1

Numerical Accuracy

Theoretically, if the same **DCF!** is used, all the algorithms mentioned in section **??** should give exactly the same energy $\mathcal{F}_{\text{exc}}$ and gradient γ, in the perfect condition that all the quantities are continuous (with infinite points of grid), the basis sets of **FFT!** and **FGSHT!** are complete (infinite number of $\mathbf{k}$ or order of expansion $n_{\max}$), and the system is border-free (infinite length of box). However, this is obviously impossible, and the errors due to such effects can be far from imperceptible within the level of discretization allowed by the computing capacity nowadays. Commonly, we consider the results to be the same, if the difference of them is at machine precision (10^{-13} to 10^{-15}, or 10^{-7} if single precision **DCF!** is used), the accuracy lost above this precision can have a visible impact on the final results. This section works on all the possible effects and parameters that have an influence to the accuracy, e.g. the usage of different algorithms, or different discretizing parameters; and the goal of this study is to determine in which conditions the errors can be regarded as negligible, thus gives a global view of the credibility for the results given by this code.

1.1 Generalized spherical harmonics transform

As discussed in §**??**, a discretized function $F(\Theta,\Phi,\Psi)$ after a forward-backward **GSHT!** process (eq. (**??**-**??**))

$$F^m_{\mu'\mu}=\frac{f_m}{8\pi^2}\sum_{i=0}^{m_{\max}} w_i \sum_{j=0}^{2m_{\max}} \sum_{k=0}^{2\lfloor m_{\max}/s\rfloor} F(\Theta_i,\Phi_j,\Psi_k)R^{m*}_{\mu'\mu}(\Theta_i,\Phi_j,\Psi_k) \tag{1.1}$$

$$F(\Theta_i,\Phi_j,\Psi_k)=\sum_{m=0}^{n_{\max}} f_m \sum_{\mu'=-m}^{m} \sum_{\substack{\mu=-m\\ \text{mod }(\mu,s)=0}}^{m} F^m_{\mu'\mu}R^m_{\mu'\mu}(\Theta_i,\Phi_j,\Psi_k) \tag{1.2}$$

only remains the same when:

1. F can be expanded on **GSH!**s of order at most $n_{\max}$ in eq. (1.2), thus it is always the case that F is a polynomial of both $\cos\Theta$, $\cos\Phi$ and $\cos\Psi$ of order n, where $n\le n_{\max}$;
2. The order of quadrature $m_{\max}$ used in the **GSH!** expansion in eq. (1.1) should be larger than the order of the polynomial F, which means $n\le m_{\max}$;

The two rules are quite evident. It should be also noted that as we do first a forward then a backward transform, when $m_{\max}<n_{\max}$, even the input function is of order at most $m_{\max}$; the output function is of order $n_{\max}$ in the presence of $R^m_{\mu'\mu}$ which is of order $n_{\max}$. That means F is no longer the same. Therefore, the forward-backward error is also controlled by the relation between the two discretizing parameters:

3. $m_{\max}\ge n_{\max}$ is needed to ensure the absence of accuracy lost.

In reality, the density variable $\rho(\mathbf{r},\mathbf{\Omega})$ and the gradient $\gamma(\mathbf{r},\mathbf{\Omega})$ that should be expanded via **FGSHT!** are never a simple polynomial. It is important to understand how much

the choice of $m_{\max}$ and $n_{\max}$ will affect the accuracy, as they are tightly linked to the performance. Therefore we chose some simple functions below to see what happens when the function does not meet the three conditions. Note that the **FFT!** process leads to strictly no accuracy lost (at machine precision), which means the **FGSHT!** process will have strictly the same result with the **GSHT!** process. Here we do not need to distinguish the two.

1.1.1 $m_{\max}$ *and* $n_{\max}$ *of projections*

The numerical error tests of a forward-backward **GSHT!** process with different order $n_{\max}$ of **GSH!** and $m_{\max}$ of quadrature are shown in table 1.1 for various polynomials, the absolute error

$$E_{\mathrm{a}}(\mathbf{\Omega}) = \left| f^{\mathrm{before}}(\mathbf{\Omega}) - f^{\mathrm{after}}(\mathbf{\Omega}) \right| \tag{1.3}$$

being defined as the norm of difference in function $f(\mathbf{\Omega})$ after a forward-backward **GSHT!** process. The maximum absolute error $E_a^{\max}$ is the maximum value in $E_{\mathrm{a}}(\mathbf{\Omega})$.

From table 1.1 we can see, as function $f(\mathbf{\Omega}) = 1 = R_{00}^0$, it should verify $m_{\max} \geq 0$, $n_{\max} \geq 0$ and $m_{\max} \geq n_{\max}$ to have a null difference. The same, function $f(\mathbf{\Omega}) = \cos 3\Theta = 4\cos^3\Theta - 3\cos\Theta$ is a polynomial of $\cos\Theta$ of order 3, which can be expanded on R_{00}^3 term and R_{00}^1 term; it should satisfy $m_{\max} \geq n_{\max} \geq 3$. $f(\mathbf{\Omega}) = \cos 3\Phi$ is a polynomial of $\cos\Phi$ of order 3, but it cannot be expanded on **GSH!**s. In fact, for all the functions in which the order of $\cos\Phi$ and $\cos\Psi$ is greater than $\cos\Theta$, it cannot be expanded on a finite number of **GSH!**s. The R_{30}^3 itself is an expansion on itself, thus a polynomial of order 3 requiring $m_{\max} \geq n_{\max} \geq 3$.

We can see that, all the $E_a^{\max}$ where $m_{\max} < n_{\max}$ is horrible, and fortunately we use only $m_{\max} \geq n_{\max}$ in **FGSHT!** for numerical reason. Yet the $E_a^{\max}$ for $m_{\max} \geq n_{\max}$ is not negligible neither, knowing that the values of trigonometric functions are in between $[-1, 1]$. However, if the most portion of the function is a polynomial within the required order, the extra part does not play a great role, such that the total mean error is not as mush as seen now.

1.1.2 *From* ρ *to* γ

The same way, the maximum absolute error of the density variable $\rho(\mathbf{r}, \mathbf{\Omega})$, which is not a combination of **GSH!**s, can be huge. It even gives the appearance of unphysical density $\rho(\mathbf{r}, \mathbf{\Omega}) < 0$ (i.e. $\Delta\rho(\mathbf{r}, \mathbf{\Omega})/\rho_0 < -1$) at certain points after a forward-backward **GSHT!** process, as shown in figure 1.1.

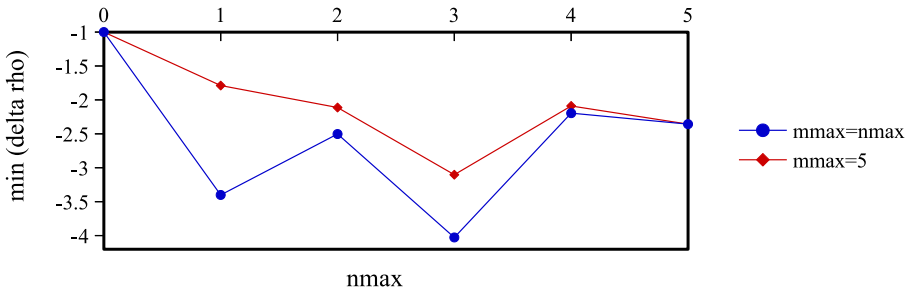


Figure 1.1: The minimum value of $\Delta\rho(\mathbf{r}, \mathbf{\Omega})/\rho_0$ after a forward-backward **GSHT!** process with respect to $n_{\max}$. Computed for a 45^3 grid ($L = 25$) for a converged density of an artificial charged LJ center $\mathrm{CH}_4^{+0.4}$.

$m\backslash n$	0	1	2	3	4	5
0	**0 (0)**	9.00 (3.00)	34.00 (18.00)	83.00 (39.00)	164.00 (84.00)	285.00 (139.00)
1	**0 (0)**	**0 (0)**	0 (1.67)	4.34 (6.07)	7.06 (13.63)	14.88 (17.30)
2	**0 (0)**	**0 (0)**	**0 (0)**	0 (0)	0 (0)	5.65 (2.71)
3	**0 (0)**	**0 (0)**	**0 (0)**	**0 (0)**	0 (0)	0 (0)
4	**0 (0)**	**0 (0)**	**0 (0)**	**0 (0)**	**0 (0)**	0 (0)
5	**0 (0)**	**0 (0)**	**0 (0)**	**0 (0)**	**0 (0)**	**0 (0)**

(a) $f(\mathbf{\Omega}) = 1$

$m\backslash n$	0	1	2	3	4	5
0	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)
1	0.96 (0.96)	0 (0)	0 (0)	2.56 (6.99)	10.76 (14.15)	13.83 (21.21)
2	0.46 (0.46)	0 (0)	0 (0)	0 (0)	0 (0)	1.36 (0.50)
3	0.86 (0.86)	0.66 (0.66)	0.66 (0.66)	**0 (0)**	0 (0)	0.66 (0.66)
4	0.99 (0.99)	0.80 (0.80)	0.80 (0.80)	**0 (0)**	**0 (0)**	0 (0)
5	0.83 (0.83)	1.01 (1.01)	1.01 (1.01)	**0 (0)**	**0 (0)**	**0 (0)**

