{j>i=1}^N \quad (35)$$

This is done by introducing relative and center-of-mass coordinates $\mathbf{x}^T = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N)$, where x_N is chosen to be the center-of-mass coordinate and the rest of the $N - 1$ coordinates are independent relative coordinates. They are related to the single-particle $\mathbf{r}^T = (\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_N)$ coordinates through a linear transformation:

$$\mathbf{x}_i = \sum_{j=1}^N U_{ij} \mathbf{r}_j, \quad (i = 1, \dots, N) \quad (36)$$

For this thesis the Jacobi coordinate set has been used, which is defined by the transformation matrix ([8], p. 10):

$$U_J = \begin{bmatrix} 1 & -1 & 0 & \dots & 0 \\ \frac{m_1}{m_{12}} & \frac{m_2}{m_{12}} & -1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{m_1}{m_{12\dots N-1}} & \frac{m_2}{m_{12\dots N-1}} & \dots & \dots & -1 \\ \frac{m_1}{m_{12\dots N}} & \frac{m_2}{m_{12\dots N}} & \dots & \dots & \dots \frac{m_N}{m_{12\dots N}} \end{bmatrix} \quad (37)$$

Where $m_{12\dots i} = m_1 + m_2 + \dots + m_i$.

After transforming to this coordinate set, one removes the energy associated with the center-of-mass removing the last column and row of the transformed correlation matrix.

As an example, here is the procedure for Hydrogen. For Hydrogen, $N = 2$ and it is a one-body potential. Its correlation matrix is then given by:

$$A = \sum_{i < j=1}^2 \alpha_{ij} w_{ij} w_{ij}^T = \alpha_{12} w_{12} w_{12}^T = \alpha_{12} \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \quad (38)$$

The Jacobian coordinates are:

$$\mathbf{x}_1 = \mathbf{r}_2 - \mathbf{r}_1 \quad \text{and} \quad \mathbf{x}_2 = \frac{m_1 \mathbf{r}_1 + m_2 \mathbf{r}_2}{m_1 + m_2} \quad (39)$$

Where \mathbf{x}_1 is the relative coordinate that determines the potential and \mathbf{x}_2 is the center-of-mass coordinate. Collecting these in a vector, we have:

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} \quad (40)$$

Using our transformation matrix, we have:

$$\mathbf{r}^T A \mathbf{r} = \mathbf{x}^T U_J^T A U_J \mathbf{x} = \mathbf{x}^T A' \mathbf{x} \quad (41)$$

Where:

$$A' = U_J^T A U_J = \begin{bmatrix} 1 & \frac{m_1}{m_1+m_2} \\ -1 & \frac{m_2}{m_1+m_2} \end{bmatrix} \begin{bmatrix} \alpha_{12} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & -1 \\ \frac{m_1}{m_1+m_2} & \frac{m_2}{m_1+m_2} \end{bmatrix} = \begin{bmatrix} \alpha_{12} & -\alpha_{12} \\ -\alpha_{12} & \alpha_{12} \end{bmatrix} \quad (42)$$

To throw away the center-of-mass coordinate (\mathbf{x}_2), simply remove the last row and column of A' and end up with a $[1] \times [1]$ matrix:

$$A'_{-cmH} = \alpha_{12} \quad (43)$$

The same procedure for Helium under the assumption of an infinitely heavy nucleus yields the correlation matrix:

$$A'_{-cmHe} = \begin{bmatrix} \alpha_{12} + \alpha_{13} & -\alpha_{12} \\ -\alpha_{12} & \alpha_{12} + \alpha_{23} \end{bmatrix} \quad (44)$$

Both of these matrices are indeed positive-definite if $\alpha_{12}, \alpha_{13}, \alpha_{23} > 0$. This can be shown by calculating $r^T A r$, with $r = \begin{bmatrix} r_1 \\ r_2 \end{bmatrix}$ where $r_1 \neq 0$ and $r_2 \neq 0$.

$$r^T A r = \begin{bmatrix} r_1 & r_2 \end{bmatrix} \begin{bmatrix} \alpha_{12} + \alpha_{13} & -\alpha_{12} \\ -\alpha_{12} & \alpha_{12} + \alpha_{23} \end{bmatrix} \begin{bmatrix} r_1 \\ r_2 \end{bmatrix} = r_1^2 \alpha_{13} + r_2^2 \alpha_{23} + \alpha_{12} (r_1^2 + r_2^2 - 2r_1 r_2) \quad (45)$$

It follows this is greater than zero from the Cauchy-Schwarz inequality, that is $2r_1 r_2 \leq r_1^2 + r_2^2$. Thus, the correlation matrix is indeed positive definite.

4. Code and optimization-methods

Exponential distribution

For the stochastic variational method we need to generate random parameters for the correlation matrices of the Gaussian basis. These random parameters are generated using an exponential distribution of the form:

$$f(x, \mu) = \frac{1}{\mu} e^{-\frac{x}{\mu}} \quad (46)$$

In this thesis μ was chosen to be equal to 1.

4.1 Optimization

The goal of the optimization method is to either get a more precise result (in this case an energy closer to the experimentally measured ground state of a given system) or to get the result faster (that is, reducing computational time).

Optimization algorithms work by finding extremum-points of a given objective function - in the case of ground-state energies, one is searching for minima. However, a parameter space may have multiple minima, and the goal is to find the lowest one. A distinction is made between a local and a global minimum.

A local minimum is the point where the function reaches a minimum value on a finite interval of variables. The issue is, as the amount of parameters go up, so do the amount of local minima ([8], p. 40-41).

A global minimum is the point where the function reaches a minimum value on an infinite interval of variables - that is, it is the lowest possible value of the function.

Given an objective function f , the condition for a minimum is:

$$\frac{\partial f}{\partial x_i} = 0 \quad i = 1, \dots, n \quad (47)$$

For an n -dimensional argument.

In optimization, algorithms can be divided into two categories: the deterministic- and the stochastic-optimization.

A deterministic optimization moves downwards on the slope of a function given a starting point according to a given algorithm - in our case, we will be looking Quasi-Newton or downhill gradient method (see section 4). This approach is great for converging to a minimum but has a few drawbacks: typically, they are very time-consuming as they need to evaluate more code (such as the analytic gradients used in our method); also, even though they converge very fast, they converge only to whichever local minimum they first encounter. This is an issue, as a local minimum could still yield an energy much higher than the global minimum, which is what one is seeking when trying to estimate the true ground state energy of a given system.

A stochastic optimization addresses the problem of finding the global minimum well - it works by guessing parameters at random. Unlike the deterministic optimization, the stochastic does not get stuck in local minima and thus has a much greater chance of getting one in the valley of the global minimum - however, hitting the exact global minimum by chance is very time consuming as well, and simply being in the valley of the global minimum does not guarantee a better energy/result than any given local minima.

We thus have a dichotomy defined by these two approaches - on one hand, one needs to spend time evaluating extra code and getting stuck in local minima; on the other hand however, one has a (small) chance to hit a global minimum and getting closer and closer to the bottom of the valley of the global minimum is also very time consuming.

A solution to this is being tested in this thesis: a hybrid approach. The idea is to run the stochastic optimization first and get within the valley of the global minimum and then switch to the deterministic optimization to converge to the global minimum. In section 3.1 we showed that the eigenvalue problem we need to optimize is given by:

$$\mathcal{H}c = E\mathcal{N}c \quad (48)$$

The Matlab-routine is set to solve equations of the above form. The code makes a random guess at trial wave functions, then Cholesky decomposition is employed to find the eigen-energy.

