

Instruction on how to write a Hartree-Fock Program: an H₂ version

李睿

September 2, 2019

1 引言

如何让一个学生对计算化学充满恐惧？让他写 Hartree-Fock 程序。高斯积分手写的那种。

—————李睿

建设一流大学从 HF 做起。

—————王林军教授

Hartree-Fock 方法作为计算化学的开山鼻祖，对于一个想要从事计算化学相关工作的学生而言是一个必须要过去的坎。当然，原则上它作为一种工具你并不需要对它的所有细节都要彻底地把握，但如果并不深入了解，也可以说是十分令人叹息的事——Hartree-Fock 写出来确实算是小程序，但它仍然能称得上是人类智慧的结晶。

刚开始着手量子化学的时候，或许在编程、数学、物理等方面的功底还不足以写出一个漂亮的 Hartree-Fock 程序——也有可能只是单纯的时间不够的问题。但如果我们把目标局限在只计算某一个特定碱厂的氢分子的能量——那可行性还是有很多的¹。因此在王林军教授的强烈要求下，我尝试从我原来的程序库 (<https://github.com/Walter-Feng/Hartree-Fock>) 的 manual 中为 H₂ 进行了特化（也就是专门拎出 *s* 轨道部分的所有信息），做成这样的较为简练**非常详实**的教程，同时隐去了所有的程序细节让大家思考²。至于矩阵对角化的问题，二阶矩阵可以直接手算，而三阶矩阵及以上又只能借助于数学库³，在这里不再进行展开。

¹更何况当时我尝试在量化课上写 Hartree（没有 Fock）程序的时候可没有这么完整的 documentation 呢对吧。

²但其实去我那个程序库差不多就被刷透得差不多子

³*Mathematica* 当然可以，如果你和我一样主位是 C/C++ 的话强烈推荐 GSL(GNU Scientific Library)，当然 *Eigen* 库和 *armadillo* 库也不是不可以。

2 程序流程概述

虽然 Hartree-Fock 程序相对而言较小，但由于需要照顾许多细节所以总体上仍旧会显得相对庞大，直接从程序源代码入手还并不容易窥见其全貌，因此在这里用文字做一个整体的流程介绍⁴。

流程概括地来讲分为：

1. 读取指定基组；
2. 从文件读取所有剩余的参数，包括所有原子的信息；
3. 将所有信息打包成为程序喜欢的数据结构；
4. 进行 Hartree-Fock 方法计算；
5. 将结果进行输出。

Hartree-Fock 方法采用的流程为：

1. 计算原子核排斥能；
2. 计算重叠矩阵（即 Overlap matrix, 用 S 进行表示）；
3. 计算动能矩阵（即 Kinetic energy matrix, 用 T 进行表示）；
4. 计算原子核-电子相互作用势能矩阵（即 Nuclear attraction energy matrix, 用 Z 进行表示）；
5. 计算电子排斥能张量（为四阶张量，用 v 进行表示）；
6. 从单电子哈密顿矩阵（ $h = T + Z$ ）得到初始系数矩阵（即 Coefficient matrix, 用 C 进行表示）；
7. 进行 RHF 迭代：
 - (a) 从 C 以及电子数目得到电子密度矩阵（即 density matrix, 用 D 表示）；
 - (b) 结合 h , D 与 v 得到 Fock 矩阵（用 F 进行表示）；
 - (c) 结合 F, S 解本征值问题，得到对应特征值（分子轨道能量）及此时的 C ；
 - (d) 结合 h , D 与 v 得到对应体系能量；
 - (e) 将得到的新的 C 与旧的 C 进行混合（mixing）并代入新一轮的迭代。
8. 判断每次循环的能量，最后进行输出。

⁴在 L^AT_EX 里做流程图很累 de

3 程序流程讲解

于是我们就开始剖析整个程序所经历的流程了。

3.1 读取基组

在这里贴出 H 的 STO-3G、3-21G、6-31G 基组中的参数 [4]:

STO-3G:

```
N: 1
Name: H

L= 0
total= 3
Exponents:
3.42525091 0.62391373 0.16885540
Coefficients:
0.15432897 0.53532814 0.44463454
```