(b) $f(\mathbf{\Omega}) = \cos 3\Theta$

$m\backslash n$	0	1	2	3	4	5
0	0 (0)	9.00 (3.00)	34.00 (18.00)	83.00 (39.00)	164.00 (84.00)	285.00 (139.00)
1	0 (0)	0 (0)	0 (1.67)	4.34 (6.07)	7.06 (13.63)	14.88 (17.30)
2	1.00 (1.00)	1.00 (1.00)	0.50 (0.50)	1.53 (1.53)	1.15 (1.15)	3.65 (0.89)
3	1.00 (1.00)	1.00 (1.00)	1.00 (1.00)	0.83 (0.83)	1.10 (1.10)	1.11 (1.11)
4	1.00 (1.00)	1.00 (1.00)	1.00 (1.00)	0.90 (0.90)	0.90 (0.90)	0.69 (0.69)
5	1.00 (1.00)	1.00 (1.00)	1.00 (1.00)	0.94 (0.94)	0.94 (0.94)	0.80 (0.80)

(c) $f(\mathbf{\Omega}) = \cos 3\Phi$

$m\backslash n$	0	1	2	3	4	5
0	0 (0)	5.03 (1.68)	19.01 (10.06)	46.40 (21.80)	91.68 (46.96)	- (77.70)
1	0 (0)	0 (0)	0 (0.51)	1.32 (1.85)	2.15 (4.15)	4.53 (5.26)
2	0.56 (0.56)	0.56 (0.56)	0.07 (0.07)	0.55 (0.55)	0.76 (0.76)	2.05 (1.00)
3	0.47 (0.47)	0.47 (0.47)	0.47 (0.47)	**0 (0)**	0.46 (0.46)	0.46 (0.46)
4	0.56 (0.56)	0.56 (0.56)	0.56 (0.56)	**0 (0)**	**0 (0)**	0 (0)
5	0.51 (0.51)	0.51 (0.51)	0.51 (0.51)	**0 (0)**	**0 (0)**	**0 (0)**

(d) $f(\mathbf{\Omega}) = R^3_{30}(\mathbf{\Omega})$

Table 1.1: Maximum absolute error $E_a^{\max}$ of some function $f(\mathbf{\Omega})$ introduced by a forward-backward **GSHT!** process, with $s = 1$ outside the parentheses and $s = 2$ inside the parentheses; s being the **MRSO!** defined in §**??** concerning the C_{2v} symmetry. Differences that should be theoretically null are shown in bold character and double underlined.

Theoretically, we expect this minimum value to approach zero when increasing $m_{\max}$ or $n_{\max}$. This is not exactly the case. That means the order of expansion within the computing capacity ($n_{\max} \leq 5$) is still far from finding a tendency. If we look at the rotational invariant expansion of $\rho(\mathbf{r}, \mathbf{\Omega})$, which gives the projections $\rho_{0\nu}^{0nl}(r)$ (0 as the solute is spherical) shown in figure 1.2:

It is normal that the curves are a little noised because of the irregular grid histogram. Note that the projections $f_{\mu\nu}^{mnl}$ are purely real if f is real and μ, ν are even number.

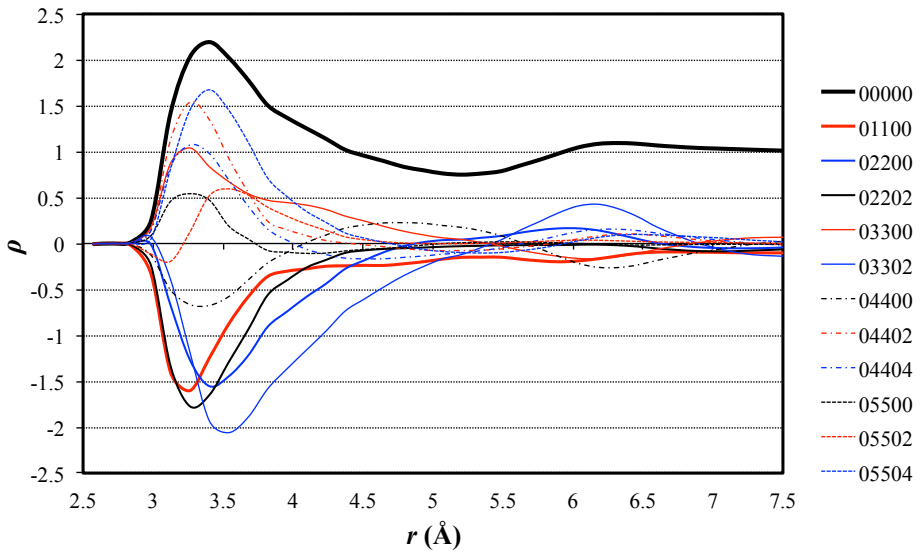


Figure 1.2: The projections $\rho_{0\nu}^{0nl}(r)$ computed for a 45^3 grid ($L = 25$) for a converged density of an artificial charged LJ center $CH_4^{+0.4}$

We observe that the first peaks of projections of higher order are still non-negligible. That gives the tendency in figure 1.1. The first projection $\rho_{00}^{000}(r)$ is purely positive, such that the minimum of ρ is zero; then the more negative projections are add on to the combined function, this error will go to negative. The minimum will have a tendency to converge only if the order of projections are above the current order 5.

That means within the computing capacity, we cannot rightly expand the density ρ on **GSH!**s. However, in the code MDFT, the density ρ is generated by minimization process determined by the gradient of energy γ. Note that $\hat{\gamma}(\mathbf{k}, \mathbf{\Omega})$ is a convolution product, where $\Delta\hat{\rho}(\mathbf{k}, \mathbf{\Omega})$ and the **DCF!** $\hat{c}(k, \mathbf{\Omega}_1, \mathbf{\Omega}_2)$ can be both expanded on **GSH!**s and rotational invariants. Therefore, the higher order terms vanishes more easily in $\hat{\gamma}$, as they are the product of higher order terms of $\Delta\hat{\rho}(\mathbf{k}, \mathbf{\Omega})$ and $\hat{c}(k, \mathbf{\Omega}_1, \mathbf{\Omega}_2)$. We see in figure 1.3 that the first terms of γ is much stronger than the terms of $n_{\max} = 3, 4, 5$. Therefore we can consider that the expansion of γ already converge within $n_{\max} \leq 5$, even $n_{\max} \leq 3$.

1.2 COMPARISON BETWEEN BRANCHES

The algorithms mentioned in section **??** should give the same result if the same **DCF!** is used, in the condition that the error due to discretization is not fatal. The most direct comparison can be done with the free energy and structure obtained in the end of minimization. To be more strict, it is also worthwhile to study only the $\mathcal{F}_{\text{exc}}$ functional evaluation during one iteration with out minimization, i.e. the process shown in figure **??**. In this scope, $\gamma(\mathbf{r}, \mathbf{\Omega})$ becomes a better detailed criteria than $\mathcal{F}_{\text{exc}}$ to be compared.

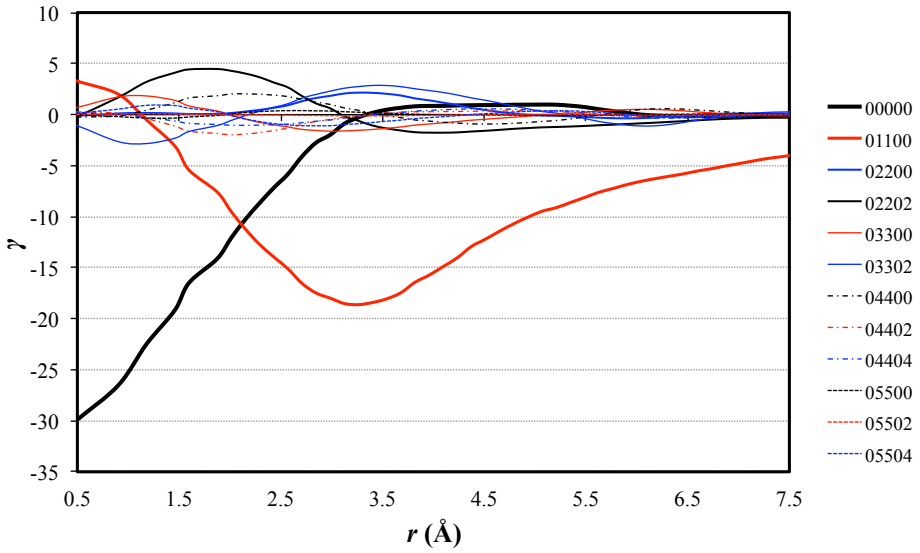


Figure 1.3: The projections $\gamma_{0\nu}^{0nl}(r)$ computed for a 45^3 grid ($L = 25$) for a converged density of an artificial charged LJ center $CH_4^{+0.4}$

1.2.1 *Difference in energy evaluation*

Illustrated in table 1.2, the methods using the same **DCF!** give nearly the same total free energy in the end of minimization.

METHOD	$n_{\max}$	DCF	FREE ENERGY (kJ/mol)
`dipole`	1	[**zhao_accurate_2013**]	13.9646297500206664
`naive_dipole`	1	[**zhao_accurate_2013**]	13.9646299010436010
`convolution_standard`	1	[**zhao_accurate_2013**]	13.9646299010438639
`naive_standard`	1	[**puibasset_bridge_2012**]	19.2241714278933635
`naive_interpolation`	1	[**puibasset_bridge_2012**]	19.4344733154547598
`naive_nmax1`	1	[**puibasset_bridge_2012**]	19.2245193794924099
`convolution_standard`	1	[**puibasset_bridge_2012**]	19.2245193794925164
`convolution_asymm`	1	[**puibasset_bridge_2012**]	19.2245193794927829
`convolution_pure_angular`	1	[**puibasset_bridge_2012**]	19.2245193794925378
`naive_standard`	3	[**puibasset_bridge_2012**]	26.1046424470133296
`naive_interpolation`	3	[**puibasset_bridge_2012**]	26.9706267349261175
`convolution_standard`	3	[**puibasset_bridge_2012**]	26.1046424470137062
`convolution_asymm`	3	[**puibasset_bridge_2012**]	26.1046424470142888
`convolution_pure_angular`	3	[**puibasset_bridge_2012**]	26.1046424470141005

Table 1.2: Minimized free energy via different branches MDFT of a charged $CH_4^{+0.33}$ LJ center calculated for a 33^3 ($L = 20$Å) grid. Gauss-Legendre quadrature is used as Θ angle grid, with $m_{\max} = n_{\max}$.

