Project 2: Hydrogen atom with a Gaussian basis

Mione Filippo, Rigo Mauro, Slongo Francesco

April 7, 2020

University of Trento
Department of Physics
Via Sommarive, 14, 38123 Povo
Trento, Italy

Abstract

We studied the ground state of the Hydrogen atom using the variational method. We projected the trial wave function over an s-wave set composed of Gaussian functions and then we numerically minimized the energy with respect to the parameters of the Gaussians. We obtained $\varepsilon = -0.424\,E_h$, $-0.486\,E_h$ and $-0.497\,E_h$ for 1, 2 and 3 s-waves respectively. The energy that we obtained with 3 Gaussians is closer to the true ground state energy than the one obtained with the STO-3G set. Lastly we explained and showed numerically why the addition of the p-wave set is detrimental to the estimation of the ground state energy and instead we used this set to approximate the 2p state energy, obtaining $\varepsilon_{2p} = -0.1247\,E_h$ as best estimate with 3 p-waves.

1 Introduction to the physical problem

The purpose of the following report is to numerically estimate the lowest energy states of the hydrogen atom. We are especially interested in the ground state energy.

A stationary state of the system is governed by the time-independent Schrödinger equation:

$$\hat{H}\psi(\boldsymbol{r}) = E\psi(\boldsymbol{r}) \tag{1}$$

where $\boldsymbol{r} = (x,\, y,\, z)$ and $\hat{H}$ is the Hamiltonian of the system. For simplicity we consider the proton, the mass of which is about 2000 times that of the electron, to be fixed, so that H assumes the form (in atomic units)

$$\hat{H} = -\frac{1}{2}\nabla^2 - \frac{1}{r} \tag{2}$$

This problem can be solved analytically and it is possible to find that the bound state energies are

$$E_n = -\frac{1}{2n^2} \tag{3}$$

thus ground state energy is

$$E_1 = -\frac{1}{2} \tag{4}$$

The ground state wave function is known as the 1s wave function and it has the form

$$\psi_{1s}(r) = \frac{1}{\sqrt{\pi}}e^{-r} = \frac{1}{\sqrt{\pi}}e^{-\sqrt{x^2+y^2+z^2}} \tag{5}$$

To estimate the ground state energy of the system, one can exploit the variational method. Given a generic trial wave function Ψ, this method states that

$$E_{gs} \leq \frac{\langle\Psi|H|\Psi\rangle}{\langle\Psi|\Psi\rangle} = \langle H\rangle = E[\Psi] \tag{6}$$

where E_{gs} is the true ground state energy of H. If Ψ is normalized, $\langle\Psi|\Psi\rangle = 1$. The previous inequality becomes an equality if and only if Ψ is the true ground state wave function. In general one puts forward a rather general expression for the trial wave function with a dependence on a set of parameters $\{\alpha_i\}$, then proceeds by minimizing the energy functional $E[\Psi]$ with respect to those parameters. To simplify

the research of the trial function, it is customary to expand it on a suitable orthogonal basis $|\chi_p\rangle$:

$$|\Psi\rangle = \sum_p c_p \, |\chi_p\rangle \tag{7}$$

Since it is not possible to consider a sum of infinite functions of the basis, one usually considers only a subspace of the total Hilbert space by limiting the number of basis functions N, then studies the behaviour of $E[\Psi]$ as a function of N hoping for a fast convergence of the result. With the expansion in Eq. (7) the variational principle can be rewritten in the following way:

$$E[\Psi] = \frac{\sum_{pq} c_p^* c_q H_{pq}}{\sum_{pq} c_p^* c_q \delta_{pq}} \tag{8}$$

where $H_{pq} = \langle \chi_p | H | \chi_q \rangle$. With some algebra (and requiring Ψ to preserve its normalization), one can see that the minimum of $E[\Psi]$ is given by the eigenvalue equation

$$\sum_q \left(H_{pq} - \varepsilon_p \delta_{pq} \right) c_q = 0 \qquad \Longrightarrow \qquad \boldsymbol{H}\boldsymbol{c} = \varepsilon \, \boldsymbol{c} \tag{9}$$

where $\boldsymbol{c}$ is the vector of the coefficients of the expansion of $|\Psi\rangle$ over the states $|\chi_p\rangle$ and ε is the corresponding eigenvalue. The smallest ε gives the minimum $E[\Psi]$ obtainable using the (finite) basis $|\chi_p\rangle$.

However, some complete orthonormal basis may yield cumbersome expressions for the matrix elements H_{pq}; additionally, if we want to find an energy value close to the true ground state, we need to consider wave functions with the same symmetries of the true ground state wave function. Therefore, we might want to expand $|\Psi\rangle$ over a convenient subset of a non-orthonormal basis. To calculate the ground state energy of the Hamiltonian in Eq. (2), a Gaussian basis is often used, because the elements of $\boldsymbol{S}$ and $\boldsymbol{H}$ can be evaluated analytically. This is the choice we adopt in what follows. Since the basis is not orthonormal, though, Eq. (9) becomes

$$\sum_q \left(H_{pq} - \varepsilon_p S_{pq} \right) c_q = 0 \qquad \Longrightarrow \qquad \boldsymbol{H}\boldsymbol{c} = \varepsilon \, \boldsymbol{S}\boldsymbol{c} \tag{10}$$

where $S_{pq} = \langle \chi_p | \chi_q \rangle$ is the so called *overlap matrix*. The problem of finding $\boldsymbol{c}$ in this case is called *generalized eigenvalue problem*.

2 Introduction to the computation problem

When performing numerical calculations, it is often wise to use suitable measurement units in order to deal with quantities close to unit and avoid inconsistent results due to a computer's internal representation of floating point numbers. For this reason we decided to use atomic units and we will express energies in Hartrees E_h and lengths in Bohr radii a_0.

To solve the generalized eigenvalue problem, if $\boldsymbol{S}$ is positive definite, we may proceed in the following way. Let us introduce a matrix $\boldsymbol{C}$ given by the column vectors $\boldsymbol{c}$ that solve Eq. (10) and a vector $\boldsymbol{E}$ which contains the eigenvalues of $\boldsymbol{H}$. Then Eq. (10) is written as

$$\boldsymbol{H}\boldsymbol{C} = \boldsymbol{E}\boldsymbol{S}\boldsymbol{C} \tag{11}$$

Now, since $\boldsymbol{S}$ is symmetric, it can be always diagonalized. Let us thus call $\boldsymbol{U}$ the matrix built from the eigenvectors of $\boldsymbol{S}$ and $\boldsymbol{\Lambda}$ the diagonal matrix with the eigenvalues of $\boldsymbol{S}$ on its diagonal. We define the

matrix $\boldsymbol{V}$ in the following way:

$$\boldsymbol{V} = \boldsymbol{U}\boldsymbol{\Lambda}^{-1/2} \tag{12}$$

where the inverse square root must be interpreted as applied to the single elements of $\boldsymbol{\Lambda}$. Note that, as $\boldsymbol{S}$ is positive definite, the square root of its eigenvalues is real. With this definition, it is easy to prove that

$$\boldsymbol{V}^T\boldsymbol{S}\boldsymbol{V} = \mathbb{1} \tag{13}$$

The generalised eigenvalue equation may then be rewritten as

$$\boldsymbol{H}'\boldsymbol{C}' = \boldsymbol{E}\boldsymbol{C}' \tag{14}$$

where $\boldsymbol{H}' = \boldsymbol{V}^T\boldsymbol{H}\boldsymbol{V}$ and $\boldsymbol{C}' = \boldsymbol{V}^T\boldsymbol{C}$. This way the eigenvalues of $\boldsymbol{H}'$ are equal to those in the original generalised eigenvalue problem, while its eigenvectors can be obtained from $\boldsymbol{C} = \boldsymbol{V}\boldsymbol{C}'$. Note that in general it is not true that $\boldsymbol{V}\boldsymbol{V}^T = \mathbb{1}$, therefore the eigenvectors that we obtain are not normalised. If one wants to get an estimate of the ground state wave function, we have to normalize the columns of $\boldsymbol{V}\boldsymbol{C}'$ (and remember to take into account the fact that the basis could be not normalised).

