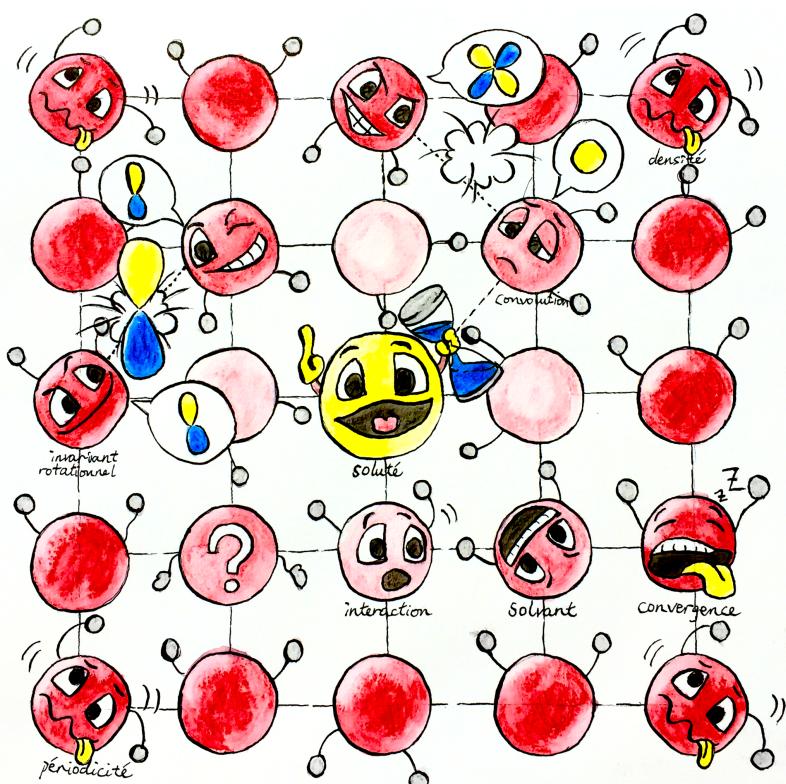


MOLECULAR DENSITY FUNCTIONAL THEORY UNDER HOMOGENEOUS REFERENCE FLUID APPROXIMATION

LU DING

Under the direction of
DANIEL BORGIS, LUC BELLONI & MAXIMILIEN LEVESQUE



SYSTEMATIC PREDICTION OF SOLVATION PROPERTIES
WITH MOLECULAR-SCALE LIQUID THEORY

ABSTRACT

In this thesis, two algorithms of excess energy functional evaluation are proposed, one is extension of the previous work, one is ... The new algorithm combines the molecular Ornstein-Zernike equation method with MDFT

It is shown that ...

The new method is able to calculate ...

RÉSUMÉ

*When you are studying any matter, or considering any philosophy, ask yourself only,
what are the facts and what is the truth that the facts bear out.
Never let yourself be diverted either by what you wish to believe, or by what you think
would have beneficent social effects if it were believed.
But look only, and solely, at what are the facts.*

— Bertrand Russell

ACKNOWLEDGEMENTS

I would love to express my most respectful gratitude to my thesis advisors Daniel Borgis and Luc Belloni, who developed the main theory used in this thesis, and whose great knowledge of liquid theory as well as the genius way of thinking and explaining have given me a solid guide on doing this research. I would also love to thank them for their being engaged in the endless work of correcting this manuscript.

I'm also grateful to Maximilien Levesque, who also belong to this project and joined the supervision of my thesis. Thanks for sharing some results and data that is useful to this thesis.

I wish to acknowledge all the great heads having accepted to evaluate and give ideas about my work: (**noms des jurés**), thank you to be part of the jury of this thesis.

For a master student issue from pure chemistry speciality, a lack of knowledge about informatics brought me a lot of difficulties. I would like to express my sincere gratitude to my colleagues Pierre Kestener, Matthieu Haefele, Yacine Ould-Rouis, and Martial Mancip for huge aide in informatics and very useful advices during this thesis.

This thesis undergoes at *Maison de la Simulation, CEA Saclay*, financially supported by Bourse Idex-CEA. I'm grateful to all the organizations and staffs that gave me the chance to have this three-year experience.

I am also grateful to ... for the help in correcting huge amount of grammar faults in the manuscript.

I'm deeply indebted to my tutors of license and master, Monsieur Hongwei Tan and Madame Michelle Gupta, whose clear logic and warm encouragement gave me all the necessities to be in loved with theoretical chemistry.

Finally I would like to thank my father, who made the right decision to send me here in France and support me in every aspect.

CONTENTS

1	INTRODUCTION	1
1.1	Simulation of solvent effects,	1
1.2	Scope of this thesis,	3
I	STATE OF THE ART: SOLVATION, MODELS AND METHODS	5
2	MODEL OF SOLUTION SYSTEM	7
2.1	Continuum Solvation Models,	7
2.1.1	Poisson–Boltzmann methods,	8
2.1.2	Born/Onsager/Kirkwood models ,	10
2.1.3	Other models adopt widely in QM calculation,	11
2.2	Model potential of rigid molecule,	12
2.2.1	Interaction of spherical particle,	13
2.2.2	Rigid molecule model,	14
2.2.3	SPC/E water and other water models,	15
2.2.4	Model flexible and polarizable,	16
2.3	Model of solute,	16
3	STATISTICAL MECHANICS OF CLASSICAL FLUIDS	19
3.1	Hamiltonian of a micro-state,	19
3.2	Time averages and ensemble averages / Partition functions and Thermo-dynamics,	20
3.3	Distribution functions,	21
4	INTEGRAL EQUATION THEORY	23
4.1	Ornstein-Zernike equation,	23
4.2	Closure,	24
4.3	RISM and 3D-RISM,	24
4.4	Molecular solvent,	24
4.5	Bulk properties and bridge function,	24
5	CLASSICAL DENSITY FUNCTIONAL THEORY	25
5.1	Equilibrium Classical DFT,	25
5.2	Variation principle,	25
5.3	Ideal free energy,	27
5.4	External free energy,	27
5.4.1	Electrostatic potential,	27
5.5	Excess free energy,	28
5.5.1	Homogenous Reference Fluid Approximation,	28
5.5.2	nn_cs,	28
5.5.3	polarisation,	29
5.6	OZ Equation in MDFT and IEM Formalism,	29

5.7 Statistical mechanics,	30
5.8 Principe of Variation,	30
5.9 Ideal free energy functional,	30
5.10 External free energy functional,	30
5.11 Homogenous Reference Fluid Approximation,	30
5.12 OZ Equation in MDFT and IEM Formalism,	30
5.13 Supercell discretization,	31
5.14 cg_vect,	31
5.14.1 Multipolar polarization,	31
5.15 Poisson solver : Iterative solvers for linear equations,	32
5.16 Jacobi's method,	32
5.17 vext q,	32
5.18 Code MDFT,	33
5.18.1 Minimizer L-BFGS-B,	33
5.18.2 Treatment to avoid unphysical density,	34
5.18.3 Evaluation of V_{ext} ,	34
5.18.4 Evaluation of V_{exc} ,	36
6 MOLECULAR DYNAMICS & MONTE CARLO SIMULATION	37
6.1 Molecular Dynamics,	37
6.1.1 Principle,	37
6.1.2 Determination of free energy: umbrella sampling,	37
6.2 Monte Carlo Method,	37
6.3 Direct correlation function of water,	37
6.3.1 Dipole DCF from molecular dynamics simulation,	37
6.3.2 Extraction of DCF from bulk Monte Carlo simulation,	37
6.3.3 Conventions,	37
6.3.4 Comparison with non-coupling dipole DCF in MDFT ($n_{\max} = 1$),	38
6.3.5 Comparison with respect to n_{\max} ,	38
6.3.6 Transform between ck_angular and projections,	38
6.4 Umbrella sampling,	38
II THEORY: HRF APPROXIMATION, FOR MOLECULAR SOLVENT	39
7 REDUCTION OF SPATIAL CONVOLUTION BY FFT	41
7.1 Using intermolecular DCF,	42
7.1.1 Zero-order interpolation of DCF,	44
7.1.2 Linear interpolation of DCF,	44
7.2 Direct calculation of DCF from rotational invariant projections,	45
7.2.1 Using projections in form of $\hat{c}_{\mu\nu}^{mn}(k)$,	45
7.2.2 Using projections in form of $\hat{c}_{\mu\nu,\chi}^{mn}(k)$,	46
8 ANGULAR CONVOLUTION, A BETTER ALGORITHM	47
8.1 Angular convolution using Blum's reduction,	47
8.2 Fast generalized spherical harmonic transform,	48
8.2.1 Equivalence of order in angular quadratures and projections,	49
8.2.2 Integration of Φ, Ψ using FFT,	50
8.3 Operational algorithm,	52

8.3.1 Reduction by symmetry,	53
8.3.2 Commutativity between operations,	53
9 FREE ENERGY AND RELATED THERMODYNAMIC QUANTITIES	57
9.1 Free energy correction for single ion,	57
9.1.1 Correction of type B,	57
9.1.2 Correction of type C,	58
9.2 Some related thermodynamic quantities,	58
9.2.1 Solubility,	59
9.2.2 Pressure,	59
10 SOLVATION STRUCTURE	61
10.1 Radial distribution function and site-site distribution function,	61
10.2 Radial polarization functions,	61
10.3 Rotational invariant expansion,	61
10.4 Rebuilt of density in a certain orientation,	61
10.4.1 Radical distribution function of numeric density,	61
10.4.2 Radical distribution function of polarization,	61
III IMPLEMENTATION	63
11 ALGORITHMS AND BRANCHES	65
11.1 Branches "naive" ,	65
11.2 Branches "convolution",	66
11.3 Testing branches for nmax=1,	66
11.4 Other paths,	66
12 NUMERICAL AND PHYSICAL ACCURACY	67
12.1 Generalized spherical harmonics transform,	67
12.1.1 FFT,	67
12.1.2 m_{\max} and n_{\max} of projections,	67
12.1.3 From ρ to γ ,	67
12.2 Comparison between branches,	68
12.2.1 Error evaluation in $\hat{c}(\mathbf{k}, \Omega_1, \Omega_2)$ calculation,	68
12.2.2 A single k-kernel,	68
12.2.3 k-border effect,	69
12.3 Intrinsic variation of free energy,	70
12.3.1 Spatial grid: length and resolution,	70
12.3.2 Angular grid: effect of psi,	70
12.4 Series of charged LJ centre,	70
12.4.1 Box length dependance and charge dependance of free energy,	70
12.4.2 Comparison with IET,	71
12.4.3 Comparison with DM ,	71
12.5 Premier conclusion,	71
13 COMPUTING PERFORMANCE OF SEQUENTIAL CODE	73
13.1 FFT,	73

13.2 FGSHT,	74
13.2.1 Computing time of GSHT and FGSHT,	74
13.2.2 Performance with respect to m_{\max} ,	74
13.2.3 Performance with respect to n_{\max} ,	74
13.3 K-kernel,	74
13.3.1 Comparison between paths,	74
13.3.2 m_{\max} and n_{\max} dependance of OZ equation,	74
13.4 Entire iteration of \mathcal{F}_{exc} evaluation,	74
13.4.1 Comparison between “naive_standard” and “convolution_pure_angular”,	
74	
13.4.2 Comparison between “convolution_standard” and “convolution_asymm”,	
75	
13.4.3 Comparison between “convolution_standard” and “convolution_pure_angular”,	
75	
13.5 Global view of the sequential code performance,	75
14 COMPUTING PERFORMANCE OF PARALLEL CODE	77
14.1 Node-level Parallelization,	77
14.2 Parallelization on Several Nodes,	77
IV APPLICATIONS	79
15 IONS AND SMALL MOLECULES COMPARED TO MD SIMULATION	81
15.0.1 ions,	81
16 HYDROGEN TRANSFER REACTION OF MN-OXO: ROLE OF SOLVENT IN REACTION SIMULATION	83
16.1 Reinvestigation of the Manganese-oxo ,	83
V CONCLUSION AND PERSPECTIVES	85
17 CONCLUSION	87
18 PERSPECTIVES	89
18.1 Reduce memory use in MDFT,	89
18.1.1 Pass to simple precision,	89
18.1.2 MPI of the L-BFGS-B minimizer,	89
18.2 Polarisable solute,	89
18.3 MDFT Viewer,	89
18.4 Classical SCF method,	89
18.5 Other branches of development about MDFT,	89
VI APPENDIX	91
A BASIC CONCEPTS ABOUT COMPUTING PERFORMANCE	93
A.1 Algorithm complexity,	93
A.2 Roofline model and memory delay,	94

A.3 Scalability of parallelized code,	95
A.4 Profiling and tracing toolkits,	96
B EQUIVALENCE OF QUADRATURE-PROJECTION ORDER	97
B.1 Gaussian quadrature,	97
B.2 Angular integration in GSHT,	97
C ROTATIONAL INVARIANT EXPANSION	99
c.1 Orthogonality of Φ ,	99
c.2 Rotational invariance of Φ ,	100
c.3 Transform in local frame,	101
c.3.1 Transform between $F_{\mu\nu}^{mn l}(r)$ and $F_{\mu\nu,\chi}^{mn}(r)$,	101
c.3.2 Rotational invariant transform with FFT-3D,	102
c.4 Transform in fixed frame,	103
c.4.1 Expansion of $F(\mathbf{r}, \Omega)$ on rotational invariants,	103
c.4.2 Rebuilt of $g(\mathbf{r}, \Omega)$ from projections,	104
c.5 Symmetry,	104
c.5.1 Symmetric rules of $F(\omega_1, \omega_2)$ in intermolecular form,	104
c.5.2 Symmetric rules of rotational invariant projections,	105
D CALCULATION OF ROTATION MATRIX ELEMENTS $R_{\mu\mu'}^m$ BY RECURRENCE	107
d.1 Case of $m_{\max} \leq 1$,	107
d.2 Case of $m_{\max} > 1$,	108
d.2.1 Recurrence relation for $-m + 1 \leq \chi' \leq m - 1$,	108
d.2.2 Recurrence relation for $-m + 2 \leq \chi' \leq m$,	108
d.2.3 Symmetries,	109
E PROPERTIES OF WIGNER 3J-SYMBOL AND GSH	111
e.1 Properties of Wigner 3j-Symbol,	111
e.2 Properties of GSH,	112
F ERROR EVALUATION OF INTERPOLATION STRATEGIES FOR DCF IN LOCAL FRAME	115
G ORIGINAL DATA OF MDFT IMPLEMENTATION	117
g.1 Series of charged CH ₄ center,	117
g.1.1 Branch “naive” result,	117
g.1.2 Branch “standard” result,	117
BIBLIOGRAPHY	119

LIST OF FIGURES

- Figure 1.1 The solvation process, 2
Figure 2.1 Continuum model, 7
Figure 2.2 Solvent-accessible surface area, 8
Figure 2.3 LJ pair potential, 13
Figure 2.4 Euler angles, 14
Figure 2.5 Water models, 15
Figure 2.6 Hierarchy of models, 17
Figure 5.1 Flowchart of code MDFT, 27
Figure 5.2 Main structure of code MDFT, 34
Figure 5.3 Charge density projected onto grids. (a) Solute. (b) Solvent., 35
Figure 7.1 Molecule 1 and 2 in different coordinate systems, 42
Figure 7.2 Rotation matrices, 43
Figure 7.3 Rotation to k-frame, 43
Figure 7.4 $\phi_1 - \phi_2$ distribution, 44
Figure 8.1 Indices arrangement in a complete forward-backward process of FFT-2D, 51
Figure 8.2 mmax to nmax, 52
Figure 8.3 Distribution of points to be calculated according to symmetry in a 2D plan, 53
Figure 8.4 Commutativity of operations, 54
Figure 9.1 IQ model and summation scheme, 58
Figure 10.1 Structure of a POINCARE node, 63
Figure 11.1 Possible algorithms for γ evaluation, 65
Figure 12.1 Schema of a k-kernel test , 68
Figure 12.2 k-border effet, 70
Figure 12.3 structure of gamma in rotational invariants, 70
Figure 12.4 free energy (without correction) of charged CH₄ centre (-1.0 to 1.0) with respect to the box length, 71
Figure 12.5 Parabolic charge dependence of free energy of CH₄ centre series, 71
Figure 12.6 Linear dependence of charge in the comparison to IET. Old algorithm / new algorithm, without correction, 72
Figure 12.7 Comparison to IET. Old algorithm / new algorithm, with P-scheme correction, 72
Figure 13.1 timing FFT, 73
Figure 13.2 Computing time of GSHT and FGSHT (per 1000 times), 74
Figure 13.3 , 74
Figure 13.4 , 75
Figure 13.5 , 75
Figure A.1 Function growth, 93
Figure A.2 The roofline model and performance pattern, 94
Figure A.3 Memory usage in hardware level, 95
Figure C.1 Symmetry operations of a 2-molecule system, 104

Figure F.1 Error of finding $\hat{c}(\mathbf{k}, \Omega_1, \Omega_2)$ by interpolation compared to direct calculation, 115

LIST OF TABLES

Table 1.1	Theories of solvation simulation,	3
Table 2.1	Parameters for SPC and SPC/E water,	16
Table 8.1	Number of FE needed by OZ equation of different form,	49
Table 11.1	Branch option in MDFT,	65
Table 12.1	Maximum absolute error introduced by a forward-backward (real-complex-real) DFT1D process on a 100-value double precision array using FFTW3 package,	67
Table 12.2	Logarithm of maximum absolute error $\lg(E_a^{\max})$ introduced by a forward-backward GSHT process.	, 68
Table 12.3	Maximum absolute error introduced by paths of a k-kernel test,	69
Table 12.4	Error in $\gamma(\mathbf{r}, \Omega)$ and \mathcal{F}_{exc} before border correction,	69
Table 12.6	Parameters of charged Lenard-Jones centre (modified from CH ₄) ,	
		71

LISTINGS

$\mathcal{F}[\rho]$	solvation free energy functional
\mathcal{F}_{exc}	excess free energy functional
$\rho(\mathbf{r}, \Omega)$	density of solvent
$\gamma(\mathbf{r}, \Omega)$	gradient of excess free energy functional,

ACRONYMS

DCF direct correlation function

DFT discret Fourier transform, also refer to density functional theory

FGSHT fast generalized spherical harmonic transform

HRF homogeneous reference fluid (approximation)

IET integral equation theory

MC Monte Carlo

MD molecular dynamics

MDFT molecular density functional theory

OZ Ornstein-Zernike (equation)

QM quantum mechanics

RISM reference interaction site model

INTRODUCTION

This thesis is about to develop an original numerical toolkit for physical chemists and structural biologists, based on the molecular density functional theory (MDFT), which makes it possible to predict efficiently, and with a microscopic accuracy, the solvation properties of arbitrary molecular objects in arbitrary molecular solvents (mainly water). This introduction will help to understand the objective of this thesis, it explains why people are interested in the nature of solvation, and where we are in the computing trends in solvation simulation.

1.1 SIMULATION OF SOLVENT EFFECTS

Solvation is a fundamental phenomenon in chemistry. The chemical behavior of lots of systems has a strong dependence on the nature of solvent. For example, for some popular issue as metal-organic reacting center [13, 38], or pharmaceutical etudes [24, 39, 40]. The solvation properties required by etudes are very variable, such as the Gibbs free energy of solvation, solubility, partition coefficient, saturated vapor pressure, pH value, as well as the 3D solvation structure, etc. Overall, the interest of these solvation properties comes from many domains, such as chemistry, biochemistry, pharmaceutics, medicine, environmental and agrochemical industries. Unlike the well-studied quantum mechanics (QM) for chemical interaction and macroscopic finite element model for physical process, the theories of solvation are very variable and still under developing, owing to the ambiguous compromise between the accuracy and the computing cost. In a word, the studies in this domain are quite important and vibrant.

To change a phenomenon to a model, we must understand its process. Solvation is defined as the process of moving a molecule from the gas phase (or vacuum) to a condensed phase (figure 1.1), which builds a stabilizing interaction with the solute (or solute moiety like protein residues, interfaces, etc.) [34]. Such interactions are mostly classical, involving electrostatic forces and van der Waals forces, with also chemically more specific effects such as hydrogen bond formation, and quantic effects for some small solvents whose vibrational or rotational energy states is at the same magnitude as $k_B T$, etc. [18].

As not all kinds of interactions is important in applications, according to the usage, different models and methods are developed.

In the most of the 20th century [9], the study of solvation effects has been dominated by continuum (implicit) models, which is simply depending on the dielectric constants and not costly on computation resource. They provide an accurate way to treat the strong, long-range electrostatic interactions which dominate many solvation phenomena, but lack of detail informations in the first solvation shell. The later, which mainly includes the cavity formation energy and solute-solvent van der Waals interactions, is often rudely treat by introducing an artificial form of cavity, that links to the form of solute. And the methods for electrostatic interactions involves like generalized Born model, or through better estimates via Poisson-Boltzmann calculations. They are widely integrated within QM simulations of the solvent, by add extra solvation terms onto the Fock or Kohn-Sham operator [17, 26, 46]. However, the improper treatment of the first-shell, where the microscopic interactions are primarily located, often introduce sometimes huge error

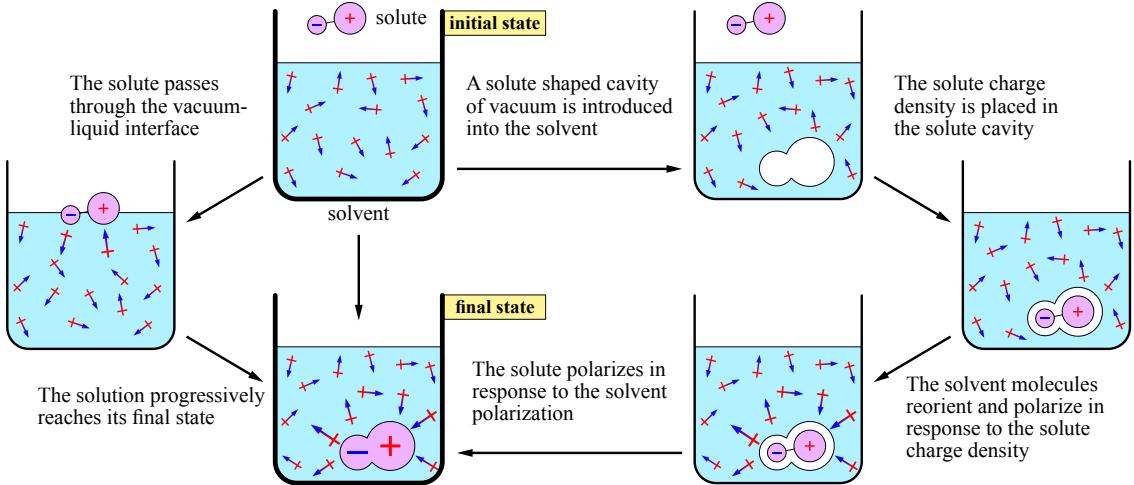


Figure 1.1: The solvation process. To a thermodynamic system, whose properties only depends on the initial and final states, it can go through different paths. The physical process of solvation (left path) takes the solute from vacuum into bulk solvent, progressively passing through the vacuum-liquid interface. Theoretically, the solvation energy is defined as the energy consumed in such a progress. In theoretical studies, the process can be decomposed to some artificial unphysical process (right path), involving the growth of an uncharged solute-sized cavity within the bulk solvent, the transfer of the solute charge distribution from vacuum into the cavity, and the interaction between the solute and solvent, etc.

in free energy evaluation, especially for polar solvents (like water), despite the accuracy that the QM calculation alone can achieve. Therefore, classical molecular simulations, which describing the individual solvent molecules (explicit), particularly the molecular dynamics (MD) and Monte Carlo method (MC), became the alternative solution during the last few decades. They generate trajectories and configurations, then estimate free energy changes by statistical mechanic technics, such as free energy perturbation (FEP) theory or thermodynamic integration (TI) [27]. These calculation is very demanding on computing cost, due to the requirement of many (hundreds or thousands of) solvent molecules to form a realistic model.

Recently, a third domain of theory to describe solvent, based on the statistical mechanics of fluid, is growing rapidly. It is generally called liquid theory, involves mainly the integral equation theory (IET), and the classical density functional theory for liquids. These approaches are cope to give the molecular nature of the first-shell, but without calculate all the instantaneous micro-states with respect to time, which can be integrated over positions and momentums theoretically. Therefore, they are of magnitudes faster than those simulations by micro-states.

The integral equation theory (IET) is about solving the Ornstein-Zernike (OZ) equation with a specific closure equation [18, 21]. It was firstly limited to so called “simple liquid” - a system of spherical particles. A part, Chandler and Andersen in 1971 [6] developed the reference interaction site model (RISM), which discretizes the distribution and correlation functions into a site-site set of functions, and solve the OZ equation in matrix [23]. Another part, Blum [3, 4], Fries and Patey [14] extend the OZ equation to molecular case, where the distribution and correlation functions depend on both position and orientation. In their theory, the orientation part of OZ equation is simplified by expending the distribution and correlation functions on Wigner generalized spherical harmonics.

The classical density functional theory approach deal with inhomogeneous liquids, which uses the same variation principle and minimization strategy [12, 20, 35] as elec-

tronic density functional theory DFT that treats electric interactions and has a great success in computational chemistry. It gives the Helmholtz free energy and the equilibrium solvent density, by minimizing the free energy functional of the solvent density in the presence of a given external potential. Borgis and collaborators [too many ref] have recently generalized it into molecular case, named molecular density functional theory (MDFT), where the solvent density depends on both position and orientation, $\rho(\mathbf{r}, \Omega)$. The main theoretical difficulty lies in the definition of well-founded and reliable functionals of the excess free energy $\mathcal{F}_{\text{exc}}[\rho]$, according to the geometric complexity of the solvent molecule. Some recent researches have shown that it is cope with linear solvents like acetonitrile, but still have little non-satisfaction with the most complex solvent, i. e. water. MDFT can be proved to be mathematically equivalent to the two-component molecular IET.

The majority of work of all these theories have been focused on water, since it is one of the most difficult systems to model due to its molecular geometry, ineligible multi-body interaction, quantum effect, hydrogen bond, etc. The importance of including instantaneous polarization in potential functions is also an issue [10, 45]. However, since polarizable force fields are not yet in common use, the simulations by micro-states and the liquid theory which feed on force field also have their own limit, compared to the continuum model which can be polarizable. The advantages and disadvantages of each branch of theory are listed in table 1.1.

THEORY	SPEED	LONG-RANGE	FIRST-SHELL	POLARIZABLE SOLVENT
Continuum model	fast	yes	no	fully
Simulation by time	costly	yes	yes	partially, very costly
Liquid theory	fast	yes	yes	partially

Table 1.1: Theories of solvation simulation

This thesis consists in the development of the MDFT, focusing on the generalization and algorithmic acceleration of the excess free energy functional \mathcal{F}_{exc} evaluation under homogenous reference fluid (HRF) approximation, which will be discussed in detail in later chapters.

1.2 SCOPE OF THIS THESIS

Chapter I reviews a selection of models and methods to the solvent effect. It includes the mainly used continuum model, the basic of liquid theory, as well as its two frontier research domains, IET and MDFT. The code structure of MDFT, which all the development in this thesis is based on, is also presented. There is also a brief introduction to MD and MC, as well as the generation of direct correlation function (DCF) used in this thesis by such methods.

Chapter II presents all the theory developed and newly used in this thesis. In this thesis, two algorithms of excess energy functional evaluation are proposed, one is extension of the previous algorithm, other is a new algorithm, that combines the molecular OZ equation treatment of angular part with MDFT. The output solvation properties is mainly the two: free energy, and solvent structure.

