

## Lab Course of Scientific Computing

---

# Quantum Many-Body Simulations in Haskell using the Hartree-Fock Method

prepared by

**David Schlegel**

at the Institute of Numerical and Applied Mathematics  
Georg-August-Universität Göttingen

Supervisor: Dr. Jochen Schulz



# Contents

<b>1. Introduction</b>	<b>1</b>
<b>2. Simple Quantum Systems</b>	<b>2</b>
2.1. Discretization of the kinetic Energy . . . . .	2
2.1.1. Method of Finite Differences . . . . .	3
<b>3. The Hartree-Fock Method</b>	<b>6</b>
3.1. The Hartree-Fock Method and Equations . . . . .	7
3.2. The Roothaan Equation . . . . .	8
3.3. Closed and open shell systems . . . . .	10
3.4. Basis set and basis functions . . . . .	11
3.5. Program Structure . . . . .	13
<b>Appendices</b>	<b>16</b>
<b>A. Basis Functions</b>	<b>16</b>

# 1. Introduction

Applying numerical methods in quantum mechanics has always been necessary in analyzing complex structures of quantum mechanical systems. The technical progress of computer performance has enabled physicists and mathematicians to simulate complex many-body systems. With these methods tangible progress in quantum physics can be made to analyze quantum phenomena on the level of many-particle interactions.

This article tackles the implementation of the Hartree-Fock method for many-body simulations in the functional programming language Haskell. Functional programming languages are getting more and more interesting for physicists through their mathematical way of implementation. In this article simple quantum systems are simulated first to show how a simple one-body system can be simulated straight forward in Haskell.

In chapter 3, the main work in this article will be presented, namely the attempt to provide a Haskell function library for simulating physical many-body systems. A general overview of how the implementation can be done as well as problems will be treated in this chapter. Here it is important to note that the full implementation still needs to be completed. Thus, within the progress of this work beyond this report, other problems may arise and might be done in a different manner to increase performance, structure, or simplicity. The implementation so far will be presented as well as the problem of calculating necessary Gaussian integrals.

Although a lot of effort still needs to be put in this project, I hope to provide a structural attempt to show that the implementation of the Hartree-Fock Method in Haskell stands out in its simplistic way, with the strong advantage of reproducibility – due to its functional operations.

The progress of this project can be tracked on the following repositories:

- GWDG: [https://gitlab.gwdg.de/scientific\\_practical/project\\_qm](https://gitlab.gwdg.de/scientific_practical/project_qm) (University Access only)
- GitHub: <https://github.com/davidschlegel/hartree-fock>

A detailed html documentation about important functions for the Hartree-Fock Method can be found in the `/dist/doc/html/hartree-fock/` directory.

## 2. Simple Quantum Systems

In this chapter simple quantum systems will be studied. Here, we consider a particle in a three-dimensional potential  $V(\vec{x})$ . The corresponding wave function  $\psi(\vec{x}, t)$  is the solution of the Schrödinger equation

$$i\hbar \frac{\partial}{\partial t} \psi(\vec{x}, t) = H\psi(\vec{x}, t) = -\frac{\hbar^2}{2m} \Delta \psi(\vec{x}, t) + V(\vec{x})\psi(\vec{x}, t), \quad (1)$$

where  $\Delta$  is the Laplacian differential operator:  $\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}$ . For a time-independent potential  $V(\vec{x})$ , the Schrödinger equation can be formally solved by

$$\psi(\vec{x}, t) = U(t, t_0)\psi(\vec{x}, t_0) = \exp \left\{ -\frac{i(t-t_0)}{\hbar} H \right\} \psi(\vec{x}, t_0). \quad (2)$$

For a time-dependent potential, like an oscillating laserfield, the time evolution of the wave function becomes

$$\begin{aligned} \psi(\vec{x}, t) &= U(t, t_0)\psi(\vec{x}, t_0) = \hat{T}_t \exp \left\{ \frac{i}{\hbar} \int_{t_0}^t H(\tau) d\tau \right\} \psi(\vec{x}, t_0) \\ &= \sum_{n=0}^{\infty} \frac{1}{n!} \left( \frac{-i}{\hbar} \right)^n \int_{t_0}^t dt_1 \int_{t_0}^{t_1} dt_2 \cdots \int_{t_0}^{t_{n-1}} dt_n \hat{T}_t \{ H(t_1)H(t_2) \dots H(t_n) \}, \end{aligned} \quad (3)$$

where  $\hat{T}_t$  is the time ordering operator. A simple approach is to divide the interval  $[0 \dots t]$  into a sequence of  $N$  steps so that

$$U(t, t_0) = U(t, t_{N-1}) \dots U(t_2, t_1)U(t_1, t_0) \quad (4)$$

and to neglect small deviations of the Hamiltonian in the small interval  $\Delta t = t_n - t_{n-1}$ .