However, Cholesky decomposition requires that the overlap matrix is positive semi-definite ([4]). This is only the case if all possible pairs of basis-elements are different enough. This is assured by requiring

that for each proposed set of parameters on the j th function the overlap is below a given, empirically determined threshold.

$$\frac{\langle g_i | g_j \rangle}{\sqrt{\langle g_i | g_i \rangle \langle g_j | g_j \rangle}} < T \text{ for all } i \neq j \quad (49)$$

For Helium, $T = 0.97$, that is, if the i th and j th element need to differ 97 percent. For Hydrogen, $T = 0.95$ as the likelihood of two parameters being the same increases due to the fact that Hydrogen only has 1 parameter per Gaussian, whereas Helium has 3.

4.2 Stochastic variational approach

The stochastic variational approach is a random approach to try to minimize the energy; it works as follows.

Parameters are randomly generated for the Gaussians which creates a basis. The Schrödinger equation is then solved for this basis and the eigenvalue (energy) is found. A new set of parameters are then generated at random and interchanged with one basis-element, and the Schrödinger equation is solved again. If the energy is lower with the next basis, the new basis element is accepted and the old is rejected - if not, the new basis element is rejected and the old is kept. Repeating the process over all the elements of the basis and doing multiple sweeps allows us to come closer and closer to the energy. To summarize:

1. Generate a random basis.
2. Solve the Schrödinger equation.
3. Generate a new set of parameters for one (or more) basis element(s) at a time.
4. Solve the Schrödinger equation again in this new basis.
5. If the energy is smaller with the new basis, keep it - if not, reject it.
6. Repeat step 3-5 until the runtime is over.

If a better energy has not been found, the time it took to get the lowest energy is noted as the convergence time and the energy itself as the convergence energy.

For this thesis both local- and global- optimization have been tested for the stochastic variational approach, that is, optimizing one basis element at a time and all basis elements at a time.

The stochastic variational approach is great in comparison with other minimization algorithms (such as Nelder-Mead or Quasi-Newton) in the sense that it does not get stuck in the local minima of the parameter space. Nelder-Mead- or Quasi-Newton- like approaches start in a given point and find the closest minimum - this is (very) likely a local minimum, whereas we are interested in a global minimum. As we change the basis each time, the stochastic variational approach tends not to get "stuck" in local minima.

4.2 Quasi-Newton method (analytic downhill gradient optimization)

The Quasi-Newton, or as it is better known (and referred to in the abstract), the downhill-gradient method, is an extension of the minimization strategy proposed by Newton.

Newton's method is based on the quadratic approximation of the objective function f in the vicinity of the suspected minimum:

$$f(\mathbf{x} + \Delta\mathbf{x}) \approx f(\mathbf{x}) + \nabla f(\mathbf{x})^T \Delta\mathbf{x} + \frac{1}{2} \Delta\mathbf{x}^T \mathbf{H}(\mathbf{x}) \Delta\mathbf{x} \quad (50)$$

Where $\nabla f(\mathbf{x}) = \frac{\partial f(\mathbf{x})}{\partial x_i} |_{i=1, \dots, n}$ is the gradient of the objective function at the point \mathbf{x} and $\mathbf{H}(\mathbf{x}) = \frac{\partial^2 f(\mathbf{x})}{\partial x_i \partial x_j} |_{i, j \in 1, \dots, n}$ is the Hessian matrix, a square matrix containing the second order partial derivatives of the objective function at the point \mathbf{x} .

The condition for the minimum of the quadratic form is the point where the gradient with respect to $\Delta\mathbf{x}$ vanishes, that is

$$\nabla f(\mathbf{x}) + \mathbf{H}(\mathbf{x}) \Delta\mathbf{x} = 0 \quad (51)$$

or

$$-\mathbf{H}^{-1} \nabla f(\mathbf{x}) = \Delta\mathbf{x} \quad (52)$$

This gives an approximation of the step toward a minimum known as Newton's step. Newton's method is then an iteration of this process:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \mathbf{H}(\mathbf{x}_k)^{-1} \nabla f(\mathbf{x}_k) \quad (53)$$

The Quasi-Newton method is a variation of the Newton Method in which one avoids recalculating the Hessian matrix at each iteration. Instead, the iteration of the Hessian matrix is determined by the analytic gradient vector. The update $\delta\mathbf{H}$ is typically chosen to satisfy the equation:

$$\nabla f(\mathbf{x} + \mathbf{s}) = \nabla f(\mathbf{x}) + (\mathbf{H} + \delta\mathbf{H})\mathbf{s} \quad (54)$$

This equation is known as the secant equation: it is a Taylor expansion of the gradient. In analytic downhill gradient optimization, it works as follows:

1. Generate a random basis.
2. Optimize one (or more) basis element(s) by converging to the nearest minima using analytic downhill gradient optimization on equation 8.
3. If the point converged to offers a smaller energy than the previous iteration, it is saved - if not, it is discarded and the process is repeated for the next Gaussian.
4. Repeat step 1-3 until the runtime is over.

4.3 Hybrid optimization

In Hybrid optimization one employs two (or more) optimization algorithms in succession in an attempt to get a better (meaning faster or more accurate) result.

In this thesis the stochastic variational approach followed by the Quasi-Newton approach using analytic gradients is tested - that is, combining a deterministic approach with a variational approach.

The rationale behind this is that the stochastic variational approach is great at getting us in the vicinity of the global minimum, which is the quantity one is interested in for the solution to the Schrödinger equation - however, the stochastic approach converges slowly due to its random nature. The Quasi-Newton method has a tendency to get stuck in local minima - however, the method is designed to find the minimum once you are in the vicinity of it. The stochastic variational method gets us in the vicinity of a global minimum. After we are in the vicinity of a global minimum, switching to the Quasi-Newton method should converge much quicker than proceeding with the stochastic variational method, due to its deterministic nature.

To summarize, the method is to simply run stochastic variational approach for x-amount of time and take the best result of this as the starting point for the downhill analytic gradient optimization.

5. Results and data analysis

In this section, we will present the results of the different optimization algorithms, namely the stochastic variational approach and the two types of hybrid optimization: firstly running the stochastic variational approach for x amount of time and then switching to downhill gradient method for y amount of time and secondly having the Matlab-routine make a guess, converge using analytic gradients and reiterate for x amount of time. We will be comparing both the accuracy of the results (closer to experimental value is better) as well as the convergence time (lower is better).

The standard runtime is set to 60s for all codes such that the results can be compared easily.

Note: the code will go over the maximum amount of time to complete an iteration - for instance, if the code is starting an iteration at 59 sec that takes 3 sec to complete, the convergence time will be 62 sec. Due to the statistical nature of stochastic simulations and random start points, each algorithm is said to run 10 times and an average is taken of both the convergence time and convergence energy. The figures presented in this section will be an example of what one of these (energy,time) graphs look like for a given algorithm.

5.1 Number of Gaussians and convergence

The more Gaussians used in the basis, the closer one gets to the true wave function - however, the larger the parameter space becomes as well, meaning it takes more and more computer power (time) to run an iteration. A rough estimate of parameters is given as follows: an N -particle system has $N(N - 1)/2$ pr. Gaussian ([8], p. 41). Thus, the total amount of parameters using a linear combination of K functions, the total amount of parameters is $K(N(N - 1)/2)$.

One needs to compromise here - that is, one needs to sacrifice some accuracy the sake of speed. The amount of Gaussians used has been investigated by looking at the convergence energy of both Hydrogen and Helium (see figure 1) as a function of Gaussians and thus decided empirically. The non-switch hybrid optimization was used to determine the amount of Gaussians used for all hybrid optimization algorithms (global, switching etc.). The code for He is set to have a maximum runtime of 60 seconds - this is an arbitrary choice.