In table 1.2, we can see that apart from `naive_interpolation`, all branches give the same result at machine precision if other parameters are fixed. The slight difference due

to the interpolation error is also acceptable (in this case); and the error due to the **GSH!** expansion (difference between for instance **`naive_standard`** and **`convolution_standard`**) seems to be well compensated during the minimization. This also supports the conclusion above that in fact we do not need as much as order of **GSH!** expansion for γ than for $\Delta\rho$.

1.2.2 *A single k-kernel*

If we want to compare within one evaluation of $\mathcal{F}_{\text{exc}}$ as shown in figure **??**, we can be first interested in the local paths from $\Delta\hat{\rho}^m_{\mu'\mu}(\mathbf{k})$ to $\hat{\gamma}^m_{\mu'\mu}(\mathbf{k})$ that can be tested independently for a given $\mathbf{k}$. Referring to figure 1.4, four algorithms are available for such a purpose:

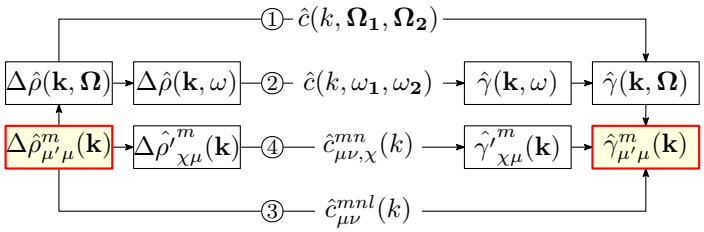


Figure 1.4: A k-kernel

A program in the purpose to compare each element of $\hat{\gamma}^m_{\mu'\mu}(\mathbf{k})$ issued from these four algorithms for a given $\Delta\hat{\rho}^m_{\mu'\mu}(\mathbf{k})$ shows that the $\hat{\gamma}^m_{\mu'\mu}(\mathbf{k})$ for the four algorithms are strictly identical, i.e. the maximum error is at machine precision. This means the final result of energy and structure is independent to the choice of path inside a k-kernel, if $\Delta\hat{\rho}(\mathbf{k}, \mathbf{\Omega})$ can be fully expanded on **GSH!**s.

1.2.3 *k-border effect*

The next step is to test the whole process shown in figure **??**. Theoretically, if $\Delta\rho(\mathbf{r}, \mathbf{\Omega})$ is generated from a recombination of **GSH!** projections $\Delta\rho^m_{\mu'\mu}(\mathbf{r})$, all the branches should give mathematically the same gradient $\gamma(\mathbf{r}, \mathbf{\Omega})$.

The detailed error value of γ here has not been noted, as it was regarded as a bug in the code at that time, then the code has since been modified. But in fact, the choice to add corrections or not on the even number grid is not something about right and wrong.

First off, we compare the three **`convolution`** algorithms passing by **GSH!** expansion. For a 64^3 grid, $n_{\max} = 3$, the three algorithms **`convolution_standard`**, **`convolution_asymm`**, and **`convolution_pure_angular`** give the same $\mathcal{F}_{\text{exc}}$ on a naked eye, but somewhat different results when comparing each element of $\gamma(\mathbf{r}, \mathbf{\Omega})$. The perceived difference seems to decrease when increasing the number of grid points. Moreover, $\gamma(\mathbf{r}, \mathbf{\Omega})$ recombined with projections $\gamma^m_{\mu'\mu}(\mathbf{r})$, which should be purely real as explained in §**??**, have a slight imaginary part. Surprisingly, for a 65^3 grid, it gives numerically the same $\gamma(\mathbf{r}, \mathbf{\Omega})$ for all three algorithms at machine precision. The theory behind this behavior is found to be a special k-border effect linking to even number of nodes in any dimension of the grid.

As the symmetry

$$\Delta\hat{\rho'}^m_{\chi\mu}(\mathbf{k}) = (-)^{m+\mu+\chi}\Delta\hat{\rho'}^{m*}_{\chi,-\mu}(-\mathbf{k}) \tag{1.4}$$

is generated by two symmetries

$$\Delta\hat{\rho}^m_{\mu'\mu}(\mathbf{k}) = (-)^{\mu'+\mu}\Delta\hat{\rho}^{m*}_{-\mu',-\mu}(-\mathbf{k}) \tag{1.5}$$

$$R^m_{\mu'\chi}(\hat{\mathbf{k}}) = (-)^{m+\mu'+\chi}R^m_{-\mu',\chi}(-\hat{\mathbf{k}}) \tag{1.6}$$

For the k points "at border", i.e. after the **FFT!** where the point having $\pm k_i = k_i^{\max}$, $i = 1, 2, 3$, for example for k_1,

$$\Delta\hat{\rho}^m_{\mu'\mu}(\pm k_1, k_2, k_3) = \Delta\hat{\rho}^m_{\mu'\mu}(k_1^{\max}, k_2, k_3) \tag{1.7}$$

is naturally put in the same array by FFT for the grids of an even number, as shown in figure 1.5.

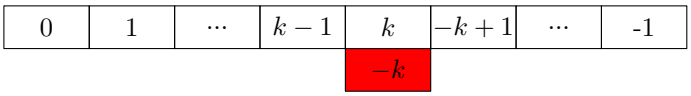


Figure 1.5: k-border effect

For example, for a grid 1D, the FFT having 6 points gives the values for indices 0,1,2,3,-2,-1, and the FFT having 7 points gives the values for 0,1,2,3,-3,-2,-1.

As **FFT!** possesses periodicity, the symmetry 1.10 can always be respected at the border. However, as

$$R^m_{-\mu',\chi}(-\hat{k} \equiv (-k_1,-k_2,-k_3)) \neq R^m_{\mu',\chi}(k_1^{\max},-k_2,-k_3) \tag{1.8}$$

the symmetries (1.6) and (1.4) are not respected for these points. In the backward process, if we make sense of all the $\gamma^m_{\mu'\mu}(\mathbf{k})$, as

$$\gamma^m_{\mu'\mu}(-\hat{k} \equiv (-k_1,-k_2,-k_3)) \neq \gamma^m_{\mu'\mu}(k_1^{\max},-k_2,-k_3) \tag{1.9}$$

the symmetry

$$\gamma^m_{\mu'\mu}(\mathbf{k}) = (-)^{\mu'+\mu}\gamma^{m*}_{-\mu',-\mu}(-\mathbf{k}) \tag{1.10}$$

is not totally respected, and this imposes that $\gamma^m_{\mu'\mu}(\mathbf{r})$ has a imaginary part. This imaginary part has been omitted implicitly in the "real to complex" **FFT!** process used in, for example, **FGSHT!** in `convolution_standard`, or **FFT!**3D in `convolution_pure_angular`. That is to say, we keep only the part of nonnegative $\mathbf{k}$ or nonnegative μ, supposing that the part we omit respects the symmetry.

In the purpose that the three algorithm gives the same result, we can artificially impose at the border:

$$R^m_{\mu',\chi}(k_i^{\max}) = \frac{1}{2}\left[R^m_{\mu',\chi}(k_i) + R^m_{\mu',\chi}(-k_i)\right] \tag{1.11}$$

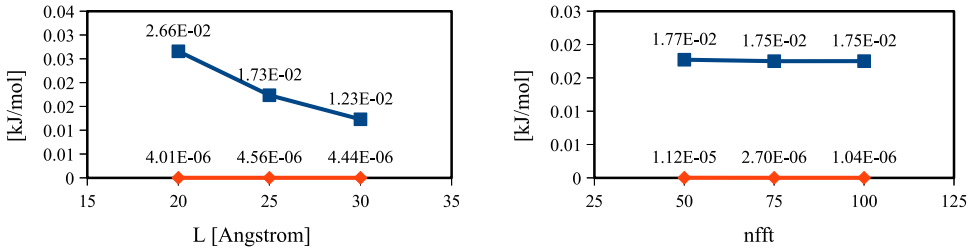
where i is the conflict index in figure 1.5. If more than one dimension is in conflict, this process can be done twice (4 terms for "edges" of the cube) or three times (8 terms for "vertices"). The point $\mathbf{k} = \hat{0}$ is different; as it was defined along z axes to avoid implementation crash, it does not respect eq. (1.6) and (1.4), neither. However, this point is proven to be negligible compared to the hundreds of thousands of total points.

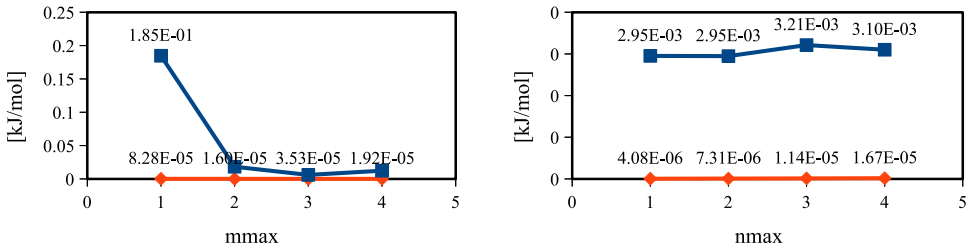
1.2.4 *"naive_standard" and "convolution_standard"*

In the same way, we compare the $\mathcal{F}_{\text{exc}}$ and $\gamma(\mathbf{r},\mathbf{\Omega})$ given by branches `naive_standard` and `convolution_standard`. The $\mathcal{F}_{\text{exc}}$ of these two branches are identical for a 65^3 and $n_{\max} = 3$ grid, but the elements of $\gamma(\mathbf{r},\mathbf{\Omega})$ have a difference at order of 10^{-2} to 10^{-3} which seems to be random. A test redone for a 45^3 grid is shown in figure 1.6.