The `GSL` library implemented in `C` provides useful and efficient routines to perform calculations involving matrices such as products and diagonalization, thus we shall use it frequently throughout our work.

To find the best estimate for the ground state energy using a given set of wave functions, one should then solve the generalized eigenvalue problem and then minimize the smallest element of $\boldsymbol{E}$ by imposing $\partial \min(\boldsymbol{E})/\partial\alpha_i = \partial\varepsilon/\partial\alpha_i = 0$ and $\partial^2\varepsilon/\partial\alpha_i^2 > 0\ \forall i$. Of course, the expression of $\boldsymbol{E}$ is in general not known analytically, therefore one has to utilize (multi-dimensional) minimization algorithms. There is a huge variety of such algorithms, but for the sake of simplicity we shall only implement the method of steepest descent. This method consists in starting from an initial point $\boldsymbol{\alpha}^0 = \{\alpha_i^0\}$ and iteratively performing small steps in the opposite direction of the gradient:

$$\boldsymbol{\alpha}^{(i+1)} = \boldsymbol{\alpha}^{(i)} - \boldsymbol{\nabla}\varepsilon\big|_{\boldsymbol{\alpha}^{(i)}} h \tag{15}$$

where h is a user-defined step. The algorithm stops when the modulus of the gradient is smaller than a given threshold, meaning that the current point lies in a valley. Clearly, also the gradient needs to be calculated numerically. In this regard, we use the the five-point method:

$$\nabla_i f(x_i) = \frac{1}{12\tilde{h}}\Big(f(x_i - 2\tilde{h}) - 8f(x_i - \tilde{h}) + 8f(x_i + \tilde{h}) - f(x_i + 2\tilde{h})\Big) + \mathcal{O}(\tilde{h}^4) \tag{16}$$

where by $f(x_i)$ we denote a multi-dimensional function $f(x_1, ..., x_n)$ with $x_{j\neq i}$ kept constant and by ∇_i the derivative with respect to x_i. As value of the step we chose $\tilde{h} = 10^{-8}$.

3 Discussion

3.1 Point 1

Here we introduce the s-wave set that we shall use to give an estimate of the ground state energy of the Hydrogen atom. Since the ground state must be rotational invariant, we could in principle use only one s-wave:

$$\langle \boldsymbol{r}|\phi_i\rangle = e^{-\alpha_i r^2} \tag{17}$$

Fig. 1 portrays a comparison between a decaying exponential e^{-x} (which is proportional to the ground state wave function) and a Gaussian e^{-x^2}. As we can see, the Gaussian has lower tails, but a much higher central region. We can tune α_i in such a way that we can approximate better the tails or the central peak, but we do not expect to do be able to do both at the same time.

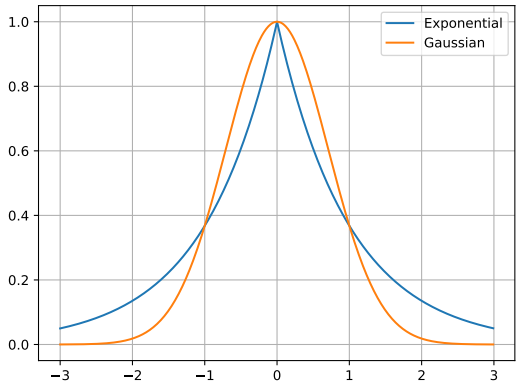

Figure 1: Comparison between a decaying exponential and a Gaussian.

In order to solve Eq. (10), we first need to calculate the matrix elements of the overlap matrix $\boldsymbol{S}$ and of the Hamiltonian $\boldsymbol{H}$. For the s-wave set we have

$$S_{ij} = \langle \phi_i | \phi_j \rangle = \left(\frac{\pi}{\alpha_j + \alpha_i} \right)^{3/2} \qquad H_{ij} = \langle \phi_i | H | \phi_j \rangle = \frac{3\hbar^2 \pi^{3/2}}{m} \frac{\alpha_i \alpha_j}{(\alpha_i + \alpha_j)^{5/2}} - \frac{2\pi e^2}{\alpha_i + \alpha_j} \tag{18}$$

The results are symmetric in the indices i and j as expected, since the α_i are real. A simple dimensional check proves that the results are correct (up to some constants). The full calculations are reported in Appendix B.

3.2 Point 2

In order to solve the generalised eigenvalue problem, we need to write a code that is able to handle matrix multiplication and matrix inversion. In this section we shall develop step by step the method explained in Sec. (2). As aforementioned, we used the `GSL` library.

We first initialize the matrices $\boldsymbol{S}$ and $\boldsymbol{H}$:

```
const int n = 3; // Number of Gaussians and dimension of the problem
double mat[n][n] = {{2, -1, 0}, {-1, 2, -1}, {0, -1, 2}}; // Test matrices
double mat2[n][n] = {{1, 3, 0}, {3, -1, 2}, {0, 2, -2}};
gsl_matrix *Sqp = gsl_matrix_alloc(n, n); // Overlap matrix
gsl_matrix *Hqp = gsl_matrix_alloc(n, n); // Hamiltonian matrix

// Assignement of matrix elements
for (int i = 0; i < n; i++)
{
    for (int j = 0; j < n; j++)
    {
        gsl_matrix_set(Sqp, i, j, mat[i][j]);
        gsl_matrix_set(Hqp, i, j, mat2[i][j]);
    }
}
```

We then calculate the eigenvalues and the eigenvectors of $\boldsymbol{S}$, in order to define the matrix $\boldsymbol{V}$:

```
// Diagonalize S
gsl_matrix *evecs = gsl_matrix_alloc(n, n); // Sqp eigenvectors
gsl_vector *evals = gsl_vector_alloc(n); // Sqp eigenvalues
gsl_eigen_symmv_workspace *ws = gsl_eigen_symmv_alloc(n);
gsl_eigen_symmv(Sqp, evals, evecs, ws);
gsl_eigen_symmv_free(ws);
gsl_eigen_symmv_sort(evals, evecs, GSL_EIGEN_SORT_VAL_ASC);
```