This should make the iteration with many Gaussians less accurate as each iteration takes longer time; thus, a bad starting guess at a many-Gaussian iteration means one would get a worse energy despite using more Gaussians, due to the time limit set on the routine. One thus expects a fairly obvious convergence at the first additions of Gaussians, which will then become more random as more and more Gaussians are added. The point here is to get an idea for a reasonable amount of Gaussians to use with this algorithm.

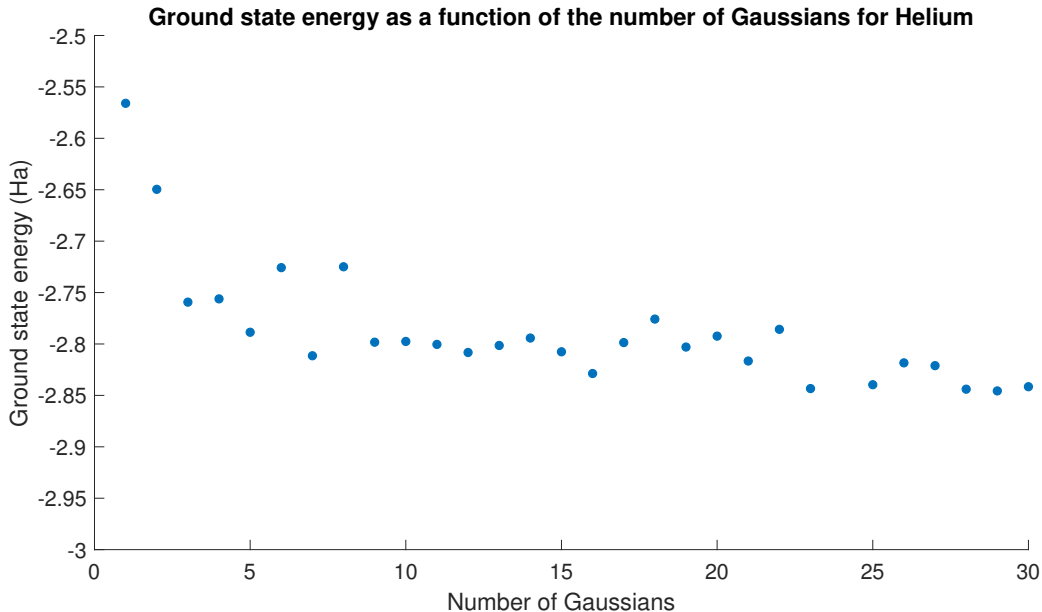


Figure 1: Ground state energy of Helium as a function of the number of Gaussians used. Runtime is 60 seconds. The maximum amount of Gaussians tested was $n_{gmax} = 30$. Step size tolerance was $tol = 10^{-4}$. For the 24th Gaussian the Cholesky-decomposition failed and the result has been discarded.

The energy converges around $n_g = 10$ for Helium and $n_g = 5$ for Hydrogen (not shown). These are the amount of Gaussians that will be used for the codes involving analytic gradients (except for the hybrid method which like stochastic method will be tested using a number different amounts of Gaussians).

The tolerance was determined in section 5.3 and used here because an lower tolerance requires each iteration to take more time which is a larger factor the more Gaussians used (see section 5.5.2). Thus, the tolerance was tested for a reasonable amount of Gaussians (10) and the tolerance found used to see how many Gaussians are needed for convergence of the energy.

5.2 Paired basis vs. random basis

In this thesis, two different approaches to generating a basis of Gaussians have been tested (for Helium, as one cannot use a paired basis with only one parameter pr. Gaussian) - the random basis and the paired basis.

The results of this are summarized in 3 and 4 in section 5.4.2.

The correlation matrix for Helium is of the form:

$$A = \begin{bmatrix} \alpha_{12} + \alpha_{13} & -\alpha_{12} \\ -\alpha_{12} & \alpha_{12} + \alpha_{23} \end{bmatrix} \quad (55)$$

Where the parameters are chosen such that the matrix is positive-definite.

However, the random basis does not take into account any correlation between the parameters a_{11} and a_{22} .

The paired basis is designed to take this into account - it makes twice the amount of Gaussians from the same amount of parameters as the random basis. It does so by swapping the diagonals, thus creating another correlation matrix from the same parameters:

$$A = \begin{bmatrix} \alpha_{12} + \alpha_{23} & -\alpha_{12} \\ -\alpha_{12} & \alpha_{12} + \alpha_{13} \end{bmatrix} \quad (56)$$

This matrix is still positive-definite.

Thus, one gets twice the amount of Gaussians for the same amount of parameters, which should lead to better results. This was indeed seen to be the case with the stochastic simulations (see section 5.3.1 and 5.3.2) and was also found to be the case by [7]. Moving forward, the paired basis has been employed for all algorithms involving analytic gradients.

5.3 Step size tolerance for fminunc

The fminunc function that evaluates the gradients is set to stop converging for a minima if the size of the current step is less than a given tolerance and move on to the next Gaussian. This is a tricky, arbitrary choice - one needs a tolerance large enough to have the program not waste time converging but also not so low that it stops converging too early. For the case of Helium using 10 Gaussians and a runtime of 60, different tolerance were tested to see which gave the best result to indicate which tolerance to use. The results are summarized in table 1. Based on the results of table 1 a step size tolerance of $\text{tol} = 10^{-4}$

Tolerance	Convergence time	Convergence energy	Deviation
10^{-3}	73.6760 s	-2.7828 Ha	4.1736%
10^{-4}	75.0547 s	-2.7857 Ha	4.0737%
10^{-5}	75.0189 s	-2.7684 Ha	4.6694%
10^{-6} (default)	108.7805 s	-2.7872 Ha	4.0220%
10^{-7}	49.0416 s	-2.7394 Ha	5.6680%
10^{-8}	55.3615 s	-2.7511 Ha	5.2652%
10^{-9}	61.1941 s	-2.7418 Ha	5.5854%
10^{-10}	77.6696 s	-2.7673 Ha	4.7073%
10^{-11}	76.2782 s	-2.7506 Ha	5.2824%

Table 1: Results of analytic downhill gradient optimization using a paired with various stepsize tolerances for Helium. The runtime for downhill gradient optimization is 60 sec and the amount of Gaussians used was 10.

has been employed for Helium for this thesis, as this tolerance offered the second lowest deviation in a

time frame much more comparable with the set runtime of 60 sec.

For Hydrogen a stepsize tolerance of $\text{tol} = 10^{-50}$ has been used, due to the significantly smaller parameter space, as a smaller parameter space has less possible local minima to converge to ([8], p. 41).

5.4 Stochastic optimization

For the stochastic approach, the amount of Gaussians used were varied as to test if one could beat the Hybrid optimization time and accuracy merely by using more Gaussians. For Hydrogen, 4-, 5-, 6- and 7- Gaussians were tested. For Helium, 5-, 10-, 20- and 30- Gaussians were tested. In general one needs more Gaussians the more bodies in the system. This is why a bigger jump has been made for Helium than for Hydrogen in terms of the amount of Gaussians tested. For both systems the amount of Gaussians are around their expected Gauss convergence found using analytic downhill gradient optimization.