3-21G:

```
N: 1
Name: H

L= 0 #1s
total= 2
Exponents:
5.4471780 0.8245470
Coefficients:
0.1562850 0.9046910

L= 0 #2s
total= 1
Exponents:
0.1831920
Coefficients:
1.0000000
```

6-31G:

```
L= 0 #1s
total= 3
Exponents:
18.7311370 2.8253937 0.6401217
Coefficients:
0.03349460 0.23472695 0.81375733

L= 0 #2s
total= 1
```

Exponents:
0.1612778
Coefficients:
1.00000000

其中归一化所用的函数为 [3]

$$N(\alpha, a_x, a_y, a_z) = \left(\frac{2\alpha}{\pi}\right)^{3/4} \frac{(4\alpha)^{(a_x+a_y+a_z)/2}}{[(2a_x-1)!!(2a_y-1)!!(2a_z-1)!!]^{1/2}}. \quad (3.1)$$

3.2 高斯积分

符号注释:

$|\mathbf{0}_A\rangle$ 代表某一个 s 轨道的高斯函数 (即无角动量), 指数部分为 α (或 ζ_A), 中心为 \vec{A} . 相应地, $|\mathbf{0}_B\rangle$ 代表另一个 s 轨道的高斯函数, 指数部分为 β (或 ζ_B), 中心为 \vec{B} .

Theorem 3.1. 两个高斯函数的乘积可转换为一系列高斯函数的和 [2], 比如:

$$g(\vec{r}, \alpha, \mathbf{0}, \vec{A})g(\vec{r}, \beta, \mathbf{0}, \vec{B}) = \exp(-\xi|\overline{AB}|^2)g(\vec{r}, \zeta, \mathbf{0}, \vec{P}), \quad (3.2)$$

其中 $\zeta = \alpha + \beta$, $\xi = \frac{\alpha\beta}{\zeta}$, $\vec{P} = \frac{\alpha\vec{A} + \beta\vec{B}}{\zeta}$, $\overline{AB} = \vec{A} - \vec{B}$. 对于含有多项式的高斯函数。可以利用

$$(x - A_x)^{a_x} = [(x - P_x) + \overline{PA}_x]^{a_x} = \sum_{i=0}^{a_x} C_{a_x}^i (x - P_x)^i P A_x^{a_x-i} \quad (3.3)$$

从而可以展开两个高斯函数的乘积为

$$g_1 g_2 = \exp(-\xi|AB|^2) \exp(-\zeta|\vec{r} - \vec{P}|^2) \times \prod_{k=x,y,z} \sum_{i=0}^{i=a_k+b_k} (k - P_k)^i f_i(a_k, b_k, \overline{PA}_k, \overline{PB}_k), \quad (3.4)$$

其中

$$f_i(a_k, b_k, \overline{PA}_k, \overline{PB}_k) = \sum_{n=0}^{n=i} C_{a_k}^n C_{b_k}^{i-n} \overline{PA}_k^{a_k-n} \overline{PB}_k^{b_k-n+i} \quad (3.5)$$

3.2.1 重叠积分

定义是

$$S_{ij} = \langle \phi_i | \phi_j \rangle = \int \phi_i(\mathbf{r}) \phi_j(\mathbf{r}) d\mathbf{r}. \quad (3.6)$$

初始条件:

$$(\mathbf{0}_A | \mathbf{0}_B) = \left(\frac{\pi}{\zeta}\right)^{3/2} \exp(-\xi|\overline{AB}|^2), \quad (3.7)$$

3.2.2 动能积分

定义是

$$T_{ij} = \langle \phi_i | -\frac{\nabla^2}{2} | \phi_j \rangle = -\frac{1}{2} \int \phi_i(\mathbf{r}) [\nabla^2 \phi_j(\mathbf{r})] d\mathbf{r}. \quad (3.8)$$

其中

$$\nabla^2 \equiv \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}. \quad (3.9)$$