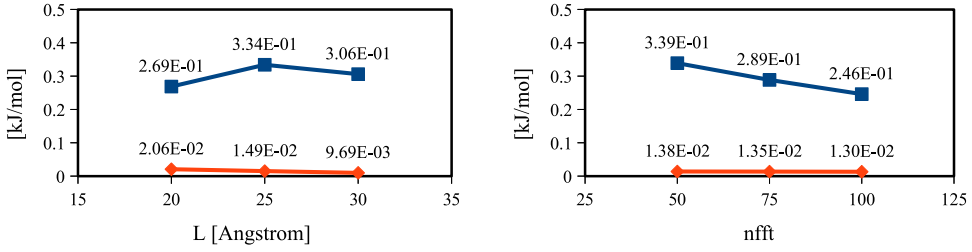
According to figure 1.6, we can hypothesize that this error depends on the angular quadrature $m_{\max}$. The dependence is natural, as the difference between algorithms `naive` and `convolution` is the treatment of the angular part. There is also a dependence on L in the k-space, but after **FFT!** it is mixed. The augmentation of error in the $n_{\max}$ chart is unnatural, implying there is still something theoretical that we have missed, or a bug in the code.

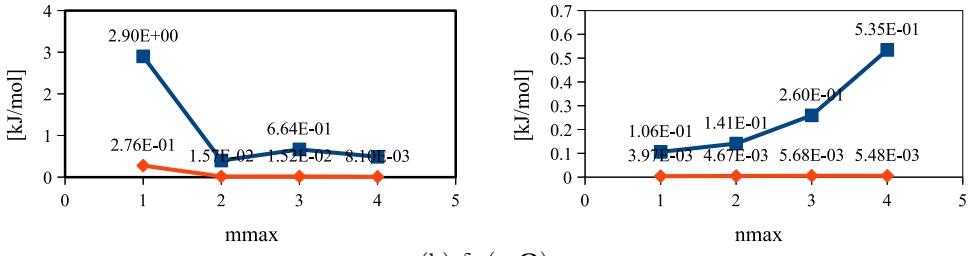
In short, this troublesome difference cannot be yet explained, as the `naive` methods do not have a k-border effect linked to symmetry, and in fact we used an odd grid. The projections $\gamma^{mnl}_{\mu\nu}(r)$ of these two algorithms seem to be identical (figure 1.7), which is to say that the global profile of the two should be almost the same, and thus the error would not be very decisive. Knowing that the difference is even more compensated with many iterations as shown in table 1.2, we can consider that this slight difference has no consequence to the final result.





(a) $\delta\hat{\gamma}(\mathbf{k}, \mathbf{\Omega})$





(b) $\delta\gamma(\mathbf{r}, \mathbf{\Omega})$

Figure 1.6: Maximum (blue) and average (red) absolute difference in $\hat{\gamma}(\mathbf{k}, \mathbf{\Omega})$ and $\gamma(\mathbf{r}, \mathbf{\Omega})$, for tests of: (1) different box length L, with nfft = 65, $m_{\max} = n_{\max} = 2$; (2) different number of grid nfft^3, with $L = 25$ Å, $m_{\max} = n_{\max} = 2$; (3) $n_{\max} = 1$ to 4 for $m_{\max} = n_{\max}$ with 45^3 grid ($L = 25$ Å); (4) $n_{\max} = 1$ to 4 for $m_{\max} = 5$ with 45^3 grid ($L = 25$ Å). The test of $m_{\max} = n_{\max} = 5$ with 45^3 grid ($L = 25$ Å) is dropped as it takes too long. All the tests uses a converged density $\rho(\mathbf{r}, \mathbf{\Omega})$ of an artificial charged LJ center $CH_4^{+0.4}$, recombined from **GSH!** projections of corresponding order $m_{\max}$ and $n_{\max}$.

1.3 INTRINSIC VARIATION OF FREE ENERGY

The comparison between branches shows that there is not so much difference between the branches, if other parameters are all fixed. Therefore it is free to study the dependence on discretizing parameters with one or another algorithm.

Before to study the dependence on order of expansion $m_{\max}$ and $n_{\max}$, we are interested in the spatial grid dependance, as well as the Ψ grid dependance if it is not automatically fixed by $m_{\max}$ (figure 1.8), which can have an influence on the tests later.

Looking at figure 1.8 (a) and (b), we show that the resolution of the spatial grid has an effect on the calculated free energy. For the charged solute $CH_4^{+0.33}$, the energy has a

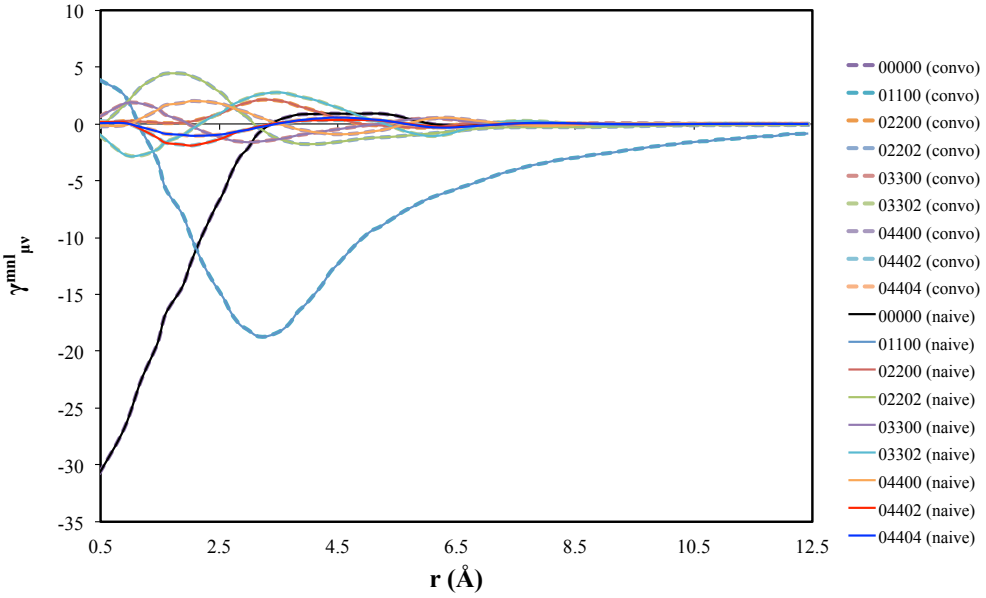


Figure 1.7: Comparison of projections $\gamma^{0nl}_{0\nu}(r)$ for branches "naive_standard" and "convolution_standard", computed with a 45^3 grid ($L = 25$) for a converged density of an artificial charged LJ center $CH_4^{+0.4}$.

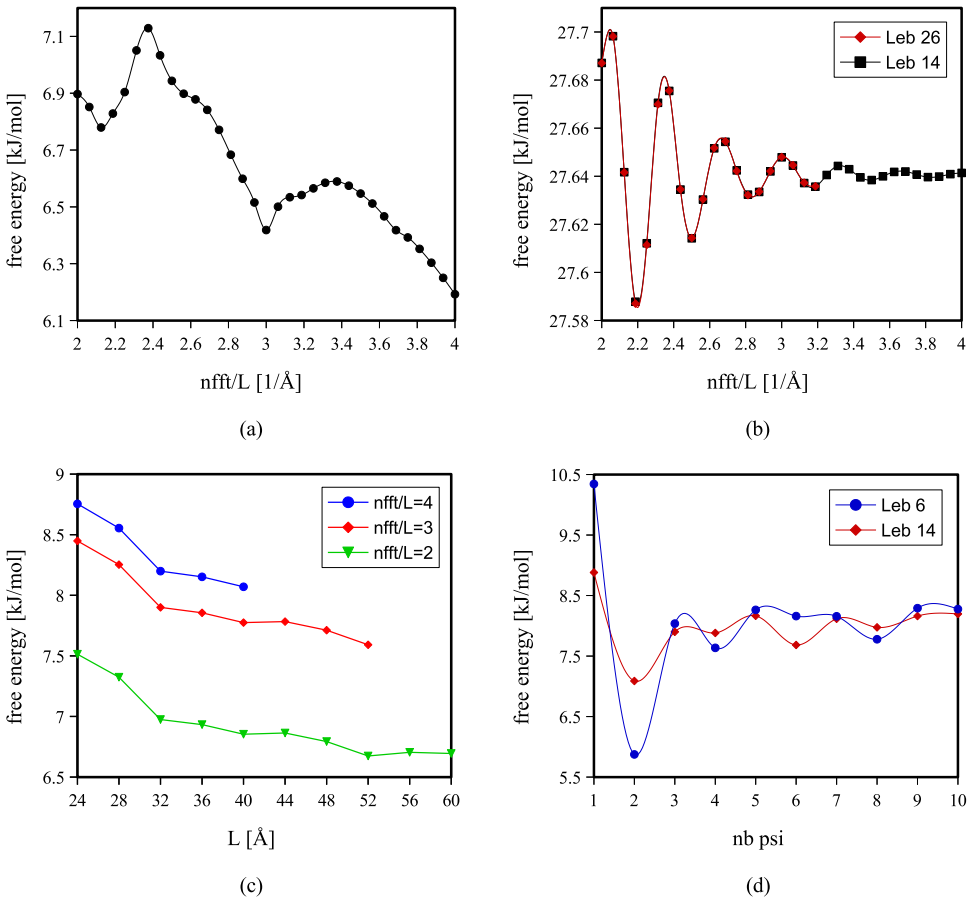


Figure 1.8: Space-grid and Ψ dependence of code **MDFT!**. (a) $CH_4^{+0.33}$ using dipole **DCF!** with $n_{\max} = 1$, $L = 32$ Å; (b) CH_4 using **DCF!** of $n_{\max} = 5$, $L = 32$ Å, Lebedev quadrature of order 2 or 3; (c) acetone using **DCF!** of $n_{\max} = 5$, Lebedev quadrature 2 and 3Ψ angles, nfft$/L = 2, 3, 4$; (d) acetone using dipole **DCF!** and Lebedev quadrature 1 or 2, varying Ψ, $L = 32$ Å, nfft $= 96$.