### 2.1. Discretization of the kinetic Energy

Dividing the Hamiltonian  $H$  into  $H = T + V$ , the non-local kinetic energy operator can be written as

$$T\psi(\vec{x}, t) = -\frac{\hbar^2}{2m} \Delta \psi(\vec{x}, t). \quad (5)$$

### 2.1.1. Method of Finite Differences

There are different methods and approaches to solve the above equation with numerically. Here we will focus on the approach using finite differences, due to its easy way of implementation, with still very good results. This approach is also computationally stable for many given potential, thus this method can be used in many applications in quantum mechanics.

Taking a grid  $(k, l, m)$  in three dimensions, the kinetic energy operator can be approximated by finite differences

$$T\psi(\vec{x}, t) \approx -\frac{\hbar^2}{2m} \left( \frac{\psi_{(k+1,l,m)}^n - 2\psi_{(k,l,m)}^n + \psi_{(k-1,l,m)}^n}{\Delta x^2} + \frac{\psi_{(k,l+1,m)}^n - 2\psi_{(k,l,m)}^n + \psi_{(k,l-1,m)}^n}{\Delta y^2} + \frac{\psi_{(k,l,m+1)}^n - 2\psi_{(k,l,m)}^n + \psi_{(k,l,m-1)}^n}{\Delta z^2} \right), \quad (6)$$

with higher order terms  $\mathcal{O}(\Delta x^2, \Delta y^2, \Delta z^2)$  where  $n$  represents the discrete time index of the wave function. Considering the time independent Schrödinger equation, we can write the operator in one dimension as a matrix satisfying the eigenvalue equation

$$\left[ \begin{pmatrix} 2 & -1 & & \dots & 0 \\ -1 & 2 & -1 & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ 0 & \dots & & -1 & 2 \end{pmatrix} + V_{kk} \right] \begin{pmatrix} \psi_1 \\ \vdots \\ \psi_k \\ \vdots \\ \psi_N \end{pmatrix} = E \begin{pmatrix} \psi_1 \\ \vdots \\ \psi_k \\ \vdots \\ \psi_N \end{pmatrix}, \quad (7)$$

where  $\vec{\psi}$  are the values of the wave function on the evaluation points  $x_k$ .  $V_{kk}$  represents the potential for each  $x_k$ , thus it is a diagonal matrix.

For the example of a particle in a box the potential is

$$V_{kk} = \begin{cases} \infty & \text{for } k = 0, k = N \\ 0 & \text{else} \end{cases}. \quad (8)$$

Solving the above eigenvalue equation yields the eigenfunctions  $\psi_n$  with eigenenergies  $E_n$ . The boundary conditions  $V = \infty$  at  $k = 0$  and  $k = N$  are satisfied even when the boundary conditions are left out. An example of the first four eigenstates is shown in figure 1. To reduce the problem of a particle in three dimensions to a

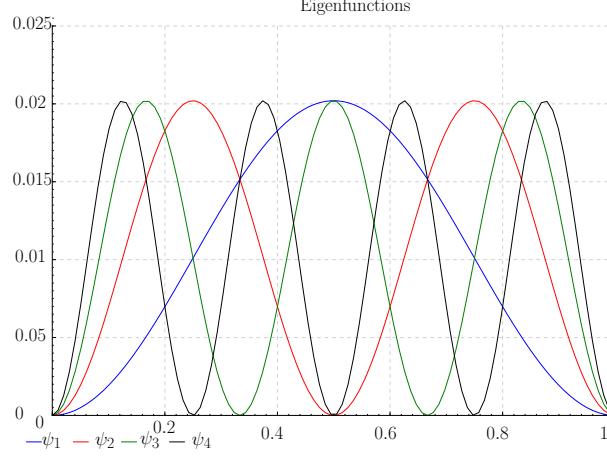


Figure 1: *Probability densities for the calculated eigenfunctions  $\psi_n$ , with the method of finite differences and a grid of  $N = 100$  for  $n = 1, 2, 3$  and 4 in one dimension. For simplicity the pre-factor  $\frac{\hbar^2}{2m}$  was set to one, obtaining non-normalized eigenfunctions.*

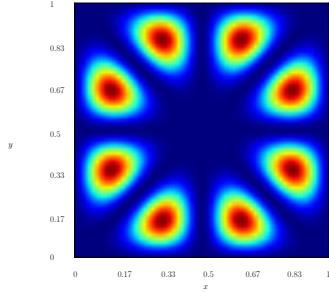
simple matrix equation, the spatial wave function  $\vec{\psi}$  can be stacked, to obtain a  $N^3$  dimensional vector through nested vectors,

$$\vec{\psi}_{k,l,m} = \begin{pmatrix} \vec{\psi}_{1,1,m} \\ \vdots \\ \vec{\psi}_{1,N,m} \\ \vdots \\ \vec{\psi}_{N,1,m} \\ \vdots \\ \vec{\psi}_{N,N,m} \end{pmatrix}. \quad (9)$$