Chapter III takes note of all the implementation result, that divided into two aspects, the “accuracy”, which involves comparisons between algorithms, and with IET and MD

results; and the “efficiency”, which evaluate the computing cost of the code, both in sequential and parallelized version.

Chapter VI gives some application to ions and molecules.

Chapter I

STATE OF THE ART: SOLVATION, MODELS AND METHODS

the computing science gives a convenient way to find this properties.

In this part, several important models and methods related to this issue are briefly reviewed.

As a huge part of this thesis involves computing science, principle and technics about performance are also presented.

MODEL OF SOLUTION SYSTEM

Computing models of solvent are broadly divided into two types: those treating the solvent as a continuous medium (implicit models) and those describing the individual solvent molecules (explicit models). In the continuum model, the solvent is characterized only by the dielectric constant ε and contains an artificial shaped cavity. The explicit models can have more specific scales. Within the scope of classical mechanics, the most expensive one can be the model explicit flexible and polarizable. While in computational chemistry, less precise models often have wider usage (for example, the proteins are treated in the unity of residues). As the theory of liquid was firstly established for spherical atom-like solvent particles, the model adopted by the theory is a rigid molecule, who only depends on its position and orientation, i. e. there is no relative movement within the solvent particles. And this approximation is proved to be reasonable [18]. In this section, we'll give a brief introduction of implicit model, which helps to understand some later discussion in solvation free energy corrections, then we'll focus on the rigid solvent model. The flexible and polarizable models will also be briefly mentioned, in the purpose to understand the limit of the rigid model.

2.1 CONTINUUM SOLVATION MODELS

Continuum models, which is popular in QM calculations, consider the solvent as a uniform polarizable medium with dielectric constant ε , with the solute M placed in the cavity within this medium [26] (figure 2.1)

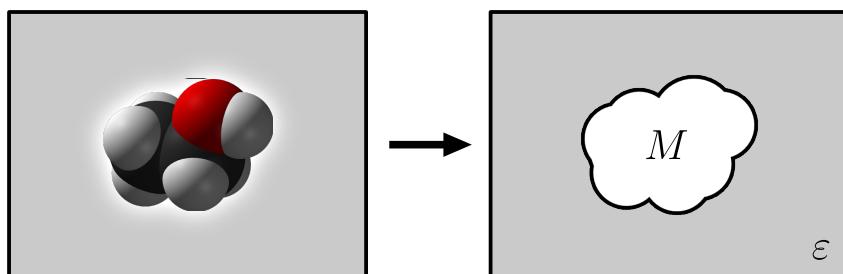


Figure 2.1: Continuum model

The solvation Gibbs free energy according to this model is

$$\Delta G_{\text{solvation}} = \Delta G_{\text{cavity}} + \Delta G_{\text{dispersion}} + \Delta G_{\text{elec}} \quad (2.1)$$

Where $\Delta G_{\text{cavity}} > 0$ is the energy costed to create a hole in the medium, and $\Delta G_{\text{dispersion}}$ is the dispersion interactions, which is roughly the van der Waals energy $\Delta G_{\text{vdW}} < 0$ between the solvent and solute. In principle, there may also be a repulsive component, and dispersion term is sometimes denoted dispersion / repulsion. $\Delta G_{\text{elec}} < 0$ is the contribution of electrostatic interaction, introduced by electric charge distribution of M who polarizes the medium, and the action back of the medium on the molecule (reaction field).

The first two terms in eq. (2.1) is linked to the configuration of first-solvation shell (cavity). The definition of cavity varies from the simplest sphere or ellipsoid to the surface

of the ensemble atomic surfaces defined by the van der Waals radius in the solute. It's somehow reasonable to consider the cavity area proportional to the number of solvent molecules in the first solvation shell [Cramer_1999_implicit_model]. This number can be calculated as the area passes through the middle region of first shell solvent. This area, named as the solvent-accessible surface area (SASA) [22, 31], is calculated by adding the radius of the probe solvent ball on the van der Waals surface area (figure 2.2).

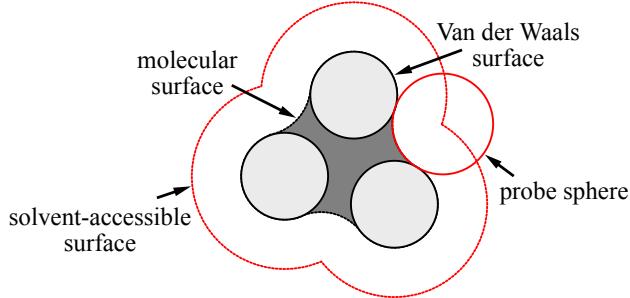


Figure 2.2: Solvent-accessible surface area

The energy required to create such a cavity and the stabilization due to van der Waals interactions between the solute and solvent, assumed to be proportional to the surface area of the cavity, is expressed as

$$\Delta G_{\text{cavity}} + \Delta G_{\text{dispersion}} = \gamma S_{\text{SASA}} + \beta \quad (2.2)$$

or parameterized by having a constant ξ specific for each atom type, with the ξ parameters being determined by fitting to experimental solvation data

$$\Delta G_{\text{cavity}} + \Delta G_{\text{dispersion}} = \sum_i^{\text{atoms}} \xi_i S_i \quad (2.3)$$

The dielectric constant ε is the only parameter characterizing the solvent, it is normally a constant value, but can for some purpose depend on the distance from the solute M . The models and methods to calculate the electrostatic contribution ΔG_{elec} have varied a lot according to its usage, here lists the most common ones.

2.1.1 Poisson–Boltzmann methods

The Poisson-Boltzmann equation (PBE) permits to calculate electrostatic potential V_{elec} for the continuum model, such that electrostatic component of the free energy

$$\Delta G_{\text{elec}} = \frac{1}{2} \int d\mathbf{r} \rho_q(\mathbf{r}) V_{\text{elec}}(\mathbf{r}) \quad (2.4)$$

$$\Delta G_{\text{elec}} = \frac{1}{2} \sum_i Q(\mathbf{r}_i) V_{\text{elec}}(\mathbf{r}_i) \quad (2.5)$$

where ρ_q is the charge distribution of the continuous medium, or in a discrete partial charge description Q .

The Maxwell-Gauss equation gives

$$\nabla \cdot D(\mathbf{r}) = \rho_q(\mathbf{r}) \quad (2.6)$$

where $D(\mathbf{r}) = \varepsilon_0 E(\mathbf{r}) + P(\mathbf{r})$ is the electric displacement field, $P(\mathbf{r})$ is the system polarization, $E(\mathbf{r})$ the electric field, ε_0 the vacuum permittivity.

$$E(\mathbf{r}) = -\vec{\nabla} V_q(\mathbf{r}) \quad (2.7)$$

where $V_q(\mathbf{r})$ is ... of unity ...

therefore

$$\nabla^2 V_q(\mathbf{r}) = -\frac{\rho_q(\mathbf{r})}{\epsilon_0} \quad (2.8)$$

The Poisson equation is a second-order differential equation, describing the connection between the electrostatic potential V_{elec} , the charge distribution ρ and the dielectric constant ϵ

$$\nabla \cdot (\epsilon(\mathbf{r}) \nabla V_{\text{elec}}(\mathbf{r})) = -4\pi\rho(\mathbf{r}) \quad (2.9)$$

This equation cannot be solved analytically for complex geometries (such as a electrostatic interactions in a dielectric medium. cav pol vdw protein)

Therefore, it is done numerically using numerical methods, such as finite differences

Note that the dielectric “constant” may depend on the position. When it is independent of the position (i.e. truly a constant), eq. (14.52) becomes eq. (14.53).

$$\nabla^2 \phi(\mathbf{r}) = -\frac{4\pi}{\epsilon} \rho(\mathbf{r}) \quad (2.10)$$

If the charge distribution is a point charge, the solution of eq. (14.53) reduces to the Coulomb interaction.

The Poisson equation can be modified by taking into account a (thermal) Boltzmann distribution of ions in the solvent. The negative ions will accumulate where the potential is positive, and vice versa, subject to a thermal fluctuation. The charge densities from a collection of ions with charges q and $-q$ and concentration c are given by eq. (14.54).

$$\begin{aligned} \rho_+ &= qce^{-q\phi/kT} \\ \rho_- &= -qce^{-q\phi/kT} \end{aligned} \quad (2.11)$$

Addition of these contributions to eq. (14.52) leads to the Poisson-Boltzmann Equation (PBE).

$$\begin{aligned} \nabla \cdot (\epsilon(\mathbf{r}) \nabla \phi(\mathbf{r})) - \kappa^2 \left(\frac{kT}{q} \right) \sinh \left(\frac{q\phi(\mathbf{r})}{kT} \right) &= -4\pi\rho(\mathbf{r}) \\ \kappa^2 &= \frac{8\pi q^2 I}{kT} \end{aligned} \quad (2.12)$$

Here I is the ion strength of the solution, and the κ^2 factor is inversely related to the Debye-Hückel length, measuring how far the electrostatic effects extend into the solution. The $\sinh(q\phi(\mathbf{r})/kT)$ term only applies for the region corresponding to the solvent, i.e. for \mathbf{r} outside the cavity. Since $q\phi(\mathbf{r})/kT$ is dimensionless, the PBE is often written in terms of a reduced potential u instead.

$$\nabla \cdot (\epsilon(\mathbf{r}) \nabla \phi(\mathbf{r})) - \kappa^2 \sinh(u(\mathbf{r})) = -4\pi\rho(\mathbf{r})$$

The Poisson Eq. Ž43. can be solved numerically by mapping the system onto a discrete grid and using a finite-difference relaxation algorithm 72,73 . Alternatively, a surface boundary element method can be used 74 76 . The numerical problem is made easier by making the approximation that the surrounding solvent has an infinite dielectric constant corresponding to that of a conductor 77,78 . A different approach to incorporate the influence of a polar solvent has been proposed and developed by Warshel and co-workers 79 , in which the solvent is modeled by a

All of these equations ((14.52)–(14.57)) are differential equations that must be solved numerically, typically by a grid representation, and the results give information about the electrostatic potential at any point in space, which is also used by later theory. It can be mapped onto the surface of the solute where it may suggest regions for interaction with other polar molecules. It can also be used for generating the reaction field, defined as the difference between the potential in the presence of a solvent ($\epsilon = 78$) and in vacuum ($\epsilon = 1$), i.e. $\text{freac} = \text{fsolv} - \text{fvac}$. Multiplication of the reaction field with the solute charges in either a continuous (ρ) or partial charge (Q) description gives the

2.1.2 Born/Onsager/Kirkwood models

For certain special cases, however, the Poisson equation (14.52) can be solved analytical, and this forms the basis for many approximate models for estimating the electronic component in eq. (14.49).

The simplest reaction field model is a spherical cavity, where only the lowest order electric moment of the molecule is taken into account. For a net charge q in a cavity of radius a , the difference in energy between a vacuum and a medium with a dielectric constant of ϵ is given by the Born model.

$$\Delta G_{\text{elec}}(q) = - \left(1 - \frac{1}{\epsilon}\right) \frac{q^2}{2a} \quad (2.13)$$

It can be noted that the Born model predicts equal solvation energies for positive and negative ions of the same size, which is not the observed behavior in solvents such as water. Furthermore, the reciprocal dependence on the dielectric constant means that the calculated solvent effect is sensitive to the variation of ϵ in the low dielectric limit but virtually unaffected by large differences in the high dielectric limit. Changing ϵ from 1 to 2 gives a factor of 1/2 in eq. (14.59) but there is virtually no difference between a solvent with a dielectric constant of 30 (e.g. acetonitrile) and one with a dielectric constant of 78 (e.g. water), although in actual experiments there may be a significant difference.

The interaction between point charges is given by the Coulomb potential, with ϵ being a dielectric constant.

$$E_{\text{el}}(R^{AB}) = \frac{Q^A Q^B}{\epsilon R^{AB}} \quad (2.14)$$

Using partial atomic charges in eq. (14.59) is often called the generalized Born model, which has been used especially in connection with force field methods in the Generalized Born/Surface Area (GB/SA) model. In this case, the Coulomb interaction between the partial charges (eq. (2.20)) is combined with the Born formula by means of a function f_{ij} depending on the internuclear distance and Born radii for each of the two atoms, a_i and a_j .

$$\Delta G_{\text{elec}}(Q_i, Q_j) = - \left(1 - \frac{1}{\epsilon}\right) \frac{Q_i Q_j}{f_{ij}} \quad (2.15)$$

$$f_{ij} = \sqrt{r_{ij}^2 - a_{ij}^2 e^{-D}} \quad (2.16)$$

$$a_{ij}^2 = a_i a_j, D = \frac{r_{ij}^2}{4a_{ij}^2}$$

The effective Born radius for a given atom depends on the nature and position of all the atoms.

The dipole in a spherical cavity is known as the Onsager model, which for a dipole moment of m leads to an energy stabilization given by eq. (14.61). (14.61) The Kirkwood model⁶⁸ refers to a general multipole expansion in a spherical cavity, while the Kirkwood–Westheimer model arises for an ellipsoidal cavity.

The Born/Onsager/Kirkwood models are widely used in for example Marcus theory for electron-transfer reactions, solvatochromism, electronic structure in solution, bio-molecules i . e. protein as well as nucleic acids in water. [23]

Generalized Born. A pair-wise approximation to the PB theory

The Born formula generalized to a system of many atoms of arbitrary shape is called the generalized Born model. The generalized Born equation captures the physics of the Poisson-Boltzmann equation, while improving the speed of calculations. • For atoms in a molecule, they also interact with the solvent, but part of the solvent has no been replaced by the other atoms of the molecule. The basic idea is to assign, to each atom, an effective radius such that the solvation free energy can be calculated using the Born formula. • So, the most important thing when using any GB-method is to accurately calculate effective Born radii

Accurate calculations of the Boir= $n r_1 \text{radii}$ can be obtained from the Poisson- Boltzmann equation, since

Poisson-Boltzmann solvers (accurate but numerical and slow). • Generalized Born models (faster, can be analytical).

The idea of a distributed multipole expansion to represent the charge distribution is also employed by the so-called generalized Born (GB) approach to continuum solvation. In this instance, however, only monopoles (i.e., atomic partial charges) are employed and instead of solving the Poisson equation with this charge distribution, one uses the generalized Born approximation,^{41,44,88,96,211,214-226} whichs when used with the dielectric descreening algorithm of [Still et al. 1990] has been demonstrated to give results very close to those obtained from solution of the Poisson equation²²⁷⁻²³² or from explicit molecular simulations.²³³

Density functional method based on the minimisation of polarisation density have been introduced too. [32]

2.1.3 Other models adopt widely in QM calculation

The critical physical concept for treating solute polarization in solution is the reaction field. The reaction field is the electric field exerted on the solute by the solvent that has polarized.

The critical physical concept for treating solute polarization in solution is the reaction field. The reaction field is the electric field exerted on the solute by the solvent that it has polarized.Including this in the solute Hamiltonian predicts a new (“distorted”) solute electronic structure, which further alters the polarization of the solvent. Iterating to self-consistency is called the self-consistent reaction field (SCRF) method. [Cramer_1999_implicit_model]

PCM. An alternative to the use of finite differences or finite elements to discretize the differential opera- tor is to use boundary element methods. The most popular of these is the polarized continuum model (PCM), developed primarily by the Pisa group of Tomasi and co-workers,^{3,142-145} which casts the quan- tum mechanical SCRF equations into a boundary element problem with apparent surface charges (ASCs) on the solute cavity surface. There are cur- rently three different approaches for carrying out PCM calculations. The original method, called di- electric PCM^{142,143} (D-PCM), an alternative model in which the surrounding medium is modeled as a conductor instead of a dielectric¹⁴⁶ (C-

PCM, cf. the COSMO model below), and an implementation whereby the PCM equations are recast in an integral equation formalism^{147,148} (IEF-PCM). The latter method has lent itself to the calculation of various molecular gradient and response properties, as detailed in section 4.2.

The self-consistent reaction field (SCRF) models [17, 26] are widely used by theoretical chemistry applications, owing to the popularity of Gaussian toolkit.

The Polarizable Continuum Model (PCM) using the integral equation formalism variant (IEFPCM)

Other available models are IPCM, which uses a static isodensity surface for the cavity the Self-Consistent Isodensity PCM (SCIPCM) model [Foresman96], and

the Onsager model [Kirkwood34, Onsager36, Wong91, Wong91a, Wong92, Wong92a], which places the solute in a spherical cavity within the solvent reaction field.

the term SCRF is quite generic and it does not by itself indicate a specific model.

PCM

CPCM Performs a PCM calculation using the CPCM polarizable conductor calculation model [Barone98, Cossi03].

Dipole Performs an Onsager model reaction field calculation.

IPCM Performs an IPCM model reaction field calculation. Isodensity is a synonym for IPCM.

SCIPCM Performs an SCIPCM model reaction field calculation, i.e. the SCRF calculation uses a cavity determined self-consistently from an isodensity surface.

2.2 MODEL POTENTIAL OF RIGID MOLECULE

First we define what we mean by classical fluids. These are fluids for which the thermal de Broglie wavelength...

Here presents the model frequently used in theory of liquid [18, 21].

This case is important only for little solvent of atom size, where quantum effect cannot be ignored.

As there's no chemical interaction of solvent

For water, most of publications have already shown... at the approximation level in this thesis, we have good reason to ignore quantum effect.

It should be noticed that quantum effect appears at low temperature and for atom-size molecules. Here we don't discuss...

$u(\mathbf{r}, \Omega_1, \Omega_2)$ pair potential of solvent model, depends on intermolecular separation and orientations. The polyatomic solvent model used in this thesis is based on 3 approximations:

Rigid molecule assumption, Classical treatment of translational and rotational motions, and pair

However, for important solvent such as H₂O and NH₃, the rotational potential energy effects become none-negligible at room temperature. P11

quantum corrections

for MD and IEM, we also use these approximation,

It is quite realistic for small solvent molecules, such as ... as their vibrational states is too huge compared to kT and ...

This assumption implies that the solvent molecule stays in its ground vibrational state, which is quite realistic for some small solvents that have large separation of vibrational states compared to $k_B T$. For water, the ratio of T/θ_v at 298K, where $\theta_v = h\nu/k$ is the characteristic vibrational temperature, is 2290K [18].

rotational temperature 40.1K

That diverges the huge amount of rigid water models [ref], and the non-rigidity [ref] and quantum effect [ref] are also well studied.

vibrational motions energy to great

margin

ignore quantum effets

This assumption fits fir many small molecule solvents, such as H₂O, N₂, CO, CO₂,etc. In these case, the characteristic vibrational temperature θ_v is quite large compares to the temperature T , is to say that the molecules will stay in their ground vibrational state [18]. And the internal rotational is too small can be averaged

2.2.1 Interaction of spherical particle

The simplest model of a fluid is the hard sphere model, with the pair potential

$$u(r) = \begin{cases} \infty & r < d \\ 0 & r > d \end{cases} \quad (2.17)$$

where d is the hard-sphere diameter. This model can represent some physical systems, such as ... However, the absence of attractive force

More realistic neutral particle models, like Lenard-Jones model, have a potential energy curve that has the same shape as the real interaction of rare gas, as shown in figure .

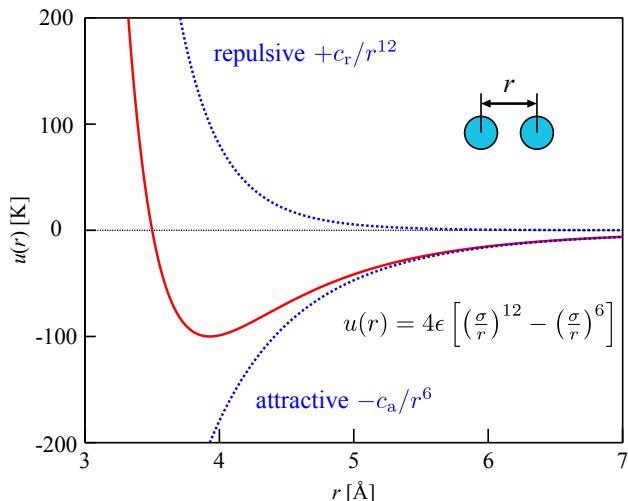


Figure 2.3: LJ pair potential A Plot of Potential Energy versus Internuclear Distance for the Interaction between Two Gaseous Hydrogen Atoms. At long distances, both attractive and repulsive interactions are small. As the distance between the atoms decreases, the attractive electron–proton interactions dominate, and the energy of the system decreases. At the observed bond distance, the repulsive electron–electron and proton–proton interactions just balance the attractive interactions, preventing a further decrease in the internuclear distance. At very short internuclear distances, the repulsive interactions dominate, making the system less stable than the isolated atoms.

Lennard-Jones (LJ) interaction

$$u_{LJ}(r) = 4\epsilon \left[\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right] \quad (2.18)$$

r is the distance from central to central. all terms in the multipole series represent attractive contributions to the potential. the leading term, varying as r^{-6} , describes the dipole-

dipole interaction. Higher-order terms represent dipole-quadrupole (r^{-8}), quadrupole-quadrupole (r^{-10}) interactions, and so on, but these are negligible compares to r^{-6} . The short-range interaction is difficult to calculate , and it's defined as r^{12} in LJ model. The collision diameter σ (of unity [\AA]) is the separation of particles where $u(r) = 0$, and ϵ is the well depth of the potential (of unity kJ/mol), where the minimum occurs at $r_{\min} = 2^{1/6}\sigma$ and $u(r_{\min}) = -\epsilon$, ... These two parameters can be measured by experiment.

electrostatic interaction between charged particles coulomb

three-body interaction

polar liquids where dipole-dipole interaction is superposed on the spherically symmetric potential

$$u(1, 2) = u_0(r) - \boldsymbol{\mu}_1 \cdot \mathbf{T}(\mathbf{r}) \cdot \boldsymbol{\mu}_2 \quad (2.19)$$

where \mathbf{r} is the vector separation of the molecular centers, $u_0(r)$ is the spherically symmetric term as discussed above, $\boldsymbol{\mu}_i$ is the dipole moment vector of particle i and $\mathbf{T}(\mathbf{r})$ is the dipole-dipole interaction tensor:

$$\mathbf{T}(\mathbf{r}) = 3\mathbf{rr}/r^5 - \mathbf{I}/r^3 \quad (2.20)$$

where \mathbf{I} is the unit tensor.

models in which the molecule is represented by a set of discrete interaction sites. the total potential energy is a sum of spherical interaction potentials. let $\mathbf{r}_{i\alpha}$ the coordinates of site α in molecule i . the total intermolecular potential energy is

$$u(1, 2) = \frac{1}{2} \sum_{\alpha} \sum_{\beta} u_{\alpha\beta}(|\mathbf{r}_{2\beta} - \mathbf{r}_{1\alpha}|) \quad (2.21)$$

the full treatment of molecular fluid is below.

2.2.2 Rigid molecule model

The rigid molecular approximation assumes that the intermolecular potential $\mathcal{U}(\mathbf{r}^N, \Omega^N)$ depends only on the positions of the N molecular centers $\mathbf{r}^N \equiv \mathbf{r}_1 \mathbf{r}_2 \dots \mathbf{r}_N$ and on their orientation Ω^N , where $\Omega \equiv (\Theta, \Phi, \Psi)$ represents the Euler angles (figure 2.4).

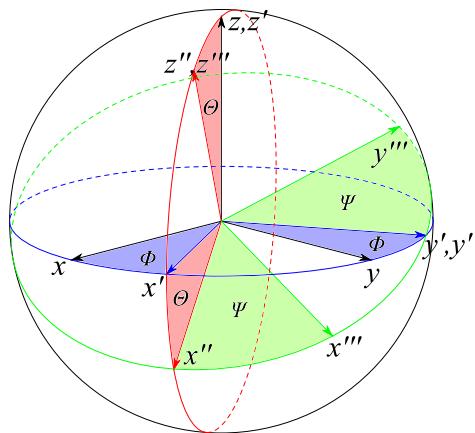


Figure 2.4: Euler angles. The result basis vectors of the new orientation are obtained by 3 sequential operations: (1) A rotation ϕ ($0 < \phi < 2\pi$) about the z -axis, bringing the frame of axes from the initial position S into the position ... (2) A rotation θ ($0 < \theta < \pi$) about the y -axis of the frame S' (3) A rotation ψ ($0 < \psi < 2\pi$) about the z -axis of the frame...

the choice of centre is often the LJ centre ... ?

Pairwise additivity of intermolecular forces

The multi-body potential is often not ignorable, can be taken into account by an effective pair potential.

The effective pair potential can be measured by experiments [18] or being semi-empirical with statistical mechanical calculations.

We assume

According to the rigid particle approximation, the pair potential

$$\mathcal{U}(\mathbf{r}^N, \Omega^N) = \frac{1}{2} \sum_{i \neq j} u(\mathbf{r}_{ij}, \Omega_i, \Omega_j) = \sum_{i < j} u(\mathbf{r}_{ij}, \Omega_i, \Omega_j)$$

only depends on the intermolecular separation \mathbf{r} and on the molecular orientations Ω_1 and Ω_2 , the dependence on orientations gives properties such as...P5

this equation is quasi-exact for low density gas, the three and more body term decrease rapidly, but it is not exact for dense fluids.

higher order corrections

$$\mathcal{U}(\mathbf{r}^N, \Omega^N) = \sum_{i < j} u(ij) + \sum_{i < j < k} u(ijk) + \sum_{i < j < k < l} u(ijkl) + \dots$$

According to the rotational invariance, u can also be reduced to $u(r, \omega_1, \omega_2)$, when \mathbf{r} is referred as polar axis.

three-body omission can cause surface tension problem,
other forces such as magnetic, multipolar, dispersion and induction intermolecular forces are usually negligible compares to the $u_{\mu\mu}$.

Coulomb point charge interaction

$$u(\mathbf{r}, \Omega_1, \Omega_2) = \sum_{\alpha\beta} \frac{q_\alpha q_\beta}{r_{\alpha\beta}} \quad (2.22)$$

Lots of model calculated with LJ and coulomb interaction

site-site model

Three-body interactions

2.2.3 SPC/E water and other water models

There is a great work done by Martin Chaplin [7], which summarized all frequent water models. As water can't be perfectly described by the pair potential due to multi-body effects, quantum effects, hydrogen bond, etc., there develops various models to fit certain properties. The models contain several sites, which can placed else than the center of atom 2.5. The more site number the model has, the more precise it can be.

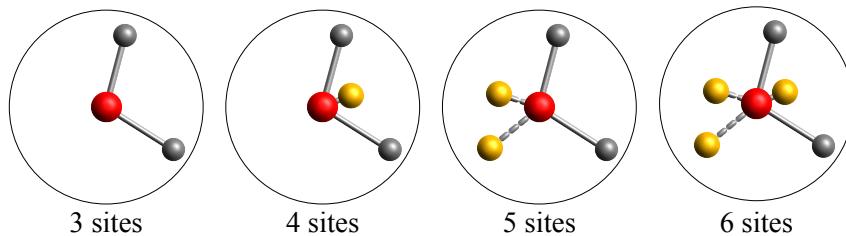


Figure 2.5: Water models

In this thesis, we use the extended simple point charge model (SPC/E) of water [2] as solvent. It is a 3-site water model, The electrostatic interaction is modeled using Coulomb's law and the dispersion and repulsion forces using the Lennard-Jones potential. Any similar model possessing ... is compatible with this theory, as acetonitrile used in [ref].