5.4.1 Stochastic variational method for Hydrogen

For Hydrogen, the results are summarized in table 2.

n_{gauss}	Convergence time	Convergence energy	Deviation
4	59.3399 s	-0.4993 Ha	0.1400%
5	59.5298 s	-0.4996 Ha	0.08000%
6	59.2133 s	-0.4997 Ha	0.06000%
7	59.5429 s	-0.4998 Ha	0.04000%

Table 2: Results of stochastic optimization with a random basis for Hydrogen

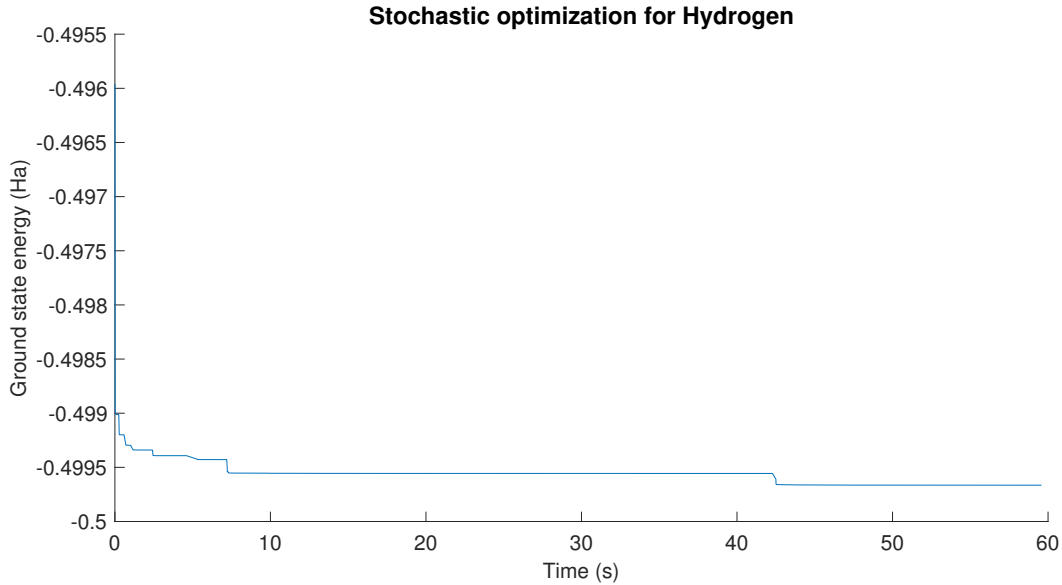


Figure 2: The ground state energy as a function of time for Hydrogen using the stochastic optimization algorithm with 5 Gaussians. For this example the convergence energy was $E = -0.4997$ Ha and was found in $t = 59.5756$ s.

5.4.2 Stochastic variational optimization for Helium

For Helium, the results are summarized in table 3 and 4.

Note the improved results using the paired basis (smaller deviation in all cases and faster convergence time in all but the case of using 20 Gaussians) - thus, for the hybrid optimization, the paired basis will be used exclusively in agreement with the results of [7].

n_{gauss}	Convergence time	Convergence energy	Deviation
5	55.1400 s	-2.8007 Ha	3.5572%
10	54.0046 s	-2.8494 Ha	1.8802%
20	55.7271 s	-2.8677 Ha	1.2500%
30	58.2705 s	-2.8702 Ha	1.1639%

Table 3: Results of stochastic optimization with a random basis for Helium

n_{gauss}	Convergence time	Convergence energy	Deviation
5	48.3285 s	-2.8484 Ha	1.9146%
10	50.7632 s	-2.8688 Ha	1.2121%
20	57.4681 s	-2.8768 Ha	0.9366%
30	57.3349 s	-2.8749 Ha	1.0021%

Table 4: Results of stochastic optimization with a paired basis for Helium

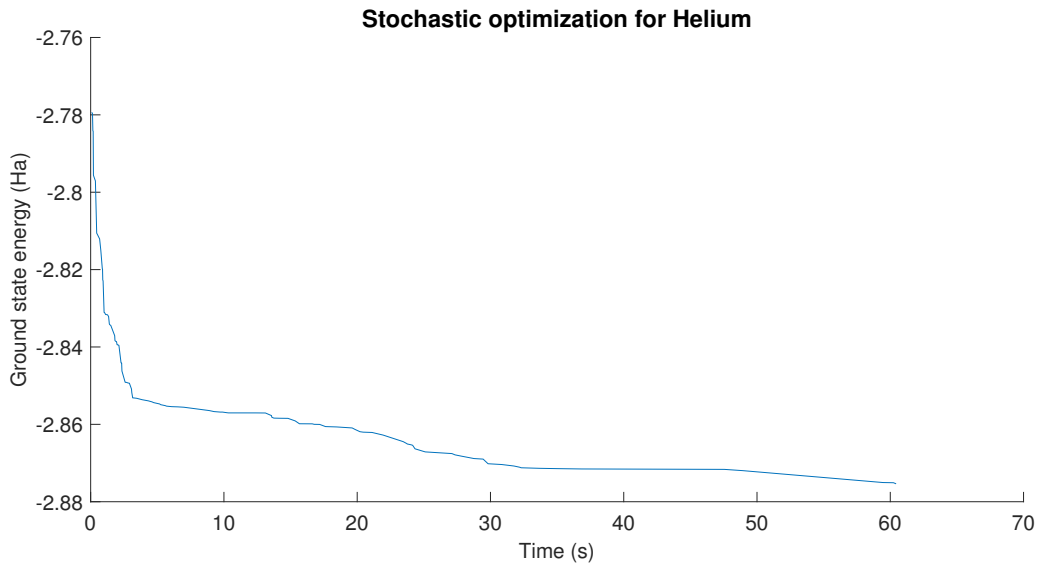


Figure 3: The ground state energy as a function of time for Helium using the stochastic optimization algorithm with 20 Gaussians. For this example the convergence energy was $E=-2.8754$ Ha and was found in $t=60.4393$ s.

For the sake of comparison with the results of section 5.6.1, the code was run with a maximum runtime of 15 s. The results were a convergence energy of $E=-2.8637$ Ha and was found in $t=12.3502$ s.

5.5 Analytic downhill gradient optimization

For the analytic downhill gradient method the program is fed a random starting point (for a Gaussian) and does downhill gradient- optimization on every single iteration. If the next iteration provides a lower energy, that starting point is saved and the process is repeated for the next basis-element (Gaussian).

5.5.1 Analytic downhill gradient optimization for Hydrogen

The result for the analytic downhill gradient optimization for Hydrogen using a runtime of 60s and 5 Gaussians is a convergence time of $t=46.1017$ s and a convergence energy of $E=-0.4988$ Ha. This deviates 0.24% from the expectation value.

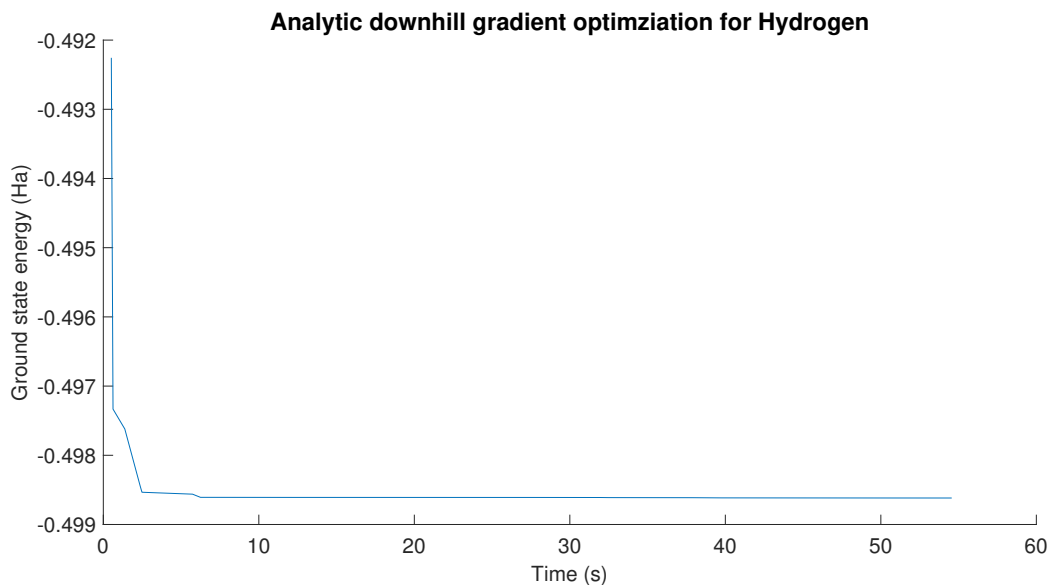


Figure 4: The ground state energy as a function of time for Hydrogen using the analytic downhill gradient optimization algorithm with 5 Gaussians. For this example the convergence energy was $E=-0.4986$ Ha and was found in $t=54.5578$ s.