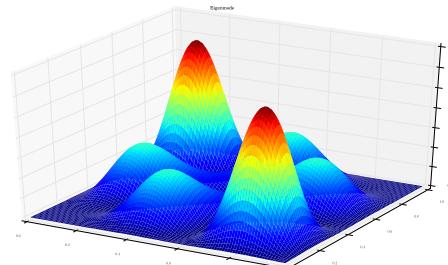
For two dimensions the approximated kinetic energy operator  $T$  thus becomes the tridiagonal block matrix

$$T \approx \begin{pmatrix} 4 & -1 & & & & & 0 \\ & \ddots & & & & & \ddots \\ & -1 & 4 & & & & -1 \\ -1 & & 4 & -1 & & & \ddots \\ & \ddots & & \ddots & & & \\ 0 & & -1 & & -1 & & 4 \end{pmatrix}. \quad (10)$$

In figure 2(a) an example of an eigenstate for a particle in a two dimensional box is shown.



(a)  $\psi_{12}$ ,  $N = 60$ .



(b)  $\psi_{34}$ ,  $N = 100$ .

Figure 2: Probability densities for the calculated eigenfunctions with the method of finite differences and a grid of  $N = 100$  and  $N = 60$  in two dimensions. For simplicity the pre-factor  $\hbar^2/2m$  was set to one, obtaining non-normalized eigenfunctions.

An implementation of this eigenvalue problem can be done in the following way:

```
--index function
index :: (Int, Int) -> Int -> Double
index (i,j) n
  | i == j      = 4
  | i == j + n = -1
  | i == j - n = -1
  | i == j + 1 = -1
  | i == j - 1 = -1
  | otherwise    = 0

-- Laplace Matrix in 2 dimensions
t :: Matrix Double
t = buildMatrix (n*n) (n*n) (\(i,j) -> index (i,j) n)

--Compute eigenvalues and corresponding eigenvectors
e = sort(toList(mapVector realPart (fst(eig(t)))))
psi = toColumns (mapMatrix realPart (snd(eig(t))))
```

In three dimensions, we obtain a  $N^3$ -dimensional Matrix, for example for  $N = 2$ :

$$T \approx \begin{pmatrix} & & & & & & 0 \\ 6 & -1 & -1 & & -1 & & \\ -1 & 6 & -1 & -1 & & -1 & \\ -1 & -1 & 6 & -1 & -1 & & -1 \\ & -1 & -1 & 6 & -1 & -1 & -1 \\ -1 & & -1 & -1 & 6 & -1 & -1 \\ & -1 & & -1 & -1 & 6 & -1 \\ 0 & & & -1 & & -1 & 6 \end{pmatrix}$$

Referring back to the time expansion in chapter 2, the time evolution of an eigenfunction of the Hamiltonian is trivial, since  $\psi_t$  only differs in phase from the initial state  $\psi_0$ , so  $|\psi_t|^2$  is time-independent, although linear combinations of eigenfunctions depend on time, which will not be calculated here.

If we take the computing time into account, it is obvious that even with a small grid size  $N < 100$  and parallelized computing the simulation of a particle in a square well potential leads to unexecutable applications. The increase of dimensions with constant gridsize results in a growing computing time higher than  $\mathcal{O}(t_0^n)$ .

To proceed to coupled quantum mechanical systems it seems to be mandatory to use efficient approximations.

### 3. The Hartree-Fock Method

We consider a real physical system, where particles are not independent, like in atoms, ions, molecules, etc. Assuming the system consists only of  $K$  nuclei and  $N$  electrons, the Hamiltonian consists of the terms

$$H = T_i + T_n + V_{ii} + V_{in} + V_{nn} ,$$

where  $T_i$  and  $T_n$  are the kinetic energies of the electrons and nuclei, respectively.  $V_{ii}$  represents the Coulomb repulsion between electrons,  $V_{nn}$  between the nuclei and  $V_{in}$

the attraction between electrons and nuclei. So the Hamiltonian reads

$$H = \sum_{i=1}^N \frac{p_i^2}{2m} + \sum_{n=1}^K \frac{P_n^2}{2M_n} + \frac{1}{4\pi\epsilon_0} \frac{1}{2} \sum_{i,j \neq 1, i \neq j}^N \frac{e^2}{|\mathbf{r}_i - \mathbf{r}_j|} \\ - \frac{1}{4\pi\epsilon_0} \sum_{i=1}^K \sum_{i=1}^N \frac{Z_n e^2}{|\mathbf{r}_j - \mathbf{R}_n|} + \frac{1}{4\pi\epsilon_0} \frac{1}{2} \sum_{n,n'=1; n \neq n'}^K \frac{Z_n Z_{n'} e^2}{|\mathbf{R}_n - \mathbf{R}_{n'}|},$$