SPC							
SPC/E							
experiment							

Table 2.1: Parameters for SPC and SPC/E water

The SPC/E model adds an average polarization correction to the potential energy function:

$$E_{\text{pol}} = \frac{1}{2} \sum_i \frac{(\mu - \mu^0)^2}{\alpha_i}$$

where μ is the dipole of the effectively polarized water molecule (2.35 D for the SPC/E model), μ^0 is the dipole moment of an isolated water molecule (1.85 D from experiment), and α_i is an isotropic polarizability constant, with a value of $1.608 \times 10^{-40} \text{ F m}^2$. Since the charges in the model are constant, this correction just results in adding 1.25 kcal/mol (5.22 kJ/mol) to the total energy. The SPC/E model results in a better density and diffusion constant than the SPC model.

The SPC/E model is successful in ...

Model all atom

numerical density, dielectric constant vary with temperature

The parameters are listed in table 2.1, compared with its relative SPC model.

DC [30]

2.2.4 Model flexible and polarizable

Extra degrees of freedom

interaction site can deal with non-rigidity of water

polarization

the full force field in classical

Many atoms → expensive Long runs required to equilibrate solvent to solute Often solvent and solute are not polarizable. Large fluctuations due to use of small system size

2.3 MODEL OF SOLUTE

The model of solute also give an influence to the energy and structure of solvation. The compromise to have a better model of solvent or solute is discussable, it varies according to the application. For example, we never use a quantum solvent model in the case of an implicit solute, it won't be profitable even the solute is of simple geometry (wall). In the case of molecular solute, we generally require the solute to have a model at the same scale of description or more precise (figure 2.6).

Most of QM calculations in apolar solvents (toluene, etc.) uses implicit SCRF model, or even without solvent correction, and is proved to be well working. The interaction between solute and the solvent is mainly the electrostatic interaction as described above. Therefore

The quantum interaction

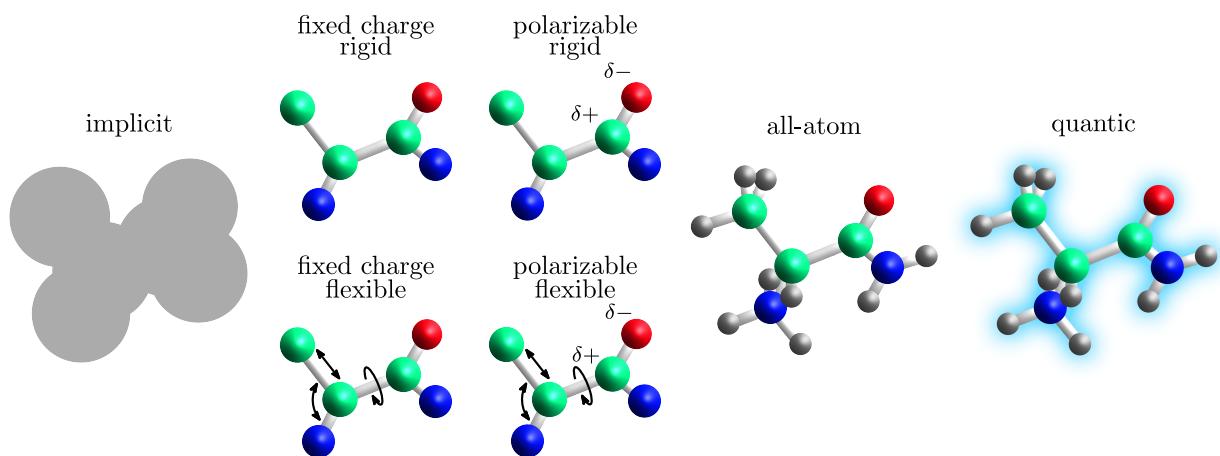


Figure 2.6: Hierarchy of models

In this thesis, we use also rigid model to describe solute to be coherent with IET, which can't treat the solvent and solute in different scale of description.

Polarizable, flexible model, and coupling with QM is described in perspective.

3

STATISTICAL MECHANICS OF CLASSICAL FLUIDS

the link between statistical mechanics and thermodynamics

From pair distribution functions and interaction potential we can get all informations about solvent equilibrium state.

equilibrium statistical mechanics of classical fluids.

one component bulk fluids. thermodynamic state point is specified by ρ and temperature T.

3.1 HAMILTONIAN OF A MICRO-STATE

effective Hamiltonian, i.e. the theorist constructs a model fluid.

In classical mechanics, the instantaneous state (phase point) of an N -particle solvent system is specified by $3N$ coordinates $\mathbf{r}^N \equiv \mathbf{r}_1, \dots, \mathbf{r}_N$ and $3N$ momenta $\mathbf{p}^N \equiv \mathbf{p}_1, \dots, \mathbf{p}_N$. The Hamiltonian os the system is

$$H_N(\mathbf{r}^N, \mathbf{p}^N) = K_N(\mathbf{p}^N) + V_N(\mathbf{r}^N) + V_N^{\text{ext}}(\mathbf{r}^N) \quad (3.1)$$

where

$$K_N(\mathbf{p}^N) = \sum_{i=1}^N \frac{\mathbf{p}_i^2}{2m} \text{ is the kinetic energy}$$

$$V_N(\mathbf{r}^N) = \sum_{i < j}^N u(|\mathbf{r}_i - \mathbf{r}_j|) + 3 \text{ body} + \dots \text{ is the interatomic potential energy } \mathcal{U}(\mathbf{r}^N), \text{ One then has a pair}$$

$$V_N^{\text{ext}}(\mathbf{r}^N) = \sum_{i=1}^N V_{\text{ext}}(\mathbf{r}_i) \text{ is the potential energy arising from the interaction of the particles with the exterr}$$

The distribution of phase points of systems of **the ensemble** is described by a phase space probability density $f^{[N]}(\mathbf{r}^N, \mathbf{p}^N; t)$, such that

$$\int f^{[N]}(\mathbf{r}^N, \mathbf{p}^N; t) d\mathbf{r}^N d\mathbf{p}^N = 1 \quad (3.2)$$

for all t .

subset of particles of size n , reduced phase space distribution function

$$f^{[n]}(\mathbf{r}^n, \mathbf{p}^n; t) = \frac{N!}{(N-n)!} \int f^{[N]}(\mathbf{r}^N, \mathbf{p}^N; t) d\mathbf{r}^{N-n} d\mathbf{p}^{N-n} = 1 \quad (3.3)$$

where $\mathbf{r}^n \equiv \mathbf{r}_1, \dots, \mathbf{r}_n$ and $\mathbf{r}^{N-n} \equiv \mathbf{r}_{n+1}, \dots, \mathbf{r}_N$. the probability of finding a subset of n particles in the reduced phase space element.... the factor is the number of ways of choosing a subset of size n .

The Liouville theorem shows that the probability density is independent of time.

The Bogoliubov–Born–Green–Kirkwood–Yvon hierarchy express $f^{(n)}$ in terms of $f^{(n+1)}$, approximation closure

3.2 TIME AVERAGES AND ENSEMBLE AVERAGES / PARTITION FUNCTIONS AND THERMODYNAMICS

The classical canonical partition function for a one component fluid is given by,

$$Z_N(\beta, V) = \frac{h^{-dN}}{N!} \int d\mathbf{r}^N d\mathbf{p}^N e^{-\beta H_N} \quad (3.4)$$

d is the dimensionality and V is the volume of the system. We can integrate over the momenta to obtain

$$Z_N(\beta, V) = \Lambda^{-dN} Q_N \quad (3.5)$$

$$Q_N = \frac{1}{N!} \int d\mathbf{r}^N e^{-\beta \mathcal{U}(\mathbf{r}_1, \dots, \mathbf{r}_n)} \quad (3.6)$$

is the configurational partition function. Note that the potential energy $\mathcal{U}(\mathbf{r}_1, \dots, \mathbf{r}_n)$ may still include an external field contribution.

The Helmholtz free energy is simply

$$F_N(\beta, V) = -\beta^{-1} \ln Z_N$$

which leads to entropy

$$S = \left(\frac{\partial F_N}{\partial T} \right)_V$$

pressure

$$p = \left(\frac{\partial F_N}{\partial V} \right)_T$$

for bulk fluid

For an ideal (non-interacting) gas, where $\Phi \rightarrow 0$, in $d = 3$???

$$\beta F_N = \ln(N! \Lambda^{3N} V^{-N}) = N \ln(\Lambda^3 \rho) - N$$

Sterling equation... number density $\rho = N/V$ (a uniform ideal classical gas.)

Th grand function We consider open systems with fixed temperature T and chemical potential μ . The partition function for the grand canonical ensemble is

$$\Xi(\beta???, \mu, T) = \sum_{N=0}^{\infty} \frac{e^{\beta \mu N}}{h^{3N} N!} \iint d\mathbf{r}^N d\mathbf{p}^N e^{-\beta H_N(\mathbf{r}^N, \mathbf{p}^N)} = \sum_{N=0}^{\infty} \frac{1}{N!} \int d\mathbf{r}^N e^{-\beta V_N(\mathbf{r}^N)} \left(\prod_{i=1}^N \frac{e^{\beta u(\mathbf{r}_i)}}{\Lambda^{3N}} \right) \quad (3.7)$$

with $\Lambda = (2\pi \hbar^2/m)^{\frac{1}{2}}$ and $u(\mathbf{r}) = \mu - V_{\text{ext}}(\mathbf{r})$.

The equilibrium density of probability

$$f(\mathbf{r}^N, \mathbf{p}^N; N) = \frac{1}{h^{3N} N! \Xi} e^{-\beta [H_N(\mathbf{r}^N, \mathbf{p}^N) - \mu N]}$$

and the grand potential

$$\Omega = -\beta^{-1} \ln \Xi(\beta???, \mu, T)$$

which for the case of a uniform fluid, reduces to $\Omega = -pV$.

$$\begin{aligned} \Omega[u] &= \beta^{-1} \left\langle \ln \left(h^{3N} N! f(\mathbf{r}^N, \mathbf{p}^N; N) \right) + K_N(\mathbf{p}^N) + V_N(\mathbf{r}^N) - \sum_{i=0}^N u(\mathbf{r}_i) \right\rangle \\ u(\mathbf{r}_i) &= \int \delta(\mathbf{r} - \mathbf{r}_i) u(\mathbf{r}) d\mathbf{r} \end{aligned}$$

$$\rho(\mathbf{r}) = \left\langle \sum_{i=1}^N \delta(\mathbf{r} - \mathbf{r}_i) \right\rangle$$

$$\Omega[u] = \beta^{-1} \left\langle \ln \left(h^{3N} N! f(\mathbf{r}^N, \mathbf{p}^N; N) \right) + K_N(\mathbf{p}^N) + V_N(\mathbf{r}^N) \right\rangle - \int \rho(\mathbf{r}) u(\mathbf{r}) d\mathbf{r}$$

free energy of Helmholtz

$$\Omega[u] = F + \mu \langle N \rangle = F + \mu \int \rho(\mathbf{r})$$

3.3 DISTRIBUTION FUNCTIONS

the direct correlation function hierarchy:

$$\begin{aligned} c^{(1)}(\mathbf{r}) &= -\frac{\delta(\beta \mathcal{F}_{\text{exc}}[\rho])}{\delta \rho(\mathbf{r})} \\ c^{(2)}(\mathbf{r}_1, \mathbf{r}_2) &= \frac{\delta c^{(1)}(\mathbf{r}_1)}{\delta \rho(\mathbf{r}_2)} = -\frac{\delta^2(\beta \mathcal{F}_{\text{exc}}[\rho])}{\delta \rho(\mathbf{r}_1) \delta \rho(\mathbf{r}_2)} = c^{(2)}(\mathbf{r}_2, \mathbf{r}_1) \\ c^{(n)}(\mathbf{r}_1, \dots, \mathbf{r}_n) &= \frac{\delta c^{(n-1)}(\mathbf{r}_1, \dots, \mathbf{r}_{n-1})}{\delta \rho(\mathbf{r}_n)} \end{aligned}$$

for bulk solvent

$\mathcal{F}_{\text{exc}}[\rho]$ is the excess (over ideal) **Helmholtz free energy** functional arising from the interactions.

4

INTEGRAL EQUATION THEORY

density-density pair correlation function $\langle v(\mathbf{r})v(\mathbf{r}') \rangle \equiv \rho(\mathbf{r}, \mathbf{r}')$, when the fluid is uniform, the quantity can be expressed by a function of only the distance between the two places , such that $\rho(\mathbf{r}, \mathbf{r}') \rightarrow \rho(|\mathbf{r} - \mathbf{r}'|)$ [23]

4.1 ORNSTEIN-ZERNIKE EQUATION

The Ornstein-Zernike equation for simple (atomic) liquid

$$h(\mathbf{r}, \mathbf{r}') = c(\mathbf{r}, \mathbf{r}') + \int_V c(\mathbf{r}, \mathbf{r}'') \rho(\mathbf{r}'') h(\mathbf{r}'', \mathbf{r}') d\mathbf{r}'' \quad (4.1)$$

The potential energy of the liquid system

$$U'_N(\mathbf{r}_1, \dots, \mathbf{r}_N) = \sum_{i=1}^N \varphi(\mathbf{r}_i) + U_N(\mathbf{r}_1, \dots, \mathbf{r}_N) \quad (4.2)$$

where $\varphi(\mathbf{r}_i)$ is the interaction potential of i -th molecule of the system with an external field, and $U_N(\mathbf{r}_1, \dots, \mathbf{r}_N)$ is the intermolecular potential energy. the local activity $z(\mathbf{r}) = z \exp [-\beta \varphi(\mathbf{r})]$, where $z = \exp [\mu / k_B T]$, $\beta = 1 / k_B T$, μ is the chemical potential, and k_B is the Boltzmann constant.

The grand canonical partition function

$$\Xi = \sum_N \frac{1}{N!} \int \dots \int \prod_i^N z(\mathbf{r}_i) \exp (-\beta U_N) d\mathbf{r}_1 \dots d\mathbf{r}_N \quad (4.3)$$

single particle and pair correlation functions of density

$$\rho(\mathbf{r}) = \frac{z(\mathbf{r})}{\Xi} \frac{\delta \Xi}{\delta z(\mathbf{r})} \quad (4.4)$$

$$\rho^{(2)}(\mathbf{r}, \mathbf{r}') = \frac{z(\mathbf{r})z(\mathbf{r}')}{\Xi} \frac{\delta^2 \Xi}{\delta z(\mathbf{r}) \delta z(\mathbf{r}')} \quad (4.5)$$

with the aid of the mathematical theorem concerning the functional derivative

$$\frac{\delta z(\mathbf{r})}{\delta z(\mathbf{r}')} = \delta(\mathbf{r} - \mathbf{r}') \quad (4.6)$$

the relation between

$$\frac{\delta \rho(\mathbf{r})}{\delta \ln z(\mathbf{r}')} = \rho^{(2)}(\mathbf{r}, \mathbf{r}') - \rho(\mathbf{r})\rho(\mathbf{r}') + \delta(\mathbf{r} - \mathbf{r}')\rho(\mathbf{r})$$

we define the direct correlation function

$$\frac{\delta \ln z(\mathbf{r})}{\delta \rho(\mathbf{r}')} = \frac{\delta(\mathbf{r} - \mathbf{r}')}{\rho(\mathbf{r}')} - c(\mathbf{r} - \mathbf{r}')$$

inserting eq... into the chain-rule theorem of functional derivatives

$$\int \frac{\delta \rho(\mathbf{r})}{\delta \ln z(\mathbf{r}'')} \frac{\delta \ln z(\mathbf{r}'')}{\delta \rho(\mathbf{r}')} d\mathbf{r}'' = \delta(\mathbf{r} - \mathbf{r}')$$

one obtains the Ornstein-Zernike equation.

4.2 CLOSURE

The general relation to close the OZ equation

$$g(\mathbf{r}, \mathbf{r}') = \exp [-\beta u(\mathbf{r}, \mathbf{r}') + t(\mathbf{r}, \mathbf{r}') + b(\mathbf{r}, \mathbf{r}')] \quad (4.7)$$

$b(\mathbf{r}, \mathbf{r}')$ is the bridge function
hypernetted-chain (HNC) approximation

$$g(\mathbf{r}, \mathbf{r}') = \exp [-\beta u(\mathbf{r}, \mathbf{r}') + t(\mathbf{r}, \mathbf{r}')] \quad (4.8)$$

suitable ...

Percus-Yevick (PY) approximation

$$g(\mathbf{r}, \mathbf{r}') = \exp [-\beta u(\mathbf{r}, \mathbf{r}')] [1 + t(\mathbf{r}, \mathbf{r}')] \quad (4.9)$$

suitable...

Another type of closure, used for

$$u(r) = \begin{cases} \infty & (r \leq \sigma) \\ w(r) & (r > \sigma) \end{cases}$$

in the mean-spherical approximation (MSA)

$$h(r) = -1, r \leq \sigma$$

$$c(r) = -\beta w(r), r > \sigma$$

4.3 RISM AND 3D-RISM

4.4 MOLECULAR SOLVENT

Fries and Patey, Blum

4.5 BULK PROPERTIES AND BRIDGE FUNCTION

Luc

5

CLASSICAL DENSITY FUNCTIONAL THEORY

Classical density functional theory provides a framework for determining thermodynamic properties and correlation functions of a wide variety of inhomogeneous (model) fluids starting from a microscopic basis, i.e. the Hamiltonian describing interactions between particles. DFT is based on the result that the grand potential of a specified inhomogeneous fluid is a functional of the average one-body density,

$$\rho(\mathbf{r}) = \left\langle \sum_{i=1}^N \delta(\mathbf{r} - \mathbf{r}_i) \right\rangle$$

where \mathbf{r}_i is the position coordinate of particle i .

This article describes all the fundamental theories behind MDFT different algorithms involved in this thesis to evaluate excess free energy functional \mathcal{F}_{exc} under HRF approximation.

5.1 EQUILIBRIUM CLASSICAL DFT

5.2 VARIATION PRINCIPLE

The key idea is that $\mathcal{F}[\rho]$ is a unique functional of $\rho(\mathbf{r})$; its form does not depend on the external potential $V(\mathbf{r})$. !!!

variation principe: par minimization

In molecular density functional theory (MDFT), the grand potential density functional corresponding to an inhomogeneous fluid density $\rho(\mathbf{r}, \Omega)$ is given by

$$\Theta[\rho(\mathbf{r}, \Omega)] = \Theta[\rho_0] + \mathcal{F}[\rho(\mathbf{r}, \Omega)] \quad (5.1)$$

Where $\Theta[\rho_0]$ is the correspondent reference bulk fluid grand potential. And ρ is the fluid density function variable of 3 to 6 dimensions, including 3 coordinations for the position part, and 0 to 3 for the angular part. For instance, in an isotropic fluid:

$$\rho(\mathbf{r}, \Omega) = \begin{cases} n(\mathbf{r}) & \text{if atomic, } \Omega \equiv 1 \\ n(\mathbf{r})/4\pi & \text{if linear, } \Omega \equiv (\theta, \phi) \\ n(\mathbf{r})/8\pi^2 & \text{if non-linear, } \Omega \equiv (\theta, \phi, \psi) \end{cases} \quad (5.2)$$

Here the 4π and $8\pi^2$ is the normalization factor who equals to $\int d\Omega$. The 6D definition of the latest case in eq. 5.2 of $\rho(\mathbf{r}, \Omega)$ is needed for arbitrary solvent, which results the most complex form of the functional.

According to the variation principle (c.f. Evans), the equilibrium density can be found by minimizing the free energy functional $\mathcal{F}[\rho]$ regarding to $\rho(\mathbf{r}, \Omega)$:

$$\frac{\delta \mathcal{F}[\rho]}{\delta \rho(\mathbf{r}, \Omega)}|_{\rho=\rho_0} = 0 \quad (5.3)$$

And this functional is defined as a sum of functional contributions:

$$\mathcal{F}[\rho] = \mathcal{F}_{\text{id}}[\rho] + \mathcal{F}_{\text{ext}}[\rho] + \mathcal{F}_{\text{exc}}[\rho] \quad (5.4)$$

The ideal term $\mathcal{F}_{\text{id}}[\rho]$ is deduced from the particle interaction-free condition:

$$\mathcal{F}_{\text{id}}[\rho] = \beta^{-1} \int d\mathbf{r} d\Omega \left[\ln \left(\frac{\mathbf{a}(\mathbf{r}, \boldsymbol{\Omega})}{\mathbf{a}_0} \right) - \rho(\mathbf{r}, \boldsymbol{\Omega}) + \rho_0 \right] \quad (5.5)$$

Where $\beta = (K_B T)^{-1}$ and ρ_0 is the reference bulk density of pure solvent.

The external potential term calculates the contribution of solute external potential V_{ext} :

$$\mathcal{F}_{\text{ext}}[\rho] = \int d\mathbf{r} d\Omega V_{\text{ext}}(\mathbf{r}, \boldsymbol{\Omega}) \rho(\mathbf{r}, \boldsymbol{\Omega}) \quad (5.6)$$

$$V_{\text{ext}}(\mathbf{r}, \boldsymbol{\Omega}) = \sum_{j \in \text{solvent}} \left\{ q_j V_q(\mathbf{r}_j) + \sum_{i \in \text{solute}} 4\epsilon_{ij} \left[\left(\frac{\sigma_{ij}}{r_{ij}} \right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}} \right)^6 \right] \right\} \quad (5.7)$$

Where V_{ext} is pre-calculated and stored as a 6-dimension double precision (8 bytes per real value) table. It contains the contribution of Lennard-Jones interaction and electrostatic potential.

This two terms $\mathcal{F}_{\text{id}}[\rho]$ and $\mathcal{F}_{\text{ext}}[\rho]$ are physically exact.

The excess term $\mathcal{F}_{\text{exc}}[\rho]$ depend on the exact correlation function, which is a priori unknown:

$$C(\mathbf{r}_1, \mathbf{r}_2, \boldsymbol{\Omega}_1, \boldsymbol{\Omega}_2; \rho) \equiv \frac{\beta \delta^2 \mathcal{F}_{\text{exc}}[\rho]}{\delta \rho(\mathbf{r}_1, \boldsymbol{\Omega}_1) \delta \rho(\mathbf{r}_2, \boldsymbol{\Omega}_2)} \quad (5.8)$$

The ideal term $\mathcal{F}_{\text{id}}[\rho]$ is deduced from the condition that the interaction potential of particles $\Phi = 0$

$$Z_N(\beta, V) = \frac{\lambda^{-3N}}{N!} \int_V d\mathbf{r}^N e^{-\beta \Phi} = \frac{\lambda^{-3N}}{N!} V^N$$

$$F_N(\beta, V) = -\beta^{-1} \log Z_N = \beta^{-1} \log (N! \lambda^{3N} V^{-N}) = \beta^{-1} (N \log \lambda^3 \rho - N)$$

(also available for the in homogenous case.) Since we define

$$\mathcal{F}_{\text{id}}[\rho] = \int d\mathbf{X} f_{\text{id}}[\rho(\mathbf{X})]$$

We have

$$f_{\text{id}}[\rho(\mathbf{X})] = \beta^{-1} \Delta \rho (\log \lambda^3 \Delta \rho - 1)$$

and

$$\mathcal{F}_{\text{id}}[\rho] = \beta^{-1} \int d\mathbf{X}_1 [\rho(\mathbf{X}_1) \log (\frac{\rho(\mathbf{X}_1)}{\rho_0}) - \rho(\mathbf{X}_1) + \rho_0]$$

The external potential term

$$\mathcal{F}_{\text{ext}}[\rho] = \int d\mathbf{X}_1 V_{\text{ext}}(\mathbf{X}_1) \rho(\mathbf{X}_1)$$

Where V_{ext} is given by table.

The exact (in an approximation that C do not depend on ρ) form of the excess term \mathcal{F}_{exc} is

$$\mathcal{F}_{\text{exc}}[\rho] = \beta^{-1} \int d\mathbf{X}_1 d\mathbf{X}_2 \Delta \rho(\mathbf{X}_1) \Delta \rho(\mathbf{X}_2) C(\mathbf{X}_1, \mathbf{X}_2; \rho)$$

where $C(\mathbf{X}_1, \mathbf{X}_2; \rho)$ is the correlation function, a priori unknown, which definition is:

$$C(\mathbf{X}_1, \mathbf{X}_2; \rho) = \frac{\beta \delta^2 \mathcal{F}_{\text{exc}}[\rho]}{\delta \rho(\mathbf{r}_1) \delta \rho(\mathbf{r}_2)}$$

Figure 5.1: Flowchart of code MDFT

structure MDFT

The code MDFT is built for minimizing the free energy functional $\mathcal{F}[\rho]$ by minimizer L-BFGS (ref). It's main structure is shown in fig.

The minimizer requires the functional and its gradient as input, and produce at each iteration a new fluid density $\rho(\mathbf{r}, \Omega)$ nearer to the equilibrium density. The objective for **branch hesper** is to calculate the excess term $\mathcal{F}_{\text{exc}}[\rho]$ of the functional in eq.??, as well as its gradient:

$$\gamma(\mathbf{r}_1, \Omega) = \frac{\delta \mathcal{F}_{\text{exc}}}{\delta \rho} = -\beta^{-1} \int d\mathbf{r}_2 d\Omega_2 \Delta \rho(\mathbf{r}_2, \Omega_2) c^{(2)}(\mathbf{r}_{12}, \Omega_1, \Omega_2) \quad (5.9)$$

5.3 IDEAL FREE ENERGY

5.4 EXTERNAL FREE ENERGY

5.4.1 Electrostatic potential

The direct evaluation of the Coulomb sum for N particles is

$$U_C = \sum_{i < j} a \frac{q_i q_j}{\|\mathbf{r}_i - \mathbf{r}_j\|}, \quad (5.10)$$

where a is a constant that gives U_C the dimension of an energy, and q_i, \mathbf{r}_i are the charge and position of particle i . The computation of equation 5.10 is demanding because it requires a large number of distances $r_{ij} \equiv \|\mathbf{r}_i - \mathbf{r}_j\|$ to be computed. In our case, we have $n_{\text{fft1}} \times n_{\text{fft2}} \times n_{\text{fft3}} \times N_\Omega \times N_\psi \times N_{qv}$, where N_{qv} is the number of point charges of the solvent molecule, to be computed per point charge of the solute. This is typically 10^9 distances for MDFT.

L'énergie électrostatique s'écrit

$$\mathcal{F}_q = \frac{1}{2} \iiint \frac{\rho_c^{\text{soluté}}(\mathbf{r}') \rho_c^{\text{solvant}}(\mathbf{r}, \Omega)}{4\pi\epsilon_0 \|\mathbf{r} - \mathbf{r}'\|} d\mathbf{r} d\mathbf{r}' d\Omega \quad (5.11)$$

$$= \frac{1}{2} \frac{1}{4\pi\epsilon_0} \iiint \frac{\rho_c^{\text{soluté}}(\mathbf{r}')}{\|\mathbf{r} - \mathbf{r}'\|} [\rho^{\text{solvant}} * \sigma](\mathbf{r}, \Omega) d\mathbf{r} d\mathbf{r}' d\Omega \quad (5.12)$$

$$= \frac{1}{2} \frac{1}{4\pi\epsilon_0} \iint [\rho_c^{\text{soluté}} * \|\mathbf{r}\|^{-1}](\mathbf{r}, \Omega) [\rho^{\text{solvant}} * \sigma](\mathbf{r}, \Omega) d\mathbf{r} d\Omega \quad (5.13)$$

$$= \quad (5.14)$$

On définit une densité de charge d'une molécule $\sigma(\mathbf{r}, \Omega)$:

$$\sigma(\mathbf{r}, \Omega) = \sum_{m=1}^{\text{sites du solvant}} q_m \delta(\mathbf{r} - \mathbf{s}_m(\Omega)). \quad (5.15)$$

$\mathbf{s}_m(\Omega)$ désigne la position du $m^{\text{ième}}$ site de la molécule de solvant quand elle a l'orientation Ω .