We thus obtain $\boldsymbol{V}$:

```
// Build V
gsl_matrix *V = gsl_matrix_alloc(n, n);
for (int i = 0; i < n; i++)
{
    for (int j = 0; j < n; j++)
    {
        gsl_matrix_set(V, i, j, gsl_matrix_get(evecs, i, j) ...
        ... / sqrt(gsl_vector_get(evals, j)));
    }
}
```

Now we can calculate $\boldsymbol{H'}$. The function `gsl_blas_dgemm`(op_A, op_B, α, $\boldsymbol{A}$, $\boldsymbol{B}$, β, $\boldsymbol{C}$) calculates the product of $\boldsymbol{A}$ and $\boldsymbol{B}$ as $\boldsymbol{C} = \alpha\, op_A(\boldsymbol{A}) op_B(\boldsymbol{B}) + \beta \boldsymbol{C}$, where $op(\boldsymbol{A}) = \boldsymbol{A}$, $\boldsymbol{A}^T$ or $\boldsymbol{A}^\dagger$.

```
//Product between H and V
gsl_matrix *tempH = gsl_matrix_alloc(n, n);
gsl_matrix_set_zero(tempH);
gsl_blas_dgemm(CblasNoTrans, CblasNoTrans, 1., Hqp, V, 0., tempH);

//Product between Vt and tempH
gsl_matrix *newH = gsl_matrix_alloc(n, n);
gsl_matrix_set_zero(newH);
gsl_blas_dgemm(CblasTrans, CblasNoTrans, 1., V, tempH, 0., newH); // Transpose V
```

Finally we solve the eigenvalue problem to find the energies:

```
// Diagonalize newH
gsl_matrix *Cp = gsl_matrix_alloc(n, n);
gsl_vector *epsilon = gsl_vector_alloc(n);
gsl_eigen_symmv_workspace *ws2 = gsl_eigen_symmv_alloc(n);
gsl_eigen_symmv(newH, epsilon, Cp, ws2);
gsl_eigen_symmv_free(ws2);
```

And we calculate and normalise the eigenvectors to obtain the correct projection of Ψ over the basis:

```
// Calculate normalised eigenvectors
gsl_matrix *C = gsl_matrix_alloc(n, n);
gsl_matrix_set_zero(C);
//Product between V and Cp
gsl_blas_dgemm(CblasNoTrans, CblasNoTrans, 1., V, Cp, 0., C);


double norm; // Normalize columns
for (int i = 0; i < n; i++)
{
    norm = 0.;
    for (int j = 0; j < n; j++)
    {
        norm += gsl_matrix_get(C, j, i)*gsl_matrix_get(C, j, i);
    }
```

```
        norm = sqrt(norm);
        for (int j = 0; j < n; j++)
        {
            gsl_matrix_set(C, j, i, gsl_matrix_get(C, j, i)/norm);
        }
    }
```

In truth, GSL already offers a subroutine that directly solves the generalised eigenvalue problem, which reduces all the code above to just few lines:

```
    // Using dedicated subroutine
    gsl_matrix *genEvecs = gsl_matrix_alloc(n, n);
    gsl_vector *genEvals = gsl_vector_alloc(n);
    gsl_eigen_gensymmv_workspace *genws = gsl_eigen_gensymmv_alloc(n);
    gsl_eigen_gensymmv(Hqp, Sqp, genEvals, genEvecs, genws);
    gsl_eigen_gensymmv_free(genws);
```

We tested the first code against the GSL function. It is important to remember that $\boldsymbol{S}$ must be symmetric and positive definite, otherwise both algorithms fail. Here we report some of the matrices that we tested and the relative outputs.

Test 1

$$S = \begin{pmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{pmatrix} \qquad H = \begin{pmatrix} 1 & 3 & 0 \\ 3 & -1 & 2 \\ 0 & 2 & 2 \end{pmatrix}$$

The difference between the $\boldsymbol{C}$ obtained with the two codes are (we indicate the first algorithm with *our* and the one with the dedicated GSL subroutine with *gsl*)

$$|C_{our} - C_{gsl}| = \begin{pmatrix} 7.2 & 3.3 & 0 \\ 4.4 & 4.0 & 2.2 \\ 5.8 & 3.3 & 1.1 \end{pmatrix} \times 10^{-16}$$

The eigenvalues are

λ_{gsl}	$\lambda_{our} - \lambda_{gsl}$
-1.13	1.1×10^{-15}
0.810	-1.3×10^{-15}
6.57	-1.8×10^{-15}

Table 1: Comparison between the eigenvalues for the matrices of test 1.

The time used for the algorithms are (averaged over 10000 trials)

$$t_{our} = 7.4 \times 10^{-6}\ s \qquad t_{gsl} = 5.6 \times 10^{-6}\ s$$

Test number 2

$$S = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 3 \end{pmatrix} \qquad H = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 2 & 3 \\ 3 & 3 & 3 \end{pmatrix}$$

The difference between the $\boldsymbol{C}$ obtained with the two codes are

$$|C_{our} - C_{gsl}| = \begin{pmatrix} 0 & 1.4 & 1.1 \\ 2.2 & 0 & 8.3 \\ 8.3 & 1.1 & 1.1 \end{pmatrix} \times 10^{-16}$$

The eigenvalues are instead:

λ_{gsl}	$\lambda_{our} - \lambda_{gsl}$
-1.7	2.2×10^{-16}
-0.21	-5.2×10^{-16}
2.9	-4.4×10^{-16}

Table 2: Comparison between the eigenvalues for the matrices of test number 2.

The times used by the algorithm are

$$t_{our} = 7.5 \times 10^{-6}\ s \qquad t_{gsl} = 5.9 \times 10^{-6}\ s$$

Test number 3

$$S = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 2 & 0 \\ 1 & 0 & 3 \end{pmatrix} \qquad H = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 2 \\ 3 & 2 & 1 \end{pmatrix}$$

The difference between the $\boldsymbol{C}$ obtained with the two codes are

$$|C_{our} - C_{gsl}| = \begin{pmatrix} 2.2 & 0 & 0 \\ 0.3 & 0 & 3.3 \\ 0 & 0.6 & 3.6 \end{pmatrix} \times 10^{-16}$$

The eigenvalues are instead:

λ_{gsl}	$\lambda_{our} - \lambda_{gsl}$
-2.6	2.2×10^{-16}
0.26	-5.2×10^{-16}
2.9	-4.4×10^{-16}

Table 3: Comparison between the eigenvalues for the matrices of test number 3.

The time used for the algorithms are

$$t_{our} = 7.2 \times 10^{-6}\ s \qquad t_{gsl} = 5.7 \times 10^{-6}\ s$$

As we can see from the previous tests, our algorithm gives the same eigenvalues of the `GSL` function. We get the same eigenvectors as well, up to a minus sign[1]. Moreover the `GSL` algorithm is slightly faster than ours, therefore we shall use that in the following sections.

[1] An eigenvector multiplied by -1 is still an eigenvector.