where  $m$  is the electron mass and  $M_n$  the nuclei mass. The index  $i$  refers to the electrons, the index  $n$  to the nuclei. This hamiltonian describing the system looks quite complicated and in fact, computing the dynamics of this system seem to be unsolvable even for a few particles and an efficient super computer. Therefore, approximations must be made, still comprising the important information about the system. One approach is the Born-Oppenheimer approximation. It uses the fact that the nuclei are much heavier than the electrons, so the motions of the nuclei are much slower compared to the electrons, justifying to neglect the coulomb repulsion between the nuclei and the kinetic energy of the nuclei, so the approximated Hamiltonian becomes

$$H_{\text{BO}} = \sum_{i=1}^N \frac{p_i^2}{2m} + \frac{1}{4\pi\epsilon_0} \frac{1}{2} \sum_{i,j=1, i \neq j}^N \frac{e^2}{|\mathbf{r}_i - \mathbf{r}_j|} - \frac{1}{4\pi\epsilon_0} \sum_{i=1}^K \sum_{i=1}^N \frac{Z_n e^2}{|\mathbf{r}_j - \mathbf{R}_n|}. \quad (11)$$

The positions of the nuclei can be varied to find the minimum of the total energy.

### 3.1. The Hartree-Fock Method and Equations

In equation (11) the antisymmetry of the fermion wave functions was not taken into account. Fock extended the so called *Hartree equation* by taking antisymmetry into account. This approach is based on introducing antisymmetry of the wavefunction by constructing the so called Slater determinant, which is a determinant from single-particle wave functions:

$$\Psi_{\text{AS}}(\mathbf{x}_1, \dots, \mathbf{x}_N) = \frac{1}{\sqrt{N!}} \begin{vmatrix} \psi_1(\mathbf{x}_1) & \psi_2(\mathbf{x}_1) & \dots & \psi_N(\mathbf{x}_1) \\ \psi_1(\mathbf{x}_2) & \psi_2(\mathbf{x}_2) & \dots & \psi_N(\mathbf{x}_2) \\ \vdots & \vdots & & \vdots \\ \psi_1(\mathbf{x}_N) & \psi_2(\mathbf{x}_N) & \dots & \psi_N(\mathbf{x}_N) \end{vmatrix}$$

Interchanging spatial coordinates of two particles thus changes the sign of the slater determinant. It is important to note that the slater determinant already imposes a

restriction to the full wave function. Further improvements could be made to also take linear combinations of slater determinants into account.

The derivation of the *Hartree* and *Hartree-Fock equations* will not be given here, since the main aim here is the implementation of the method. Therefore see e.g. [11, p. 56-60] and [7, chapter 3]. The *Fock operator* is in natural units given by

$$\begin{aligned}\mathcal{F}\psi_k = & \left[ -\frac{1}{2}\nabla^2 - \sum_n \frac{Z_n}{|\mathbf{r} - \mathbf{R}_n|} \right] \psi_k(\mathbf{x}) + \sum_{l=1}^N \int d\mathbf{x}' |\psi_l(\mathbf{x}')|^2 \frac{1}{|\mathbf{r} - \mathbf{r}'|} \psi_k(\mathbf{x}) \\ & - \sum_{l=1}^N \int d\mathbf{x}' \psi_l^*(\mathbf{x}') \frac{1}{|\mathbf{r} - \mathbf{r}'|} \psi_k(\mathbf{x}') \psi_l(\mathbf{x}),\end{aligned}\quad (12)$$

satisfying the *Hartree-Fock equation*

$$\mathcal{F}\psi_k = \epsilon_k \psi_k.$$

The fourth term in equation (12) is called the exchange term [11, p. 55] and is non-local since the operator is acting on  $\psi_k$ , but the value at the position  $\mathbf{r}$  is determined by the value assumed by  $\psi_k$  at all possible positions  $\mathbf{r}'$ . The eigenvalues  $\epsilon_k$  of the Fock operator are related to the total energy by

$$\begin{aligned}E = & \frac{1}{2} \sum_k [\epsilon_k + \langle \psi_k | h | \psi_k \rangle], \text{ with} \\ \langle \psi_k | h | \psi_k \rangle = & \int d^3\mathbf{x} \psi_k^*(\mathbf{x}) \left[ -\frac{1}{2}\nabla^2 - \sum_n \frac{Z_n}{|\mathbf{r} - \mathbf{R}_n|} \right] \psi_k(\mathbf{x}).\end{aligned}$$

It is obvious that the Fock operator in equation (12) which acts on  $\psi_k$  also depends on  $\psi_k$  itself.