5.5 EXCESS FREE ENERGY

In the molecular density functional theory (MDFT), the excess free energy functional \mathcal{F}_{exc} can be developed via Taylor expansion

$$\mathcal{F}_{\text{exc}} [\rho] \equiv \mathcal{F}_{\text{exc}} [\rho_0] + \int d\mathbf{X}_1 \frac{\delta \mathcal{F}_{\text{exc}} [\rho]}{\delta \rho(\mathbf{X}_1)} \Delta \rho(\mathbf{X}_1) + \frac{1}{2} \int d\mathbf{X}_1 d\mathbf{X}_2 \frac{\delta^2 \mathcal{F}_{\text{exc}} [\rho]}{\delta \rho(\mathbf{X}_1) \delta \rho(\mathbf{X}_2)} \Delta \rho(\mathbf{X}_1) \Delta \rho(\mathbf{X}_2) + \mathcal{O}(\Delta \rho^3) \quad (5.16)$$

where $\mathbf{X} \equiv (\mathbf{r}, \Omega)$, $\beta^{-1} = k_B T$, $\Delta \rho = \rho - \rho_0$, and $\rho(\mathbf{X})$ the single-particle density of the fluid.

5.5.1 Homogenous Reference Fluid Approximation

According to the variation principle, the first derivative in eq. (5.16) is zero

$$\left. \frac{\delta \mathcal{F}_{\text{exc}} [\rho]}{\delta \rho(\mathbf{X}_1)} \right|_{\rho=\rho_{\text{eq}}} = 0 \quad (5.17)$$

And $\mathcal{F}_{\text{exc}} [\rho_0]$ can be compensated by the grand potential of bulk fluid (to be detailed), we can then focus on only $\mathcal{O}(\Delta \rho^2)$ and higher order terms.

As \mathcal{F}_{exc} is a generating functional of $c^{(n)}(\mathbf{X}^n)$, the direct correlations functions (DCF) [21], for example

$$c^{(2)}(\mathbf{X}_1, \mathbf{X}_2) = -\beta \frac{\delta^2 \mathcal{F}_{\text{exc}} [\rho]}{\delta \rho(\mathbf{X}_1) \delta \rho(\mathbf{X}_2)} \quad (5.18)$$

eq. (5.16) can be rewritten as

$$\begin{aligned} \mathcal{F}_{\text{exc}} [\rho] &= -\frac{\beta^{-1}}{2} \int d\mathbf{X}_1 d\mathbf{X}_2 c^{(2)}(\mathbf{X}_1, \mathbf{X}_2) \Delta \rho(\mathbf{X}_1) \Delta \rho(\mathbf{X}_2) + \mathcal{O}(\Delta \rho^3) \\ &\simeq -\frac{\beta^{-1}}{2} \int d\mathbf{X}_1 d\mathbf{X}_2 c^{(2)}(\mathbf{X}_1, \mathbf{X}_2) \Delta \rho(\mathbf{X}_1) \Delta \rho(\mathbf{X}_2) \end{aligned} \quad (5.19)$$

If we take further approximation, to assume that

$$c^{(2)}(\mathbf{X}_1, \mathbf{X}_2) \simeq c_0^{(2)}(\mathbf{X}_1, \mathbf{X}_2) \quad (5.20)$$

where $c_0^{(2)}(\mathbf{X}_1, \mathbf{X}_2)$ is the DCF of the bulk fluid, it's called the homogenous reference fluid (HRF) approximation.

5.5.2 nn_cs

We define $r = \|\mathbf{r} - \mathbf{r}'\|$, and $\Delta n(\mathbf{r}) = n(\mathbf{r}) - n_0$, and n_0 the density (given in dft.in) of the homogeneous fluid of reference, e.g., 0.0332891 molecule per Å³ for water.

We also define $n(\mathbf{r}) = \int \rho(\mathbf{r}, \Omega) d\Omega$. We have

$$F_{\text{exc}} = -\frac{1}{2} k_B T \iint \Delta n(\mathbf{r}) \Delta n(\mathbf{r}') c(r) dr dr', \quad (5.21)$$

Now, we consider the convolution in the right hand side of the equation, $\gamma \equiv (\Delta n * c)$, that can be computed much efficiently than in $O(N^2)$ by fast Fourier transform in $O(N \log N)$.

Inputs:

- $\rho(\mathbf{r}, \Omega)$

- $c_s(k)$, with $k \equiv \|\mathbf{k}\|$
- functions to Fast Fourier Transform (FFT) and inverse Fast Fourier Transform (FFT^{-1})
- n_0 , the density of the homogeneous fluid of reference
- T the temperature in Kelvin
- k_B the Boltzmann constant.

5.5.3 polarisation

$$\mathcal{F}_{\text{exc}} = -\frac{k_B T}{2} \iiint_{\mathcal{R}^3} \int_{\theta=0}^{\pi} \int_{\phi=0}^{2\pi} \int_{\psi=0}^{2\pi} \Delta\rho(\mathbf{r}_1, \boldsymbol{\Omega}_1) c^{(2)}(\mathbf{r}_1, \mathbf{r}_2, \boldsymbol{\Omega}_1, \boldsymbol{\Omega}_2) \Delta\rho(\mathbf{r}_2, \boldsymbol{\Omega}_2) d\mathbf{r}_1 d\mathbf{r}_2 d\boldsymbol{\Omega}_1 d\boldsymbol{\Omega}_2 \quad (5.22)$$

where $\rho = n / (8\pi^2)$ and we define

$$\mathbf{P}(\mathbf{r}) = \int \mathbf{p}\rho(\mathbf{r}, \boldsymbol{\Omega}) d\boldsymbol{\Omega} \quad (5.23)$$

with $\mathbf{p} = p\boldsymbol{\Omega}$ the dipolar moment of a water molecule.

5.6 OZ EQUATION IN MDFT AND IEM FORMALISM

Ornstein-Zernike (OZ) equation is a fundamental equation to the theory of liquids, which defined $c^{(2)}(\mathbf{X}_1, \mathbf{X}_2)$. Its most common form is as shown [21]:

$$h^{(2)}(\mathbf{X}_1, \mathbf{X}_2) = c^{(2)}(\mathbf{X}_1, \mathbf{X}_2) + \int c^{(2)}(\mathbf{X}_1, \mathbf{X}_3) \rho(\mathbf{X}_3) h^{(2)}(\mathbf{X}_3, \mathbf{X}_2) d\mathbf{X}_3 \quad (5.24)$$

where $h^{(2)}(\mathbf{X}_1, \mathbf{X}_2)$ is the pair correlation function (PCF). When the fluid is homogenous, eq. (5.24) becomes:

$$h^{(2)}(\mathbf{X}_1, \mathbf{X}_2) = c^{(2)}(\mathbf{X}_1, \mathbf{X}_2) + \rho_0 \int c^{(2)}(\mathbf{X}_1, \mathbf{X}_3) h^{(2)}(\mathbf{X}_3, \mathbf{X}_2) d\mathbf{X}_3 \quad (5.25)$$

which gives

$$\gamma(\mathbf{X}_1, \mathbf{X}_2) = h^{(2)}(\mathbf{X}_1, \mathbf{X}_2) - c^{(2)}(\mathbf{X}_1, \mathbf{X}_2) = \rho_0 \int c^{(2)}(\mathbf{X}_1, \mathbf{X}_3) h^{(2)}(\mathbf{X}_3, \mathbf{X}_2) d\mathbf{X}_3 \quad (5.26)$$

If we take \mathbf{X}_2 as origin in the laboratory coordinate system, eq. (5.26) becomes:

$$\gamma(\mathbf{X}_1) = \int c^{(2)}(\mathbf{X}_1, \mathbf{X}_3) \Delta\rho(\mathbf{X}_3) d\mathbf{X}_3 \quad (5.27)$$

which coincides with the gradient of $\mathcal{F}_{\text{exc}}[\rho]$ in eq. (5.19).

It can be further proved that the homogenous reference fluid (HRF) approximation of molecular density functional theory (MDFT) is mathematically equivalent to the hypernetted chain (HNC) approximation to integral equations method (IEM) if the fluid is homogenous. (To be detailed.)

5.7 STATISTICAL MECHANICS

5.8 PRINCIPE OF VARIATION

5.9 IDEAL FREE ENERGY FUNCTIONAL

5.10 EXTERNAL FREE ENERGY FUNCTIONAL

5.11 HOMOGENOUS REFERENCE FLUID APPROXIMATION

In the molecular density functional theory (MDFT), the excess free energy functional \mathcal{F}_{exc} can be developed via Taylor expansion

$$\mathcal{F}_{\text{exc}} [\rho] \equiv \mathcal{F}_{\text{exc}} [\rho_0] + \int d\mathbf{X}_1 \frac{\delta \mathcal{F}_{\text{exc}} [\rho]}{\delta \rho(\mathbf{X}_1)} \Delta\rho(\mathbf{X}_1) + \frac{1}{2} \int d\mathbf{X}_1 d\mathbf{X}_2 \frac{\delta^2 \mathcal{F}_{\text{exc}} [\rho]}{\delta \rho(\mathbf{X}_1) \delta \rho(\mathbf{X}_2)} \Delta\rho(\mathbf{X}_1) \Delta\rho(\mathbf{X}_2) + \mathcal{O}(\Delta\rho^3) \quad (5.28)$$

where $\mathbf{X} \equiv (\mathbf{r}, \Omega)$, $\beta^{-1} = k_B T$, $\Delta\rho = \rho - \rho_0$, and $\rho(\mathbf{X})$ the single-particle density of the fluid.

According to the variation principle, the first derivative in eq. (5.28) is zero

$$\left. \frac{\delta \mathcal{F}_{\text{exc}} [\rho]}{\delta \rho(\mathbf{X}_1)} \right|_{\rho=\rho_{\text{eq}}} = 0 \quad (5.29)$$

And $\mathcal{F}_{\text{exc}} [\rho_0]$ can be compensated by the grand potential of bulk fluid (to be detailed), we can then focus on only $\mathcal{O}(\Delta\rho^2)$ and higher order terms.

As \mathcal{F}_{exc} is a generating functional of $c^{(n)}(\mathbf{X}^n)$, the direct correlations functions (DCF) [21], for example

$$c^{(2)}(\mathbf{X}_1, \mathbf{X}_2) = -\beta \frac{\delta^2 \mathcal{F}_{\text{exc}} [\rho]}{\delta \rho(\mathbf{X}_1) \delta \rho(\mathbf{X}_2)} \quad (5.30)$$

eq. (5.28) can be rewritten as

$$\begin{aligned} \mathcal{F}_{\text{exc}} [\rho] &= -\frac{\beta^{-1}}{2} \int d\mathbf{X}_1 d\mathbf{X}_2 c^{(2)}(\mathbf{X}_1, \mathbf{X}_2) \Delta\rho(\mathbf{X}_1) \Delta\rho(\mathbf{X}_2) + \mathcal{O}(\Delta\rho^3) \\ &\simeq -\frac{\beta^{-1}}{2} \int d\mathbf{X}_1 d\mathbf{X}_2 c^{(2)}(\mathbf{X}_1, \mathbf{X}_2) \Delta\rho(\mathbf{X}_1) \Delta\rho(\mathbf{X}_2) \end{aligned} \quad (5.31)$$

If we take further approximation, to assume that

$$c^{(2)}(\mathbf{X}_1, \mathbf{X}_2) \simeq c_0^{(2)}(\mathbf{X}_1, \mathbf{X}_2) \quad (5.32)$$

where $c_0^{(2)}(\mathbf{X}_1, \mathbf{X}_2)$ is the DCF of the bulk fluid, it's called the homogenous reference fluid (HRF) approximation.

5.12 OZ EQUATION IN MDFT AND IEM FORMALISM

Ornstein-Zernike (OZ) equation is a fundamental equation to the theory of liquids, which defined $c^{(2)}(\mathbf{X}_1, \mathbf{X}_2)$. Its most common form is as shown [21]:

$$h^{(2)}(\mathbf{X}_1, \mathbf{X}_2) = c^{(2)}(\mathbf{X}_1, \mathbf{X}_2) + \int c^{(2)}(\mathbf{X}_1, \mathbf{X}_3) \rho(\mathbf{X}_3) h^{(2)}(\mathbf{X}_3, \mathbf{X}_2) d\mathbf{X}_3 \quad (5.33)$$

where $h^{(2)}(\mathbf{X}_1, \mathbf{X}_2)$ is the pair correlation function (PCF). When the fluid is homogenous, eq. (5.33) becomes:

$$h^{(2)}(\mathbf{X}_1, \mathbf{X}_2) = c^{(2)}(\mathbf{X}_1, \mathbf{X}_2) + \rho_0 \int c^{(2)}(\mathbf{X}_1, \mathbf{X}_3) h^{(2)}(\mathbf{X}_3, \mathbf{X}_2) d\mathbf{X}_3 \quad (5.34)$$

which gives

$$\gamma(\mathbf{X}_1, \mathbf{X}_2) = h^{(2)}(\mathbf{X}_1, \mathbf{X}_2) - c^{(2)}(\mathbf{X}_1, \mathbf{X}_2) = \rho_0 \int c^{(2)}(\mathbf{X}_1, \mathbf{X}_3) h^{(2)}(\mathbf{X}_3, \mathbf{X}_2) d\mathbf{X}_3 \quad (5.35)$$

If we take \mathbf{X}_2 as origin in the laboratory coordinate system, eq. (5.35) becomes:

$$\gamma(\mathbf{X}_1) = \int c^{(2)}(\mathbf{X}_1, \mathbf{X}_3) \Delta \rho(\mathbf{X}_3) d\mathbf{X}_3 \quad (5.36)$$

which coincides with the gradient of $\mathcal{F}_{\text{exc}}[\rho]$ in eq. (5.31).

It can be further proved that the homogenous reference fluid (HRF) approximation of molecular density functional theory (MDFT) is mathematically equivalent to the hypernetted chain (HNC) approximation to integral equations method (IEM) if the fluid is homogenous. (To be detailed.)

5.13 SUPERCELL DISCRETIZATION

$L_x \times L_y \times L_z$ Å³ space is discretized on a regular grid of nfft₁ × nfft₂ × nfft₃ nodes. Angular grid is discretized with Lebedev (L) or Gauss-Legendre (GL) quadrature for $\Omega \equiv \{\theta, \phi\}, \theta \in [0, \pi], \phi \in [0, 2\pi]$, and regular quadrature for $\psi \in [0, \pi]$.

TRANSLATE_SOLUTE_TO_CENTER = {T,F}

This translates your solute to the center of the supercell: all solute coordinates are moved by {Lx/2., Ly/2., Lz/2.}, with {Lx,Ly,Lz} the length of the supercell.

5.14 CG_VECT

5.14.1 Multipolar polarization

We define a charge density $\sigma(\mathbf{r}, \Omega)$ and polarization density $\mathbf{p}(\mathbf{r}, \Omega)$ of a molecule at the origin of the cartesian reference frame with orientation Ω .

$$\sigma(\mathbf{r}, \Omega) = \sum_m q_m \delta(\mathbf{r} - \mathbf{s}_m(\Omega)) \quad (5.37)$$

in the general case, and

$$\sigma(\mathbf{r}, \Omega) = q_O \delta(\mathbf{r}) + q_H \delta(\mathbf{r} - \mathbf{s}_{H_1}(\Omega)) + q_H \delta(\mathbf{r} - \mathbf{s}_{H_2}(\Omega)) \quad (5.38)$$

for a single SPC-type water molecule with oxygen at the origin.

5.15 POISSON SOLVER : ITERATIVE SOLVERS FOR LINEAR EQUATIONS

Given a square system of n linear equations:

$$\mathbf{A}\mathbf{x} = \mathbf{b}. \quad (5.39)$$

Some (very slow) direct techniques exist that involve algebra. We will solve this equation by an iterative (also called relaxation) method. We will use a sequence of $\mathbf{x}^{(k)}$ that converges toward the fixed point \mathbf{x} , solution of our problem.

In conclusion, we want to write, for a given $\mathbf{x}^{(0)}$, a sequence $\mathbf{x}^{(k+1)} = F(\mathbf{x}^{(k)})$ with $k \in \mathbb{N}$.

5.16 JACOBI'S METHOD

We decompose $A = D - L - U$ where D is an invertible matrix.

$$\mathbf{A}\mathbf{x} = \mathbf{b} \Leftrightarrow \mathbf{D}\mathbf{x} = (\mathbf{L} + \mathbf{U})\mathbf{x} + \mathbf{b} \quad (5.40)$$

$$\Leftrightarrow \mathbf{x} = \mathbf{D}^{-1}(\mathbf{L} + \mathbf{U})\mathbf{x} + \mathbf{D}^{-1}\mathbf{b} \quad (5.41)$$

$$= F(\mathbf{x}) \quad (5.42)$$

where F is a linear function of x .

It is shown that we can solve this equation by iteratively solving, where k indicates the iteration number

$$\mathbf{x}^{(k+1)} = \mathbf{D}^{-1}(\mathbf{L} + \mathbf{U})\mathbf{x}^{(k)} + \mathbf{D}^{-1}\mathbf{b}. \quad (5.43)$$

Since $\text{diag}((\mathbf{L} + \mathbf{U})\mathbf{x}^{(k)}) = \mathbf{0}$, each $x_i^{(k+1)}$ is independent of $x_i^{(k)}$. It is instead a weighted average of other $x_{j \neq i}$. Since we don't know the solution *a priori*, we define the residual at each step k as $\mathbf{r}^{(k)} = \mathbf{b} - \mathbf{A}\mathbf{x}^{(k)}$. See algorithm 5.1.

5.17 VEXT Q

L'énergie électrostatique s'écrit

$$\mathcal{F}_q = \frac{1}{2} \iiint \frac{\rho_c^{\text{soluté}}(\mathbf{r}') \rho_c^{\text{solvant}}(\mathbf{r}, \Omega)}{4\pi\epsilon_0 \|\mathbf{r} - \mathbf{r}'\|} d\mathbf{r} d\mathbf{r}' d\Omega \quad (5.44)$$

$$= \frac{1}{2} \frac{1}{4\pi\epsilon_0} \iiint \frac{\rho_c^{\text{soluté}}(\mathbf{r}')}{\|\mathbf{r} - \mathbf{r}'\|} [\rho^{\text{solvant}} * \sigma](\mathbf{r}, \Omega) d\mathbf{r} d\mathbf{r}' d\Omega \quad (5.45)$$

$$= \frac{1}{2} \frac{1}{4\pi\epsilon_0} \iint [\rho_c^{\text{soluté}} * \|\mathbf{r}\|^{-1}](\mathbf{r}, \Omega) [\rho^{\text{solvant}} * \sigma](\mathbf{r}, \Omega) d\mathbf{r} d\Omega \quad (5.46)$$

$$= \quad (5.47)$$

On définit une densité de charge d'une molécule $\sigma(\mathbf{r}, \Omega)$:

$$\sigma(\mathbf{r}, \Omega) = \sum_{m=1}^{\text{sites du solvant}} q_m \delta(\mathbf{r} - \mathbf{s}_m(\Omega)). \quad (5.48)$$

$\mathbf{s}_m(\Omega)$ désigne la position du $m^{\text{ième}}$ site de la molécule de solvant quand elle a l'orientation Ω .

Algorithm 5.1 Jacobi's algorithm

Output

- \mathbf{x} , an approximate solution to $\mathbf{Ax} = \mathbf{b}$ where \mathbf{A} and \mathbf{b} are known.

Inputs

- \mathbf{A}
- \mathbf{b}
- $\mathbf{x}^{(0)}$ including boundary conditions. No restriction on $\mathbf{x}^{(0)}$.
- ϵ , the tolerance or convergence criteria. A suitable tolerance might be $\|\mathbf{r}^{(k)}\| < \epsilon$ or $\|\mathbf{r}^{(k)} - \mathbf{r}^{(k-1)}\| < \epsilon$.

Restrictions

- \mathbf{A} must be strictly diagonaly dominant, *i.e.*, $|A_{i,i}| > \sum_{j=1,n;j \neq i} |A_{i,j}| \Rightarrow A_{i,i} \neq 0$.
- $\epsilon > 0$

Algo

1. $\mathbf{D} \leftarrow \text{diag}(\mathbf{A})$
 2. $\mathbf{R} \leftarrow \mathbf{D} - \mathbf{A}$
 3. $\mathbf{D}^{-1} \leftarrow 1/\mathbf{D}$
 4. do while $\xi \geq \epsilon$
 - a) $\mathbf{x}_{\text{old}} \leftarrow \mathbf{x}$
 - b) $\mathbf{x} \leftarrow \mathbf{D}^{-1} \mathbf{R} \mathbf{x}_{\text{old}} + \mathbf{D}^{-1} \mathbf{b}$
 - c) $\mathbf{r} \leftarrow \mathbf{Ax} - \mathbf{b}$
 - d) $\xi \leftarrow \|\mathbf{r}\|$
-

5.18 CODE MDFT

The code MDFT is the implementation of MDFT theory. It reads the force field (pair potential) $u(\mathbf{r}, \Omega)$ of solute and solvent as input, as well as necessary parameters like the temperature T , number density of solvent n_0 , etc. It minimizes the functional [ref] and gives the equilibrium density $\rho(\mathbf{r}, \Omega)$, then computes output properties. The main structure of code is shown in figure 5.2.

5.18.1 Minimizer L-BFGS-B

The minimizer adopted by MDFT is the L-BFGS-B [5, 49] package version 3.0 written in Fortran 77, implementing the limited-memory Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm with constraints of the form $l \leq x \leq u$ to the variable x . **During the evaluation of the initial code which use L-BFGS**, the constraint function is not used.

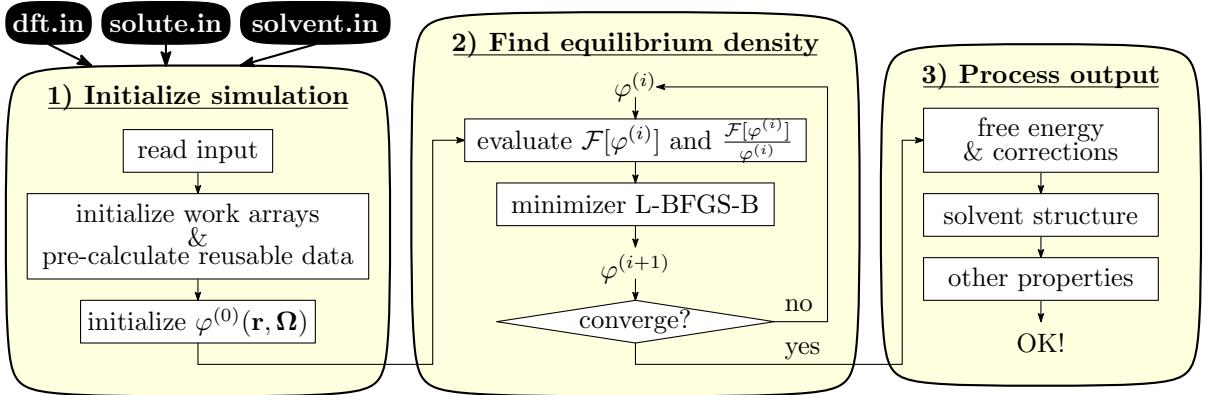


Figure 5.2: Main structure of code MDFT

The functional $\mathcal{F}[x_i]$ and the gradient of functional $\nabla\mathcal{F}[x_i] = \frac{\delta\mathcal{F}}{\delta x_i}(x_i)$ is needed by L-BFGS to minimize the functional. It saves the variables x_i and gradient of the past m iterations, which is memory eater.

The functional in MDFT to be minimized is eq. (ref), and its gradient is

$$\frac{\delta\mathcal{F}[\rho]}{\delta\rho(\mathbf{r}, \Omega)} = \beta^{-1} \ln \left(\frac{\rho(\mathbf{r}, \Omega)}{\rho_0} \right) + V_{\text{ext}}(\mathbf{r}, \Omega) + V_{\text{exc}}(\mathbf{r}, \Omega) \quad (5.49)$$

where ρ_0 is the angular density of bulk solvent $\rho_0 = n_0 / (8\pi^2)$

5.18.2 Treatment to avoid unphysical density

During the minimization, the density variable $\rho(\mathbf{r}, \Omega)$ can have unphysical negative number, which also cause the divergence of the minimization. To avoid this phenomenon, a normalized $\varphi(\mathbf{r}, \Omega)$ is used as variable during the minimization in the place of $\rho(\mathbf{r}, \Omega)$, so that

$$\rho(\mathbf{r}, \Omega) = \rho_0 \varphi^2(\mathbf{r}, \Omega) \quad (5.50)$$

According to the definition (5.50) we see

$$\frac{\delta\rho(\mathbf{r}, \Omega)}{\delta\varphi} = 2\rho_0\varphi(\mathbf{r}, \Omega) \quad (5.51)$$

Therefore the gradient to feed the L-BFGS minimizer (but in the code there is additional $\text{drd}\Omega$??? for all the three part)

$$\frac{\delta\mathcal{F}}{\delta\varphi} = \frac{\delta\mathcal{F}}{\delta\rho} \cdot \frac{\delta\rho}{\delta\varphi} = 2\rho_0\varphi(\mathbf{r}, \Omega) \cdot [\beta^{-1} \ln \varphi^2 + V_{\text{ext}} + V_{\text{exc}}] \quad (5.52)$$

5.18.3 Evaluation of V_{ext}

In eq. (ref) we define the external potential V_{ext} as the gradient of external free energy functional due to the solute, of unity []. When the solute is a molecule with force field, it is contains two components

$$V_{\text{ext}}(\mathbf{r}, \Omega) = V_{\text{LJ}}(\mathbf{r}) + V_{\text{coul}}(\mathbf{r}, \Omega) \quad (5.53)$$

The Lennard-Jones potential is given by

$$V_{\text{LJ}}(\mathbf{r}) = \sum_u \sum_v 4\epsilon_{uv} \left[\left(\frac{\sigma_{uv}}{r_{uv}} \right)^{12} - \left(\frac{\sigma_{uv}}{r_{uv}} \right)^6 \right] \quad (5.54)$$

where u stands for solute, and v stands for solvent, $\epsilon_{uv} = \sqrt{\epsilon_u \epsilon_v}$ and $\sigma_{uv} = (\sigma_u + \sigma_v)$ are the geometric and arithmetic average Lennard-Jones parameters between solute and solvent. r_{ij} is the norm of relative site-site vector

$$\mathbf{r}_{uv} = \mathbf{r} + \mathbf{R}(\Omega) \mathbf{s}_v - \mathbf{r}_u \quad (5.55)$$

where \mathbf{r}_u and \mathbf{s}_j are the coordinates of solute / solvent molecule in the molecular frame, and $\mathbf{R}(\Omega)$ the rotation matrix of the Euler angles Ω .

In the case where the solvent site wears only one LJ centre, eq. (5.55) reduces to

$$\mathbf{r}_{uv} = \mathbf{r} - \mathbf{r}_u \quad (5.56)$$

which is what we use exactly in the code as the solvent is SPC/E water.

...

The Coulomb interaction is calculated by [solving the Poisson equation \[32\]](#).