理论上最不用动脑子的方法是直接对高斯函数求导然后利用重叠积分得到值⁵。但考虑到写一个简单的 H_2 程序还要引入不定长高斯函数成本过大，不妨将其移到纸面工作⁶。

为了降低难度，在这里尝试给出所有可能需要的公式⁷：

$$\frac{\partial}{\partial x} |\mathbf{0}_B\rangle = -2\beta |\mathbf{0}_B\rangle + 4\beta^2 |\mathbf{2}_B^{x,x}\rangle. \quad (3.10)$$

$$\langle \mathbf{0}_A | \mathbf{2}_B^{x,x} \rangle = \frac{\pi^{3/2} (\alpha + \beta + 2\alpha^2(x_1 - x_2)^2) \exp(-\xi \overline{AB}^2)}{2(\alpha + \beta)^{7/2}} \quad (3.11)$$

其中

$$|\mathbf{2}_B^{x,x}\rangle \equiv (x - x_B)^2 |\mathbf{0}_B\rangle. \quad (3.12)$$

3.2.3 核吸引积分

定义是

$$Z_{ij} = \langle \phi_i | \frac{1}{|\mathbf{r} - \mathbf{r}_l|} | \phi_j \rangle = \int \frac{\phi_i(\mathbf{r}) \phi_j(\mathbf{r})}{|\mathbf{r} - \mathbf{r}_l|} d\mathbf{r} \quad (3.13)$$

其中 \mathbf{r}_l 为原子核所在的坐标。

$$\begin{aligned} (\mathbf{a} + \mathbf{1}_i | \frac{1}{|\mathbf{r} - \mathbf{r}_l|} | \mathbf{b})^{(m)} &= \frac{\beta}{\zeta} \overline{BA}_i (\mathbf{a} | \frac{1}{|\mathbf{r} - \mathbf{r}_l|} | \mathbf{b})^{(m)} \\ &+ \left(C_i - A_i - \frac{\beta}{\zeta} \overline{BA}_i \right) (\mathbf{a} | \frac{1}{|\mathbf{r} - \mathbf{r}_l|} | \mathbf{b})^{(m+1)} \\ &+ \frac{a_i}{2\zeta} \left[(\mathbf{a} - \mathbf{1}_i | \frac{1}{|\mathbf{r} - \mathbf{r}_l|} | \mathbf{b})^{(m)} - (\mathbf{a} - \mathbf{1}_i | \frac{1}{|\mathbf{r} - \mathbf{r}_l|} | \mathbf{b})^{(m+1)} \right] \\ &+ \frac{b_i}{2\zeta} \left[(\mathbf{a} | \frac{1}{|\mathbf{r} - \mathbf{r}_l|} | \mathbf{b} - \mathbf{1}_i)^{(m)} - (\mathbf{a} | \frac{1}{|\mathbf{r} - \mathbf{r}_l|} | \mathbf{b} - \mathbf{1}_i)^{(m+1)} \right] \end{aligned} \quad (3.14)$$

以及初始条件

$$(\mathbf{0}_A | \frac{1}{|\mathbf{r} - \mathbf{r}_l|} | \mathbf{0}_B)^{(m)} = \frac{2\pi}{\zeta} \exp(-\xi \overline{AB}^2) F_m(T), \quad (3.15)$$

⁵因为如果你正在写一个比较大型的程序的时候你肯定会有相应的数据结构应对不定数量的高斯函数线性组合，此时引入一个求导的函数并不困难。

⁶也就是回家作业

⁷从 *Mathematica* 直接推导而出，正确与否无法判定

其中

$$T = \zeta \overline{PC}^2, \quad \vec{C} \equiv \mathbf{r}_l, \quad (3.16)$$

以及

$$F_m(T) = \int_0^1 t^{2m} \exp \{-Tt^2\} dt. \quad (3.17)$$

对于 $m = 0$, 其转化为误差函数

$$F_0(T) = \frac{\sqrt{\pi}}{2\sqrt{T}} \text{Erf}(\sqrt{T}). \quad (3.18)$$

3.2.4 双电子积分/库仑积分

定义为

$$J_{ij} = \langle \phi_i | \frac{1}{|\mathbf{r}_1 - \mathbf{r}_2|} | \phi_j \rangle = \int \frac{\phi_i(\mathbf{r}) \phi_j(\mathbf{r})}{|\mathbf{r}_1 - \mathbf{r}_2|} d\mathbf{r}. \quad (3.19)$$