### 3.2. The Roothaan Equation

For molecules numerical basis sets are not efficient, so we use atom centered basis sets to describe the molecular orbital

$$\psi_k(\vec{x}) = \sum_{p=1}^M C_{pk} \chi_p(\vec{x}), \quad (13)$$

with  $k = 1, \dots, M$ , where  $M$  is the number of basis states. So the Hartree-Fock equation reads

$$\mathcal{F}(\vec{x})\psi_k(\vec{x}) = \epsilon_k\psi_k(\vec{x}). \quad \text{Using Eq. (13) yields}$$

$$\mathcal{F}(\vec{x}) \sum_{p=1}^M C_{pk}\chi_p(\vec{x}) = \epsilon_k \sum_{p=1}^M C_{pk}\chi_p(\vec{x}).$$

We can now multiply from the left by a specific basis function  $\chi_\nu$ . Integrating over the space yields

$$\sum_{p=1}^M C_{pk} \int \chi_\nu^*(\vec{x}) \mathcal{F}(\vec{x}) \chi_p(\vec{x}) d^3\vec{x} = \epsilon_k \sum_{p=1}^M C_{pk} \int \chi_\nu^*(\vec{x}) \chi_p(\vec{x}) d^3\vec{x}.$$

This can be written in matrix notation of the Hartree-Fock equations, expressed in the atomic orbital basis:

$$\sum_{p=1}^M C_{pk} F_{\nu p} = \epsilon_k \sum_{p=1}^M C_{pk} S_{\nu p}$$

$$\mathbf{FC}_k = \epsilon_k \mathbf{SC}_k, \quad (14)$$

which is called the *Roothaan equation*. Since the Fock operator itself depends on  $\mathbf{C}_k$  we have the problem of self consistency, because  $\mathcal{F}$  has the solution already inside. The Self Consistent Field Method (SCF) uses the approach to solve the above equation iteratively, until the solution deviates only small from the next iteration.

Equation (14) looks similar to the Hartree-Fock equation, but here  $\mathbf{S}$  is the overlap matrix for the given orbital basis  $\chi_p(\mathbf{r})$ . The  $\mathbf{S}$ -matrix does not vanish, since the used orbital basis does not need to be normalized. In comparison with eq. (12) we can write the Fock matrix as [11]

$$F_{pq} = h_{pq} + \sum_k \sum_{rs} C_{rk}^* C_{sk} (2 \langle pr|g|qs \rangle - \langle pr|g|sq \rangle),$$

where  $h_{pq}$  is identified with the first term of the Fock operator

$$h_{pq} = \langle p|h|q \rangle = \int d^3r \chi_p^*(\mathbf{r}) \left[ -\frac{1}{2} \nabla^2 - \sum_n \frac{Z_n}{|\mathbf{R}_n - \mathbf{r}|} \right] \chi_q(\mathbf{r}).$$

The terms  $\langle pr|g|qs\rangle$  and  $\langle pr|g|sq\rangle$  are the two-electron integrals given by

$$\langle pr|g|qs\rangle = \int d^3r_1 d^3r_2 \chi_p^*(\mathbf{r}_1) \chi_r^*(\mathbf{r}_2) \frac{1}{|\mathbf{r}_1 - \mathbf{r}_2|} \chi_q(\mathbf{r}_1) \chi_s(\mathbf{r}_2).$$

Here  $p, q, r$  and  $s$  label the basis functions and  $k$  labels the orbitals  $\psi_k$ . For simplicity and implementation it is convenient to define the density matrix as

$$P_{pq} = 2 \sum_k C_{pk}^* C_{qk},$$

where the factor 2 is conventionally due to the spin [11]. Now the Fock matrix reads

$$F_{pq} = h_{pq} + \sum_{rs} P_{sr} (2 \langle pr|g|qs\rangle - \langle pr|g|sq\rangle).$$

In this orbital representation the energy is given by

$$E = \sum_{pq} P_{pq} h_{pq} + \frac{1}{2} \sum_{pqrs} P_{pq} P_{sr} \left[ \langle pr|g|qs\rangle - \frac{1}{2} \langle pr|g|sq\rangle \right].$$

### 3.3. Closed and open shell systems

Since also the spin plays important role in atomic structures, further restriction on the spin states can be made, leading to closed and open-shell systems, referred to as Unrestricted Hartree-Fock, and Restricted Hartree-Fock. In a closed shell, the orbitals are occupied by two electrons with opposite spin, whereas in an open shell, the energy levels are partially filled where levels contain only one electron. Since the occupied levels in atoms strictly follow the Aufbau principle, the restriction of a closed-shell might lead to unphysical results in some cases. Even if the number of electrons is even, the electrons do not necessarily occupy closed shells – for an odd number of electrons, an open shell is always present. For a closed shell, and thus for even  $N$ , the slater determinant is

$$\Psi_{\mathbf{R}} = \det [\alpha\psi_1, \dots, \alpha\psi_{N/2}, \beta\psi_1, \dots, \beta\psi_{N/2}] (\mathbf{x}_1, \dots, \mathbf{x}_N),$$

which is referred to as **restricted closed shell Hartree-Fock** (RHF). When allowing the number of spin-up and spin-down states to be different, the slater-

determinant becomes:

$$\Psi_{\mathbf{R}} = \det [\alpha\psi_1, \dots, \alpha\psi_{N_\alpha}, \beta\psi_1, \dots, \beta\psi_{N_\beta}] (\mathbf{x}_1, \dots, \mathbf{x}_N).$$