tendency to decrease when increasing the resolution of grid (nfft). This decrease does not link to the border correction mentioned in §**??**, as both the box length and the charge remain the same for the whole set of test. From (b) we consider that at least a 3 points grid in 1 dimension (each direction) per Angstrom is needed to reduce the uncertainty due to grid resolution. Figure (c) .... (need to be verified: with dipole DCF (recalculated), energy is at 8 kJ/mol, but with Luc DCF (recalculated), energy is at 40kJ/mol, but acetone is soluble!) is neutral, i.e. does not need any border correction neither. Figure (d) fixed the Lebedev quadrature for Θ and Φ, but left varying the Ψ. We can also see a dependence on Ψ. which does not completely have a tendency to vanish when increasing the resolution of grid. Since during the whole thesis the Ψ is theoretically fixed in the same order (as) with Θ and Φ, this remains an issue for further verification. We can roughly conclude that an error around 1 kJ/mol is common for this code.

1.4 SERIES OF CHARGED LJ CENTER

To validate the method by comparing compare with **IET!**, as well as to study the dependance on $m_{\max}$ and $n_{\max}$, we firstly chose a series of LJ centers, which possess the LJ parameters of CH_4 in [**asthagiri_role_2008**], and have a various charge from -1.0 to 1.0 (table 1.3).

*For both IET and DM results in this thesis, 298K is used according to habitude instead of 303K recommended in reference [**SPC/E**]. For MDFT, 300K and 298K are used.*

SOLUTE	$\mathfrak{q}$	σ [Å]	ϵ [kJ · mol^{-1}]
CH_4	-1.0 to 1.0	3.73	1.23

Table 1.3: Parameters of charged CH_4 LJ centers for test usage

1.4.1 *Box length dependance and charge dependance of free energy*

As discussed in section **??**, for single ions, two types of corrections need to be added on the free energy, which depend on the box length and charge of the ion. To verify these dependencies, we implement a systematic calculation from charge using 3 different methods, the parameters of which are shown in table 1.4. It should be noted that, the **`naive_interpolation`** only used 14 Lebedev and 3 Ψ angles to converge, which gives exactly the same result with 26 Lebedev and 4 Ψ angles. That means the **`naive`** methods do not need an order of quadrature $m_{\max}$ to be greater than the order of **DCF!** $n_{\max}$.

METHOD	nfft/L	$m_{\max}$	$n_{\max}$
`naive_nmax1`	3	1 (Leb), 3 angles for Ψ	1
`naive_interpolation`	3	2 (Leb), 3 angles for Ψ	5
`convolution_standard`	3	1	1

Table 1.4: Methods and parameters for CH_4 series test. Leb is Lebedev quadrature (6 angles of Θ and Φ for $m_{\max} = 1$ and 14 for $m_{\max} = 2$), which is mathematically equivalent with Gauss-Legendre quadrature but only ~2/3 angles.

Some points of negative charge diverged; (in **IET!**, the negative charge also have difficulty to converge), and all the converged results are presented.

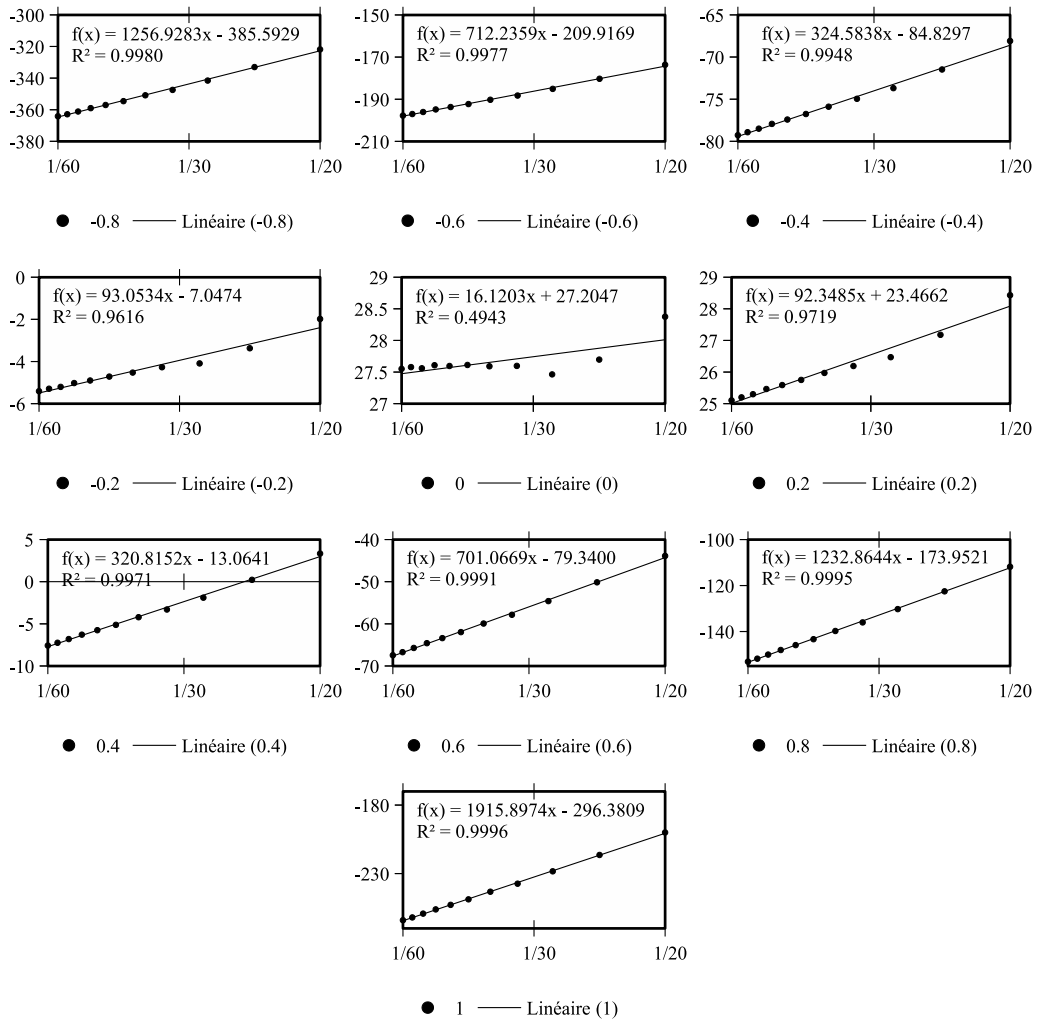


Figure 1.9: Free energy (without correction) of charged CH_4 center (-1.0 to 1.0) with respect to the box length, for **`naive_nmax1`** method, with 6 angles of Lebedev quadrature angles for Θ and Φ, 3 for Ψ, DCF of $n_{\max} = 1$, at 300K.

The collections the raw results issue directly from the code MDFT are shown in figure 1.9, 1.10 and 1.11. We can see that the dependence of box length for each charge is almost linear, except for the charge between $[-0.2, 0.2]$ (where grid dependence dominated compared to other effects). This means the influence of box length is much greater than the intrinsic variation of result mentioned in 1.3. The charge dependency of the slopes in these figures is traced in figure 1.12 with respect to $\mathfrak{q}^2$, square of the corresponding number charge. A linear regression is done to give the slope in figure 1.12 at 1937.8 $\mathrm{kJ \cdot mol^{-1} \cdot Å}$. This slope corresponds to the correction of type-B:

$$\frac{f_Q \xi}{2}\left(1 - \frac{1}{\varepsilon}\right) = 1943.2\,\mathrm{kJ \cdot mol^{-1} \cdot Å} \tag{1.12}$$

where $f_Q = q_e^2 10^{-3} N_\mathrm{A}/(4\pi\varepsilon_0 10^{-10})$ is the electrostatic potential unit so that $f_Q \cdot \mathfrak{q}^2/r$ is in $[\mathrm{kJ \cdot mol^{-1}}]$.