3.3 Point 3

If we limit the number of Gaussians to 1, it is actually possible to find the analytic expression of $E[\Psi]$ (from now on, we shall indicate the ground state estimate as ε) and therefore also the value of α that minimizes it. In fact, in this case the generalized eigenvalue equation Eq. (10) reduces to (using the expressions for the matrix elements of $\boldsymbol{S}$ and $\boldsymbol{H}$ found above)

$$\left(\frac{3\pi^{3/2}}{4\sqrt{2}}\frac{1}{\sqrt{\alpha}} - \frac{\pi}{\alpha}\right)\mathbf{c} = \varepsilon\left(\frac{\pi}{2\alpha}\right)^{3/2}\mathbf{c} \tag{19}$$

Since we are dealing with simple numbers, we can simplify $\mathbf{c}$ on both sides of the equation and solve for ε, obtaining

$$\varepsilon = \frac{3\alpha\sqrt{2\pi} - 8\sqrt{\alpha}}{2\sqrt{2\pi}} \tag{20}$$

From this equation we can find the minimum of the energy by imposing the derivative of ε to be equal to zero:

$$\frac{\partial\varepsilon}{\partial\alpha} = \frac{3\sqrt{2\pi} - 4\alpha^{-1/2}}{2\sqrt{2\pi}} = 0 \quad \rightarrow \quad \alpha = \frac{8}{9\pi} \simeq 0.283\, a_0^{-2} \tag{21}$$

which corresponds to an estimate of the ground state energy

$$\varepsilon = -\frac{4}{3\pi} \simeq -0.424\, E_h \tag{22}$$

This case could be useful to test the minimization algorithm we employ and study the landscape of ε as a function of the set of parameters $\{\alpha_i\}$. Fig. 2 portrays the behaviour of the energy functional ε as a function of α.

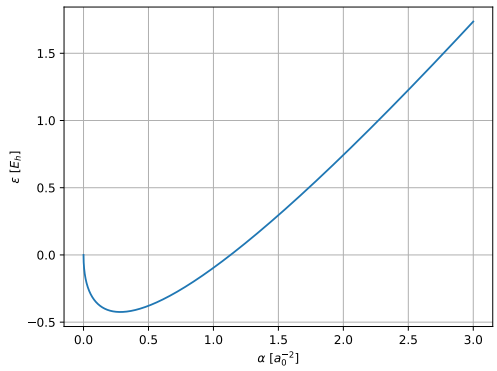

Figure 2: Behaviour of ε as a function of α.

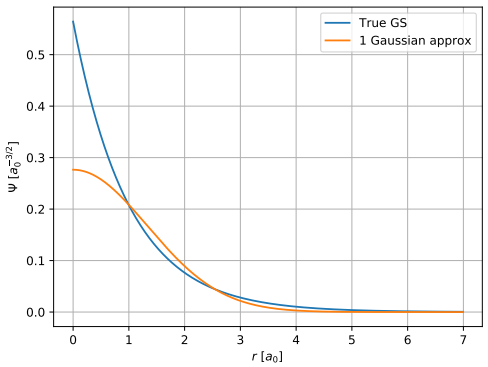

Figure 3: Comparison between the true ground state wave function and the best approximation using 1 Gaussian.

It can be noticed that $\varepsilon(\alpha)$ does not display multiple local minima. As we shall see, this seems to be a general feature also at higher numbers of Gaussians. For completeness, in Fig. 3 we also portray the best approximation using a single Gaussian to allow a comparison with those using more Gaussians.

Other than changing the thresholds and the steps for the minimization algorithm, it is also possible to implement it in different ways. For instance, we can choose to normalize the gradient in Eq. (15) and thus move the point of a quantity h at every step. One could then decrease h as the number of steps increases, such that, assuming to have reached the proximity of a minimum, the algorithm refines

the search up to the desired threshold for the modulus of the gradient. Alternatively, we could choose *not* to normalize the gradient and keep h constant, such that the point moves more when ε is steeper. The former algorithm may be more efficient in case of very flat functions with rather deep and localized minima, while the latter could be more refined in the search for the minimum and in general might be more adaptive. Moreover, the efficiency of the first algorithm depends on the choice of the behaviour of h, the best of which depends on the overall behaviour of the function as well as on the initial position, while the second algorithm may be more general and has less parameters to fix. Of course, there are many other changes we could make as well as algorithms we could use; however, as mentioned above, we choose to keep things simple. Again for the sake of simplicity, we preferred the second algorithm over the first one as it requires tweaking less parameters. Anyway, in all cases we did not study in detail the dependence of the results on these parameters, rather we slightly tuned the parameters and kept the most accurate numerical values until repeating this procedure would mostly worsen the results. For the cases in which the analytical results are not available, we considered to be most accurate the results with the lowest estimate for the energy eigenvalue, as the variational method suggests doing. In what follows, we shall use "best values" in reference to the values obtained with this procedure.

In the case of one Gaussian, we tested the minimization algorithm mainly to verify its functioning. We tweaked the initial conditions and the step h; we found setting $h = 0.1$ to be beneficial both in terms of efficiency and accuracy of the solution. Additionally, we set a threshold for the modulus of the gradient of 10^{-10} and imposed a maximum number of iterations for the minimization algorithm of 5×10^6. In Tab. 4 we report the best result for the difference of the numerical parameters ε and α with respect to the analytical values.

$\Delta\varepsilon\ [E_h]$	$\Delta\alpha\ \left[a_0^{-2}\right]$
-5.6×10^{-17}	-4.4×10^{-11}

Table 4: Difference between best numerical values and analytical ones for 1 Gaussian.

We note that the discrepancy of the energy eigenvalue may be entirely given by the precision of the machine, since other initial values of α yielded different final results for this parameter but equal energy discrepancy (by equal we mean that *every* digit was the same, at least within the 20 significant digits outputted) or even a discrepancy of 0[2]. We thus consider the difference in the energy eigenvalue reported to be just a rough estimate. It is also worth mentioning that changing the initial condition on α did not modify its final value of more than one order of magnitude, even considering e.g. $\alpha_0 = 50$.

For the case of 2 Gaussians, it is still possible to plot the behaviour of ε as a function of the parameters α_1, α_2 (Fig.s 4 and 5, the black line marks $\alpha_1 = \alpha_2$, a region not allowed due to the $\boldsymbol{S}$ matrix not being positive definite).

As we can see, the function once again doesn't display multiple local minima as for the 1 Gaussian case. We infer this might be the case also for 3 (or more) Gaussians. The figures also portray one minimization run starting from $(\alpha_1, \alpha_2) = (1, 1.6)$; a detail of the run reaching the minimum is shown in Fig. 6. It is also possible to notice that the function in the neighbourhood of the the minimum is very flat: in fact, the modulus of the gradient in this region is very small (also of order 10^{-10}) and so we expect that rather significant changes in (α_1, α_2) around the minimum may lead to negligible changes in ε. Therefore, even if the minimum finding algorithm is very inefficient in the proximity of the minimum (since the modulus of the gradient is very small), the energy functional does not change in a significant fashion for our purposes. We also expect to obtain noticeable variations of the final parameters (α_1, α_2) even with very similar final energies.

[2]This would also explain the reason why this value is negative. In all other cases we obtained positive differences.