This represents the **restricted open shell Hartree-Fock** (ROHF). However, in the general case, also the one-particle wave functions have spin-dependence, requiring two distinct basis sets, for spin-down and for spin-up states, respectively, yielding to the **unrestricted Hartree-Fock** (UHF) method, with slater determinant

$$\Psi_{\mathbf{R}} = \det [\alpha\psi_1^\alpha, \dots, \alpha\psi_{N_\alpha}^\alpha, \beta\psi_1^\beta, \dots, \beta\psi_{N_\beta}^\beta] (\mathbf{x}_1, \dots, \mathbf{x}_N).$$

In this general case, the spin-dependence of the basis set is reflected in the expansion coefficients  $\mathbf{C}_k$ , since they also depend on the spin, yielding the so-called *Pople-Nesbet* equations:

$$\begin{aligned}\mathbf{F}^\alpha \mathbf{C}_k^\alpha &= \epsilon_k^\alpha \mathbf{S} \mathbf{C}_k^\alpha \\ \mathbf{F}^\beta \mathbf{C}_k^\beta &= \epsilon_k^\beta \mathbf{S} \mathbf{C}_k^\beta\end{aligned}$$

Here, each equation for spin-up and spin-down orbitals has to be solved independently. However, we will restrict ourselves to RHF for simplicity.

### 3.4. Basis set and basis functions

A key for obtaining reasonable and accurate results in Hartree-Fock theory is the correct choice of the orbital set which on which the calculation is based. As mentioned in section. 3.2, the one-particle wave functions are expanded in the form:

$$\psi_k(\vec{r}) = \sum_{p=1}^M C_{pk} \chi_p(\vec{r}),$$

which is a linear combination of molecular orbitals (LCAO). Since analytic forms of the desired atomic orbitals are only known for the hydrogen atom, these orbitals need to be approximated in a suitable manner. Following the analytic solution for the hydrogen orbital, we get a general form of atomic orbital basis functions, whith the nucleus centered at  $\mathbf{R}_A$ :

$$\chi_\alpha(\mathbf{r}) = r^m P_l(x, y, z) e^{-\alpha|\mathbf{r}-\mathbf{R}_A|},$$

, where  $\alpha$  defines the range of the orbital. Here,  $P_l$  is a polynomial in  $x, y$  and  $z$  of degree  $l$ . This orbital type is also known as a Slater type orbital (STO). However, in Hartree-Fock calculations these orbital types are highly disadvantageous, since integrals containing products of STOs are analytically very difficult and in most cases impossible to compute.

Instead, we can avoid this problem, by replacing the simple exponential by a Gaussian function, yielding:

$$\chi_\alpha(\mathbf{r}) = P_M(x, y, z) e^{-\alpha(\mathbf{r}-\mathbf{R}_A)^2}$$

These functions are called *primitive basis functions*. In this case, products of Gaussian type orbitals (GTO) centered at different nuclei are still Gaussian functions. Thus, the necessary integrals, such as the kinetic integrals, the overlap integrals, the nuclear repulsion integrals as well as the electron-electron interaction integrals can be calculated analytically. To accurately approximate a Slater-type orbital one can use a linear combination of Gaussian-type orbitals, which is called a *contracted Gaussian orbital*, with contraction coefficients  $a_i$ . Denoting a primitive Gaussian with  $G(\alpha, A, l, m, n)$ , a contraction is simply given by a set of primitive Gaussians:

$$G^{\text{Contr.}} = \sum_i^n a_i G(\alpha_i, A, l, m, n)$$

In Fig. 3, a comparison of different contractions for a 1s orbital is shown. Choosing and finding the right basis set is a rather difficult part and there is a huge data base of different basis sets available (see, e.g. [3, 10] and the EMSL Basis Set Exchange database). Here, we use the basis set provided by the EMSL Basis Set Exchange database for our calculations.

For a given basis set, the following integrals need to be computed in the Hartree-Fock method:

- **Overlap integral**

$$S_{ij} = \langle \alpha_1, \mathbf{A}, l_1, m_1, n_1 | \alpha_2, \mathbf{B}, l_2, m_2, n_2 \rangle = \int \psi_1^*(\mathbf{r}) \psi_2(\mathbf{r}) d\mathbf{r}$$

- **Kinetic energy integral**

$$T_{ij} = \langle \alpha_1, \mathbf{A}, l_1, m_1, n_1 | -\frac{1}{2} \nabla^2 | \alpha_2, \mathbf{B}, l_2, m_2, n_2 \rangle = \int \psi_1^*(\mathbf{r}) \left( -\frac{1}{2} \nabla^2 \right) \psi_2(\mathbf{r}) d\mathbf{r}$$