$$\begin{aligned} (\mathbf{a} + \mathbf{1}_i | \frac{1}{r_{12}} | \mathbf{b}) = & \overline{PA}_i(\mathbf{a} | \frac{1}{r_{12}} | \mathbf{b})^{(m+1)} \\ & + \frac{a_i}{2\alpha} \left\{ (\mathbf{a} - \mathbf{1}_i | \frac{1}{r_{12}} | \mathbf{b})^{(m)} - \frac{\beta}{\zeta} (\mathbf{a} - \mathbf{1}_i | \frac{1}{r_{12}} | \mathbf{b})^{(m+1)} \right\} \\ & + \frac{b_i}{2\zeta} (\mathbf{a} | \frac{1}{r_{12}} | \mathbf{b} - \mathbf{1}_i)^{(m+1)} \end{aligned} \quad (3.20)$$

初始条件:

$$(\mathbf{0}_A | \frac{1}{r_{12}} | \mathbf{0}_B)^{(m)} = \frac{2\pi^{\frac{5}{2}}}{\alpha\beta\zeta^{\frac{1}{2}}} F_m(T), \quad (3.21)$$

其中

$$T = \xi |\overline{AB}|^2, \quad (3.22)$$

以及

$$F_m(T) = \int_0^1 t^{2m} \exp \{-Tt^2\} dt. \quad (3.23)$$

对于 $m = 0$, 其转化为误差函数

$$F_0(T) = \frac{\sqrt{\pi}}{2\sqrt{T}} \text{Erf}(\sqrt{T}) \quad (3.24)$$

值得一提的是在体系中应该是以双电子积分的形式存在, 不过利用 Gaussian Product Theorem 可以把两边的高斯函数乘积转化成为一系列高斯函数之和, 然后问题就简单了。但这个成本仍旧较大, 我们直接列出双电子积分版本 [3]

$$(\mathbf{0}_A \mathbf{0}_B | \mathbf{0}_C \mathbf{0}_D)^{(0)} = (\zeta + \eta)^{-1/2} K(\zeta_a, \zeta_b, \mathbf{A}, \mathbf{B}) K(\zeta_c, \zeta_d, \mathbf{C}, \mathbf{D}) F_0(T), \quad (3.25)$$

其中

$$K(\zeta, \zeta', \mathbf{R}, \mathbf{R}') = 2^{1/2} \frac{\pi^{5/4}}{\zeta + \zeta'} \exp \left[-\frac{\zeta \zeta'}{\zeta + \zeta'} (\mathbf{R} - \mathbf{R}')^2 \right], \quad (3.26)$$

以及

$$\eta = \zeta_c + \zeta_d \quad (3.27)$$

3.3 Hartree-Fock 迭代

终于到了最为核心的 Hartree-Fock 部分！在解决了所有电子积分相关问题以后后面就是纯粹的线性代数问题了。

3.3.1 非正交基组下的特征解问题——Löwdin 方法

初始想法很简单——我们做这么一个变化：

$$FC = SCE \Rightarrow FS^{-\frac{1}{2}}S^{\frac{1}{2}}C = S^{\frac{1}{2}}S^{\frac{1}{2}}CE \Rightarrow S^{-\frac{1}{2}}FS^{-\frac{1}{2}}S^{\frac{1}{2}}C = S^{\frac{1}{2}}CE,$$

我们就能够看到对基组进行一个 $S^{1/2}$ 的变换，对 F 进行变换 $F' = S^{-\frac{1}{2}}FS^{-\frac{1}{2}}$ ，我们就能够完美抵消 S 的存在，最终就能够转换成为

$$F'C' = C'E. \quad (3.28)$$

于是问题转化成为如何求得 $S^{-\frac{1}{2}}$ 及 $S^{\frac{1}{2}}$ 。注意到 S 为实对称矩阵（而且是正定矩阵），

$$SU = sU \Leftrightarrow S = UsU^T \quad (3.29)$$

其中 s 为 S 对角化后的矩阵， U 为对应的特征向量组。然后约定 $s^{1/2}$ 表示每个特征值取根号值，那么

$$Us^{1/2}U^TUs^{1/2}U^T = UsU^T = S \quad (3.30)$$

因此我们可以判定 $Us^{1/2}U^T$ 为我们想要的 $S^{\frac{1}{2}}$ 。同理，从而有

$$S^{\frac{1}{2}} = Us^{1/2}U^T, \quad S^{-\frac{1}{2}} = Us^{-1/2}U^T. \quad (3.31)$$