5.5.2 Analytic downhill gradient optimization for Helium

The result for the analytic downhill gradient optimization using a runtime of 60 sec and 10 Gaussians is a convergence time of $t=75.0547$ s and a convergence energy of $E=-2.7857$ Ha. This deviates 4.0737 % from the expectation value.

The fact that the program goes over the runtime by 15.0547 s (25%) indicates that it does not have time to complete a lot of iterations with this amount of Gaussians.

For completeness, and to see if the above deduction is correct, the method was also tested using a runtime of 60 s and 20 Gaussians. To prove the claim in section 5.1 that the tolerance matters more the more Gaussians used, this method was tested using both a step size tolerance $\text{tol} = 10^{-10}$ and $\text{tol} = 10^{-4}$.

The result for $\text{tol} = 10^{-10}$ was a convergence time of $t=595.41$ s and a convergence energy of $E=-2.8075$ Ha. This deviates 3.3230 % from the expectation value.

The result for $\text{tol} = 10^{-4}$ was a convergence time of $t=334.20$ s and a convergence energy of $E=-2.8129$ Ha. This deviates 3.1371 % from the expectation value.

This almost halving of the convergence time whilst still getting a similar (slightly better) result shows that for more Gaussians the lower tolerance is still better, justifying the reasoning behind calculating the tolerance before the Gauss convergence.

The almost fivefold increase in convergence time and the only slightly improved deviation from the expectation value confirms that about 10 Gaussians is the most optimal to use, in agreement with 1. This long of a convergence time given a runtime of 60 s also shows that for 20 Gaussians the program is only able to complete one iteration, hinting at analytic gradients taking very long to calculate and evaluate. Furthermore, one iteration is obviously not enough to refine and get within the valley of the global minimum, thus the program is merely being fed a starting point and converging (for each Gaussian) to the local minima.

This indicates testing the program with a runtime of 60 s and 5 Gaussians could be beneficial. The result of this is a convergence time of $t=50.9651$ s and a convergence energy of $E=-2.7702$ Ha. This deviates 4.6074% from the expectation value.

Compared to the 10 Gaussians, the algorithm using 5 Gaussians deviate 0.53370% more but also converge 24.0896 s faster. The time can not be commented on, due to the fact that code used to create figure 1 shares the tendency to go over the runtime. However, it is in agreement with more Gaussians being more accurate, and the result is deemed to be in agreement with figure 1, that 10 Gaussians seems to be optimal.

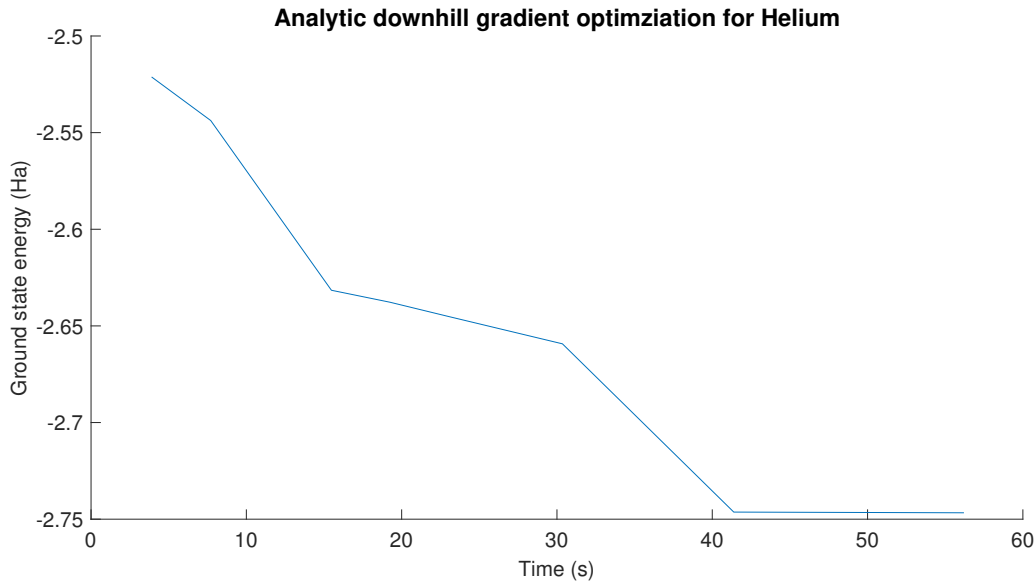


Figure 5: The ground state energy as a function of time for Helium using the analytic downhill gradient optimization algorithm with 10 Gaussians. For this example the convergence energy was $E=-2.7467$ Ha and was found in $t=56.2049$ s.

5.6 Hybrid optimization

For the hybrid optimization method, the program runs stochastic optimization for x amount of time and then takes its best guess of parameters and uses as a starting point for the downhill gradient method. The idea is to get within the valley of the global minimum and have the quick convergence that the downhill method provides.

To get an estimate of how long it takes the stochastic approach to get within the valley of the global minimum, the code was tested using 5 and 10 Gaussians for Hydrogen and Helium respectively and various different stochastic runtimes (followed by 60 sec of downhill gradient optimization). The results can be seen in table 5 for Helium. For Hydrogen, no improvement was seen switching to the analytic gradient method for any stochastic runtime - thus, hybrid optimization for Hydrogen has been omitted from this thesis.

Stochastic runtime	Convergence time	Convergence energy	Deviation	stochastic runtime/total runtime
10 s	20.5289 s	-2.8455 Ha	2.0145%	49%
20 s	27.2093 s	-2.8464 Ha	1.9835%	74%
30 s	30.4774 s	-2.8546 Ha	1.7011%	98%
40 s	51.8087 s	-2.8536 Ha	1.7355%	77%
50 s	72.7095 s	-2.8593 Ha	1.5393%	69%
60 s	77.8766 s	-2.8591 Ha	1.5461%	77%

Table 5: Results of hybrid optimization with various stochastic optimization runtimes for Helium using 10 Gaussians. The runtime for downhill gradient optimization is 60 s.

For Helium there is only a minor reduction in error using more time than 10 sec. To make this optimization strategy more competitive (that is, to reduce its total runtime), the stochastic optimization will be set to run for 10 s.

Also note that the convergence time only goes slightly above the stochastic runtime - that signals that the analytic gradient method is not very effective, as it does not find a better result given an entire 60 s than what the stochastic optimization has already found. Overall this tendency gets worse and worse the more stochastic runtime the program has (see table 5).