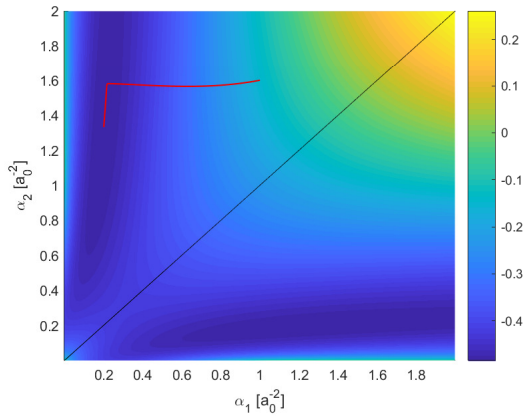

Figure 4: Surface plot of ε as a function of α_1 and α_2.

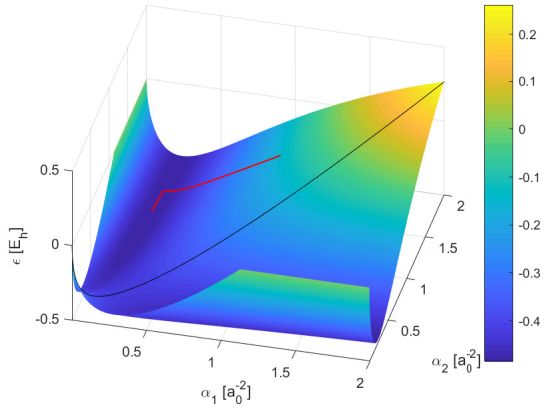

Figure 5: Angled view of the surface plot of ε as a function of α_1 and α_2.

As for 1 Gaussian, we used $h = 0.1$, a threshold for the modulus of the gradient of 10^{-10} but a maximum iteration number of 1×10^6. The best results are reported in Tab. 5.

ε $[E_h]$	α_1 $\left[a_0^{-2}\right]$	α_2 $\left[a_0^{-2}\right]$
-0.485812716616275	1.3324998	0.20152963

Table 5: Best numerical values for 2 Gaussians.

Note that the results are reported up to the last significant digit that remained constant considering different initial conditions[3]. However, the last digits of these results may be affected by the machine precision; for the α_i, at first order, a variation $\delta\alpha_i \sim 10^{-7}$ with a gradient threshold of 10^{-10} gives an energy variation of order 10^{-17}, where the results are indeed affected by machine precision.

Also in this case we report the best approximation for the ground state wave function along with the single s-waves that compose it in Fig. 7. The coefficients of the two components are simply given by the eigenvector in $\boldsymbol{C}$ associated with ε. The improvement with respect to the approximation using 1 Gaussian is evident.

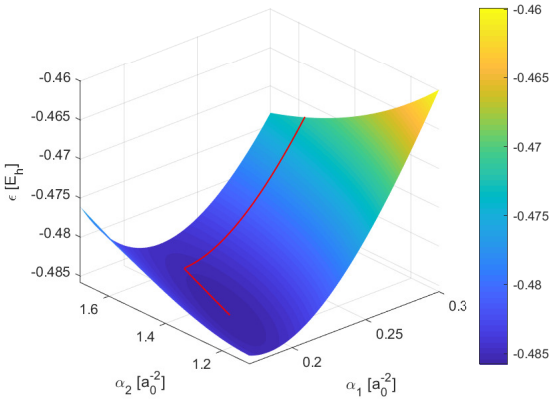

Figure 6: Detail of the path of the minimization algorithm.

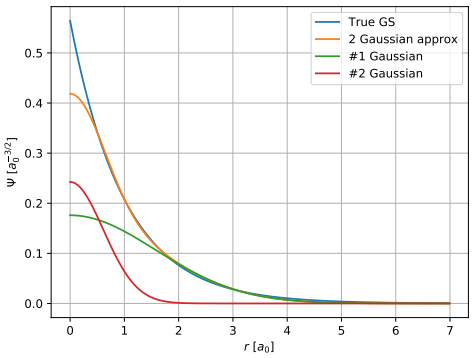

Figure 7: Comparison between the true ground state wave function and the best approximation using 2 Gaussians. The two s-waves are also shown.

[3]The same is also true for all of the following tables.

The case of 3 Gaussians is analogous to the previous ones. We limit ourselves to reporting the results for the best results in Tab. 6 and showing the approximation of the ground state with its components in Fig. 8. We shall discuss the comparison between the approximation and the true ground state in detail in Sec. 3.4.

$\varepsilon\ [E_h]$	$\alpha_1\ \left[a_0^{-2}\right]$	$\alpha_2\ \left[a_0^{-2}\right]$	$\alpha_3\ \left[a_0^{-2}\right]$
-0.4969792527050511	0.6812892	0.15137639	4.500362

Table 6: Best numerical values for 3 Gaussians.

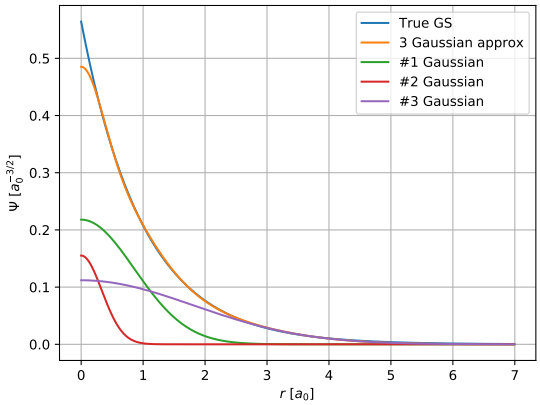

Figure 8: Comparison between the true ground state wavefunction and the best approximation using 3 gaussians. The 3 s-waves are also shown.

With these three results it is possible to appreciate the overall improvement both in the approximated wave functions and in the estimates of the ground state energies as the limited Hilbert space is expanded. Adding more Gaussians would surely improve the results, though this analysis is beyond the purpose of this report.

3.4 Point 4

The STO-nG sets are minimal basis sets, where n primitive Gaussian integrals are fitted to a single-Slater determinant. The STO-3G basis for the ground state of the Hydrogen is made by three Gaussian functions with exponents

$$\alpha_1 = 0.109818\, a_0^{-2} \qquad \alpha_2 = 0.405771\, a_0^{-2} \qquad \alpha_3 = 2.22776\, a_0^{-2} \tag{23}$$

Using these exponents in our code we find as coefficients for the three Gaussian respectively

$$C_1 = 0.220688 \quad C_2 = 0.692622 \quad C_3 = 0.686711$$

Use these results to estimate the ground state energy, we obtain

$$\varepsilon_{\text{STO-3G}} = -0.495011\, E_h \tag{24}$$

This energy estimate is rather good, but not as close to the true value as the one we get with the parameters from the minimization algorithm. In particular, the differences between the ground state energy estimates and the true one (we indicate the best estimation from the minimisation algorithm with

our) are

$$\varepsilon_{gs} - \varepsilon_{\text{STO-3G}} = 0.004989\, E_h \qquad \varepsilon_{gs} - \varepsilon_{our} = 0.003021\, E_h$$

We can also compare the wave functions obtained with the two sets or parameters. We can calculate the discrepancy between the estimated wave functions and the real ground state one as $\xi = 1 - \langle \Psi_{gs} | \Psi_{est} \rangle$, the results are

$$\xi_{\text{STO-3G}} = 0.00020 \qquad \xi_{our} = 0.00062$$

This means that actually the STO-3G wave function globally follows more accurately the behaviour of the true ground state one. Fig. 9 compares the true ground state wave function and the ones calculated with the exponents from the minimisation algorithm and the STO-3G set. We can notice that our approximation follows better the ground state at low r. On the contrary, the STO-3G describes more accurately the tail. In general, for a good estimation of the energy, it is more important to describe closely the correct wave function at low values of r[4], while discrepancies in the tail can be neglected. This explains why our estimation of the energy is more accurate with respect to the STO-3G one, despite having a larger discrepancy on the wave function.