The intercept values in each of figures 1.9 to 1.11 correspond to the free energy of an infinite box. The **IET!** results to be compared are done with $R_{\max} = 102.4$Å, and need a correction of $-2.556 k_\mathrm{B}T$ to obtain the free energy in the infinite system. (“oui! préciser qu’il s’agit de calculs HNC "1D" 1 distance + 5 angles.”: je ne comprends pas...) The difference in free energy between **MDFT!** and **IET!** of the infinite system are given in

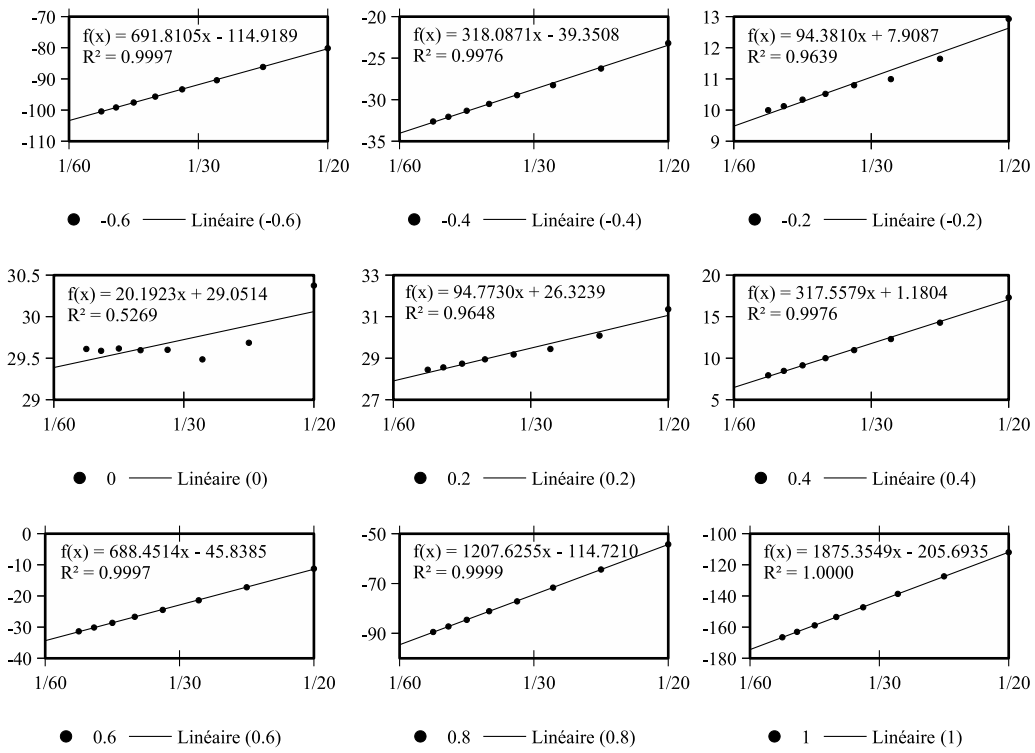


Figure 1.10: Free energy (without correction) of charged CH_4 center (-1.0 to 1.0) with respect to the box length, for **`naive_interpolation`** method, with 14 angles of Lebedev quadrature angles for Θ and Φ, 3 for Ψ, DCF of $n_{\max} = 5$, at 300K.

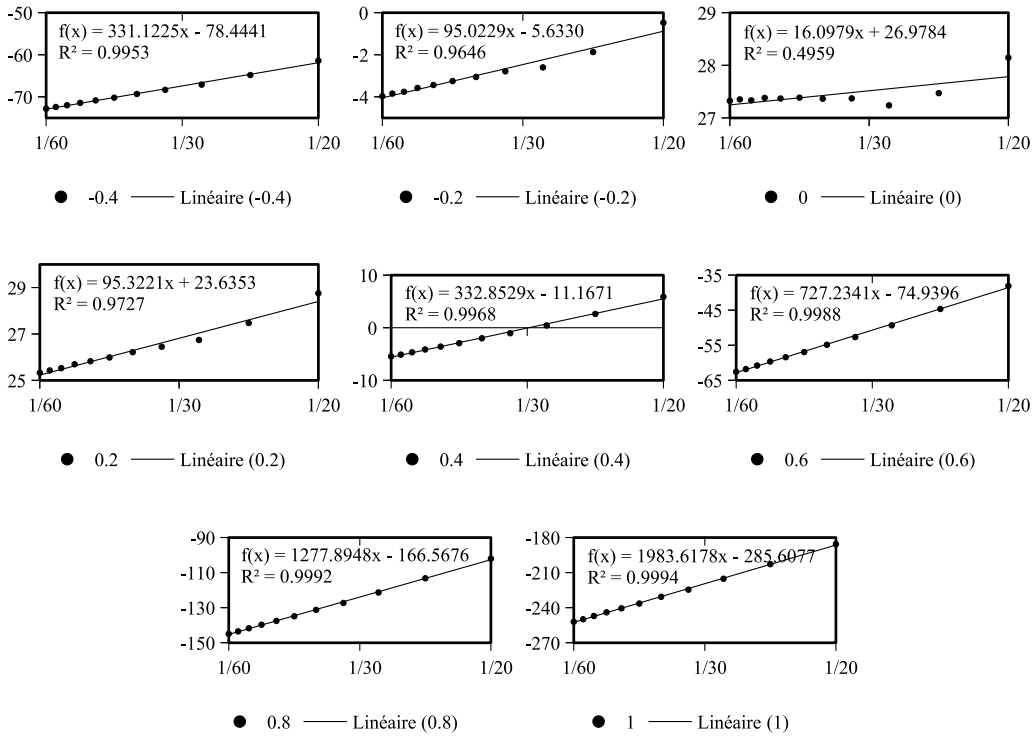


Figure 1.11: Free energy (without correction) of charged CH_4 center (-1.0 to 1.0) with respect to the box length, for **`convolution_standard`** method, with $m_{\max} = n_{\max} = 1$, at 298.15K.

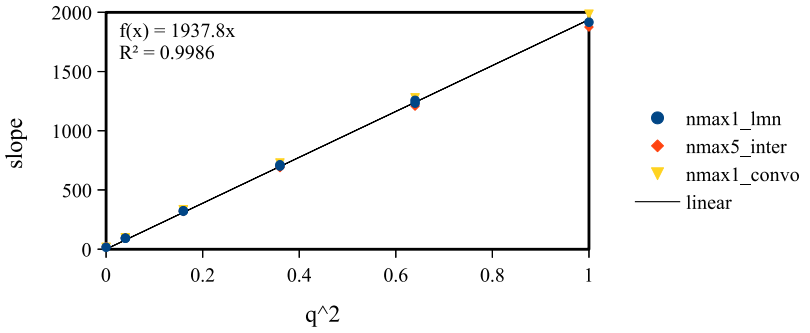


Figure 1.12: Quadratic charge dependence of free energy in CH_4^q series

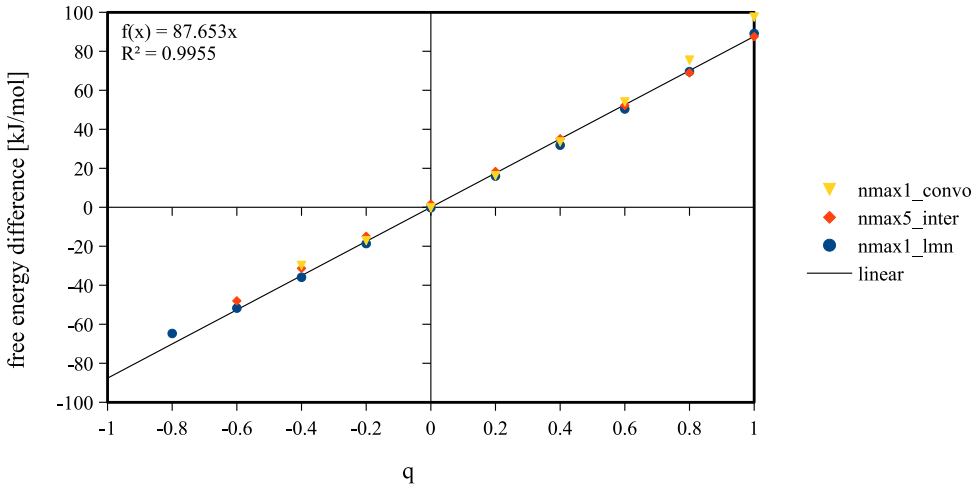


Figure 1.13: Free energy (extrapolated to infinite box length) of charged CH_4 compared to **IET!**, without P-scheme correction

figure 1.13. The linear regression is done with all existing points in this figure, and its slope 87.653 $\mathrm{kJ \cdot mol^{-1}}$ corresponds to the correction of type-C:

$$\frac{2\pi}{3} f_Q \eta \gamma = 82.104 \mathrm{kJ \cdot mol^{-1}} \tag{1.13}$$

The measured number and the theoretical one are a little different, it can be principally due to the lack of point at the -1 charge side.

1.4.2 *Comparison with IET after corrections*

The points in figure 1.13 after correction with eq. (1.13), as well as the results of $m_{\max} = n_{\max} = 5$ given by **`convolution_standard`** using theoretical corrections are shown in figure 1.14. Note that they are not perfectly in agreement with each other. The $n_{\max} = 1$ methods have a larger difference when the charge increases, especially the **`convolution_standard`** using **GSH!** expansion. We see that during one iteration, the **`naive_nmax1`** and the **`convolution_standard`** only give a slight difference in free energy (table 1.2). The $n_{\max} = 5$ has an energy shift of about 2 $\mathrm{kJ \cdot mol^{-1}}$, which cannot be explained. But overall, to have 2 $\mathrm{kJ \cdot mol^{-1}}$ per 100 $\mathrm{kJ \cdot mol^{-1}}$ is already a good result.

The case of $m_{\max} = 5$, $n_{\max} = 0, \ldots, 5$ with **`convolution_standard`** is shown in figure **??**. It is interesting to see how energy evaluates with $n_{\max}$ while fixing $m_{\max}$. As we said, γ is smoother than ρ; that means we can have $n_{\max} < m_{\max}$ to economize computing cost. Results show that within $n_{\max} \geq 3$ for $n_{\max} = 5$, the error is acceptable.