5.6.1 Hybrid optimization for Helium

This code was tested for 5, 10, 20 and 30 Gaussians. Based on the results in table(5 the code is set to run 10 s stochastic optimization and 60 s downhill gradient optimization. The results are summarized

n_{gauss}	Stochastic time	Total time	Stochastic energy	Total energy	Improvement	Deviation
5	8.6123 s	14.097 s	-2.8200 Ha	-2.8218 Ha	0.063800%	2.8306%
10	8.6772 s	14.309 s	-2.8395 Ha	-2.8401 Ha	0.021100%	2.2004%
20	11.3974 s	20.9714 s	-2.8372 Ha	-2.8379 Ha	0.024700%	2.2782%
30	15.317 s	15.317 s	-2.8445 Ha	-2.8445 Ha	0%	2.0489%

Table 6: Results of hybrid optimization for Helium using 10 stochastic optimization runtime and 60 s downhill gradient optimization runtime

in table 6.

Notice that for 30 Gaussians no improvement is seen from the switch, likely due to the amount of time it takes to evaluate the gradients in such a large parameter space.

5.7 Global optimization

Global optimization works by making a new guess for all Gaussians at once, opposed to replacing one Gaussian at a time, testing to see if it lowers the energy and moving on to the next Gaussian - also known as local optimization.

Global optimization has been tested on both the analytic gradient optimization and on the stochastic optimization.

5.7.1 Global analytic gradient optimization

5.7.1.1 Global analytic gradient optimization for Hydrogen This algorithm was tested using 5 Gaussians and a runtime of 60 s. The result was as follows: the convergence time was $t=24.4662$ s and the convergence energy $E=-0.4980$ Ha, which is a 0.4000% deviation from the expectation value.

Comparing with the local optimization done in section 5.5.1 ($t=55.4842$ s, $E=-0.4991$ Ha) we note a 0.2200 increase in deviation from the expectation value despite a 31.0180 s decrease in runtime.

To test if this global optimization is better, we give the local optimization a runtime of 25 sec and see if it can find a more accurate result. One obtains a convergence time of $t=18.1370$ s and a convergence energy of $E=-0.4985$ Ha, which is a 0.3000% deviation from the expectation value.

5.7.1.2 Global analytic gradient optimization for Helium This algorithm was tested using 10 Gaussians and a runtime of 60 sec. The results are as follows: the convergence time was $t=71.4165$ s and the convergence energy $E=-2.7636$ Ha, which is a 4.8347% deviation from the expectation value.

Comparing with the local optimization done in section 5.5.2 ($t=75.0547$ s, $E=-2.7857$ Ha (4.0737% deviation)) we note a 0.7610% increase in deviation from the expectation value and a 3.6382 s decrease in convergence time.

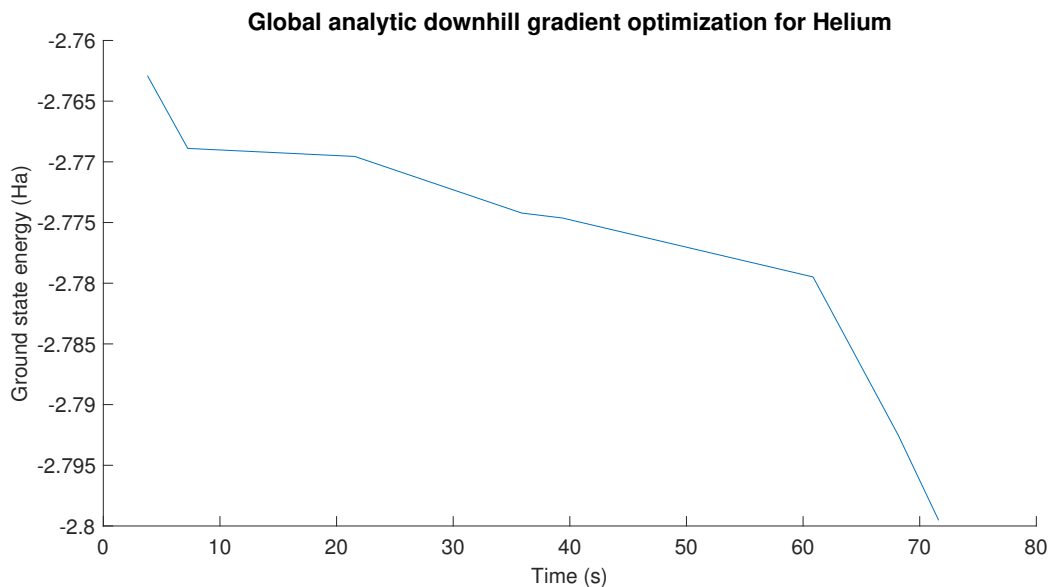


Figure 6: The ground state energy as a function of time for Helium using the global analytic downhill gradient optimization algorithm with 10 Gaussians. For this example the convergence energy was $E=-2.7995$ Ha and was found in $t=71.6248$ s.

5.7.2 Global stochastic optimization

This algorithm works by doing global optimization for the stochastic optimization as explained in section 5.7.

5.7.2.1 Global stochastic optimization for Hydrogen This algorithm was tested using 5 Gaussians and a runtime of 60 s. The result was as follows: the convergence time was $t=59.4537$ s and the convergence energy was $E=-0.4996$ Ha, which is a 0.08000% deviation from the expectation value. Comparing with local optimization done in section 5.4.1 ($t=59.5298$ s, $E=-0.4996$ Ha), no significant change in convergence time (a 0.0761 s decrease) is observed as well as no change in deviation from the expectation value.

This shows that for Hydrogen, it does not matter whether one uses local or global optimization when using the stochastic variational method (when using 5 Gaussians).

5.7.2.2 Global stochastic optimization for Helium This algorithm was tested using 10 Gaussians and a runtime of 60 sec. The result was as follows: the convergence time was $t=54.3319$ s and the convergence energy was $E=-2.8554$ Ha, which is a 1.6736% deviation from the expectation value. Comparing with the local optimization (table 4) ($t=50.7632$ s, $E=-2.8688$ Ha) we note both an increase in convergence time (3.5687 s) and deviation from the expectation value (0.4615%).

This result shows that for the stochastic variational method, local optimization is better than global optimization.

6. Discussion

In this section, the results of the various algorithms and their implications for analytic gradients will be discussed, in particular if any algorithm involving them are useful as opposed to the stochastic variational method, which is the standard method today or even the numerical gradients used in [7].

Most of the analysis will pertain to Helium, as this is a more complicated system and much more interesting to analyze.

The algorithms will be judged on both their speed (convergence time) and accuracy (convergence energy/deviation).

Had there been any ambiguity (had, for instance the convergence time been better but the result been worse), one could test which algorithm was best by doing the following: set the runtime of the slower (but more precise) algorithm equal to the convergence time of the faster (but less precise) algorithm and see if the result is still more precise. If so, that algorithm is the best.

There are two main factors that will influence the results - the amount of local minima in the parameter space will affect the accuracy, whereas the computing power needed to calculate and evaluate the gradients will affect the speed.

To test the gradients, the best result of the stochastic optimization method is upheld to the best results of the algorithms involving analytic gradients. This is done by looking at the result of the algorithms involving analytic gradients and seeing if the stochastic algorithm can find a result that deviates less from the expectation value in the same or less time.

6.1 Hydrogen

For Hydrogen, both local and global stochastic optimization were tested - it was found that global optimization did not offer any advantage over local optimization, as the change in deviation from the expectation value was the same and the convergence times were very similar (0.0761 s decrease using global).

For Hydrogen, two different algorithms involving analytic gradients were tested; the (local) analytic downhill gradient optimization and the global analytic downhill gradient optimization.

For the case of local vs. global analytic downhill gradient optimization, it was found in section 5.7.1.1 that the local optimization offered a more precise result (0.3000% deviation vs 0.4000%) in a shorter time (18.1370 s vs 24.4662 s) - thus, local optimization works better.