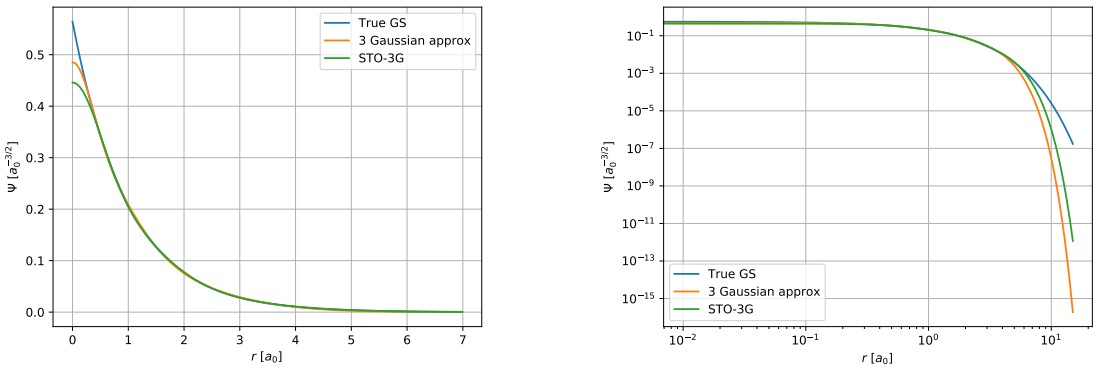

Figure 9: Comparison between wave functions of the STO-3G set and our best set of parameters, both approximating the true Hydrogen ground state energy wave function, also plotted here.

3.5 Point 5

In this section we consider more functions for our basis set. In particular, we focus our attention on the influence of a p-wave set. The functions of this set are

$$\langle \boldsymbol{r} | \phi_i^x \rangle = x e^{-\alpha_i r^2} \qquad \langle \boldsymbol{r} | \phi_i^y \rangle = y e^{-\alpha_i r^2} \qquad \langle \boldsymbol{r} | \phi_i^z \rangle = z e^{-\alpha_i r^2} \tag{25}$$

As for the s-wave set, the matrix elements of $\boldsymbol{S}$ and $\boldsymbol{H}$ considering only one direction are given by (the results are clearly also valid for ϕ_i^y and ϕ_i^z)

$$\begin{aligned} S_{ij}^{xx} &= \left\langle \phi_i^x \middle| \phi_j^x \right\rangle = \frac{\pi\sqrt{\pi}}{2} (\alpha_i^x + \alpha_j^x)^{-5/2} \\ H_{ij}^{xx} &= \left\langle \phi_i^x \middle| H \middle| \phi_j^x \right\rangle = \frac{5}{2} \frac{\hbar^2 \pi^{3/2}}{m} \frac{\alpha_i^x \alpha_j^x}{(\alpha_j^x + \alpha_i^x)^{7/2}} - \frac{2\pi e^2}{3} \frac{1}{(\alpha_j^x + \alpha_i^x)^2} \end{aligned} \tag{26}$$

[4] This is a consequence of the chosen Hamiltonian.

Notice that by construction $|\phi_i^x\rangle$, $|\phi_i^y\rangle$ and $|\phi_i^z\rangle$ are orthogonal to each other, therefore the cross elements (e.g. S_{ij}^{xy}, H_{ij}^{yz} etc.) are all null[5]. Once again, the dimensions of the final expression are the expected ones (note that they differ from those of H_{ij} of the s-wave set due to the wave functions themselves having different dimensions). The full calculations are in Appendix B.

In the choice of the basis set for the wave function, it is important to take into account the symmetries of the problem and of the basis set itself. The ground state and all the states with angular momentum $l = 0$ are rotationally invariant. The single elements of the p-wave set are not invariant under rotation around the origin (the wave function in one direction has a different behaviour than in the other directions), therefore it is not possible to approximate the ground state only using p-wave states along the same direction (e.g $|\phi_i^x\rangle$, $|\phi_j^x\rangle$ and $|\phi_k^x\rangle$). One could wonder if, by combining all the elements of the p-wave set, it could be possible to obtain a radial symmetry, but this is not the case. This can be clearly seen if we restrict ourselves to two dimensions and we plot the behaviour of a linear combination of $|\phi_i^x\rangle$ and $|\phi_j^y\rangle$. Of course, we require the wave function to decay in the same fashion along x and y, therefore we have $\alpha_i = \alpha_j$ and ψ must be given by

$$|\Psi\rangle = \frac{|\phi_i^x\rangle + |\phi_i^y\rangle}{\sqrt{2}} \tag{27}$$

From Fig. (10) we can clearly see that this state is not invariant under rotation around the origin.

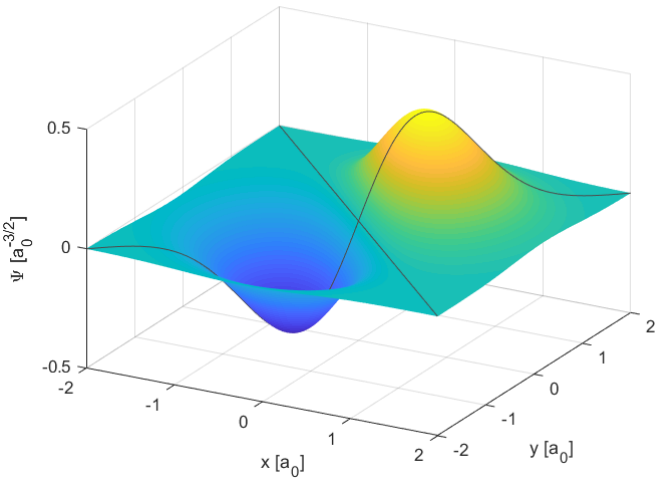

Figure 10: Plot in 2D of $|\Psi\rangle$ as linear combination of $|\phi_i^x\rangle$ and $|\phi_i^y\rangle$ ($\alpha_i = 1$). The dark axes mark $x = y$ and $x = -y$.

It is thus rather clear that we can't obtain a good estimate of the ground state energy with the p-wave set. Moreover, it is detrimental to add the p-wave set to the s-wave set, again because every p-wave element break the rotational invariance. In fact, solving the generalized eigenvalue problem both using 3 s-waves and 3 p-waves in the same direction and using 3 s-waves and 3 p-waves in different directions we obtained energy eigenvalues of the order of those obtained with only 3 s-waves and null coefficients for the p-waves, just with more computational time.

This does not actually mean that the p-wave set is not useful. The variational method allows to get an estimate of the lowest energy state *with the symmetries of the basis set chosen.* Using the p-wave set, we can find the lowest energy states invariant under rotations about a fixed axis, i.e. the 2p states.