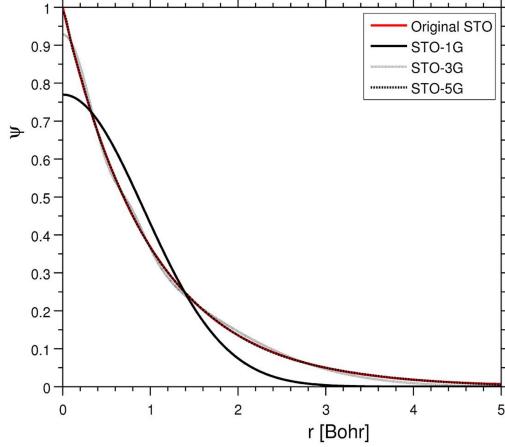


Figure 3: Approximation of a slater determinant for the 1s orbital by STO-1G, STO-3G and STO-6G basis functions. [1]

- Nuclear Attraction integrals

$$V_{ij}^C = \langle \alpha_1, \mathbf{A}, l_1, m_1, n_1 | \frac{Z_C}{r_C} | \alpha_2, \mathbf{B}, l_2, m_2, n_2 \rangle = \int \psi_1^*(\mathbf{r}) \left( \frac{Z_C}{r_C} \nabla^2 \right) \psi_2(\mathbf{r}) d\mathbf{r}$$

- Electron repulsion integral

$$\text{ERI} = \langle pr | g | qs \rangle = \int d^3r_1 d^3r_2 \psi_p^*(\mathbf{r}_1) \psi_r^*(\mathbf{r}_2) \frac{1}{|\mathbf{r}_1 - \mathbf{r}_2|} \psi_q(\mathbf{r}_1) \psi_s(\mathbf{r}_2)$$

The last integral is by far the most important and also most difficult part since the calculation involving 4 different primitive Gaussians scales with  $N^4$ , where  $N$  is the number of primitive Gaussians. Using very efficient state-of-the-art algorithms, the calculation of the electron repulsion integral can be reduced to  $\mathcal{O}(N)$ . For these integrals, analytic solutions exists. The implementation of these integrals is straightforward but very tedious. For a detailed calculation of Gaussian integral evaluation, see [4–6, 8, 9].

### 3.5. Program Structure

Here a general overview of the general program structure will be given. As outlined before it is important to consider a suitable set of basis functions describing the electrons electrons of the atoms. The accuracy and efficiency of the basis largely depend on the choice of the basis functions, so they need to be chosen carefully. For

suitable calculations we use Gaussian basis functions, so the important integrals can be calculated analytically. For a more detailed view on basis functions, see appendix A. Below the program procedure is sketched.

### 1. Input data:

Here the basis sets as well as the geometry of the atoms in the molecule will be provided and parsed to the program. Furthermore the numbers of electrons  $N$  and the atomic charge numbers  $Z_n$  are needed as input.

### 2. Calculate independent matrices

All matrices that are independent of the eigenvectors  $\mathbf{C}_k$  can be calculated:

- The overlap matrix  $S_{pq}$ ,
- The two-electron integrals  $\langle pr|g|qs \rangle$ ,
- The uncoupled one-electron Hamiltonian  $h_{pq}$ .

### 3. Make an initial guess for the density matrix $\mathbf{P}$ .

A possibility is to use  $\mathbf{P} = 0$  as an initial guess, which means that the electrons only feel the nuclei but not each other.

### 4. Self-consistency procedure

This program section is the most important, since it solves the Roothaan equation recursively. It consists of the following steps.

- Calculate the Coulomb and exchange contributions to the Fock matrix,
- Construct the Fock matrix,
- Solve the Roothaan equation by diagonalization of the Fock matrix for the given density matrix  $\mathbf{P}$ ,
- With the obtained eigenvectors a new density matrix  $\mathbf{P}$  is constructed.

### 5. Output data

The output is the converged eigenstates  $\mathbf{C}_k$  and its corresponding Fock levels.

In figure 4 a schematic representation of the general program scheme is shown.