有了这个我们就能愉快地求解非正交基组下的特征解问题了。

需要注意的是，进行变换后得到的特征向量组虽然库会帮你正交归一，但在实际使用的时候需要把它还原到原来的原子轨道基组中，也就是要乘上 $S^{-\frac{1}{2}}$ 。

3.3.2 单电子哈密顿矩阵

这一部分是不包含电子之间的排斥的能量。我们有了 T 和 Z 后就很好求了，

3.3.3 初猜

初猜——Initial guess⁸，是通过迭代方法求解某类问题时必需的东西。在本程序中通过对单电子哈密顿量求特征解来获得初始的电子轨道构型。其实可以看到，在能量的构成中单电子哈密顿量占相当大的比重，所以通过单电子哈密顿量来获得初猜也不失为一个不错的选择。我们把单电子哈密顿矩阵假装为 Fock 矩阵：

$$hC_0 = SC_0E_0 \quad (3.32)$$

这个 C_0 便成为我们的初猜构型参与战斗。

⁸这翻译不是我想出来的不关我事

3.3.4 HF 能量

一般 Hartree-Fock 的能量的求法是

$$\langle \Psi | \mathcal{H} | \Psi \rangle = \sum_{m=1}^N [m|h|m] + \sum_{m=1}^N \sum_{n>m}^N ([mm|nn] - [mn|nm]) \quad (3.33)$$

但如果从二次量子化的角度出发，会得到一个很有意思的结果 [1]。在二次量子化下体系的哈密顿量表示为

$$\hat{H} = \sum_{\alpha\beta} h_{\alpha\beta} a_{\alpha}^{\dagger} a_{\beta} + \frac{1}{2} \sum_{\alpha\beta\gamma\delta} v_{\alpha\beta\gamma\delta} a_{\alpha}^{\dagger} a_{\beta}^{\dagger} a_{\delta} a_{\gamma}. \quad (3.34)$$

而电子密度算符为

$$\hat{\rho} = \sum_{\alpha=1}^A |\alpha\rangle\langle\alpha| = \sum_{\alpha=1}^A a_{\alpha}^{\dagger} |0\rangle\langle 0| a_{\alpha} \quad (3.35)$$

即表示所有具有电子占据的轨道对应的投影矩阵相加。而一个 Slater 行列式表示为

$$|\Phi\rangle = \prod_{\alpha=1}^A a_{\alpha}^{\dagger} |0\rangle \quad (3.36)$$

这就有意思了——相互组合，并注意算符交换导致的符号变化，然后就能够得到

$$E[\rho] = \sum_{ij} h_{ij} \rho_j^i + \frac{1}{2} \sum_{ijkl} \bar{v}_{ijkl} \rho_{ji} \rho_{lk}, \quad (3.37)$$

其中

$$\bar{v}_{ijkl} = v_{ijkl} - v_{ikjl} = [ij|kl] - [ik|jl]. \quad (3.38)$$

所以实质上 Hartree-Fock 也是一种 DFT 泛函啦！

求变分，得到 Fock 矩阵

$$F_{ij} = h_{ij} + \sum_{kl} \bar{v}_{ijkl} \rho_{lk}. \quad (3.39)$$

在 RHF (Restricted Hartree-Fock) 中我们只需要考虑轨道，而自旋导致的影响只会出现在双电子积分中：库仑积分为交换积分的两倍，即：

$$F = h_{ij} + \sum_{kl} (2v_{ijkl} - v_{ikjl}) \rho_{lk}. \quad (3.40)$$

4 效果展示

下面是用我自己写的那个程序算的。氢气两个原子的距离为 1 Å. 为了方便大家 debug，我将所有信息都打出来：

STO-3G:


```
Total electrons: 2
Nuclear repulsions: 0.529177
Fock Matrix:
```

```
-0.247300 -0.477659
-0.477659 -0.247300
```

```
Energy: -1.595286
```

```
===== Start SCF =====
```

```
iteration = 1, energy = -1.595286
iteration = 2, energy = -1.595286
iteration = 3, energy = -1.595286
```

```
=====
```

```
SCF converged.
```

```
Fock matrix:
```

```
-0.247300 -0.477659
-0.477659 -0.247300
```

```
MOs:
```

```
MO_NUM: 0, MO_ENERGY = -0.484442 , occ = 2
MO_NUM: 1, MO_ENERGY = 0.457502 , occ = 0
```

```
Total Energy: -1.066109
```

```
MO_LABEL:
[ H1s , H1s ]
```

```
MO_COEFF:
```

```
0.578028 0.996503
0.578028 -0.996503
```