In this case, it is not beneficial to solve the generalized eigenvalue problem using only the 3 different functions in the p-wave set. This can be explained from a theoretical point of view: if we wish to estimate

[5]To be precise, the cross elements of the overlap matrix are null only due to the basis chosen, but this is not always true for the Hamiltonian matrix. This depends on the symmetries of the Hamiltonian and how it couples different directions. We shall discuss the symmetries of the Hamiltonian hereafter.

the energy of a state with an axial symmetry, we do not need to consider functions that are not rotational invariant with respect to the symmetry axis. Therefore if, say, the state is invariant for rotations about the z-axis, we should not use $|\phi_i^x\rangle$ or $|\phi_i^y\rangle$ states. This can also explained from a algebraic standpoint: using 3 functions in the 3 different directions, both the $\boldsymbol{S}$ matrix and the $\boldsymbol{H}$ matrix are already diagonal and, since we only minimize the smallest eigenvalue, only one parameter (the one which initially gives the smallest eigenvalue) will be modified, while the others will remain untouched. In this sense, considering three p-waves in different directions is identical to using only one in an arbitrary direction. Once again, the Hamiltonian being diagonal with respect to these states is due to its symmetry properties, that is, there exists no state with the same symmetry as a linear combination of the three p-states in different directions.

As for one single Gaussian, the problem can be solved analytically considering only one p-state. The generalized eigenvalue equation reads

$$\left[\frac{5}{2}\pi^{3/2}\frac{\alpha^2}{(2\alpha)^{7/2}} - \frac{2\pi}{3}\frac{1}{4\alpha^2}\right]\mathbf{c} = \varepsilon\frac{\pi^{3/2}}{2}\frac{1}{(2\alpha)^{5/2}}\mathbf{c}$$

where $\mathbf{c}$ is a number and can thus be simplified. Isolating ε we obtain

$$\varepsilon = \frac{5}{2}\alpha - \frac{4}{3}\sqrt{\frac{2\alpha}{\pi}}$$

The minimum of the energy is then

$$\frac{\partial\varepsilon}{\partial\alpha} = \frac{5}{2} - \frac{2}{3}\sqrt{\frac{2}{\pi\alpha}} \quad \rightarrow \quad \alpha = \frac{32}{225\pi} \simeq 0.0453\, a_0^{-2}$$

The corresponding energy estimate is

$$\varepsilon = -\frac{16}{45\pi} \simeq -0.113\, E_h$$

As expected, a p-wave function estimates the energy of a 2p state, that is $\varepsilon_{2p} = -0.125\, E_h$.

As we did for the s-wave set, we tested the minimization algorithm against the analytical result for a single p-wave function. The results are in Tab. (7).

$\Delta\varepsilon\ [E_h]$	$\Delta\alpha\ \left[a_0^{-2}\right]$
-4.2×10^{-17}	-1.4×10^{-11}

Table 7: Difference between best numerical values and analytical ones for 1 p-wave.

The results are of the same order of magnitude of the s-wave set. This could mean that the energy landscape is not much different between the two cases (we shall not report any plot in this case).

To improve the estimate of the energy of the first excited state, we can add p-waves along the same direction. The results are in Tab. (8) and Tab. (9).

$\varepsilon\ [E_h]$	$\alpha_1\ \left[a_0^{-2}\right]$	$\alpha_2\ \left[a_0^{-2}\right]$
-0.1232887133586300	0.0323923652	0.13927846

Table 8: Best numerical values for 2 p-waves.

$\varepsilon\ [E_h]$	$\alpha_1\ \left[a_0^{-2}\right]$	$\alpha_2\ \left[a_0^{-2}\right]$	$\alpha_3\ \left[a_0^{-2}\right]$
−0.1247276009564717	0.024685343	0.07983417	0.3370727

Table 9: Best numerical values for 3 p-waves.

As expected as the number of p-waves increases the goodness of the estimates of the 2p state energy increases. With 3 p-waves the result obtained is correct within 0.2%.

The wave function obtained for 3 p-waves is portrayed in Fig. (11) along with the true first excited state wave function.

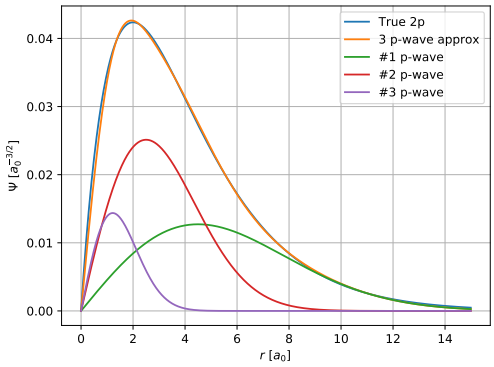

Figure 11: Comparison between the true first excited state wave function and the best approximation using 3 p-waves.

4 Conclusions

We were able to successfully estimate the ground state energy of the Hydrogen atom. We used the variational principle: we projected the trial wave function Ψ over a Gaussian s-wave set and then we restated our problem as a generalized eigenvalue equation. We wrote a code to solve a generalized eigenvalue equation given a symmetric, positive definite $\boldsymbol{S}$ matrix and we compared our result to the corresponding `GSL` routine. Our code gave the same results of the `GSL` function, but it was slower of a factor of about 1.5.

After that we minimised numerically the energy with respect to the parameters of the Gaussians, using the method of the steepest descent. We obtained $\varepsilon = -0.424\, E_h$, $\varepsilon = -0.486\, E_h$ and $\varepsilon = -0.497\, E_h$ for 1, 2 and 3 Gaussians respectively. The results are very close to the true ground state energy. These values are better than the ones we can obtain using the STO-3G basis.

Lastly we showed that we can't estimate the ground state energy using a p-wave set, because they are not rotationally invariant. However the set is invariant for rotations about the three axes and can thus be used to estimate the energy of the 2p state. With three p-waves, we obtained a minimum energy of $\varepsilon = -0.1247$ against a true value of $\varepsilon_{2p} = -0.125$.

Appendix A Code and data depository

The data were simulated using `C++`, while the plots and the data analysis were made in Python and Matlab. The code written for simulating the data of this report is available at the link `https://github.com/FrancescoSlongo/Computational-Physics---Trento/tree/master/Project_2`.