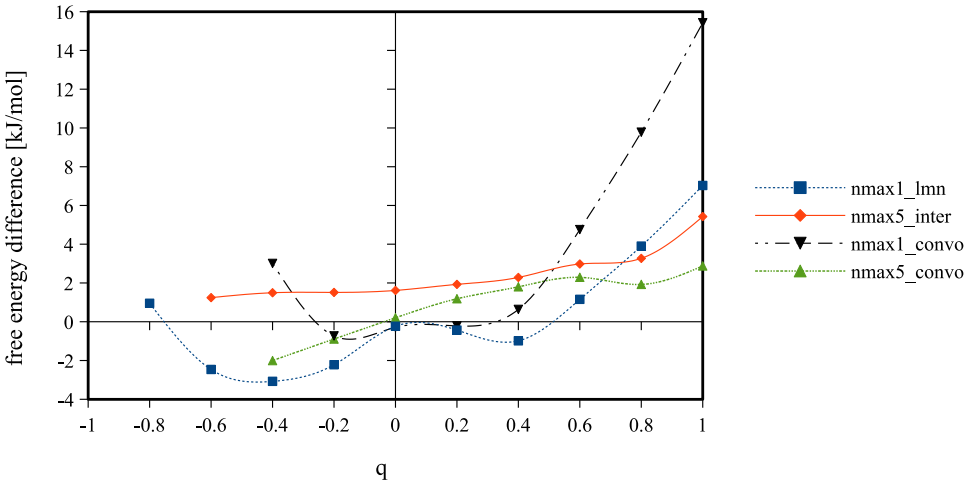


Figure 1.14: Comparison to IET, with P-scheme correction

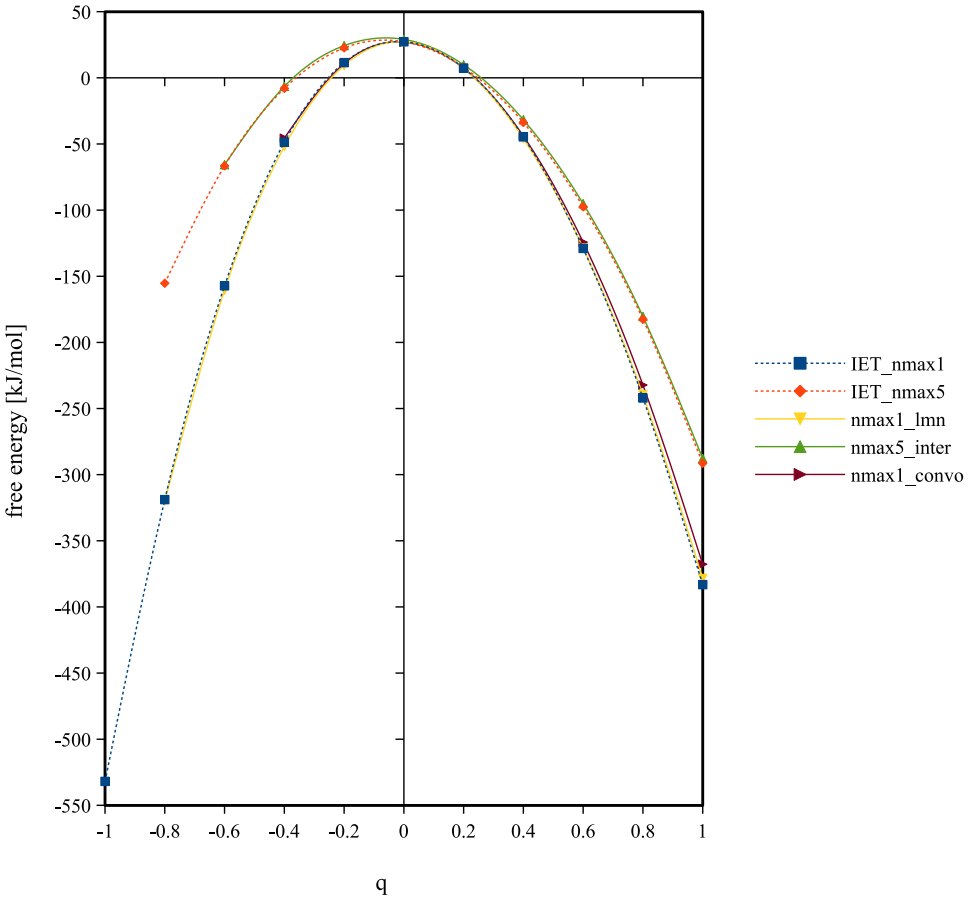


Figure 1.15: Free energy with P-scheme correction

Yet again, the dependence on q after correction is incomprehensible. And compared to **`naive_interpolation`** in figure 1.14, we see that the error for $n_{\max} = 5$ is different.

The nmax 1 not the same because of angular grid

CHARGE		-0.6			0			1	
$n_{\max}\backslash m_{\max}$	**IET!**	$= n_{\max}$	$= 5$	**IET!**	$= n_{\max}$	$= 5$	**IET!**	$= n_{\max}$	$= 5$
1	-157.48	diverge	-163.35	27.33	27.47	27.97	-382.34	-365.89	-379.01
2	-84.32	-87.56	-87.71	27.82	27.94	28.12	-301.74	-298.95	-298.67
3	-66.25	-69.58	-69.70	27.16	27.24	27.50	-294.77	-292.12	-291.87
4	-	-69.41	-69.50	-	27.85	27.91	-	-289.19	-289.04
5	-66.50	diverge	diverge	27.25	27.47	27.47	-291.60	-288.54	-288.54

Table 1.5: free energy (with correction) cmpared to IET

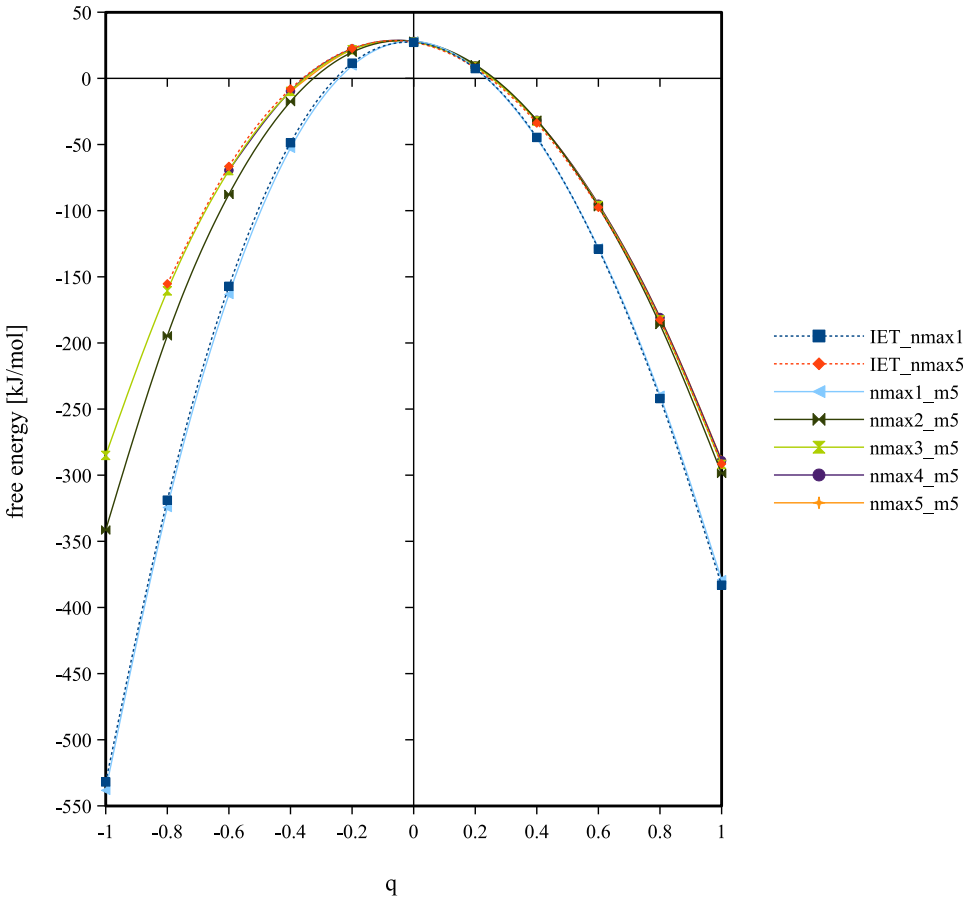


Figure 1.16: Free energy with P-scheme correction

1.4.3 *Dependence on mmax and nmax*

we take particle 2-1 formalism for **MDFT!**, and **IET!** takes the 1-2 formalism. The same issue arrives in the comparison of $g_{\mu\nu}^{mnl}$ of the two approaches, where in **MDFT!** the $g_{\mu\nu}^{mnl}$ corresponds to the $g_{\nu\mu}^{nml}$ in **IET!**.