Comparing the local analytic downhill gradient optimization results to those of the stochastic optimization, we find a convergence time 46.1017 s vs 59.5429 s and a deviation from the expectation value of 0.2400% vs 0.04000% - despite the longer runtime, the stochastic method is way more precise and a more accurate result can than a deviation of 0.2400% can be obtained in 46.1017 s using the stochastic method.

6.2 Helium

For Helium, both local and global stochastic optimization were tested - it was found that local optimization was better, as it deviated less (0.4615%) from the expectation value in a shorter convergence time (3.5687 s). This is likely due to the refining process of optimizing one Gaussian at a time is more efficient in a larger parameter space, as a completely random guess is less likely to get you in the vicinity of a minima.

For Helium, three different algorithms involving analytic gradients were tested; the (local) analytic downhill gradient optimization, the global analytic downhill gradient optimization and the hybrid optimization.

In section 5.6.1 the hybrid optimization algorithm was tested for Helium, and the improvement the gradient method made to the stochastic guess noted. It was found that the smallest deviation from the expectation value (2.0489 %) coincided with the case where the gradient method improved the result 0 % (see table 6). The fact that the best result for hybrid optimization is obtained with the purely stochastic approach directly shows that hybrid optimization using analytic gradients is not more efficient than stochastic optimization.

For the case of local vs. global analytic downhill gradient optimization, it was found in section 5.7.1.2 that global optimization not more efficient than local. The best result using analytic gradients was found in section 5.5.2 to be using 10 Gaussians ($E=-2.7857$ Ha, $t=75.0547$ s). The local stochastic optimization got a more accurate result in less time than this algorithm ($E=-2.8768$ Ha, $t=57.4671$ s) and thus pr. default also beats the global analytic downhill gradient algorithm.

A possible explanation for this is that the code required to calculate and evaluate the gradients is too complex compared to merely taking a random guess, i.e. the stochastic approach. It is simply be more computationally efficient to make random guesses than to calculate and evaluate gradients. This can be tested by looking at Matlab's code-analyzer. For simplicity, let us compare the results of the code analyzer for local stochastic optimization and local analytic downhill gradient optimization. The breakdown of the main routines can be found in figure 7 and 8.

Function Name	Calls	Total Time	Self Time*	Total Time Plot (dark band = self time)
HeStokastiskPairedBasis	1	60.142 s	0.003 s	
...airedBasis>stochastic_optimization	1	60.135 s	0.029 s	
HeStokastiskPairedBasis>paired_basis	2061	60.023 s	0.878 s	
HeStokastiskPairedBasis>build_HN	2061	58.928 s	5.918 s	
HeStokastiskPairedBasis>matrix_elements	1690020	53.010 s	47.397 s	

Figure 7: Matlab code-analyzer of the stochastic algorithm for Helium

Function Name	Calls	Total Time	Self Time*	Total Time Plot (dark band = self time)
HeAnalytiskgradientPairedBasis	1	106.653 s	0.004 s	
...entPairedBasis>hybrid_optimization	1	106.644 s	0.086 s	
fminunc	20	106.369 s	0.209 s	
...new_x+[zeros_before_a.zeros_after],i)	880	105.098 s	0.025 s	
HeAnalytiskgradientPairedBasis>myFun	880	105.072 s	0.017 s	
sfminbx	20	103.638 s	0.212 s	
...radientPairedBasis>matrix_elements	554610	101.864 s	85.287 s	
optimlib\private\sfd	220	78.836 s	0.102 s	
...tPairedBasis>paired_basis_gradient	880	69.983 s	0.205 s	
HeAnalytiskgradientPairedBasis>build_HN	1761	69.730 s	1.793 s	
...skgradientPairedBasis>paired_basis	881	35.156 s	0.212 s	
...radientPairedBasis>build_gradient	880	34.894 s	0.967 s	

Figure 8: Matlab code-analyzer of the analytic downhill gradient algorithm for Helium

For both cases, the main routine taking up time is matrix_elements. However, this routine is not the same for both algorithms. For the stochastic algorithm it is simply the energy- and overlap- calculations derived in section 3.2.1, 3.2.2 and 3.2.3. For the analytic downhill gradient algorithm it also includes the calculations derived in section 3.2.4.

The result is that for the stochastic algorithm, this routine can be called upon 35.657 times pr. second, whereas for the analytic gradients, this drops down to 6.5029 times pr. second, which is approximately 1/6th.

From this, one concludes that the extra calculations needed to make the gradients reduces the efficiency of the algorithm too much to make up for the theoretical potential of the faster convergence.

7. Conclusion

From the work presented in this thesis one can conclude that using the ECG method for few-body systems, the fastest and most accurate optimization algorithm is the stochastic one.

To obtain accuracy, the more Gaussians/parameters you use, the better. This is part of the reason why the stochastic algorithm is more effective than the downhill gradient algorithm - the more parameters used, the more time each iteration to the energy takes as more calculations are needed. The stochastic method omits most of this by simply guessing a new Gaussian and seeing if the energy is lower or not, and if it is, keeping that Gaussian and repeating the process for the next basis element. Due to the computational inefficiency of the analytic downhill gradient method as compared to the stochastic method, one can thus test many more different Gaussians with the stochastic version and thus converge quicker - and apparently more accurate, given a limited runtime.

The analytic downhill gradient method needs a random starting point and then time to converge to the nearest minima - the accuracy of this is skewed due to the fact that a given parameter space has multiple local minima and only one global. If not enough iteration can be completed due to the time it takes to evaluate the gradients, one simply does not get within the valley of this global minimum. This problem is avoided with the stochastic method, as it can run through many more Gaussians/starting points in the same amount of time.

Theoretically, combining the two methods might work - that is, run the stochastic variation method long enough to get within the valley of the global minimum and then use that as a starting point for the analytic downhill gradient method. This was explored but with no noticeable results - the improvements from the switch were small and could still easily be beaten by the pure stochastic method given the same amount of runtime.

This idea has also been explored with the numerical downhill gradient method in [7] and it was found to work better than stochastic optimization (and thus also this variant of hybrid optimization) - this suggests that the culprit is the time it takes to calculate the analytic gradients. This may be circumvented by using a faster, more efficient coding-language than what Matlab offers and is an interesting area for future study.

Furthermore, global optimization offers no advantage to local optimization in any system tested (Hydrogen and Helium) using any stochastic- or analytic downhill gradient- optimization algorithms.

In conclusion, analytic gradients are not better than traditional methods, it takes up too much computer power to calculate them and there are too many local minima for it to be efficient.

8. Summary

8.1 Summary in English

In variational calculations with correlated Gaussians the optimization of the basis (minimization of the energy) is the task which takes the longest time. One popular approach is to first run stochastic optimization in order to approximately find the global minimum and then run deterministic optimization (downhill gradient method) to quickly find the precise position of the minimum. The downhill methods converge fast but need the gradients of the matrix elements for their operations. The gradients are notoriously difficult to calculate numerically. However for certain potentials (including Coulomb) the gradients can be actually calculated analytically. So the goal of this project is to calculate analytically the gradients of the matrix elements with correlated Gaussians for the Coulomb potential, apply the analytic gradients to calculations of small atoms, and estimate how efficient they are as compared to traditional methods.

This was done by testing if more accurate results for the ground state energy of Hydrogen and Helium were obtained in the same or less time as traditional methods (i.e stochastic optimization) after having the program provide a starting guess based on stochastic optimization to the deterministic optimization (analytic downhill gradient method). By more accurate results, one means closer to the experimental value, which for Hydrogen is -0.5000 Ha and for (para)Helium is -2.9040 Ha. For each result the code is set to run 10 times and take an average to avoid lucky/unlucky starting guesses leading to false-positive results.