The charge density of the solute is projected onto space grid,

$$\rho_q(\mathbf{r}) = \sum_u q_{ijk} \quad (5.57)$$

where q_{ijk} is the charge on the space grid distributed by its nearby point charge as shown in figure 5.3a.

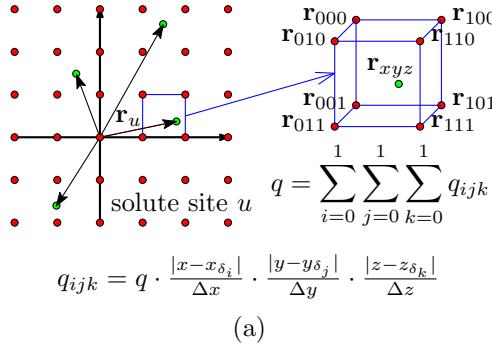


Figure 5.3: Charge density projected onto grids. (a) Solute. (b) Solvent.

$$V_{\text{coul}}(\mathbf{r}, \Omega) = \sum_v q_v V_q(\mathbf{r}_u) \quad (5.58)$$

Poisson solver

Field

$$\mathbf{E}(\mathbf{r}) = -\vec{\nabla} V_q(\mathbf{r}) \quad (5.59)$$

where $V_q(\mathbf{r})$ is ... of unity ...

Local expression of Gauss theorem

$$\nabla \cdot \mathbf{E}(\mathbf{r}) = \frac{\rho_q(\mathbf{r})}{\epsilon_0} \quad (5.60)$$

therefore

$$\nabla^2 V_q(\mathbf{r}) = -\frac{\rho_q(\mathbf{r})}{\epsilon_0} \quad (5.61)$$

$$\hat{V}_q(\mathbf{k}) = \frac{\hat{\rho}_q(\mathbf{k})}{\epsilon_0 k^2} \quad (5.62)$$

where $\hat{V}_q(\mathbf{k})$ is the Fourier transform of $V_q(\mathbf{r})$.

The Fourier transform, since the Laplacian is a linear operator

$$\nabla^2 f(\mathbf{r}) = \int d\mathbf{k} \hat{f}(k) e^{2\pi i \mathbf{r} \cdot \mathbf{k}} \quad (5.63)$$

$$= \int d\mathbf{k} \hat{f}(k) \nabla^2 e^{2\pi i \mathbf{r} \cdot \mathbf{k}} \quad (5.64)$$

$$= \int d\mathbf{k} (-4\pi^2 |\mathbf{k}|^2) \hat{f}(k) e^{2\pi i \mathbf{r} \cdot \mathbf{k}} \quad (5.65)$$

$$\mathcal{F} [\nabla^2 V_q(\mathbf{r})] = -4\pi^2 |\mathbf{k}|^2 \hat{V}_q(\mathbf{k}) \quad (5.66)$$

For Fourier series $-4\pi^2 |\mathbf{k}|^2$ is the eigenvalue of laplacian

$$\mathcal{F} [\nabla^2 V_q(\mathbf{r})] = \mathcal{F} \left[-\frac{\rho_q(\mathbf{r})}{\varepsilon_0} \right] = (i\mathbf{k})^2 \hat{V}_q(\mathbf{k}) = -4\pi \hat{\rho}_q(\mathbf{k}) \quad (5.67)$$

$$\hat{V}_{\text{Poisson}}(\mathbf{k}) = \frac{4\pi \hat{\rho}_q(\mathbf{k})}{k^2} \quad (5.68)$$

5.18.4 Evaluation of V_{exc}

V_{exc} is ... of unity []

Diople

We define $r = \|\mathbf{r} - \mathbf{r}'\|$, and $\Delta n(\mathbf{r}) = n(\mathbf{r}) - n_0$, and n_0 the density of the bulk solvent, e.g., 0.0332891 molecule per \AA^3 for water.

We also define $n(\mathbf{r}) = \int \rho(\mathbf{r}, \Omega) d\Omega$. We have

$$F_{\text{exc}} = -\frac{1}{2} k_B T \iint \Delta n(\mathbf{r}) \Delta n(\mathbf{r}') c(r) d\mathbf{r} d\mathbf{r}', \quad (5.69)$$

Now, we consider the convolution in the right hand side of the equation, $\gamma \equiv (\Delta n * c)$, that can be computed much efficiently than in $O(N^2)$ by fast Fourier transform in $O(N \log N)$.

$$\mathbf{P}(\mathbf{r}) = \int \mathbf{p} \rho(\mathbf{r}, \Omega) d\Omega \quad (5.70)$$

with $\mathbf{p} = p\Omega$ the dipolar moment of a water molecule.

HRF approximation [48]

Work by Zhao et al.

6

MOLECULAR DYNAMICS & MONTE CARLO SIMULATION

As reference method [Allen and Tildesley] [Frenkel and Smit]

The data we used in this thesis issue from MD and MC simulations are in two aspects:
(1) The DCF of bulk water produced by the work of Zhao et al. [ref] and Belloni et al. [ref].
(2) The **RDF!** (**RDF!**) of ions and molecules. The first two topics give a brief review of the principle of MD and MC, the last two gives the detail of generation of the above two data.

6.1 MOLECULAR DYNAMICS

6.1.1 *Principle*

6.1.2 *Determination of free energy: umbrella sampling*

6.2 MONTE CARLO METHOD

6.3 DIRECT CORRELATION FUNCTION OF WATER

described the direct correlation functions (DCF) used during the thesis, including different sources and forms, as well as their comparison.

6.3.1 *Dipole DCF from molecular dynamics simulation*

6.3.2 *Extraction of DCF from bulk Monte Carlo simulation*

This DCF set is calculated by Belloni *et al.* [42] and is presented here briefly for understanding. first rotational invariant of the Fourier transform of the total correlation function h

example:

$$\hat{c}^{000}(k) = \frac{\hat{h}^{000}(k)}{1 + n_0 \hat{h}^{000}(k)}$$

g accumulated, solve the inverse OZ equation to find c

norm of wave vector $\|\mathbf{q}\| = \{i \cdot \Delta_q \mid 0 \leq i \leq n, i \in \mathbb{Z}\}$, beginning from the 10th line, where Δ_q is the minimum difference between two norm q .

6.3.3 *Conventions*

whereas I take the notation of Wertheim and Hansen, or in k-space

$$\begin{aligned}
\Phi^{000} &= 1 \\
\Phi^{011} &= \hat{\mathbf{k}} \cdot \boldsymbol{\Omega}_1 \\
\Phi^{101} &= \hat{\mathbf{k}} \cdot \boldsymbol{\Omega}_2 \\
\Phi^{110} &= \boldsymbol{\Omega}_1 \cdot \boldsymbol{\Omega}_2 \\
\Phi^{112} &= 3(\hat{\mathbf{k}} \cdot \boldsymbol{\Omega}_1)(\hat{\mathbf{k}} \cdot \boldsymbol{\Omega}_2) - \boldsymbol{\Omega}_1 \cdot \boldsymbol{\Omega}_2
\end{aligned} \tag{6.1}$$

Different rotational invariant projections from Luc's c. Luc defines (according to Blum)

$$\begin{aligned}
\Phi^{000} &= 1 \\
\Phi^{011} &= i\mathbf{k} \cdot \boldsymbol{\Omega}_1 = i \cos \theta_1 \\
\Phi^{101} &= i\mathbf{k} \cdot \boldsymbol{\Omega}_2 = i \cos \theta_2 \\
\Phi^{110} &= -\sqrt{3}\boldsymbol{\Omega}_1 \cdot \boldsymbol{\Omega}_2 = -\sqrt{3}(\sin \theta_1 \sin \theta_2 \cos \phi_{12} + \cos \theta_1 \cos \theta_2) \\
\Phi^{112} &= \sqrt{\frac{3}{10}} [3(\mathbf{k} \cdot \boldsymbol{\Omega}_1)(\mathbf{k} \cdot \boldsymbol{\Omega}_2) - \boldsymbol{\Omega}_1 \cdot \boldsymbol{\Omega}_2] = \sqrt{\frac{3}{10}} (2 \cos \theta_1 \cos \theta_2 - \sin \theta_1 \sin \theta_2 \cos \phi_{12})
\end{aligned} \tag{6.2}$$

6.3.4 Comparison with non-coupling dipole DCF in MDFT ($n_{\max} = 1$)

$$c_s^{000}, c_{\Delta}^{110}, c_d^{112}$$

6.3.5 Comparison with respect to n_{\max}

first rotational invariants $m_{\max} = 1$ (4 independent projections $c_s^{000}, c_{\Delta}^{110}, c_d^{112}$ and $c_{+}^{011} = -c_{-}^{101}$).

6.3.6 Transform between ck_angular and projections

$$\hat{c}_q(\cos \theta_1, \cos \theta_2, \phi, \psi_1, \psi_2)$$

columns: $\cos \theta_1, \cos \theta_2, \psi_1, \psi_2, \phi = \phi_1 - \phi_2, \operatorname{Re}(\hat{c}), \operatorname{Im}(\hat{c})$; with $(n_{\theta} + 1)n_{\theta}n_{\phi}n_{\psi}^2/4$ items as only $\cos \theta_1 + \cos \theta_2 > 0$ and $\phi \in [0, \pi]$ is stocked, where $n_{(\text{angle})}$ is the number of angles for each dimension.

6.4 UMBRELLA SAMPLING

Chapter II

THEORY: HRF APPROXIMATION, FOR MOLECULAR SOLVENT

This chapter presents a complete theory of the \mathcal{F}_{exc} evaluation under HRF approximation:

$$\mathcal{F}_{\text{exc}} = -\frac{\beta^{-1}}{2} \int d\mathbf{r}_1 d\mathbf{r}_2 d\Omega_1 d\Omega_2 \Delta\rho(\mathbf{r}_1, \Omega_1) \Delta\rho(\mathbf{r}_2, \Omega_2) c(r_{12}, \Omega_1, \Omega_2)$$

which is based on previous work of Zhao et al. [48], where the HRF approximation has been applied to linear molecules ($\Omega \equiv (\Theta, \Phi)$). In this thesis, the method is generated to molecular solvent, using 3 Euler angles, i. e. $\Omega \equiv (\Theta, \Phi, \Psi)$, where the computing cost of the original algorithm is no longer reasonable. Further approximation is therefore made, where the density variable $\rho(\mathbf{r}, \Omega)$ is expended on generalized spherical harmonics. Theoretically, this approximation gives little loss of accuracy, but makes a great advantage in computing time and memory requirement. The prove is shown in the implementation part.

Section 7 describes the FFT treatment for the spatial convolution in the gradient γ of the excess functional \mathcal{F}_{exc} , which reduces the algorithm complexity from $O(N^2)$ to $O(N \log_2 N)$. A DCF directly issue of Monte Carlo simulation and solution HNC is used, in both intermolecular and projection form. To use the intermolecular form, the matrix which does the transform from laboratory to intermolecular coordinates system is generalized to molecular case compared to previous work, then interpolation of zero and first order is involved. To use the projection form, DCF is reconstructed with all projections. For order of projections $n_{\text{max}} = 1$, the formula of each projections are written explicitly.

Section 8 presents the treatment of angular convolution. As IEM and MDFT have mathematical equivalence, an algorithm inspired by the work of Fries [14] and Blum [3, 4] for IEM is built for MDFT. In this algorithm, the density variable $\rho(\mathbf{r}, \Omega)$ is expended on generalized spherical harmonics, then rotated onto intermolecular frame. It is shown that in this form, the OZ equation is largely simplified.

The solvent properties involved in this thesis are presented in the next two chapters. Section 9 presents some thermodynamic quantities, including the solvation free energy and its corrections, ... and section 10 gives some forms of structure that can be preformed, such as the radical distribution function, radical polarization functions, rotational invariants expansion, etc.

REDUCTION OF SPATIAL CONVOLUTION BY FFT

As presented in section 5, to complete the minimization process of MDFT, we need to evaluate the excess functional as well as its gradient:

$$\mathcal{F}_{\text{exc}} = -\frac{\beta^{-1}}{2} \int d\mathbf{r}_1 d\mathbf{r}_2 d\Omega_1 d\Omega_2 \Delta\rho(\mathbf{r}_1, \Omega_1) \Delta\rho(\mathbf{r}_2, \Omega_2) c(\mathbf{r}_{12}, \Omega_1, \Omega_2) \quad (7.1)$$

$$\gamma(\mathbf{r}_1, \Omega_1) = -\beta^{-1} \int d\mathbf{r}_2 d\Omega_2 \Delta\rho(\mathbf{r}_2, \Omega_2) c(\mathbf{r}_{12}, \Omega_1, \Omega_2) \quad (7.2)$$

To evaluate the gradient γ for each (\mathbf{r}, Ω) , $N \equiv N_r N_\Omega$ function evaluations (FE) are needed, the total number of FE is thus $N^2 = O(N^2)$, which, with typically $N_r = 64^3$ and $N_\Omega = 50 \sim 100$, is too costly for current computing technology. For this reason, Fourier transform is used to treat the spatial convolution in eq. (7.2).

A convolution

$$h(x_1) \equiv f(x_2) \otimes g(x_2) \equiv \int_a^b f(x_2) g(x_1 - x_2) dx_2 \quad (7.3)$$

has the property that

$$\mathfrak{F}[h(x_1)] = \mathfrak{F}[f(x_2)] \mathfrak{F}[g(x_2)] \quad (7.4)$$

\mathfrak{F} being the Fourier transform (FT) operation (*Attention d'avoir bien défini les FFT 3D avant*). As $\mathbf{r}_{12} = \mathbf{r}_1 - \mathbf{r}_2$, eq. (7.2) is a 3D convolution, which leads to

$$\hat{\gamma}(\mathbf{k}, \Omega_1) = -\beta^{-1} \int d\Omega_2 \Delta\hat{\rho}(\mathbf{k}, \Omega_2) \hat{c}(\mathbf{k}, \Omega_1, \Omega_2) \quad (7.5)$$

Thus the integral $\int d\mathbf{r}_2$ in eq. (7.2) is transformed to a simple product in eq. (7.5). To get $\hat{\gamma}(\mathbf{k}, \Omega_1)$ with given $\Delta\hat{\rho}(\mathbf{k}, \Omega_2)$, only $N_r N_\Omega^2$ FE are needed. To this computational cost should be added the transform from $\Delta\rho(\mathbf{r}, \Omega)$ to $\Delta\hat{\rho}(\mathbf{k}, \Omega)$ and the backward transform from $\hat{\gamma}(\mathbf{k}, \Omega)$ to $\gamma(\mathbf{r}, \Omega)$ which are both of order $N_\Omega \cdot O(N_r \log_2 N_r)$, due to the properties of Fast Fourier Transforms (FFT). The total number of FE is thus reduced from quadratic complexity $O(N_r^2 N_\Omega^2)$ to $N_r N_\Omega^2 + 2N_\Omega \cdot O(N_r \log_2 N_r) = O(N_r \log_2 N_r N_\Omega^2)$. As the total number of spatial grid N_r is of magnitude $10^5 \sim 10^6$, this procedure, which is mathematically equivalent to the direct evaluation (7.2), gives a great advantage in terms of computational efficiency (figure A.1 in section 1).

Once $\gamma(\mathbf{r}, \Omega)$ is obtained by inverse Fourier transform of $\hat{\gamma}(\mathbf{k}, \Omega)$, the excess functional can be calculated as

$$\mathcal{F}_{\text{exc}}[\rho(\mathbf{r}, \Omega)] = \frac{1}{2} \int d\mathbf{r} d\Omega \Delta\rho(\mathbf{r}, \Omega) \gamma(\mathbf{r}, \Omega) \quad (7.6)$$

It should be noticed that, the direct correlation function (DCF), $\hat{c}(\mathbf{k}, \Omega_1, \Omega_2)$, used as an input data in eq. (7.5) is very memory-costly. For instance, with a normal setting with 64^3 spatial grid and a Lebedev quadrature of order 2 (14 angles for Θ and Φ), and 3 Ψ -angles, even if the DCF is stocked in simple precision, it takes $64^3 \times 42^2 \times 4$ bytes = 1.76GB, and for a Lebedev quadrature of order 5 and correspondingly 5 Ψ -angles, it takes $64^3 \times 250^2 \times 4$ bytes = 65.5GB. As a normal PC has only 4 to 16 GB of RAM, it can cause a memory leak. Therefore, two strategies are developed to reduce the storage of the DCF.

7.1 USING INTERMOLECULAR DCF

(In the previous work, the interpolation has been done with $\Omega \equiv (\Theta, \Phi)$. Here, we discuss the case of full 3 Euler angles $\Omega \equiv (\Theta, \Phi, \Psi)$. : To be put somewhere else.)

The first strategy is to work in the so-called intermolecular frame described in Fig. 7.1 for which, in \mathbf{r} -space, the z axis is oriented along the vector $\mathbf{r}_{12} = \mathbf{r}_2 - \mathbf{r}_1$, or in \mathbf{k} -space is oriented along the vector \mathbf{k} . In the later case, an orientation $\Omega \equiv (\Theta, \Phi, \Psi)$ in laboratory frame become $\omega \equiv (\theta, \phi, \psi)$ in intermolecular frame.

To store the DCF in the intermolecular coordinates system, it can be defined as $\hat{c}(k, \omega_1, \omega_2)$, where $(\omega_1, \omega_2) \equiv (\cos \theta_1, \cos \theta_2, \phi, \psi_1, \psi_2)$ and k always oriented along the z axis (figure 7.1) such that only 6 variables are needed instead of 9 for $\hat{c}(\mathbf{k}, \Omega_1, \Omega_2)$, and the storage is considerably reduced. The transform from $\hat{c}(k, \omega_1, \omega_2)$ to $\hat{c}(\mathbf{k}, \Omega_1, \Omega_2)$ relies on the correspondence $\omega(\mathbf{k}, \Omega) \equiv (\cos \theta, \phi, \psi)$, which can be pre-calculated as a table of data.

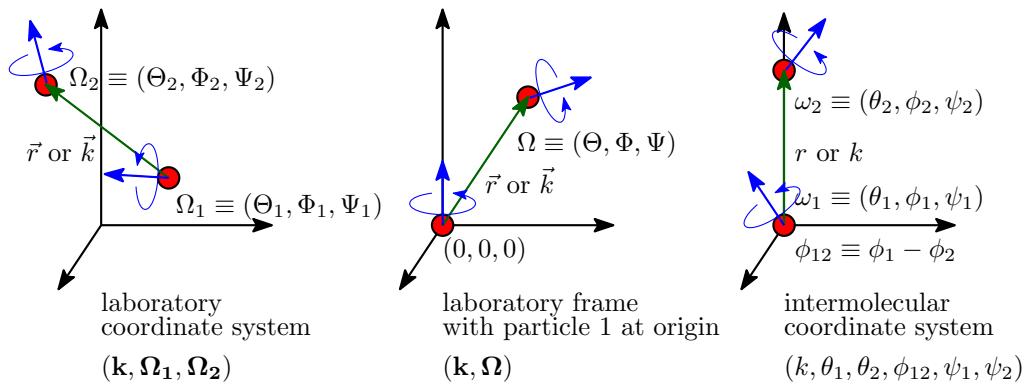


Figure 7.1: Molecule 1 and 2 in different coordinate systems

To find ω from Ω amounts to define the correspondence between the rotation matrices of the two coordinates systems.

The rotation matrix $\hat{\mathbf{R}}_\Omega$ rotates the solvent molecule from \mathbf{I} to its orientation $\hat{\mathbf{R}}_\Omega$:

$$\hat{\mathbf{R}}_\Omega \mathbf{I} = \hat{\mathbf{R}}_\Omega \quad (7.7)$$

It can be expressed by 3 rotation operations $\hat{\mathbf{R}}_\Phi$, $\hat{\mathbf{R}}_\Theta$, and $\hat{\mathbf{R}}_\Psi$ which rotates along $z - y - z$ axes (the same convention as defined in Messiah and Gray-Gubbins):

$$\begin{aligned} \hat{\mathbf{R}}_\Omega &= \begin{bmatrix} R_{xx} & R_{xy} & R_{xz} \\ R_{yx} & R_{yy} & R_{yz} \\ R_{zx} & R_{zy} & R_{zz} \end{bmatrix} \\ &= \begin{bmatrix} \cos \Phi & -\sin \Phi & 0 \\ \sin \Phi & \cos \Phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos \Theta & 0 & \sin \Theta \\ 0 & 1 & 0 \\ -\sin \Theta & 0 & \cos \Theta \end{bmatrix} \begin{bmatrix} \cos \Psi & -\sin \Psi & 0 \\ \sin \Psi & \cos \Psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} \cos \Phi \cos \Theta \cos \Psi - \sin \Phi \sin \Psi & -\cos \Phi \cos \Theta \sin \Psi - \sin \Phi \cos \Psi & \cos \Phi \sin \Theta \\ \sin \Phi \cos \Theta \cos \Psi + \cos \Phi \sin \Psi & -\sin \Phi \cos \Theta \sin \Psi + \cos \Phi \cos \Psi & \sin \Phi \sin \Theta \\ -\sin \Theta \cos \Psi & \sin \Theta \sin \Psi & \cos \Theta \end{bmatrix} \end{aligned} \quad (7.8)$$

As shown in figure 7.2, the rotation matrix to transforming the DCF from the intermolecular coordinates to laboratory coordinates $\hat{\mathbf{R}}_\omega$ can be written as:

$$\hat{\mathbf{R}}_\omega = \hat{\mathbf{R}}_k^{-1} \hat{\mathbf{R}}_\Omega \quad (7.9)$$

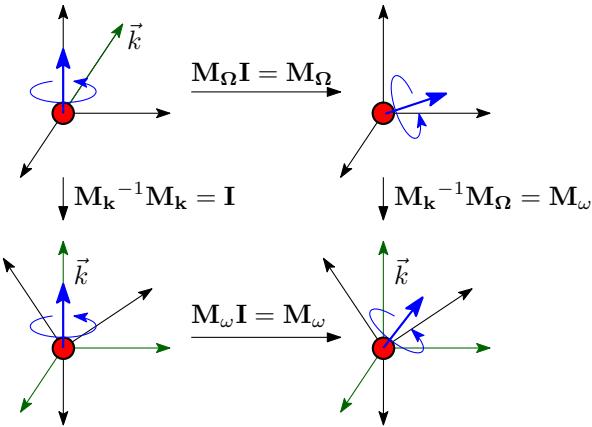


Figure 7.2: Rotation matrices

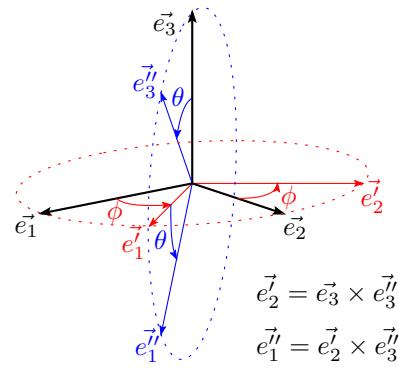


Figure 7.3: Rotation to k-frame

with the rotation matrix related to \mathbf{k} vector:

$$\hat{\mathbf{R}}_k^{-1} = \begin{bmatrix} \cos \theta_k \cos \phi_k & \cos \theta_k \sin \phi_k & -\sin \theta_k \\ -\sin \phi_k & \cos \phi_k & 0 \\ \sin \theta_k \cos \phi_k & \sin \theta_k \sin \phi_k & \cos \theta_k \end{bmatrix} \quad (7.10)$$

Here we fixe $\psi_k = 0$. As θ_k and ϕ_k are calculated from Cartesian coordinates (k_x , k_y , k_z), in the extreme cases that we cannot define θ_k (for $\|\mathbf{k}\| = 0$) and ϕ_k (for $k_x^2 + k_y^2 = 0$), we can fix arbitrarily those angles to zero.

A faster way to find the rotation matrix of \mathbf{k} , avoiding the evaluation of trigonometric functions, is shown in figure 7.3, where the matrix can be calculated by the cross products of basis vectors from z axis and \mathbf{k} vector ($\mathbf{k} = \mathbf{e}_3''$):

$$\begin{bmatrix} \mathbf{e}_1'' & \mathbf{e}_2' & \mathbf{e}_3'' \end{bmatrix} = \begin{bmatrix} \mathbf{e}_1 & \mathbf{e}_2 & \mathbf{e}_3 \end{bmatrix} \hat{\mathbf{R}}_k = \hat{\mathbf{R}}_k \quad (7.11)$$

The two ways to calculate \mathbf{k} differs only in the case of $\hat{\mathbf{k}} = [0 \ 0 \ -1]^T$, where one is the inverse of the other. This is due to the different definitions of ϕ_k (0 or π when \vec{k}_z superposes with \vec{k}_z) in the two cases. Tests have shown that it has no influence to the final result of the excess functional evaluation.

Then, the elements of $\hat{\mathbf{R}}_\omega$ can be calculated according to eq. (7.9)

$$\begin{aligned} \hat{\mathbf{R}}_\omega &= \begin{bmatrix} u_x & v_x & w_x \\ u_y & v_y & w_y \\ u_z & v_z & w_z \end{bmatrix} \\ &= \begin{bmatrix} \cos \phi \cos \theta \cos \psi - \sin \phi \sin \psi & -\cos \phi \cos \theta \sin \psi - \sin \phi \cos \psi & \cos \phi \sin \theta \\ \sin \phi \cos \theta \cos \psi + \cos \phi \sin \psi & -\sin \phi \cos \theta \sin \psi + \cos \phi \cos \psi & \sin \phi \sin \theta \\ -\sin \theta \cos \psi & \sin \theta \sin \psi & \cos \theta \end{bmatrix} \end{aligned} \quad (7.12)$$

The angles ω is thus found as

$$\begin{aligned} \cos \theta &= w_z \\ \phi &= \arccos(w_x / (w_x^2 + w_y^2)^{\frac{1}{2}}) \\ \psi &= \arccos(-u_z / (u_z^2 + v_z^2)^{\frac{1}{2}}) \end{aligned} \quad (7.13)$$

The resulting angles are between normal intervals, $\cos \theta \in [-1, 1]$, $\phi \in [0, 2\pi]$. As water possesses C_{2v} symmetry, we take $\psi \in [0, \pi]$.

(Expliquer ici -à nouveau- pourquoi tu as besoin de faire des interpolations: grille plus fine pour om dans $c(k, \omega_1, \omega_2)$ et pour chaque (k, Ω) on doit trouver le om corerspondant.)

The interpolation of these found angles to the intermolecular grid can be done with different orders: zeroth order interpolation which takes directly the nearest point, or linear interpolation.

7.1.1 Zero-order interpolation of DCF

At this order, for each possible value of \mathbf{k} and $\boldsymbol{\Omega}$, the corresponding $\cos \theta$ and ψ which relate to only one solvent molecule are stored as an index (single precision integer), which gives the nearest angle in a pre-defined table

$$\begin{aligned} i_{\cos \theta} &= \lfloor (\cos \theta + 1)(n_{\cos \theta}/2) \rfloor + 1 \\ i_\psi &= \text{mod}(\lfloor \psi(n_\psi/\pi) \rfloor, n_\psi) + 1 \end{aligned} \quad (7.14)$$

where $\lfloor f \rfloor$ is the floor function. For the angle ϕ which relate to two solvent molecules, the operation $\phi = \phi_1 - \phi_2$ introduces a double error when integer indices are used, as shown in figure 7.4.