3-21G:

```
Total electrons: 2
Nuclear repulsions: 0.529177
Fock Matrix:
```

```
0.240384 -0.402904 -0.416532 -0.405901
-0.402904 -0.274992 -0.405901 -0.341402
-0.416532 -0.405901 0.240384 -0.402904
-0.405901 -0.341402 -0.402904 -0.274992
```

Energy: -1.589576

===== Start SCF =====

```
iteration = 1, energy = -1.619842
iteration = 2, energy = -1.620547
iteration = 3, energy = -1.620563
iteration = 4, energy = -1.620563
iteration = 5, energy = -1.620563
```

SCF converged.

Fock matrix:

```
0.214398 -0.422002 -0.400401 -0.413854
-0.422002 -0.317225 -0.413854 -0.377158
-0.400401 -0.413854 0.214398 -0.422002
-0.413854 -0.377158 -0.422002 -0.317225
```

MOs:

```
MO_NUM: 0, MO_ENERGY = -0.522932 , occ = 2
MO_NUM: 1, MO_ENERGY = 0.188095 , occ = 0
MO_NUM: 2, MO_ENERGY = 1.067633 , occ = 0
MO_NUM: 3, MO_ENERGY = 1.298707 , occ = 0
```

Total Energy: -1.091386

MO_LABEL:

[H1s , H2s , H1s , H2s]

MO_COEFF:

```
0.251709 0.149322 0.883307 0.950874
0.368854 1.178273 -0.672399 -1.126288
0.251709 -0.149322 0.883307 -0.950874
```

0.368854 -1.178273 -0.672399 1.126288

6-31G:

Total electrons: 2

Nuclear repulsions: 0.529177

Fock Matrix:

```
0.117151 -0.405545 -0.430538 -0.406473
-0.405545 -0.264405 -0.406473 -0.319593
-0.430538 -0.406473 0.117151 -0.405545
-0.406473 -0.319593 -0.405545 -0.264405
```

Energy: -1.588408

===== Start SCF =====

```
iteration = 1, energy = -1.623114
iteration = 2, energy = -1.623964
iteration = 3, energy = -1.623985
iteration = 4, energy = -1.623985
iteration = 5, energy = -1.623985
```

=====

SCF converged.

Fock matrix:

```
0.086185 -0.430438 -0.417083 -0.420114
-0.430438 -0.313131 -0.420114 -0.362158
-0.417083 -0.420114 0.086185 -0.430438
-0.420114 -0.362158 -0.430438 -0.313131
```

MOs:

```
MO_NUM: 0, MO_ENERGY = -0.527541 , occ = 2
MO_NUM: 1, MO_ENERGY = 0.167710 , occ = 0
MO_NUM: 2, MO_ENERGY = 0.904280 , occ = 0
MO_NUM: 3, MO_ENERGY = 1.162273 , occ = 0
```

Total Energy: -1.094808

MO_LABEL:

[H1s , H2s , H1s , H2s]

MO_COEFF:

0.287334	0.167862	0.867287	0.982068
0.331724	1.227983	-0.698468	-1.205946
0.287334	-0.167862	0.867287	-0.982068
0.331724	-1.227983	-0.698468	1.205946

References

- [1] Trygve Helgaker, Poul Jorgensen, and Jeppe Olsen. *Molecular electronic-structure theory*. John Wiley & Sons, 2014.
- [2] Andrew James May. *Density fitting in explicitly correlated electronic structure theory*. PhD thesis, University of Bristol, 2006.
- [3] Shigeru Obara and A Saika. Efficient recursive computation of molecular integrals over cartesian gaussian functions. *The Journal of chemical physics*, 84(7):3963–3974, 1986.
- [4] 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.