The stochastic time, total time, stochastic energy and total energy were noted for both Hydrogen and Helium. The stochastic method was set to run for a maximum of 10 seconds. For Hydrogen using 5 Gaussians no improvements to the stochastic results were seen. For Helium using 10 Gaussians a 0.021100% improvement in the result was seen from the switch (stochastic time=8.6772 s, total time=14.309 s, stochastic energy=-2.8395 Ha, total energy=-2.8401 Ha). However, running the stochastic algorithm with a maximum runtime of 15 s using 10 Gaussians yielded an energy of $E=-2.8637$ Ha in a time $t=12.3502$ s, which is both closer to the experimental value of the ground state energy and was found in a shorter amount of time.

Another possible application for analytic gradients was also tested - the strictly deterministic approach, that is, a random starting point is provided and the program then converges to the nearest minima for each basis element (and keeps that point if it provides a lower energy than the previous iteration). The main result from this for Hydrogen using 5 Gaussian was an energy of $E=-0.4988$ Ha found in $t=46.1017$ s and for Helium using 10 Gaussians was an energy of $E=-2.7857$ Ha found in $t=75.0547$ s. Both of these results are beat by the stochastic algorithm which found for Hydrogen using 5 Gaussians an energy of $E=-0.4996$ Ha in $t=59.5298$ s and for Helium using 10 Gaussians an energy of $E=-2.8688$ Ha in $t=50.7632$ s.

Global optimization for both these algorithms (stochastic and analytic downhill gradient) was also tested. For the stochastic code it was found to worsen the result compared with local optimization - the result for Hydrogen using 5 Gaussians was an energy of $E=-0.4996$ Ha found in a time $t=59.4537$ s and for Helium using 10 Gaussians was an energy of $E=-2.8554$ Ha found in $t=54.3319$ s. For the analytic downhill gradient code the result worsened as well - the result for both Hydrogen and Helium was that the local optimization got a more accurate result in less time.

The main result of this thesis is thus a negative one - analytic gradients are just not better, it takes up too much computer power to calculate them and there are too many local minima in the parameter space for them to be efficient.

8.2 Resume på Dansk

I variations-beregninger indenfor den korrelerede gaussiske metode er det basis-optimeringen (minimering af energien) der tager længst tid. En populær måde at optimere sin basis er først at køre stokastisk optimering for at komme i bakkedalen af det globale minimum og derefter køre deterministisk optimering for hurtigt at konvergerer til minimumet. De deterministiske metoder konvergerer hurtigt men skal bruge gradienterne af matrix elementerne for at anvendes. Gradienter er notorisk svære at beregne numerisk, men det viser sig at for visse potentialer (herunder Coulomb) kan gradienterne beregnes analytisk. Målet med dette projekt er at udregne gradienterne af matrix elementerne analytisk med korrelerede Gauss-funktioner for et Coulomb potentiale, anvende de analytiske gradienter til udregningen af grundtilstandsenergien for små atomer, og estimerer hvor effektive disse analytiske gradienter er i forhold til traditionelle metoder, herunder primært den stokastiske metode.

Dette blev gjort ved at teste om man fik mere præcise resultater for grundtilstandsenergien af Hydrogen og Helium på kortere tid end ved traditionelle metoder (stokastisk optimering) ved at lade koden lave et start-gæt baseret på stokastisk optimering til den deterministiske optimering. Med mere præcise resultater menes der tættere på den eksperimentelle værdi af grundtilstandsenergien, som for Hydrogen er -0.5000 Ha og for (para)Helium er -2.9040 Ha. For hvert resultat er koden sat til at køre 10 gange og tage et gennemsnit for at undgå at heldige/uheldige start-gæt leder til falske konklusioner.

Tiden koden kører stokastiske optimering (den stokastiske tid), den totale tid, den stokastiske energy og den totale energien blev noteret for både Hydrogen og Helium. Den stokastiske optimering var sat til at køre i maksimum 10 sekunder. For Hydrogen med 5 Gauss-funktioner var der ingen forbedring fra skiftet til den deterministiske optimering. For Helium med 10 Gauss-funktioner så man en forbedring i resultatet på 0.021100 % (stokastisk tid= 8.6772 s, total tid= 14.309 s, stokastisk energi= -2.8395 Ha, total energi= -2.8401 Ha). Satte man dog den stokastiske algoritme til at køre med en maksimal tid på 15 s fandt man med 10 Gauss-funktioner en energi $E=-2.8637$ Ha på $t=12.3502$ s, hvilket både er tættere på den eksperimentelle værdi for grundtilstandsenergien og den blev fundet på en kortere tid.

En anden mulig brug af analytiske gradienter blev også afprøvet - den rene deterministiske metode, hvori et tilfældigt startgæt bliver lavet til et basis element og koden så konvergerer til den nærmeste minima ved brug af de analytiske gradienter (og beholder dette punkt hvis det giver en lavere energi end det forhenværende). Hovedresultatet heraf for Hydrogen med 5 Gauss-funktioner var en energi på $E=-0.4988$ Ha fundet på $t=46.1017$ s og for Helium med 10 Gauss funktioner var det en energi på $E=-2.7857$ Ha fundet på $t=75.0547$ s. Begge af disse resultater kunne slås af den stokastiske algoritme som for Hydrogen med 5 Gauss-funktioner fandt en energi på $E=-0.4996$ Ha på $t=59.4537$ s og for Helium med 10 Gauss-funktioner fandt en energi på $E=-2.8688$ Ha på $t=50.7632$ s. Global optimering for disse algoritmer (stokastisk og deterministisk) blev også efterprøvet. For den stokastiske algoritme blev resultatet værre ifht. lokal optimering - for Hydrogen med 5 Gauss-funktioner blev der fundet en energi på $E=-0.4996$ Ha på $t=59.4537$ s og for Helium med 10 Gauss-funktioner blev der fundet en energi på $E=-2.8554$ Ha på $t=54.3319$ s. For den deterministiske algoritme blev resultatet heller ikke bedre - resultatet for både Hydrogen og Helium var mere præcist og tog mindre tid at finde med lokal optimering.

Hovedresultatet af denne afhandling er altså negativt - analytiske gradienter er ikke mere effektive end traditionelle metoder, da det tager for meget computerkraft at udregne dem og der er for mange lokale minimumer for at det kan være effektivt.

References

- [1] Jim Mitroy et al. “Theory and application of explicitly correlated Gaussians.” In: *Reviews of modern physics* 85.april-june (2012), p. 713, table IV.
- [2] John Bohn. *A note on units*. URL: https://www.colorado.edu/physics/phys7550/phys7550_sp07/extras/Appendix_1.pdf. (accessed: 25.05.2018).
- [3] D.V. Fedorov. “Analytic Matrix Elements and Gradients with Shifted Correlated Gaussians.” In: (2016).
- [4] *GNU project: GNU Scientific Library Reference Manual: "Choletsky Decomposition"*. URL: https://www.gnu.org/software/gsl/manual/html_node/Cholesky-Decomposition.html. (accessed: 24.05.2018).
- [5] D. Griffiths. *Introduction to Quantum Mechanics - Second Edition*. Pearson Prentice Hall, 2005.
- [6] K. D. Fink J.H. Mathews. *Numerical Methods Using MATLAB - Third Edition*. Upper Saddle River: Prentice Hall, 1999.
- [7] Jesper B. Rasmussen. “Optimization Algorithms Correlated Gaussian Method”. In: (2017).
- [8] K. Varga Y. Suzuki. *Stochastic Variational Approach to Quantum-Mechanical Few-Body Problems*. Springer,