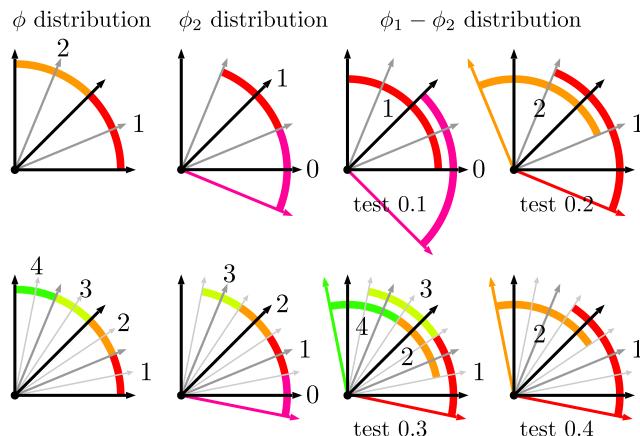


Figure 7.4: $\phi_1 - \phi_2$ distribution: Test 0.1 is the direct subtraction of ϕ established in the same way with θ and ψ , as shown in the top first schema. Test 0.2 tabulates ϕ_2 by taking the nearest point in another manner, as shown in the second schema. In test 0.3-0.4, all ϕ or only ϕ_2 is doubled.

In the actual implementation, as an integer takes 4 bytes and a real takes 8 bytes, there is no profit to tabulate ϕ in integer two times, thus ϕ is stored directly in real.

7.1.2 Linear interpolation of DCF

At this order, $\omega(\mathbf{k}, \boldsymbol{\Omega})$ is stored in double precision. All angles are stored in real type, and the correspondent DCF is calculated as

$$c(\omega) = w_0 c(\omega_0) + w_1 c(\omega_1) \quad (7.15)$$

where $w_0 = \frac{\omega_1 - \omega}{\omega_1 - \omega_0}$ and $w_1 = \frac{\omega - \omega_0}{\omega_1 - \omega_0}$. Here ω is one of the 5 dimensions in $\tilde{\omega}(\mathbf{k}, \boldsymbol{\Omega}_1, \boldsymbol{\Omega}_2) \equiv (\cos \theta_1, \cos \theta_2, \phi, \psi_1, \psi_2)$, ω_0 and ω_1 are the 2 nearest value points, while other variables

are fixed. If we express the weight for each dimension as $w_{n_i}^i$ where $i = 1, 2, 3, 4, 5$ is the i th variable, the total equation with 5 variables is

$$c(\tilde{\omega}) = \left[\sum_{n_1=0}^1 \sum_{n_2=0}^1 \sum_{n_3=0}^1 \sum_{n_4=0}^1 \sum_{n_5=0}^1 \left(\prod_i^5 w_{n_i}^i c(\tilde{\omega}_{n_1, n_2, n_3, n_4, n_5}) \right) \right] \quad (7.16)$$

These two equations are available for both interpolation and extrapolation, where the latter applies e.g. for $\cos \theta_1$ and $\cos \theta_2$.

An error evaluation of these two strategies of interpolation is shown in appendix F. Results show that the linear interpolation scheme is absolutely needed. On the other hand, as seen in eq. (7.16), it is computationally much more expensive than the simple histogram scheme as it requires $2^5 = 32$ times of operations.

7.2 DIRECT CALCULATION OF DCF FROM ROTATIONAL INVARIANT PROJECTIONS

Another strategy to calculate $\hat{c}(\mathbf{k}, \Omega_1, \Omega_2)$ is to use the DCF expressed in terms of rotational invariant projections, which takes far less memory than in the intermolecular form thanks to their angular independence and symmetric properties.

7.2.1 Using projections in form of $\hat{c}_{\mu\nu}^{mn}(k)$

As described by Blum [3, 4], $\hat{c}(\mathbf{k}, \Omega_1, \Omega_2)$ can be expanded as

$$\hat{c}(\mathbf{k}, \Omega_1, \Omega_2) = \sum_{mn\mu\nu} \hat{c}_{\mu\nu}^{mn}(k) \Phi_{\mu\nu}^{mn}(\hat{\mathbf{k}}, \Omega_1, \Omega_2) \quad (7.17)$$

where $\Phi_{\mu\nu}^{mn}(\hat{\mathbf{k}}, \Omega_1, \Omega_2)$ is the rotational invariant that depends on both the spatial and angular coordinates of the two particles (detailed in appendix C).

For projections of order $n_{\max} = 1$, the DCF can be expressed in very simple form. Only 4 projections $\hat{c}^{mn}(k)$ are independent: $\hat{c}_S \equiv \hat{c}^{000}$, $\hat{c}_\Delta \equiv \hat{c}^{110}$, $\hat{c}_D \equiv \hat{c}^{112}$ and $\hat{c}_+ \equiv \hat{c}^{011} = -\hat{c}^{101}$, with the corresponding rotational invariants expressed below both in laboratory and intermolecular frame (il faut définir ici OM_1 et OM_2 comme des vecteurs d'orientation !)

$$\begin{aligned} \Phi^{000} &= 1 \\ \Phi^{011} &= i\mathbf{k} \cdot \Omega_1 = i \cos \theta_1 \\ \Phi^{101} &= i\mathbf{k} \cdot \Omega_2 = i \cos \theta_2 \\ \Phi^{110} &= -\sqrt{3}\Omega_1 \cdot \Omega_2 = -\sqrt{3}(\sin \theta_1 \sin \theta_2 \cos \phi_{12} + \cos \theta_1 \cos \theta_2) \\ \Phi^{112} &= \sqrt{\frac{3}{10}} [3(\mathbf{k} \cdot \Omega_1)(\mathbf{k} \cdot \Omega_2) - \Omega_1 \cdot \Omega_2] \\ &= \sqrt{\frac{3}{10}} (2 \cos \theta_1 \cos \theta_2 - \sin \theta_1 \sin \theta_2 \cos \phi_{12}) \end{aligned} \quad (7.18)$$

To express the DCF at higher orders, the number of FE needed for $\Phi_{\mu\nu}^{mn}(\hat{\mathbf{k}}, \Omega_1, \Omega_2)$ becomes huge and the DCF should be calculated in intermolecular frame as indicated above.

7.2.2 Using projections in form of $\hat{c}_{\mu\nu,\chi}^{mn}(k)$

Compared to the expression of $\Phi_{\mu\nu}^{ml}(\hat{\mathbf{k}}, \Omega_1, \Omega_2)$ in laboratory frame (eq. (C.2) in appendix C), its intermolecular form has far less terms (eq. (C.18) in appendix C), such that

$$\hat{c}(k, \omega_1, \omega_2) = \frac{1}{2l+1} \sum_{mn\mu\nu\chi} \hat{c}_{\mu\nu,\chi}^{mn}(k) r_{\chi\mu}^m(\theta_1) r_{-\chi\nu}^n(\theta_2) e^{-i\chi(\phi_{12}\equiv\phi_1-\phi_2)} e^{-i\mu\psi_1} e^{-i\nu\psi_2} \quad (7.19)$$

where r is the generalized Legendre polynomial, $m, n \leq n_{\max}$, $|\mu| \leq m$, $|\nu| \leq n$, and $\chi \in [-\min(m, n), \min(m, n)]$.

$r_{\chi\mu}^m(\theta)$, $e^{-i\chi\phi}(\phi)$ and $e^{-i\mu\psi}(\psi)$ can be separately pre-tabulated for each given \mathbf{k} , to avoid double evaluation of each term.

E.q. (7.19) replaces the interpolation of eq. (7.16) by an exact formula and it requires to have stored in memory the projections $\hat{c}_{\mu\nu,\chi}^{mn}(k)$ rather than the full angular representation $\hat{c}(k, \omega_1, \omega_2)$. It requires also indeed the passage from orientations in laboratory frame to orientations in intermolecular frame, i.e. use of the formulas (7.13) for each \mathbf{k} vector.

8

ANGULAR CONVOLUTION, A BETTER ALGORITHM

In section 7, the spatial convolution is treated by FFT thanks to the transitional invariance that leads to $\mathbf{r}_{12} = \mathbf{r}_1 - \mathbf{r}_2$. However, as the angular grid is not homogeneous, the relative coordinates of two angles cannot be simply represented $\Omega_{12} = \Omega_1 - \Omega_2$, therefore we cannot take advantage of the convolution property shown in eq. (7.3-7.4). In other hand, these two-particle quantities also have rotational invariance. Proposed by Blum [3, 4] and used by Fries and Patey [14], a rotational invariant expansion technic reduces the molecular Ornstein-Zernike (MOZ) equation into smaller irreducible matrix equations. As there is an mathematical equivalence between IEM and MDFT (section 5), where eq. (7.5) can be regarded as the molecular OZ equation, this formalism can be also applied to MDFT.

8.1 ANGULAR CONVOLUTION USING BLUM'S REDUCTION

To build a relation between the irreducible form of the molecular OZ equation deduced by Blum (detailed in section 4)

$$\hat{\gamma}_{\lambda\mu,\chi}^{lm}(k) = \sum_{n=0}^{n_{\max}} \sum_{\nu=-n}^n (-)^{\chi+\nu} \Delta \hat{\rho}_{\lambda\nu,\chi}^{ln}(k) \hat{c}_{\mu\nu,\chi}^{mn}(k) \quad (8.1)$$

and the MDFT, a generalized spherical harmonic transform (GSHT) treatment is proposed by Luc Belloni, developing the functional gradient $\hat{\gamma}$ and the density $\hat{\rho}$ in eq. (7.5) on Wigner generalized spherical harmonics (GSH):

$$\hat{\gamma}(\mathbf{k}, \Omega_1) = \sum_{m\mu'\mu} f_m \hat{\gamma}_{\mu'\mu}^m(\mathbf{k}) R_{\mu'\mu}^m(\Omega_1) \quad (8.2)$$

$$\Delta \hat{\rho}(\mathbf{k}, \Omega_2) = \sum_{n\nu'\nu} f_n \Delta \hat{\rho}_{\nu'\nu}^n(\mathbf{k}) R_{\nu'\nu}^n(\Omega_2) \quad (8.3)$$

where $0 \leq m \leq m_{\max}$, $|\mu'|, |\mu| < m$ and $|\nu'|, |\nu| < n$. $f_m = (2m+1)^{\frac{1}{2}}$ is the normalization factor (according to Luc's definition).

The DCF can also be expended on rotational invariants [4], with the normalization factors according to Luc's definition:

$$\hat{c}(k, \Omega_1, \Omega_2) = \sum_{mnl\mu\nu} f_m f_n \hat{c}_{\mu\nu}^{mnl}(k) \sum_{\mu'\nu'\lambda'} \begin{pmatrix} m & n & l \\ \mu' & \nu' & \lambda' \end{pmatrix} R_{\mu'\mu}^m(\Omega_1) R_{\nu'\nu}^n(\Omega_2) R_{\lambda'0}^l(\hat{\mathbf{k}}) \quad (8.4)$$

As GSH possess orthogonality eq. (E.21) and symmetry eq. (E.15), eq. (7.5) can be rewritten by (8.2, 8.3, 8.4), which gives

$$\hat{\gamma}_{\mu'\mu}^m(\mathbf{k}) = \sum_{nl\nu} \hat{c}_{\mu\nu}^{mnl}(k) \sum_{\nu'\lambda'} (-)^{\nu'+\nu} \Delta \hat{\rho}_{\nu'\nu}^n(\mathbf{k}) \begin{pmatrix} m & n & l \\ \mu' & \nu' & \lambda' \end{pmatrix} R_{\lambda'0}^l(\hat{\mathbf{k}}) \quad (8.5)$$

thus the OZ equation is expended on GSHs and rotational invariants.

Here the projections $F_{\mu\nu,\chi}^{mn}$ is defined as eq. (appendix) with symmetries eq. (appendix). It's mathematically identical with [ref] but using $R_{\mu'\mu}^m = D_{\mu'\mu}^{m*}$.

It should be noticed that eq. (8.5) is reducible. Blum's χ -transform defines[3]

$$\hat{c}_{\mu\nu,\chi}^{mn}(k) = \sum_l \begin{pmatrix} m & n & l \\ \chi & -\chi & 0 \end{pmatrix} \hat{c}_{\mu\nu}^{ml}(k) \quad (8.6)$$

$\hat{c}_{\mu\nu,\chi}^{mn}(k)$ is well
the invariant in
eq. (8.1).

$$\hat{c}_{\mu\nu}^{ml}(k) = (2l+1) \sum_{\chi} \begin{pmatrix} m & n & l \\ \chi & -\chi & 0 \end{pmatrix} \hat{c}_{\mu\nu,\chi}^{mn}(k) \quad (8.7)$$

Invariants of form $F_{\mu\nu,\chi}^{mn}(k)$ have a very simple relation with its combined function $F(k, \omega_1, \omega_2)$ in intermolecular coordinate system (eq. (C.26, C.27)). In MDFT formalism, the projections of $\hat{\gamma}$ and $\hat{\rho}$ in local frame ($\omega_i = \hat{k}\Omega_i$) are

$$\hat{\gamma}'(\mathbf{k}, \omega_1) = \sum_{m\chi\mu} f_m \hat{\gamma}_{\chi\mu}^m(\mathbf{k}) R_{\chi\mu}^m(\omega_1) \quad (8.8)$$

$$\Delta\hat{\rho}'(\mathbf{k}, \omega_2) = \sum_{n\chi\nu} f_n \Delta\hat{\rho}_{\chi\nu}^n(\mathbf{k}) R_{\chi\nu}^n(\omega_2) \quad (8.9)$$

and with the rotation formula of GSH (eq. (E.23)), we have

$$\hat{\gamma}_{\chi\mu}^m(\mathbf{k}) = \sum_{\mu'} \hat{\gamma}_{\mu'\mu}^m(\mathbf{k}) R_{\mu'\chi}^m(\hat{\mathbf{k}}) \quad (8.10)$$

$$\Delta\hat{\rho}_{\nu'\nu}^n(\mathbf{k}) = \sum_{\chi} \Delta\hat{\rho}_{\chi\nu}^n(\mathbf{k}) R_{\nu'\chi}^{n*}(\hat{\mathbf{k}}) = \sum_{\chi} \Delta\hat{\rho}_{\chi\nu}^n(\mathbf{k}) (-)^{\chi+\nu'} R_{\nu'\chi}^n(\hat{\mathbf{k}}) \quad (8.11)$$

Using eq. (8.10), (8.5), eq. (8.11), eq. (8.7) and GSH products relation eq. (E.24) and 3j-symbol orthogonality eq. (E.7), we deduce that

$$\hat{\gamma}_{\chi\mu}^m(\mathbf{k}) = \sum_{n\nu} (-)^{\chi+\nu} \hat{c}_{\mu\nu,\chi}^{mn}(k) \Delta\hat{\rho}_{\chi\nu}^n(\mathbf{k}) \quad (8.12)$$

Eq. (8.12) is mathematically identical to eq. (8.1), as (to be verified with patient...)

$$\hat{\gamma}_{\chi\mu}^m(\mathbf{k}) = \sum_{l\lambda} \hat{\gamma}_{\lambda\mu,\chi}^{lm*}(k) R_{\chi\lambda}^l(\hat{\mathbf{k}}) \quad (8.13)$$

$$\hat{\rho}_{\chi\nu}^n(\mathbf{k}) = \sum_{l\lambda} \Delta\hat{\rho}_{\lambda\nu,\chi}^{ln*}(k) R_{\chi\lambda}^l(\hat{\mathbf{k}}) \quad (8.14)$$

And in this way, the integral of the angular part in eq. (7.5) is reduced to a sum of a few terms.

Table 8.1 shows some parameters linking to computing cost of different algorithms. It shows that the expansion en GSH (eq. (8.5)) projections doesn't gives an enormous reduction of FE comparing to its 6D function form (eq. (7.5)), but after the Blum's χ -transform, the OZ function is largely reduced. As the spatial convolution takes advantage of the transitional invariance r_{12} , the χ -transform in fact makes use of the rotational invariance.

8.2 FAST GENERALIZED SPHERICAL HARMONIC TRANSFORM

In the angular convolution algorithm above, the OZ equation is reduced to a few function evaluations by taking advantage of the orthogonality and symmetries of rotation invariants. It's analogue to the treatment of convolution with FFT for spatial grid, in which the generalized spherical harmonics transform (GSHT) is used as an alternative of FFT for inhomogeneous grid, and it *a priori* should be fast.

m_{\max}	0	1	2	3	4	5
N_{Θ}	1	2	3	4	5	6
N_{ang} (Gauss-Legendre)	1 (1)	18 (6)	75 (45)	196 (84)	405 (225)	726 (330)
N_{ang} (Lebedev $\times \psi$)	1 (1)	18 (6)	70 (42)	182 (78)	342 (190)	550 (250)
N_{proj}	1 (1)	10 (4)	35 (19)	84 (40)	165 (85)	286 (140)
FE for eq. (7.5)	1 (1)	324 (36)	5625 (2025)	38416 (7056)	164025 (50625)	527076 (108900)
FE for eq. (8.5)	1 (1)	262 (6)	4787 (483)	36588 (1932)	175989 (13157)	633490 (36882)
FE for eq. (8.12)	1 (1)	34 (6)	259 (75)	1092 (252)	3333 (877)	8294 (2002)

Table 8.1: Number of FE needed by OZ equation of different form for arbitrary solvent (outside the parentheses) and solvent possessing C_{2v} symmetry (inside the parentheses)

The FGHST provides a forward-backward transform between a general angular function $F(\boldsymbol{\Omega}) \equiv F(\cos \Theta, \Phi, \Psi)$ and its projections $F_{\mu' \mu}^m$ ($|\mu'|, |\mu| \leq m$)

$$F_{\mu' \mu}^m = \frac{f_m}{8\pi^2} \int d\boldsymbol{\Omega} F(\boldsymbol{\Omega}) R_{\mu' \mu}^{m*}(\boldsymbol{\Omega}) \quad (\text{forward}) \quad (8.15)$$

$$F(\boldsymbol{\Omega}) = \sum_{m, \mu', \mu} f_m F_{\mu' \mu}^m R_{\mu' \mu}^m(\boldsymbol{\Omega}) \quad (\text{backward}) \quad (8.16)$$

where $f_m = (2m+1)^{\frac{1}{2}} = \|R_{\mu' \mu}^m\|^{-1}$ is the normalization factor, and $R_{\mu' \mu}^m(\boldsymbol{\Omega})$ is the Wigner generalized spherical harmonics (Appendix [Ref]) being defined as

$$R_{\mu' \mu}^m(\boldsymbol{\Omega}) = r_{\mu' \mu}^m(\Theta) e^{-i(\mu' \Phi + \mu \Psi)} \quad (8.17)$$

which form a complete orthogonal set.

8.2.1 Equivalence of order in angular quadratures and projections

Suppose that $F(\boldsymbol{\Omega})$ is a polynomial of $\cos \Theta$, $\cos \Phi$ and $\cos \Psi$ of order n , ($n+1$ polynomial terms). To expand completely this function as shown in equation (8.16), at least $n_{\max} = n$ is needed. Then to evaluate exactly the integration in equation (8.15), at least $n+1$ for $\cos \Theta$ (Gauss-Legendre grid), $2n+1$ for Φ (equal-spaced grid), $2n+1$ for Ψ (equal-spaced grid) points of angular grid are needed (c. f. appendix B). In the case of water who possesses a C_2 symmetry $F(\Psi + \pi) = F(\Psi)$, only projections of even μ is nonzero

$$F_\mu = \int d\Psi F(\Psi) e^{i\mu\Psi} = \int d(\Psi + \pi) F(\Psi + \pi) e^{i\mu(\Psi+\pi)} = e^{i\mu\pi} \int d\Psi F(\Psi) e^{i\mu\Psi} = e^{i\mu\pi} F_\mu \quad (8.18)$$

$$F_\mu = \begin{cases} 0 & \mu = 2n+1, n \in \mathbb{Z} \\ F_\mu & \mu = 2n, n \in \mathbb{Z} \end{cases} \quad (8.19)$$

Therefore the function

$$F(\Psi) = \sum_\mu F_\mu e^{-i\mu\Psi} \quad (8.20)$$

can be rewritten as

$$F(\Psi_2/2 \equiv \Psi) = \sum_{\mu_2 \equiv \mu/2} F_{2\mu_2} e^{-i\mu_2\Psi_2} \quad (8.21)$$

As $|\mu_2| \leq n/2$, $F(\Psi_2/2 \equiv \Psi)$ is a polynomial of $\cos \Psi_2$ of order $\text{floor}(n/2) \equiv \lfloor n/2 \rfloor$, thus in the forward transform

$$F_{2\mu_2 \equiv \mu} = \int d\Psi F(\Psi) e^{i\mu\Psi} = \frac{1}{2} \int d\Psi_2 F(\Psi_2/2 \equiv \Psi) e^{i\mu_2\Psi_2} \quad (8.22)$$

the total degree $\cos \Psi_2$ polynomial in the integrand is $2 \lfloor n/2 \rfloor$, then $2 \lfloor n/2 \rfloor + 1$ points of Ψ_2 (or Ψ) is needed.

For further implementation, it's interesting to distinguish the order of quadrature m_{\max} and the order of projection n_{\max} .

8.2.2 Integration of Φ, Ψ using FFT

$$\begin{aligned} F_{\mu'\mu}^m &= \frac{f_m}{8\pi^2} \sum_{i=0}^{m_{\max}} \sum_{j=0}^{2m_{\max}} \sum_{k=0}^{2\lfloor m_{\max}/s \rfloor} w_i F(\Theta_i \Phi_j \Psi_k) R_{\mu'\mu}^{m*}(\Theta_i \Phi_j \Psi_k) \quad (\text{forward}) \\ F(\Omega) &= \sum_{m=0}^{n_{\max}} \sum_{\mu'=-m}^m \sum_{\mu=-m}^m f_m F_{\mu'\mu}^m R_{\mu'\mu}^m(\Omega) \quad (\text{backward}) \end{aligned}$$

To integrate eq. (8.15) in a direct way, $(m_{\max} + 1)(2m_{\max} + 1)(2 \lfloor m_{\max}/s \rfloor + 1) = N_\Theta N_{\Phi\Psi} = N_{FE}$ function evaluations (FE) are needed for each $F_{\mu'\mu}^m$ ($s = 1$ or 2 according to the symmetry of axe C_s), an overall $O(N_{FE}^2)$ process is needed and *vice versa*. A faster algorithm proposed by Numerical Recipes [41] suggests reducing this cost to $O(N_\Theta^2 N_{\Phi\Psi} \ln N_{\Phi\Psi} \simeq N_{FE}^{4/3})$ by fast Fourier transform.

Following this idea, eq. (8.15) can be rewritten as

$$F_{\mu'\mu}^m = \frac{f_m}{8\pi^2} \int d\Theta r_{\mu'\mu}^m(\Theta) F_{\mu'\mu}(\Theta) \simeq \frac{f_m}{8\pi^2} \sum_{i=1}^{m_{\max}+1} w_i r_{\mu'\mu}^m(\Theta_i) F_{\mu'\mu}(\Theta_i) \quad (8.23)$$

where w_i is the Gauss-Legendre quadrature weight with $m_{\max} + 1$ points ($\sum w_i = 2$), and $F_{\mu'\mu}(\Theta_i)$ the Φ, Ψ integration part with trapezoid (or Gauss-Chebyshev) quadrature

$$\begin{aligned} F_{\mu'\mu}(\Theta) &= \sum_{k'=0}^{2m_{\max}} \sum_{k=0}^{2\lfloor m_{\max}/s \rfloor} F(\Phi_{k'}, \Psi_k, \Theta) e^{i(\mu'\Phi_{k'} + \mu\Psi_k)} \\ &= \sum_{k'=0}^{2m_{\max}} \sum_{k=0}^{2\lfloor m_{\max}/s \rfloor} F(\Phi_{k'}, \Psi_k, \Theta) e^{2\pi i \mu' k' / (2m_{\max} + 1)} e^{2\pi i \mu k / (2 \lfloor m_{\max}/s \rfloor + 1)} \end{aligned} \quad (8.24)$$

which shares the same formula with an FFT-2D process.

Similarly, the backward process (8.16) can be rewritten as

$$\begin{aligned} F(\Theta, \Phi, \Psi) &= \sum_{m=0}^{n_{\max}} \sum_{\mu'=-m}^m \sum_{\mu=-m}^m f_m F_{\mu'\mu}^m R_{\mu'\mu}^m(\Omega) \\ &= \sum_{\mu'=-n_{\max}}^{n_{\max}} \sum_{\mu=-n_{\max}}^{n_{\max}} \sum_{m=\max(|\mu'|, |\mu|)}^{n_{\max}} f_m F_{\mu'\mu}^m R_{\mu'\mu}^m(\Omega) \\ &= \sum_{\mu'=-n_{\max}}^{n_{\max}} \sum_{\mu=-n_{\max}}^{n_{\max}} F_{\mu'\mu}(\Theta) e^{-i(\mu'\Phi + \mu\Psi)} \end{aligned} \quad (8.25)$$

with

$$F_{\mu'\mu}(\Theta) = \sum_{m=\max(|\mu'|, |\mu|)}^{n_{\max}} f_m F_{\mu'\mu}^m R_{\mu'\mu}^m(\Theta) \quad (8.26)$$

The FFTW3 library [15] is used for implementation, which performs discrete Fourier Transform (DFT) as defined

$$Y_k = \sum_{j=0}^{n-1} X_j e^{-2\pi i j k / n} \text{ (forward)} \quad (8.27)$$

$$X_j = \sum_{k=0}^{n-1} Y_k e^{2\pi i j k / n} \text{ (backward)} \quad (8.28)$$

It should be noticed that after a forward-backward Fourier transform, the original function is multiplied by a normalization factor N_k , which is the total number of nodes k .