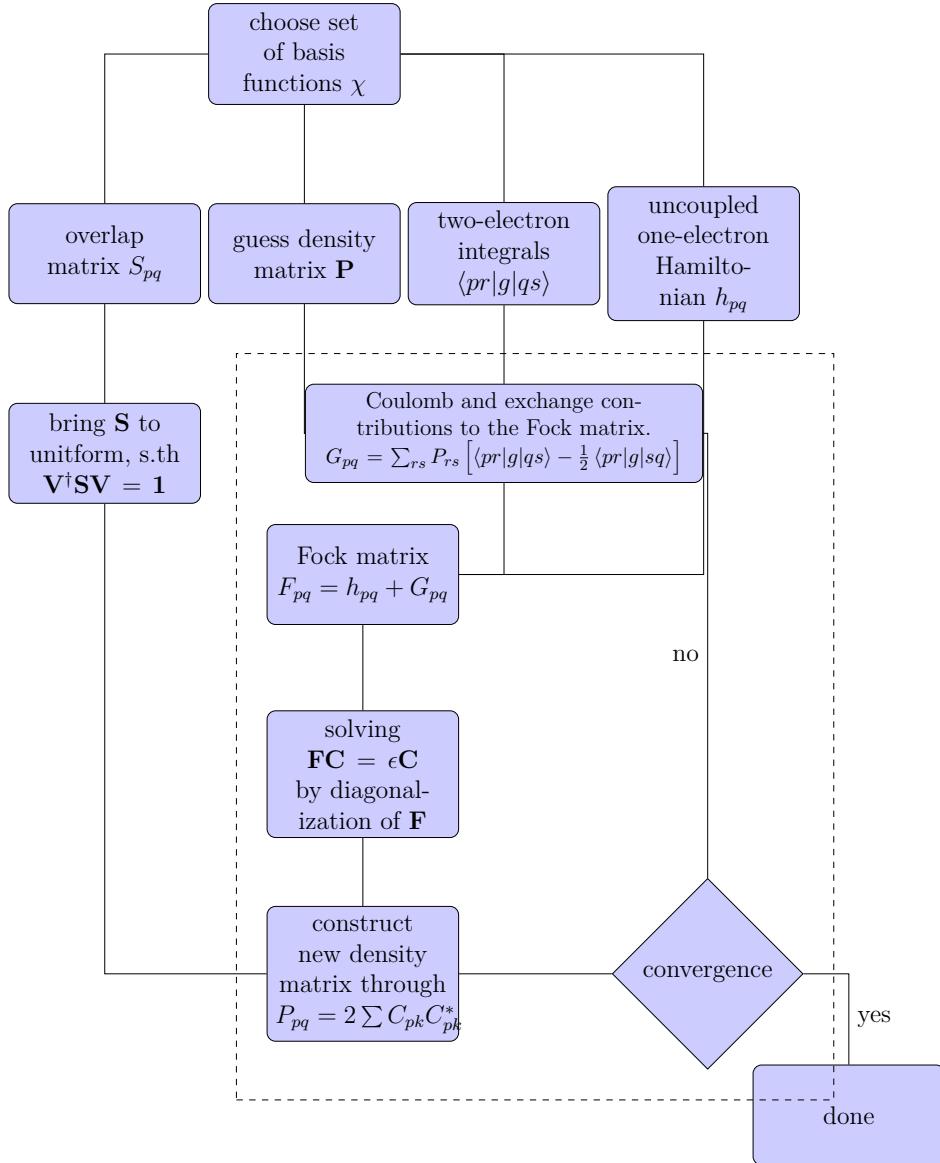


Figure 4: General scheme of the program structure.

# Appendices

## A. Basis Functions

## References

- [1] Wikimedia Commons. Comparison of different sto-*ng* and sto, 2013. URL [https://commons.wikimedia.org/wiki/File:Gauss\\_sto4.jpeg](https://commons.wikimedia.org/wiki/File:Gauss_sto4.jpeg).
- [2] Pablo Echenique and J. L. Alonso. A mathematical and computational review of Hartree-Fock SCF methods in Quantum Chemistry. pages 105(23–24), 2007.
- [3] David Feller. The role of databases in support of computational chemistry calculations. *Journal of computational chemistry*, 17(13):1571–1586, 1996.
- [4] Justin T Fermann and Edward F Valeev. Fundamentals of molecular integrals evaluation. Technical report, Tech. rep, 1997. URL [http://www.valeevgroup.chem.vt.edu/docs/int\\_s.07032015.pdf](http://www.valeevgroup.chem.vt.edu/docs/int_s.07032015.pdf).
- [5] Peter MW Gill, Martin Head-Gordon, and John A Pople. An efficient algorithm for the generation of two-electron repulsion integrals over gaussian basis functions. *International Journal of Quantum Chemistry*, 36(S23):269–280, 1989.
- [6] Sigeru Huzinaga. Basis sets for molecular calculations. *Computer Physics Reports*, 2(6):281–339, 1985.
- [7] Frank Jensen. *Introduction to Computational Chemistry* -. John Wiley & Sons, New York, 2013. ISBN 978-1-118-68162-6.
- [8] Tomas Petersson and Bo Hellsing. A detailed derivation of gaussian orbital-based matrix elements in electron structure calculations. *European journal of physics*, 31(1):37, 2009.
- [9] Richard C Raffenetti. General contraction of gaussian atomic orbitals: Core, valence, polarization, and diffuse basis sets; molecular integral evaluation. *The Journal of Chemical Physics*, 58(10):4452–4458, 1973.
- [10] Karen L Schuchardt, Brett T Didier, Todd Elsethagen, Lisong Sun, Vidhya Gurumoorthi, Jared Chase, Jun Li, and Theresa L Windus. Basis set exchange: a community database for computational sciences. *Journal of chemical information and modeling*, 47(3):1045–1052, 2007.
- [11] Jos Thijssen. *Computational Physics* -. Cambridge University Press, Cambridge, 2. edition, 2007. ISBN 978-0-521-83346-2.