Appendix B Integrals full calculations

Here we perform all the calculations form the overlap matrix $\boldsymbol{S}$ and the Hamiltonian $\boldsymbol{H}$. Starting from the s-wave set we have

$$\begin{aligned} S_{ij} = \langle \phi_i | \phi_j \rangle &= \int \mathrm{d}\boldsymbol{r}\, e^{-\alpha_i r^2} e^{-\alpha_j r^2} \\ &= \iiint_{-\infty}^{+\infty} \mathrm{d}x\, \mathrm{d}y\, \mathrm{d}z\, e^{-(\alpha_j+\alpha_i)x^2} e^{-(\alpha_j+\alpha_i)z^2} e^{-(\alpha_j+\alpha_i)y^2} \\ &= \left(\int_{-\infty}^{+\infty} \mathrm{d}x\, e^{-(\alpha_j+\alpha_i)x^2} \right)^3 \\ &= \left(\frac{\pi}{\alpha_j+\alpha_i} \right)^{3/2} \end{aligned}$$

As for the Hamiltonian, we obtain

$$\begin{aligned} H_{ij} &= \langle \phi_i | \hat{H} | \phi_j \rangle \\ &= \int \mathrm{d}\boldsymbol{r}\, e^{-\alpha_i r^2} \left(-\frac{\hbar^2}{2m}\nabla^2 - \frac{e^2}{r} \right) e^{-\alpha_j r^2} \\ &= -\frac{\hbar^2}{2m} \iiint_{-\infty}^{+\infty} \mathrm{d}x\, \mathrm{d}y\, \mathrm{d}z\, e^{-\alpha_i(x^2+y^2+z^2)} \left(\partial_x^2 + \partial_y^2 + \partial_z^2 \right) e^{-\alpha_j(x^2+y^2+z^2)} - e^2 \int \mathrm{d}\Omega \int_0^{+\infty} \mathrm{d}r\, r e^{-(\alpha_i+\alpha_j)r^2} \\ &= -\frac{3\hbar^2}{2m} \left(\iiint_{-\infty}^{+\infty} \mathrm{d}x\, \mathrm{d}y\, \mathrm{d}z\, 2\alpha_j(2\alpha_j x^2 - 1) e^{-(\alpha_i+\alpha_j)(x^2+y^2+z^2)} \right) - 4\pi e^2 \int_0^{+\infty} \mathrm{d}r\, r e^{-(\alpha_i+\alpha_j)r^2} \\ &= -\frac{3\hbar^2}{2m} \frac{2\alpha_j \pi}{(\alpha_i+\alpha_j)} \left(\frac{\sqrt{\pi}\alpha_j}{(\alpha_i+\alpha_j)^{3/2}} - \frac{\sqrt{\pi}}{(\alpha_i+\alpha_j)^{1/2}} \right) - \frac{2\pi e^2}{\alpha_i+\alpha_j} \\ &= \frac{3\hbar^2 \pi^{3/2}}{m} \frac{\alpha_i \alpha_j}{(\alpha_i+\alpha_j)^{5/2}} - \frac{2\pi e^2}{\alpha_i+\alpha_j} \end{aligned}$$

For the p-wave set we have

$$\begin{aligned} S_{ij} = \langle \phi_i^x | \phi_j^x \rangle &= \int \mathrm{d}\boldsymbol{r} x^2 e^{-\alpha_i^x r^2} e^{-\alpha_j^x r^2} \\ &= \iiint_{-\infty}^{+\infty} \mathrm{d}x\, \mathrm{d}y\, \mathrm{d}z\, x^2 e^{-(\alpha_i^x+\alpha_j^x)x^2} e^{-(\alpha_i^x+\alpha_j^x)z^2} e^{-(\alpha_i^x+\alpha_j^x)y^2} \\ &= \frac{\pi}{\alpha_i+\alpha_j} \left(\frac{\sqrt{\pi}}{2(\alpha_i^x+\alpha_j^x)^{3/2}} \right) \\ &= \frac{\pi\sqrt{\pi}}{2} (\alpha_i^x+\alpha_j^x)^{-5/2} \end{aligned}$$

The matrix elements of the Hamiltonian instead read

$$
\begin{aligned}
H_{ij} &= \langle \phi_i^x | \hat{H} | \phi_j^x \rangle \\
&= \int \mathrm{d}\boldsymbol{r}\, x\, e^{-\alpha_i^x r^2} \left(-\frac{\hbar^2}{2m}\nabla^2 - \frac{e^2}{r} \right) x e^{-\alpha_j^x r^2} \\
&= \iiint_{-\infty}^{+\infty} \mathrm{d}x\, \mathrm{d}y\, \mathrm{d}z\, x\, e^{-\alpha_i^x (x^2+y^2+z^2)} \left(-\frac{\hbar^2}{2m} \right) (\partial_x^2 + \partial_y^2 + \partial_z^2)\, x\, e^{-\alpha_j^x (x^2+y^2+z^2)} + \\
&\quad - e^2 \int_0^{2\pi} \mathrm{d}\varphi \cos^2(\varphi) \int_0^{\pi} \mathrm{d}\theta \sin^3(\theta) \int_0^{\infty} \mathrm{d}r\, r^3 e^{-(\alpha_i^x + \alpha_j^x) r^2} \\
&= -\frac{\hbar^2}{2m} \iiint_{-\infty}^{+\infty} \mathrm{d}x\, \mathrm{d}y\, \mathrm{d}z\, x\, e^{-\alpha_i^x (x^2+y^2+z^2)} 2\alpha_j^x x (2\alpha_j^x x^2 - 3) e^{-\alpha_j^x (x^2+y^2+z^2)} + \\
&\quad - \frac{2\hbar^2}{2m} \iiint_{-\infty}^{+\infty} \mathrm{d}x\, \mathrm{d}y\, \mathrm{d}z\, x\, e^{-\alpha_i^x (x^2+y^2+z^2)} 2\alpha_j^x (2\alpha_j^x y^2 - 1) x\, e^{-\alpha_j^x (x^2+y^2+z^2)} - \frac{4}{3} \frac{e^2 \pi}{2(\alpha_i^x + \alpha_j^x)^2} \\
&= -\frac{\hbar^2 \alpha_j^x}{m} \frac{\pi}{\alpha_i^x + \alpha_j^x} \left(\frac{3\alpha_j^x \sqrt{\pi}}{2(\alpha_i^x + \alpha_j^x)^{5/2}} - \frac{3\sqrt{\pi}}{2(\alpha_i^x + \alpha_j^x)^{3/2}} \right) + \\
&\quad - \frac{2\hbar^2 \alpha_j^x}{m} \sqrt{\frac{\pi}{\alpha_i^x + \alpha_j^x}} \frac{\sqrt{\pi}}{2(\alpha_i^x + \alpha_j^x)^{3/2}} \left(\frac{\alpha_j^x \sqrt{\pi}}{(\alpha_i^x + \alpha_j^x)^{3/2}} - \sqrt{\frac{\pi}{\alpha_i^x + \alpha_j^x}} \right) - \frac{2}{3} \frac{\pi e^2}{(\alpha_i^x + \alpha_j^x)^2} \\
&= \frac{3\hbar^2 \pi^{3/2}}{2m} \frac{\alpha_i^x \alpha_j^x}{(\alpha_j^x + \alpha_i^x)^{7/2}} + \frac{\hbar^2 \pi^{3/2}}{m} \frac{\alpha_i^x \alpha_j^x}{(\alpha_j^x + \alpha_i^x)^{7/2}} - \frac{2\pi e^2}{3} \frac{1}{(\alpha_j^x + \alpha_i^x)^2} \\
&= \frac{5}{2} \frac{\hbar^2 \pi^{3/2}}{m} \frac{\alpha_i^x \alpha_j^x}{(\alpha_j^x + \alpha_i^x)^{7/2}} - \frac{2\pi e^2}{3} \frac{1}{(\alpha_j^x + \alpha_i^x)^2}
\end{aligned}
$$

References

[1] F. Pederiva. *The variational method; Lecture slides for the Advanced Computational Physics Course.* University of Trento, 2020.