For real input function Y_k ($k = 0, \dots, n - 1$), FFTW3 only output elements $k = 0, \dots, \lfloor n/2 \rfloor$ ($\lfloor n/2 \rfloor + 1$ complex numbers of X_j are stocked), with the “Hermitian” symmetry

$$Y_k = Y_{n-k}^* \quad (8.29)$$

used to regenerate elements of $k > \lfloor n/2 \rfloor$. The result X_j issue from the corresponding backward transform is purely real. As the angular function $F(\Omega)$ is real, and the GSHs possess symmetry [18, 36] of

$$\begin{aligned} r_{-\mu', -\mu}^m(\Theta) &= (-1)^{\mu' + \mu} r_{\mu' \mu}^m(\Theta) \\ R_{-\mu', -\mu}^m(\Omega) &= (-1)^{\mu' + \mu} R_{\mu' \mu}^{m*}(\Omega) \end{aligned} \quad (8.30)$$

the symmetry relation between the projections are

$$F_{-\mu', -\mu}^m = (-1)^{\mu' + \mu} F_{\mu' \mu}^{m*} \quad (8.31)$$

Therefore only $\mu \geq 0$ need to be stocked, which can be calculated with only these FFTW3 output elements. The full process of FFTW3-2D real to real transform is illustrated in figure 8.1.

dim 1		in_forward / out_backward (real)	in_backward / out_forward (complex)																		
array index		<table border="1"><tr><td>1</td><td>2</td><td>3</td><td>...</td><td>$m'+1$</td><td>$m'+2$</td><td>...</td><td>$2m'$</td><td>$2m'+1$</td></tr></table>	1	2	3	...	$m'+1$	$m'+2$...	$2m'$	$2m'+1$	<table border="1"><tr><td>1</td><td>2</td><td>3</td><td>...</td><td>$m'+1$</td></tr></table>	1	2	3	...	$m'+1$				
1	2	3	...	$m'+1$	$m'+2$...	$2m'$	$2m'+1$													
1	2	3	...	$m'+1$																	
real index		<table border="1"><tr><td>0</td><td>1</td><td>2</td><td>...</td><td>m'</td><td>$m'+1$</td><td>...</td><td>$2m'-1$</td><td>$2m'$</td></tr></table>	0	1	2	...	m'	$m'+1$...	$2m'-1$	$2m'$	<table border="1"><tr><td>0</td><td>1</td><td>2</td><td>...</td><td>m'</td></tr></table>	0	1	2	...	m'				
0	1	2	...	m'	$m'+1$...	$2m'-1$	$2m'$													
0	1	2	...	m'																	
$k \leftrightarrow \mu$		<table border="1"><tr><td>1</td><td>2</td><td>3</td><td>...</td><td>$m'+1$</td><td>$m'+2$</td><td>...</td><td>$2m'$</td><td>$2m'+1$</td></tr></table>	1	2	3	...	$m'+1$	$m'+2$...	$2m'$	$2m'+1$	<table border="1"><tr><td>0</td><td>1</td><td>2</td><td>...</td><td>m'</td></tr></table>	0	1	2	...	m'				
1	2	3	...	$m'+1$	$m'+2$...	$2m'$	$2m'+1$													
0	1	2	...	m'																	
dim 2		in_forward / out_backward (complex)	in_backward / out_forward (complex)																		
array index		<table border="1"><tr><td>1</td><td>2</td><td>3</td><td>...</td><td>$m+1$</td><td>$m+2$</td><td>...</td><td>$2m$</td><td>$2m+1$</td></tr></table>	1	2	3	...	$m+1$	$m+2$...	$2m$	$2m+1$	<table border="1"><tr><td>1</td><td>2</td><td>3</td><td>...</td><td>$m+1$</td><td>$m+2$</td><td>...</td><td>$2m$</td><td>$2m+1$</td></tr></table>	1	2	3	...	$m+1$	$m+2$...	$2m$	$2m+1$
1	2	3	...	$m+1$	$m+2$...	$2m$	$2m+1$													
1	2	3	...	$m+1$	$m+2$...	$2m$	$2m+1$													
real index		<table border="1"><tr><td>0</td><td>1</td><td>2</td><td>...</td><td>m</td><td>$m+1$</td><td>...</td><td>$2m-1$</td><td>$2m$</td></tr></table>	0	1	2	...	m	$m+1$...	$2m-1$	$2m$	<table border="1"><tr><td>0</td><td>1</td><td>2</td><td>...</td><td>m</td><td>$m+1$</td><td>...</td><td>$2m-1$</td><td>$2m$</td></tr></table>	0	1	2	...	m	$m+1$...	$2m-1$	$2m$
0	1	2	...	m	$m+1$...	$2m-1$	$2m$													
0	1	2	...	m	$m+1$...	$2m-1$	$2m$													
$k' \leftrightarrow \mu'$		<table border="1"><tr><td>1</td><td>2</td><td>3</td><td>...</td><td>$m+1$</td><td>$m+2$</td><td>...</td><td>$2m$</td><td>$2m+1$</td></tr></table>	1	2	3	...	$m+1$	$m+2$...	$2m$	$2m+1$	<table border="1"><tr><td>0</td><td>1</td><td>2</td><td>...</td><td>m</td><td>$-m$</td><td>$-m+1$</td><td>...</td><td>-1</td></tr></table>	0	1	2	...	m	$-m$	$-m+1$...	-1
1	2	3	...	$m+1$	$m+2$...	$2m$	$2m+1$													
0	1	2	...	m	$-m$	$-m+1$...	-1													

Figure 8.1: Indices arrangement in a complete forward-backward process of FFT-2D. The DFT of dim 1 (Ψ_k to μ) and dim 2 ($\Phi_{k'}$ to μ') are done consequentially and *vice versa*. Array index is the one of Fortran array, real index is the one shown in eq. (8.27) and (8.28), k and k' indices shown in the left as well as μ and μ' in the right are those in eq. (8.24) and (8.25). Here $m = m_{\max}$ and $m' = \lfloor m_{\max}/s \rfloor$.

As the output array of FFTW3 is periodic

$$e^{2\pi i \mu k / n} = e^{2\pi i (\mu - n)k / n} e^{2\pi i k} = e^{2\pi i (\mu - n)k / n} \quad (8.32)$$

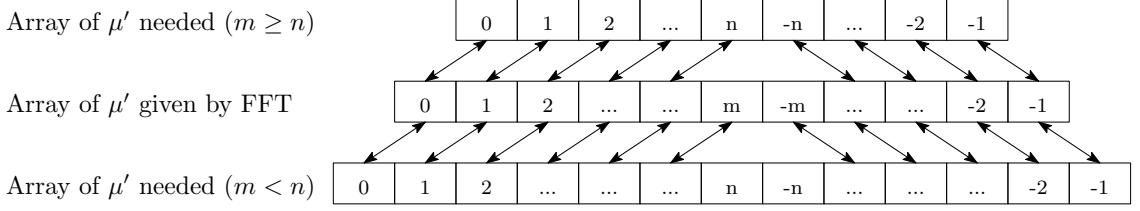


Figure 8.2: mmax to nmax

the indices $\mu = m_{\max} + 1, \dots, 2m_{\max}$ actually corresponding to $\mu = -m_{\max}, \dots, -1$. It should be noticed that eq. (8.24) and (8.25) don't possess the periodicity of eq. (8.32), just in the domain of definition of μ' and μ some intermediary functions share the same formula with FFT.

Moreover, from eq. (8.24), (8.26) and (8.31) we can verify that

$$F_{\mu'\mu}(\Theta) = F_{-\mu',-\mu}^*(\Theta) \quad (8.33)$$

The latter is used in the code because according to the definition in eq. (8.27) and (8.28), $F_{-\mu',-\mu}(\Theta)$ is calculated instead of $F_{\mu'\mu}(\Theta)$.

8.3 OPERATIONAL ALGORITHM

As described above, the whole process of γ and \mathcal{F}_{exc} functional evaluation is as shown:

Firstly, the solvent density variable $\Delta\rho(\mathbf{r}, \Omega)$ is expended on generalized spherical harmonics

$$\Delta\rho_{\mu'\mu}^m(\mathbf{r}) = \frac{f_m}{8\pi^2} \int d\Omega \Delta\rho(\mathbf{r}, \Omega) R_{\mu'\mu}^{m*}(\Omega) \quad (8.34)$$

Then the Fourier transform of these projections are computed

$$\hat{\Delta\rho}_{\mu'\mu}^m(\mathbf{k}) = \int d\mathbf{r} \Delta\rho_{\mu'\mu}^m(\mathbf{r}) e^{-i\mathbf{r}\cdot\mathbf{k}} \quad (8.35)$$

Then the projections in k-frame are then rotated into local coordinates system along the unit vector $\hat{\mathbf{k}}$

$$\hat{\Delta\rho}_{\chi\mu}^m(\mathbf{k}) = \sum_{\mu'} \hat{\Delta\rho}_{\mu'\mu}^m(\mathbf{k}) R_{\mu'\chi}^m(\hat{\mathbf{k}}) \quad (8.36)$$

where the evaluation of rotation matrix elements by recurrence is detailed [in appendix](#).

Then computing the OZ equation with Blum's reduction

$$\hat{\gamma}_{\chi\mu}^m(\mathbf{k}) = \sum_{n,\nu} (-1)^{\chi+\nu} \hat{c}_{\mu\nu,\chi}^{mn}(\mathbf{k}) \Delta\hat{\rho}_{\chi\nu}^n(\mathbf{k}) \quad (8.37)$$

Then the γ projections are then transformed back to global coordinates system

$$\hat{\gamma}_{\mu'\mu}^m(\mathbf{k}) = \sum_{\chi} \hat{\gamma}_{\chi\mu}^m(\mathbf{k}) R_{\mu'\chi}^{m*}(\hat{\mathbf{k}}) \quad (8.38)$$

Then the inverse Fourier transform of these projections

$$\gamma_{\mu'\mu}^m(\mathbf{r}) = \int d\mathbf{k} \hat{\gamma}_{\mu'\mu}^m(\mathbf{k}) e^{i\mathbf{r}\cdot\mathbf{k}} \quad (8.39)$$

Then the function in angular frame can thus be rebuilt

$$\gamma(\mathbf{r}, \Omega) = \sum_{m,\mu',\mu} f_m \gamma_{\mu'\mu}^m(\mathbf{r}) R_{\mu'\mu}^m(\Omega) \quad (8.40)$$

Finally, the functional \mathcal{F}_{exc} is computed by

$$\mathcal{F}_{\text{exc}} = \frac{1}{2} \int d\mathbf{r} d\Omega \Delta\rho(\mathbf{r}, \Omega) \gamma(\mathbf{r}, \Omega) \quad (8.41)$$

8.3.1 Reduction by symmetry

A further reduction of computing cost can be made by performing approximately only a half of operations, thanks to the symmetric relations between the projections.

In eq. (8.34), $\Delta\rho(\mathbf{r}, \Omega)$ is real. Thanks to the property of GSH (eq. (appendix))

$$R_{\mu'\mu}^m(\Omega) = (-)^{\mu'+\mu} R_{-\mu'-\mu}^{m*}(\Omega)$$

we find

$$\Delta\hat{\rho}_{\mu'\mu}^m(\mathbf{r}) = (-)^{\mu'+\mu} \Delta\hat{\rho}_{-\mu',-\mu}^{m*}(\mathbf{r}) \quad (8.42)$$

therefore only $\mu' > 0$ or $\mu > 0$ is needed to generate all information.

When $\Delta\hat{\rho}_{\mu'\mu}^m(\mathbf{r})$ is transformed into k -space, replace eq. (8.42) into eq. (8.35) gives

$$\Delta\hat{\rho}_{\mu'\mu}^m(\mathbf{k}) = (-)^{\mu'+\mu} \Delta\hat{\rho}_{-\mu',-\mu}^{m*}(-\mathbf{k}) \quad (8.43)$$

thus only the projections of $\mu' > 0$, $\mu > 0$ or \mathbf{k} where one of the dimensions $k_i > 0$ are independent.

The rotation to local frame is reignited by the relation (prove?)

$$R_{\mu'\chi}^m(\hat{\mathbf{k}}) = (-)^m R_{\mu',-\chi}^m(-\hat{\mathbf{k}}) = (-)^{m+\mu'+\chi} R_{-\mu',\chi}^m(-\hat{\mathbf{k}}) \quad (8.44)$$

which gives

$$\Delta\hat{\rho}_{\chi\mu}^m(\mathbf{k}) = (-)^{m+\mu+\chi} \Delta\hat{\rho}_{\chi,-\mu}^{m*}(-\mathbf{k}) \quad (8.45)$$

Thanks to the symmetry (eq. (appendix))

$$\hat{c}_{\mu\nu,\chi}^{mn}(k) = (-)^{m+n+\mu+\nu} \hat{c}_{\underline{\mu}\underline{\nu},\chi}^{mn*}(k) \quad (8.46)$$

$$\hat{c}_{\mu\nu,\chi}^{mn}(k) = (-)^{m+n} \hat{c}_{\nu\mu,\chi}^{nm}(k) \quad (8.47)$$

$$\hat{c}_{\mu\nu,\chi}^{mn}(k) = \hat{c}_{\underline{\mu}\underline{\nu},\chi}^{mn}(k) = (-)^{m+n} \hat{c}_{\mu\nu,\chi}^{mn*}(k) \quad (8.48)$$

$\hat{\gamma}_{\chi\mu}^m(\mathbf{k})$ possesses the same symmetry. Thus OZ equation can be reduced by a factor two.

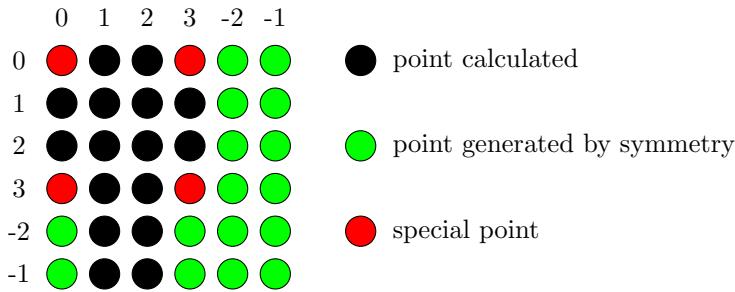


Figure 8.3: Distribution of points to be calculated according to symmetry in a 2D plan

8.3.2 Commutativity between operations

As mentioned in the operational algorithm, three types of operation are being done before and after OZ equation. They are

1. Fast Fourier transform for 3-dimensional spatial grid (FFT3D): implemented by package FFTW3 [15], mathematically leading to no accuracy lost.

2. Fast generalized spherical harmonics transform (FGSHT): have real or complex input, is exact if $F(\Omega)$ is a polynomial of $\cos \Theta$, $\cos \Phi$ and $\cos \Psi$ of order at most m_{\max} .
3. Rotation between laboratory coordinate system and local system linked to vector \mathbf{k} (RotS): can be done for both function and projections. It introduces a minus error in accuracy at origin and border of the box, it will be discussed in next chapter.

Their commutativity is shown in figure 8.4.

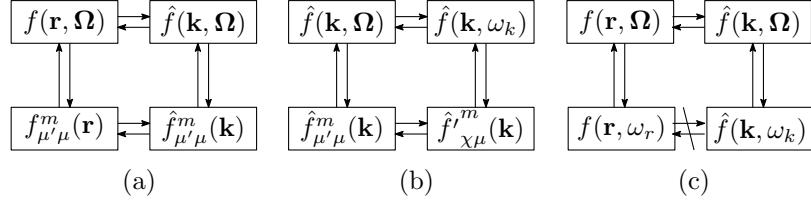


Figure 8.4: Commutativity of operations. (a) FFT3D and FGSHT; (b) RotS and FGSHT; (c) FFT3D and RotS

8.3.2.1 FFT3D and FGSHT

The FFT3D

$$f(\mathbf{r}) = \int \hat{f}(\mathbf{k}) e^{i\mathbf{k}\cdot\mathbf{r}}$$

$$\hat{f}(\mathbf{k}) = \int f(\mathbf{r}) e^{-i\mathbf{k}\cdot\mathbf{r}}$$

does not depend on the angular part of function, and

$$f(\Omega) = \sum_{m\mu'\mu} f_m f_{\mu'\mu}^m R_{\mu'\mu}^m(\Omega)$$

$$f_{\mu'\mu}^m = \int f_m f(\Omega) R_{\mu'\mu}^m(\Omega)$$

does not depend on the spatial part of function. The two operations are commutative.

8.3.2.2 FGSHT and coordinate rotation

Function \hat{f}' intermolecular frame can be deduced from the function \hat{f} in laboratory frame

$$\hat{f}(\mathbf{k}, \Omega) = \hat{f}'(\mathbf{k}, \omega_k)$$

where the two can both be expended on GSH

$$\hat{f}(\mathbf{k}, \Omega) = \sum_{m\mu'\mu} f_m \hat{f}_{\mu'\mu}^m(\mathbf{k}) R_{\mu'\mu}^m(\Omega)$$

$$\hat{f}'(\mathbf{k}, \omega_k) = \sum_{m\chi\mu} f_m \hat{f}'_{\chi\mu}^m(\mathbf{k}) R_{\chi\mu}^m(\omega_k)$$

And the relation between projections are simple

$$f'_{\chi\mu}^m(\mathbf{k}) = \sum_{\mu'} R_{\mu'\chi}^m(\hat{\mathbf{k}}) f_{\mu'\mu}^m(\mathbf{k})$$

$$f_{\mu'\mu}^m(\mathbf{k}) = \sum_{\chi} R_{\mu'\chi}^{m*}(\hat{\mathbf{k}}) f'_{\chi\mu}^m(\mathbf{k})$$

The two operations are commutative.

8.3.2.3 Coordinate rotation and FFT3D

The rotation from $f(\mathbf{r}, \Omega)$ to $f(\mathbf{r}, \omega)$ depends on the vector \mathbf{r} , of which the information is totally lost after FFT3D. The rotation from $f(\mathbf{k}, \Omega)$ to $f(\mathbf{k}, \omega)$ can only depend on the vector \mathbf{k} , they are not the same rotation. Non-commutative.

9

FREE ENERGY AND RELATED THERMODYNAMIC QUANTITIES

The solvation free energy is the most important property that we seek, as shown in previous sections, it can be calculated by the minimization of free energy functional $\mathcal{F}[\rho]$. Here is a discussion about some corrections needed for charged solute and some related thermodynamic quantities that can be obtained directly from the solvation free energy.

The solvation properties often involve the three-dimensional microscopic structure of the solvent around the dissolved molecule, as well as thermodynamics quantities such as the enthalpy, entropy, and free energy of solvation. [ref gubbins] pKa [ref]

The solvation properties can be found if the two principle properties are accurately calculated, free energy and structure. In this thesis, we focus on this two aspect.

9.1 FREE ENERGY CORRECTION FOR SINGLE ION

In the calculation of external potential as well as the total solvation free energy, the use of different convention can lead to a charge-independent offset, which introduces error for charged solutes [25, 28, 29]. This offset is mainly built by two sources: (1) due to the use of a finite system size, in our case, is a system with cubic periodic boundary conditions, where presents artificial interactions between the ion and its own periodic copies, as well as between the solvent and the periodic copies of the ion (Type-B in [29]); (2) due to the choice of convention for summing up the contributions of solvent charges to the electrostatic potential in the sample system (Type-C in [29]).

9.1.1 Correction of type B

Type B correction should be added for systems with finite size or periodic boundary conditions, accounting for the error in the solvent polarization

$$\Delta G_B = \frac{1}{8\pi\varepsilon_0} \left(1 - \varepsilon^{-1}\right) \frac{q^2}{L} \left[\xi + \frac{4\pi}{3} \left(\frac{R_I}{L}\right)^2 - \frac{16\pi}{45} \left(\frac{R_I}{L}\right)^5 \right] \quad (9.1)$$

where

ε_0 is the vacuum permittivity;

ε is the solvent permittivity (dielectric constant);

q is the solute charge;

L is the box length;

R_I is the ionic radius;

ξ is the energy per particle in a simple cubic lattice, $\xi \simeq -2.837297$ [37].

Another way to evaluate this error is to make a numerical extrapolation of the inverse of the box size ($1/L$), it's more accurate, but asking for much more calculation.

As R_I is significantly smaller than the size of the computational box, i. e. $R_I \ll L$, its quadratic as well as higher order of (R_I/L) is considered negligible, thus eq. (9.1) becomes

$$\Delta G_B = \frac{\xi}{8\pi\varepsilon_0} \left(1 - \varepsilon^{-1}\right) \frac{q^2}{L} \quad (9.2)$$

It links to Born correction.

9.1.2 Correction of type C

Type-C corrections is needed when the systems to be compared use different electrostatic summation schemes: on the basis of point charges within entire solvent molecules (M scheme) or on the basis of individual point charges (P scheme), shown in figure 9.1 (c) and (d), which brings an fixed free energy difference at the boundary.

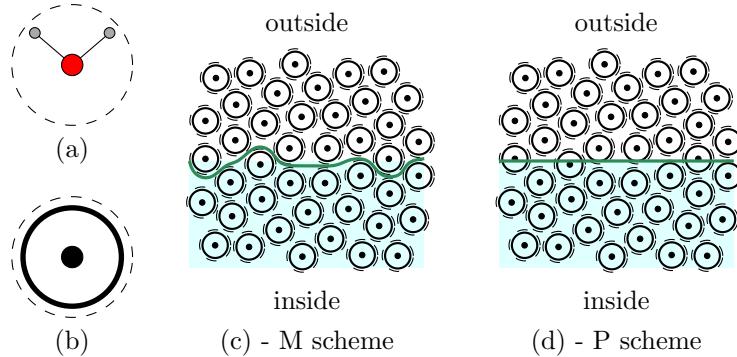


Figure 9.1: IQ model and summation scheme. (a) The solvent molecule. (b) The equivalent isotropic quadrupole (IQ) fluid model. (c) In the M scheme, one evaluates the Coulombic potential generated by the solvent charges belonging to all molecules within the boundary. (d) In the P scheme, one evaluates the Coulombic potential generated by all solvent charges within the boundary.

It can be deduced analytically by considering the solvent as a **canonical ensemble** under the orientational disorder limit (ODL) [28], which becomes an isotropic quadrupole (IQ) fluid, whose solvent molecule (figure 9.1 (b)) possesses the same quadrupole trace γ

$$\gamma = \text{tr}(\mathcal{Q}) = Q_{xx} + Q_{yy} + Q_{zz} \quad (9.3)$$

γ is also somewhere called the spheropole moment [33, 44], which is the spherical component of the quadrupole moment, and is invariant with respect to rotations.

where the quadrupole moment of the solvent molecule can be calculated by its definition [43]

$$\mathcal{Q}_{ij} = \int_V r_i r_j \rho(\mathbf{r}) dv = \sum_{\alpha=1}^N q^{(\alpha)} r_i^{(\alpha)} r_j^{(\alpha)} \quad (9.4)$$

It can be shown that charge density of solvent located within the boundary of sample system vanishes everywhere, except at the boundary in the M scheme, which results in a uniform normal surface polarization. The correction needed for **M scheme** is

$$\Delta G_C = -q \left(1 - \frac{4\pi R_I^3}{3L^3} \right) \Delta \Phi_{\text{ODL}} \quad (9.5)$$

where $\Delta \Phi_{\text{ODL}} = (6\epsilon_0)^{-1} \eta \gamma$, η being the solvent number density.

The same way, when we consider $R_I \ll L$, eq. (9.5) becomes

$$\Delta G_C = -(6\epsilon_0)^{-1} \eta \gamma q \quad (9.6)$$

9.2 SOME RELATED THERMODYNAMIC QUANTITIES

Thermodynamic quantities such as the internal energy, pressure, compressibility and heat capacity are obtained as derivatives of the classical partition function. [Evans poly]

Structure and Thermodynamic Properties of Bulk Liquids
pressure is (virial pressure equation)

$$p = \rho k_B T - \frac{\rho^2}{2} \int d\mathbf{r} g(r) \frac{r}{3} \frac{du(r)}{dr} \quad (9.7)$$

the The Gibbs free energy G is simply [Evans 1979]

$$G = \mu \int d\mathbf{r} \rho(\mathbf{r})$$

9.2.1 Solubility

9.2.2 Pressure

enthalpy, entropy [3] pH

10

SOLVATION STRUCTURE

In MDFT, all the information about solvation structure can be deduced from the solvent density $\rho(\mathbf{r}, \Omega)$. This section presents some examples of structure which are used in later chapters.

10.1 RADIAL DISTRIBUTION FUNCTION AND SITE-SITE DISTRIBUTION FUNCTION

Radial distribution function (**RDF!**) and site-site distribution function

It should be the same with

10.2 RADIAL POLARIZATION FUNCTIONS

It should be the same with

10.3 ROTATIONAL INVARIANT EXPANSION

appendix C

10.4 REBUILT OF DENSITY IN A CERTAIN ORIENTATION

appendix C

10.4.1 Radical distribution function of numeric density

Content.

10.4.2 Radical distribution function of polarization

$$F(\mathbf{r}, \Omega) = \sum_{nl\nu} F_{0\nu}^{0nl}(r) \Phi_{0\nu}^{0nl}(\mathbf{r}, \Omega)$$

conversely

$$F_{0\nu}^{0nl}(r) = \int d\hat{\mathbf{r}} d\Omega F(\mathbf{r}, \Omega) \Phi_{0\nu}^{0nl*}(\mathbf{r}, \Omega) / \int d\hat{\mathbf{r}} d\Omega \left\| \Phi_{0\nu}^{0nl}(\mathbf{r}, \Omega) \right\|^2$$

with

$$\Phi_{0\nu}^{0nl}(\mathbf{r}, \Omega) = f_n \sum_{\nu'} \begin{pmatrix} 0 & n & l \\ 0 & \nu' & -\nu' \end{pmatrix} R_{\nu'\nu}^n(\Omega) R_{-\nu',0}^l(\hat{\mathbf{r}})$$

in particular

$$\begin{aligned}\Phi_{00}^{011}(\mathbf{r}, \boldsymbol{\Omega}) &= \sqrt{3} \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 0 \end{pmatrix} R_{00}^1(\boldsymbol{\Omega}) R_{00}^1(\hat{\mathbf{r}}) \\ &+ \sqrt{3} \begin{pmatrix} 0 & 1 & 1 \\ 0 & 1 & -1 \end{pmatrix} R_{10}^1(\boldsymbol{\Omega}) R_{-10}^1(\hat{\mathbf{r}}) \\ &+ \sqrt{3} \begin{pmatrix} 0 & 1 & 1 \\ 0 & -1 & 1 \end{pmatrix} R_{-10}^1(\boldsymbol{\Omega}) R_{10}^1(\hat{\mathbf{r}})\end{aligned}$$

or

$$\begin{aligned}\Phi_{00}^{011}(\mathbf{r}, \boldsymbol{\Omega}) &= -3R_{00}^1(\boldsymbol{\Omega}) R_{00}^1(\hat{\mathbf{r}}) \\ &+ 3R_{10}^1(\boldsymbol{\Omega}) R_{-10}^1(\hat{\mathbf{r}}) \\ &+ 3R_{-10}^1(\boldsymbol{\Omega}) R_{10}^1(\hat{\mathbf{r}})\end{aligned}$$

noting

$$R_{00}^1(\boldsymbol{\Omega}) = \cos \theta$$

$$R_{10}^1(\boldsymbol{\Omega}) = -\frac{1}{\sqrt{2}} \sin \theta e^{-i\phi}$$

one finds

$$\begin{aligned}\Phi_{00}^{011}(\mathbf{r}, \boldsymbol{\Omega}) &= -3R_{00}^1(\boldsymbol{\Omega}) R_{00}^1(\hat{\mathbf{r}}) \\ &+ 3R_{10}^1(\boldsymbol{\Omega}) R_{-10}^1(\hat{\mathbf{r}}) \\ &+ 3R_{-10}^1(\boldsymbol{\Omega}) R_{10}^1(\hat{\mathbf{r}}) \\ &= -3\boldsymbol{\Omega} \cdot \hat{\mathbf{r}}\end{aligned}$$

such that

$$\begin{aligned}F_{00}^{011}(r) &= -\frac{1}{3} \int \\ &= P(\mathbf{r}) \cdot \hat{\mathbf{r}}\end{aligned}$$

proportionality coefficient to be determined precisely... One thus gets the radial projection of the polarization.

Content.

Chapter III

IMPLEMENTATION

The code MDFT developed in this thesis is based on the branch master of Git project MDFT (<https://github.com/maxlevesque/MDFT/>), version [Fri Jun 20 19:05:52 2014 +0200]. All the implementations are run on **POINCARE** machines of IDRIS, which involved two kinds of machines:

poincare[001-092]: 2 processors Sandy Bridge E5-2670 (2.60GHz, 8 cores per processor, being 16 cores per node); 32 GB of memory per node.

poincarebig[01-02]: 4 processors AMD Opteron 6282 (2.60GHz, 16 cores per processor, being 64 cores per node); 128 GB of memory per node.

The former is used for regular calculation, whose memory hierarchy is shown in figure 10.1. The later is only used in the evaluation of accuracy, in case of memory leak in the former.