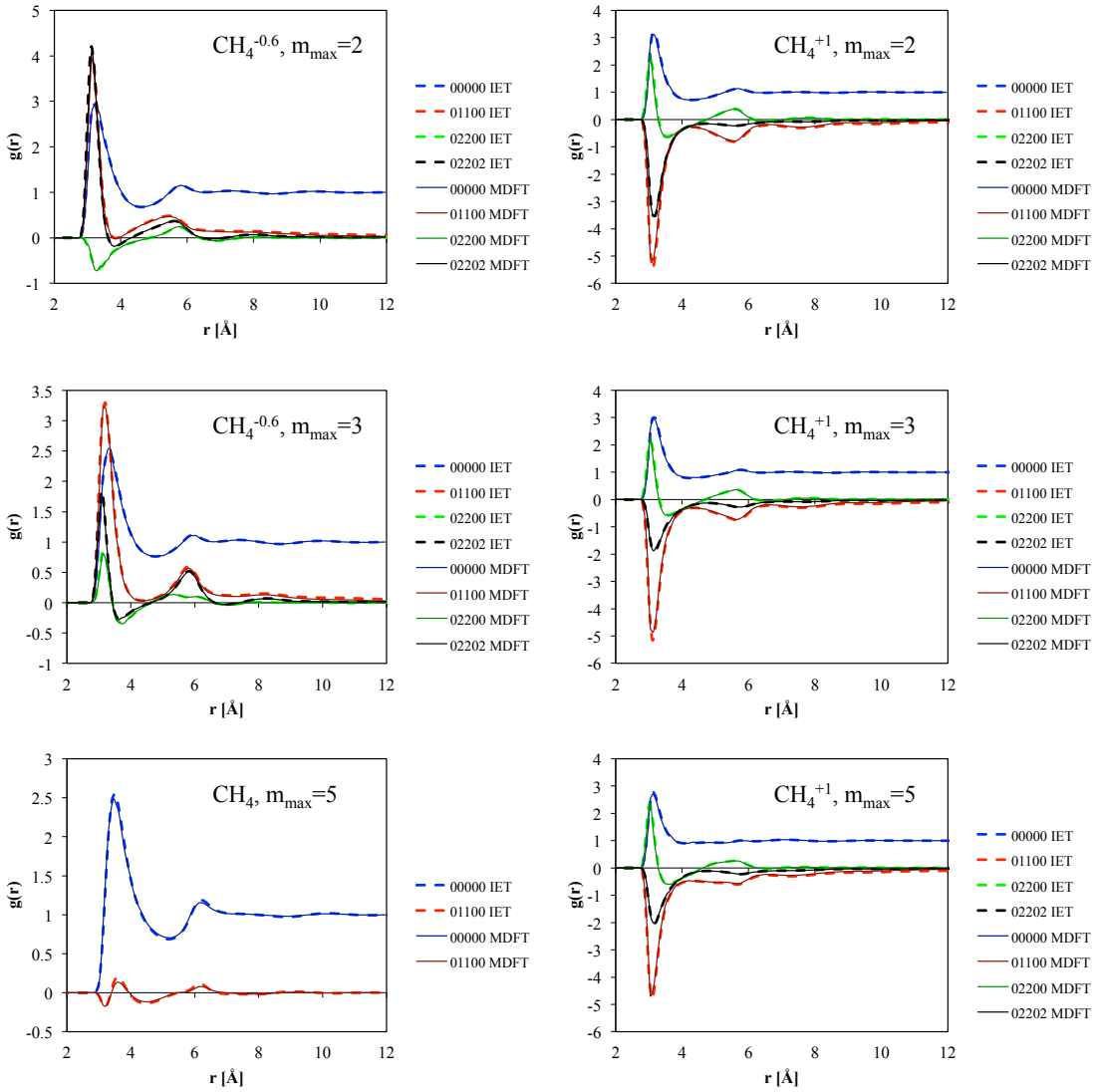
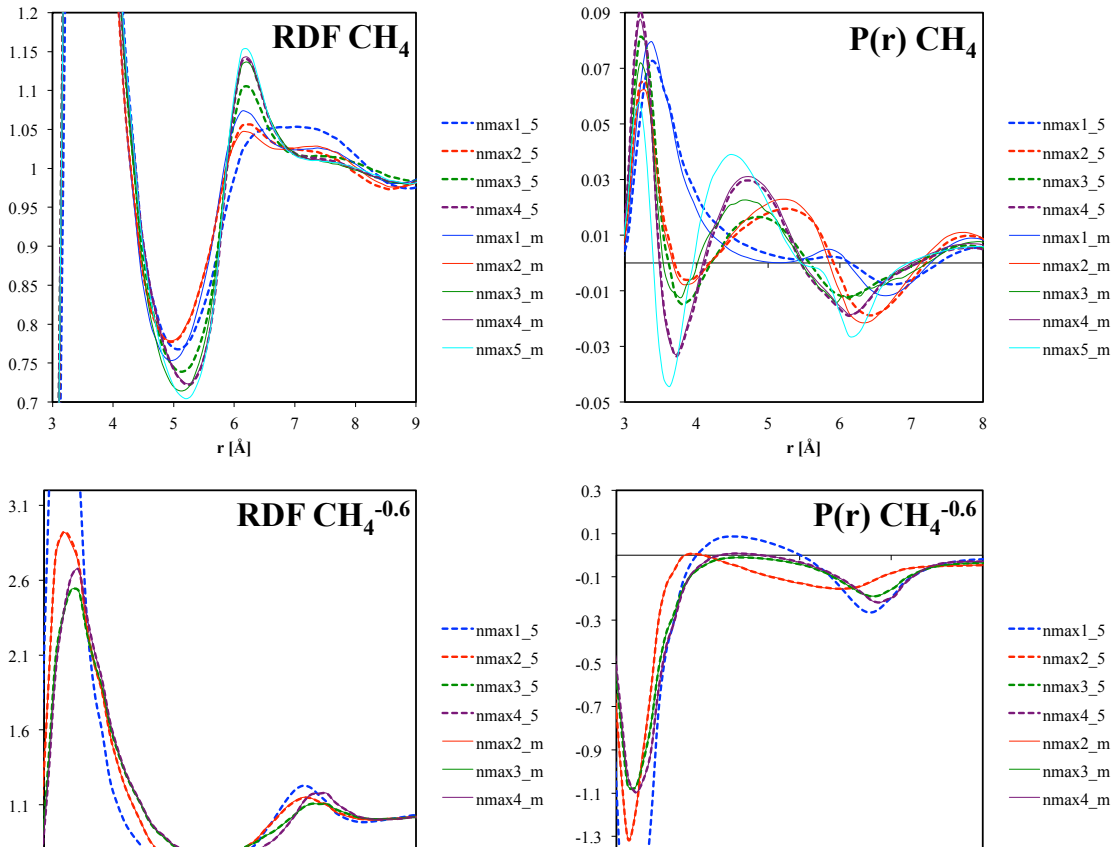


Figure 1.17: Comparison to IET. Profile of ρ in rotational invariant projections, L=24, nfft=72.



SOLUTE	σ	ϵ	dipole	1	2	3	4	5	inter	**IET!**
Neon	3.035	0.15432	18.608	18.793	19.028	18.556	18.990	18.862	20.041	18.823
Argon	3.415	1.03931	22.155	22.465	22.858	22.146	22.722	22.389	24.238	22.270
Krypton	3.675	1.4051	25.356	25.763	26.228	25.483	26.137	25.733	27.823	25.493
Xenon	3.975	1.7851	29.678	30.241	30.766	30.125	30.735	30.249	32.658	29.918

Table 1.6: free energy (with correction) cmpared to IET

The profile of ρ can be expanded on rotational invariants as discussed in §**??**. The comparison with **IET!** is done for three charges, 0, -0.6 and +1. Shown in figure 1.17. Observe that the 0 and +1 charges of $m_{\max} = 5$ corresponds well the result of **IET!**, but the -0.6 charge has a lot of noise. (In fact, this configuration had difficulty in converging, and the given energy is not good, neither. Thus it is deleted in figure **??**.) The profiles appear much better for $m_{\max} = 2, 3$. Normally, more points means more precision. It may means indicate that with $m_{\max} = 5$, there is perhaps a bug of integer overflow that prevents the convergence for high charges.

compare nmax1-5 !!!

1.5 UNCHARGED LJ CENTERS

1.6 SOLUTES

Linear solutes

compared to Luc

We test

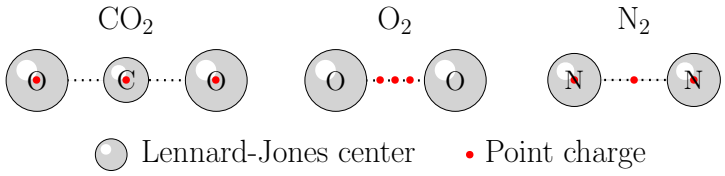


Figure 1.19: Test linear solutes

...

...

22.05

.

1.7 PREMIER CONCLUSION

From the results, we see that **MDFT!** ... "capable" to produce the same result with **IET!** for single ions. (but have more ability to calculate 3D molecules which is not suitable for spherical coordinates...)

`naive_interpolation` is more stable compared to **`convolution`** methods and can use fewer angles for convergence, although in terms of computing time it cannot compare with **`convolution`** methods, which will be discussed later.

SOLUTE	SITE	q	σ [Å]	ϵ [kJ · mol^{-1}]	x [Å]
CO_2 [**Harris_1995**]	1	0.6512	2.76	0.234	0.000
	2	-0.3256	3.03	0.67	-1.149
	3	-0.3256	3.03	0.67	1.149
O_2 [**Boutard200525**]	1	0.0	3.1062	0.36	-0.485
	2	0.0	3.1062	0.36	0.485
	3	-2.1	0.00	0.00	-0.200
	4	-2.1	0.00	0.00	0.200
	5	4.2	0.00	0.00	0.000
N_2 [ref]	1	-0.5075	3.30	0.30	-0.549
	2	-0.5075	3.30	0.30	0.549
	3	1.0150	0.00	0.00	0.000

Table 1.7: Parameters of test solutes

SOLUTE	CO_2 (20.48 [kJ · mol^{-1}])		N_2 (26.75 [kJ · mol^{-1}])	
$n_{\max}\backslash m_{\max}$	$m_{\max} = n_{\max}$	$m_{\max} = 5$	$m_{\max} = n_{\max}$	$m_{\max} = 5$
1	10.85	8.97	24.13	23.88
2	17.46	17.47	26.02	26.02
3	19.99	20.01	26.09	26.11
4	20.52	20.53	26.58	26.58
5	19.38	19.38	26.23	26.23

Table 1.8: free energy (with correction) cmpared to IET

recomented nmax3
tipically grid 64 with nmax3 is sufficient ?? (x seconde)
we can reproduce the results of IET, energy and structure