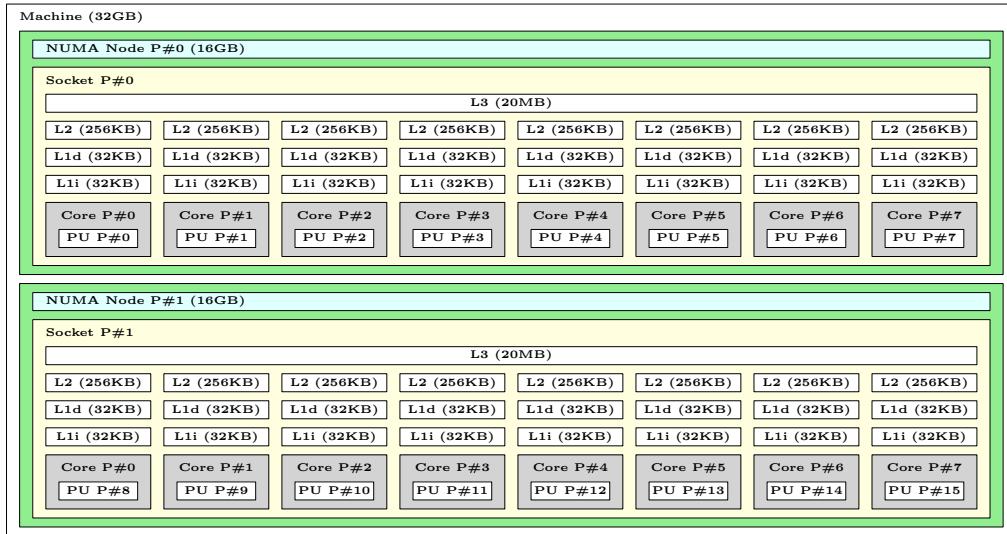


Figure 10.1: Structure of a POINCARE node

Section 11,

Section 12 discussion on precision,

Section 13 computing performance and memory limits,

Section 14 OpenMP, MPI giving the possibility to go beyond the memory limit. Due to the complexity of minimizer L-BFGS, this process is only added on the part of \mathcal{F}_{exc} evaluation. Tests of performance stability with respect to both threads and nodes are made.

ALGORITHMS AND BRANCHES

According to the commutativity of operations (see §8.3.2), the only possible algorithms to evaluate $\gamma(\mathbf{r}, \Omega)$ from $\Delta\rho(\mathbf{r}, \Omega)$ are shown in the figure 11.1.

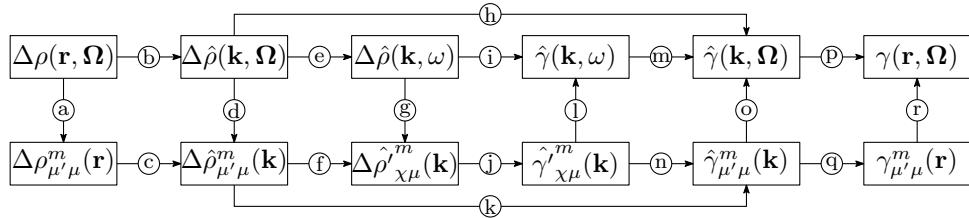


Figure 11.1: Possible algorithms for γ evaluation

Several branches are built to test and compare between algorithms, which are shown below in table 11.1 and will be detailed in the following context.

METHOD	SUB-METHOD	DESCRIPTION	THEORY
reference	dipole	calculate $n(r)$ and $P(r)$ separately	§5 [ref]
naive	standard	use $c_{\mu\nu,\chi}^{mn}(k)$ as input DCF	§7.2.2
	zero-order	use $\hat{c}(k, \omega_1, \omega_2)$ and take the nearest point	§7.1.1
	interpolation	use $\hat{c}(k, \omega_1, \omega_2)$ with linear interpolation	§7.1.2
	dipole	use c_S, c_Δ, c_D issue from [ref]	§7.2.1
	nmax1	use $c_S, c_\Delta, c_D, c_\pm$ issue from [ref]	§7.2.1
convolution	standard	algorithm with symmetry reduction	§8.3.1
	asymm	algorithm without symmetry reduction	§8.3.1
	pure-angular	inverse FFT and FGSHT	§11

These branches should give numerically the same result in certain conditions, that will be discussed in later sections.

Table 11.1: Branch option in MDFT

11.1 BRANCHES "NAIVE"

Branches **naive** are the algorithms mentioned in section 7, which go through the path

$$(b) \rightarrow (h) \rightarrow (p)$$

in figure 11.1, calculating directly $\hat{\gamma}(k, \Omega)$ from $\Delta\hat{\rho}(k, \Omega)$. The difference between branches is the way to calculate $\hat{c}(k, \Omega_1, \Omega_2)$. Branch **naive_standard** use $c_{\mu\nu,\chi}^{mn}(k)$ as input DCF. Branch **naive_zero-order** and **naive_interpolation** using $\hat{c}(k, \omega_1, \omega_2)$ with zero-order and linear interpolation, where the former is rejected in the implementation due to a lack of precision (appendix F).

The discussion of the accuracy to transform c projection to c table is given in section

...

As one is constant other has a $O(N^3)$ dependence on n_{\max} , the discussion between the computing performance and memory usage with respect to n_{\max} and it will be given in section ...

11.2 BRANCHES "CONVOLUTION"

Branches **convolution_asymm** and **convolution_standard** are operational algorithms of angular convolution show in section 8, which go through the path

$$(a) \rightarrow (c) \rightarrow (f) \rightarrow (j) \rightarrow (n) \rightarrow (q) \rightarrow (r)$$

Branches **convolution_asymm** uses the original operational algorithm (§8.3) without symmetry reduction (§8.3.1), and **convolution_standard** without it.

Branch **convolution_pure_angular** goes through the path

$$(b) \rightarrow (d) \rightarrow (f) \rightarrow (j) \rightarrow (n) \rightarrow (o) \rightarrow (p)$$

which inverse the first and last two steps of the two algorithms mentioned above.

mathematically

a discussion the comparison of computing speed in in section ...

11.3 TESTING BRANCHES FOR NMAX=1

Branches **naive_dipole**, **naive_nmax1** pass by $(b) \rightarrow (h) \rightarrow (p)$, using DCF separately in reference [ref] and [ref], whose slight difference is shown in §6.3.4. Branch **reference_dipole** use DCF in [ref], which is the original method in MDFT to calculate \mathcal{F}_{exc} via multipole expansion. In addition with branch **convolution_standard**, which can also use the two DCF mentioned above, a test of validation can be performed (§), which should at any case exactly numerically the same if the same DCF is used.

11.4 OTHER PATHS

There will also be a discussion of other paths, such as those passes by (i) and (k) , of their accuracy (§), and of the computing cost (§).

12

NUMERICAL AND PHYSICAL ACCURACY

This chapter gives a systematic comparison between algorithms for the evaluation of γ in terms of accuracy. The different methods should lead to mathematical equivalent results. The comparison for a series of CH4 with IEM and MD is done as validation of method.

12.1 GENERALIZED SPHERICAL HARMONICS TRANSFORM

12.1.1 FFT

FFT3D is implemented by package FFTW3 [15] with discrete Fourier transform (DFT) defined as:

$$Y_k = \sum_{j=0}^{n-1} X_j e^{-2\pi i j k / n} \quad (\text{forward}) \quad (12.1)$$

$$X_j = \sum_{k=0}^{n-1} Y_k e^{2\pi i j k / n} \quad (\text{backward}) \quad (12.2)$$

Mathematically, these transforms lead to no accuracy lost. It should be noticed that after a forward-backward Fourier transform, the original function is multiplied by a normalization factor N_k , which is the total number of nodes k . The numerical tests for selected functions $f(x) \in [-5, 5]$ (100 values) are shown in table 12.1.

$f(x)$	1	x^3	e^x	random number
E_a^{\max}	0	$2.84 \cdot 10^{-14}$	$3.32 \cdot 10^{-14}$	$1.80 \cdot 10^{-16}$

Table 12.1: Maximum absolute error introduced by a forward-backward (real-complex-real) DFT1D process on a 100-value double precision array using FFTW3 package

We can see that the errors are about machine precision.

The algorithmic complicity of FFT is about $\mathcal{O}(N_k \ln N_k)$ [41], and the computing time will be discussed in later sessions.

mmax/nmax

12.1.2 m_{\max} and n_{\max} of projections

The numerical error tests of a forward-backward GSHT process with different order m_{\max} (m in table) of GSH and order n of quadrature is shown in table 12.2.

It is shown that

It should be noticed that the tested functions are theoretically the same if the order of quadrature / GSH is sufficient. The real case of ρ is in later session.

12.1.3 From ρ to γ

Unphysical rho after transform, Gamma more smooth, Orientation

$m \setminus n$	0	1	2	3	4	$m \setminus n$	0	1	2	3	4
0						0					
1						1					
2						2					
3						3					
4						4					

(a) $f(\Omega) = 1$

$m \setminus n$	0	1	2	3	4	$m \setminus n$	0	1	2	3	4
0						0					
1						1					
2						2					
3						3					
4						4					

(b) $f(\Omega) = \cos 3\Theta$

$m \setminus n$	0	1	2	3	4	$m \setminus n$	0	1	2	3	4
0						0					
1						1					
2						2					
3						3					
4						4					

(c) $f(\Omega) = \cos 3\Phi$

$m \setminus n$	0	1	2	3	4	$m \setminus n$	0	1	2	3	4
0						0					
1						1					
2						2					
3						3					
4						4					

(d) $f(\Omega) = \cos \frac{3}{2}\Psi$

Table 12.2: Logarithm of maximum absolute error $\lg(E_a^{\max})$ introduced by a forward-backward GSHT process.

12.2 COMPARISON BETWEEN BRANCHES

12.2.1 Error evaluation in $\hat{c}(\mathbf{k}, \Omega_1, \Omega_2)$ calculation

In appendix F, we compared the error introduced by the interpolation strategy of $\hat{c}(\mathbf{k}, \Omega_1, \Omega_2)$ calculation from the intermolecular $\hat{c}(k, \omega_1, \omega_2)$. Here, we want to compare the error introduced by **naive_interpolation** and **naive_standard** concerned both $\hat{c}(\mathbf{k}, \Omega_1, \Omega_2)$ and the free energy and structure issued from a single-k calculation.

As shown in table ,

12.2.2 A single k-kernel

As shown in figure 12.1, four paths are presented to be tested, (1) (2) (3) (4)

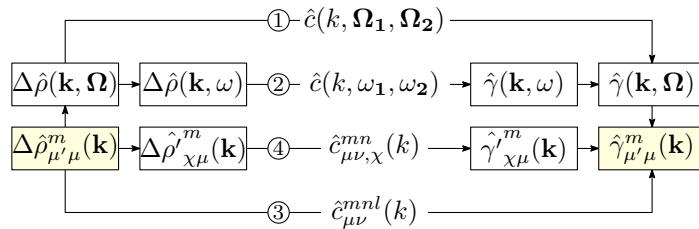


Figure 12.1: Schema of a k-kernel test

The result of error in energy for the result $\hat{\gamma}_{\mu'\mu}^m(\mathbf{k})$ with certain \mathbf{k} is presented in table. As shown, there is no accuracy lost for different paths in both 4 cases of \mathbf{k} . This means, the

final result of energy and structure is independent to the choice of path inside a k-kernel, if $\Delta\hat{\rho}(\mathbf{k}, \Omega)$ is a polynomial of ... as discussed in §12.1.

\mathbf{k}	E_2^{\max}	E_3^{\max}	E_4^{\max}
(2, 3, 5)			
(4, 0, 1)			
(0, 0, 0)			
(0, 0, 1)			

Table 12.3: Maximum absolute error introduced by paths (1) ... (4) shown figure 12.1, using an input $\Delta\hat{\rho}_{\mu'\mu}^m(\mathbf{k})$ issue of MDFT minimization. \mathbf{k} is of unity [\AA^{-3}]. For special case $\mathbf{k} = (0, 0, 0)$, ... E_i^{\max} is the absolute difference between path (1) and path (i).

12.2.3 k -border effect

	<code>convolution_standard</code>	<code>convolution_asymm</code>	<code>convolution_pure-angular</code>
$E_{\gamma}^{\max}(64)$			
$\mathcal{F}_{\text{exc}}(64)$			
$E_{\gamma}^{\max}(65)$			
$\mathcal{F}_{\text{exc}}(65)$			

Table 12.4: Maximum absolute error introduced by different branches in calculated $\gamma(\mathbf{r}, \Omega)$ and the difference in \mathcal{F}_{exc} from a given $\Delta\hat{\rho}_{\mu'\mu}^m(\mathbf{r})$ compared to `naive_standard` before border correction for a 64^3 and 65^3 spatial grid, $n_{\max} = 5$.

standard, asymm and pure-angular different because...

In MDFT, the symmetry

$$\Delta\hat{\rho}_{\chi\mu}^m(\mathbf{k}) = (-)^{m+\mu+\chi}\Delta\hat{\rho}_{\chi,-\mu}^{m*}(-\mathbf{k}) \quad (12.3)$$

is generated by two symmetries

$$\Delta\hat{\rho}_{\mu'\mu}^m(\mathbf{k}) = (-)^{\mu'+\mu}\Delta\hat{\rho}_{-\mu',-\mu}^{m*}(-\mathbf{k}) \quad (12.4)$$

$$R_{\mu'\chi}^m(\hat{\mathbf{k}}) = (-)^{m+\mu'+\chi}R_{-\mu',\chi}^m(-\hat{\mathbf{k}}) \quad (12.5)$$

For the points “at border”, it’s to say that after the FFT where the point having $\pm k_i = k_i^{\max}$, $i = 1, 2, 3$, for example for k_1 ,

$$\Delta\hat{\rho}_{\mu'\mu}^m(\pm k_1, k_2, k_3) = \Delta\hat{\rho}_{\mu'\mu}^m(k_1^{\max}, k_2, k_3)$$

is naturally put in the same array by FFT for the grids having even number.

But

$$R_{-\mu',\chi}^m(-\hat{\mathbf{k}} \equiv (-k_1, -k_2, -k_3)) \neq R_{\mu',\chi}^m(k_1^{\max}, -k_2, -k_3)$$

Thus the symmetries (12.5) and (12.3) are not respected for these points

In the backward process, if we make sense of all the $\gamma_{\mu'\mu}^m(\mathbf{k})$, as

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.

$$\gamma_{\mu'\mu}^m(-\hat{k} \equiv (-k_1, -k_2, -k_3)) \neq \gamma_{\mu'\mu}^m(k_1^{\max}, -k_2, -k_3)$$

the symmetry

$$\gamma_{\mu'\mu}^m(\mathbf{k}) = (-)^{\mu'+\mu} \gamma_{-\mu',-\mu}^{m*}(-\mathbf{k}) \quad (12.6)$$

is not respected totally, and this imposes that $\gamma_{\mu'\mu}^m(\mathbf{r})$ have a imaginary part. In the version FFTW3, we keep only the part of none-negative \mathbf{k} or none-negative, supposing that the part we omit respects the symmetry.

0	1	...	$k-1$	k	$-k+1$...	-1
				-k			

Figure 12.2: k-border effet

convolution_standard	convolution_asymm	convolution_pure-angular
$E_\gamma^{\max}(65)$		
$\mathcal{F}_{\text{exc}}(65)$		

Table 12.5: Maximum absolute error introduced by different branches after border correction

the mysterious error between **naive_standard** and the **convolution** branches cannot explained, this implied that there is probably a bug in **naive_standard**.

12.3 INTRINSIC VARIATION OF FREE ENERGY

Before study of free energy dependence on angular algorithms, we are interested in the grid dependance, with can have an influence in the follow tests.

12.3.1 Spatial grid: length and resolution

12.3.2 Angular grid: effect of psi

12.4 SERIES OF CHARGED LJ CENTRE

charged CH₄ centre

here we use 298K according to habitude instead of 303K recommended in reference [ref].

12.4.1 Box length dependance and charge dependance of free energy

As discussed in section 9, for single ions, two types of corrections need to be added on the free energy, which depend on the box length and and charge of the ion. To verify these dependence, we implement a systematic calculation from charge shown in figure

Figure 12.3: structure of gamma in rotational invariants

charge	sigma []	epsilon []	x [Å]	y [Å]	z [Å]	temperature [K]	number density of solvent
-1.0 to 1.0	3.73	1.23	0	0	0	298	0.0332891

Table 12.6: Parameters of charged Lenard-Jones centre (modified from CH₄)

Figure 12.4: free energy (without correction) of charged CH₄ centre (-1.0 to 1.0) with respect to the box length

Continuum model correction at boundary

It satisfies the Born model.

memory leak can cause divergence.

12.4.2 Comparison with IET

Old data in appendix G, which use 2m phi. It gives quite similar result, which shows the insensibility of grid.

12.4.3 Comparison with DM

12.5 PREMIER CONCLUSION

Capable to produce the same result with IEM, but have more ability to calculate 3D molecules which is not suitable for spherical coordinates.

Figure 12.5: Parabolic charge dependence of free energy of CH₄ centre series

Figure 12.6: Linear dependence of charge in the comparison to IET. Old algorithm / new algorithm, without correction

Figure 12.7: Comparison to IET. Old algorithm / new algorithm, with P-scheme correction

COMPUTING PERFORMANCE OF SEQUENTIAL CODE

This section evaluates the computing performance of the code without parallelization. Our goal is to show that the new algorithm of angular convolution is faster than the old naive one, and the huge amount of simulation has shown that it is absolutely the case. But a raw result, where the implementation goes for an indefinite number of iterations during minimization, cannot give a proper and systematic performance evolution. This gives the propose of this section.

As discussed in appendix A, two main factors give influence to the performance of a sequential code: the algorithm complexity, and the memory delay. To study the algorithm complexity, testing with respect to parameters is done to some simple but important components. The result can match the theoretical algorithm complexity, or completely different due to the overhead of function calling or the inhomogeneity of memory access. Etude of this small parts permits a further understanding of the entire code.

13.1 FFT

The **FFT!** play a great role in the implementation, which is used by the spatial convolution and the FGSHT process. As shown in figure 13.1a, the dependance on $O(N \log_2 N)$ doesn't totally exist, due to the algorithm of FFT [ref dft]. To compare between the algorithms involved in this thesis, we are not really interested in computing performance with respect to the number of spatial grid, but the **FFT!** used in FGSHT process (figure 13.1b).

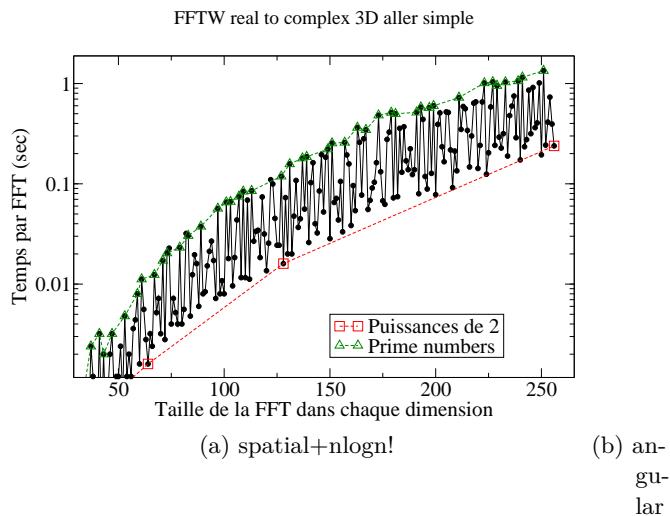


Figure 13.1: timing FFT

13.2 FGSHT

13.2.1 Computing time of GSHT and FGSHT

To serve as an alternative of FFT for angular grid, the algorithmic complicity of GSHT should be at least less than $O(N_\Omega^2)$, where N_Ω is the total number of Euler angles. But to integrate eq. (??) in a direct way, $(n+1)(2n+1)(2\lfloor n/s \rfloor + 1) = N_\Theta N_{\Phi\Psi} = N_\Omega$ function evaluations (FE) are needed for each $F_{\mu'\mu}^m$ ($s = 1$ or 2 according to the symmetry of axe C_s), an overall $O(N_\Omega^2)$ process is needed and *vice versa*. A faster algorithm proposed by Numerical Recipes [41] suggests reducing this cost to $O(N_\Theta^2 N_{\Phi\Psi} \ln N_{\Phi\Psi} \simeq N_\Omega^{4/3})$ by FFT. The implementation is detailed in appendix ???. The computing time of GSHT and FGSHT are shown in figure 13.2, for the case that Ψ possesses no symmetry.

It is shown that the computing cost reduced by FFT is about ... times of the original cost, which is agreed with the prediction. ?

Figure 13.2: Computing time of GSHT and FGSHT (per 1000 times)

13.2.2 Performance with respect to m_{\max}

13.2.3 Performance with respect to n_{\max}

psi treatment for mmax/nmax different

13.3 K-KERNEL

13.3.1 Comparison between paths

Timing paths in figure 12.1

13.3.2 m_{\max} and n_{\max} dependance of OZ equation

Theoretical listed in table 8.1.

with respect to nmax

13.4 ENTIRE ITERATION OF \mathcal{F}_{exc} EVALUATION

13.4.1 Comparison between “naive_standard” and “convolution_pure-angular”

Figure 13.3

difference after FFT.

verified the conclusion of k-kernel test.

13.4.2 Comparison between “convolution_standard” and “convolution_asymm”

Figure 13.4

The symmetries, ideally the time should be reduced by two, but as shown in §, convolution_standard need more “decoration”.

13.4.3 Comparison between “convolution_standard” and “convolution_pure_angular”

Figure 13.5

The inversion of FFT and FGSHT

we can see the other part is almost identical, but the implementation of FFT takes different time. Because in convolution_standard the number of FE we need for FFT is the number of projections, and in convolution_pure_angular it is the number of angular grid nodes. As there is less projections than angular nodes, convolution_standard reasonably takes less time.

13.5

GLOBAL VIEW OF THE SEQUENTIAL CODE PERFORMANCE

Computing time and memory limits

Hotspots and bottlenecks

14

COMPUTING PERFORMANCE OF PARALLEL CODE

14.1 NODE-LEVEL PARALLELIZATION

OpenMP, scalability with respect to the number of thread

14.2 PARALLELIZATION ON SEVERAL NODES

MPI, scalability with respect to the number of node

Chapter IV

APPLICATIONS

Only a few applications are made due to the time limit of this thesis, concluding

Section [ref] to show the capability of MDFT to calculate ions and small molecules.

Section [ref] to show a qualitative influence of polar solvent on the reaction including metal-oxo centre.

15

IONS AND SMALL MOLECULES COMPARED TO MD SIMULATION

15.0.1 *ions*

Table 15.1

[single ions P113]

16

HYDROGEN TRANSFER REACTION OF Mn-Oxo: ROLE OF SOLVENT IN REACTION SIMULATION

16.1 REINVESTIGATION OF THE MANGANESE-OXO

This is the case we talked about in the introduction, which is the thesis of university , the motivation of this thesis

Chapter V

CONCLUSION AND PERSPECTIVES

The only one publication during this thesis

CONCLUSION

Here is the publication during this thesis:

PERSPECTIVES

Due to the difficulty of this thesis and my mental state, there are a lot of un finished work and theory. Here lists some to give un idea.

18.1 REDUCE MEMORY USE IN MDFT

As shown in chapter III, memory leak can cause divergence and other problem.

18.1.1 *Pass to simple precision*

L-BFGS-B is in double precision

18.1.2 *MPI of the L-BFGS-B minimizer*

As the code is a blackbox, in Fortran 77, it is difficult to parallelize it.

18.2 POLARISABLE SOLUTE

Vext variable

18.3 MDFT VIEWER

This thesis contains a part of visualization

Viewer is an important part of code developing, which provide beautiful visualization and easy analyzing helps to popularize the code. GaussViewer is a good example.

The popular language of visualization is c++, OpenDM, ...

18.4 CLASSICAL SCF METHOD

To be more compatible with Gaussian, [Jensen]

This is only an idea, the mathematical deduction is not fully verified.

18.5 OTHER BRANCHES OF DEVELOPMENT ABOUT MDFT

3-body, polarization,

Chapter VI

APPENDIX

A

BASIC CONCEPTS ABOUT COMPUTING PERFORMANCE

In addition to the theory work, the performance of the code being developed is also an important aspect of this thesis. It is essential to have a fast and accurate method. To evaluate a code in a strict and systematic way, some basic concepts of computing performance are listed here.

a.1 ALGORITHM COMPLEXITY

Algorithm complexity is a crucial criteria for sequential code. A definition is given below.

Let f and g be two real (or even complex) functions defined over the natural numbers \mathbb{N} . We write

$$f = O(g) \quad (\text{A.1})$$

if there is a constant $c > 0$ such that from certain number $n > n_0$ we always have $|f(n)| \leq c|g(n)|$. The O is also named as the big-O notation [16], or order of growth. Figure A.1 shows the growth tendency of some frequent functions; from this we can conclude the following:

$$O(1) > O(\log_2 n) > O(n) > O(n \log_2 n) > O(n^2) > O(2^n) > O(n!) \quad (\text{A.2})$$

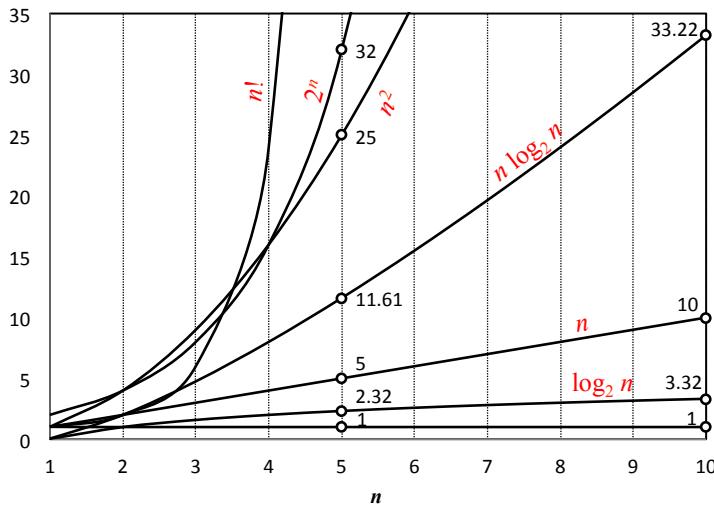


Figure A.1: Function growth

In this thesis, the big-O notation is used to measure algorithm complexity. Other notations can also be used for the same purpose, such as:

- $f = o(g)$ if $f(n)/g(n) \rightarrow 0$, $n \rightarrow \infty$
- The inverse of big-O notation $f = \Omega(g)$ if $g = O(f)$

- The notation $f = \Theta(g)$ means that both $f = O(g)$ and $g = O(f)$ hold, and we can also say they are of the same order.

In a code we always search algorithms with a lower algorithm complexity. Ideally, the implementation of code matches the model and have the same growth tendency as its complexity, but in the practical case, overheads and memory delay can also limit the performance. (part to be modified to adapt implementation results)

a.2 ROOFLINE MODEL AND MEMORY DELAY

The simplest model aiming to distinguish whether a piece of code is limited by the computing power (CPU) or the memory bandwidth (RAM to Caches) is the roofline model [47] for single loop:

$$P = \min(P_{\max}, I \cdot b_S) \quad (\text{A.3})$$

where

- P is the applicable peak performance of a loop, assuming that data comes from the level 1 cache, of unity GFlop/s.
- I is the computational intensity (“work” per byte transferred) over the slowest data path utilized, of unity Flop/Byte.
- b_S is the applicable peak bandwidth of the slowest data path utilized, of unity GByte/s.

As shown in figure A.2, the overall performance is limited by both the peak performance and the memory bandwidth. The computational intensity I depends on the code, while the other two terms in eq. (A.3) depend on hardware. The optimal use of resources occurs at the intersection point.

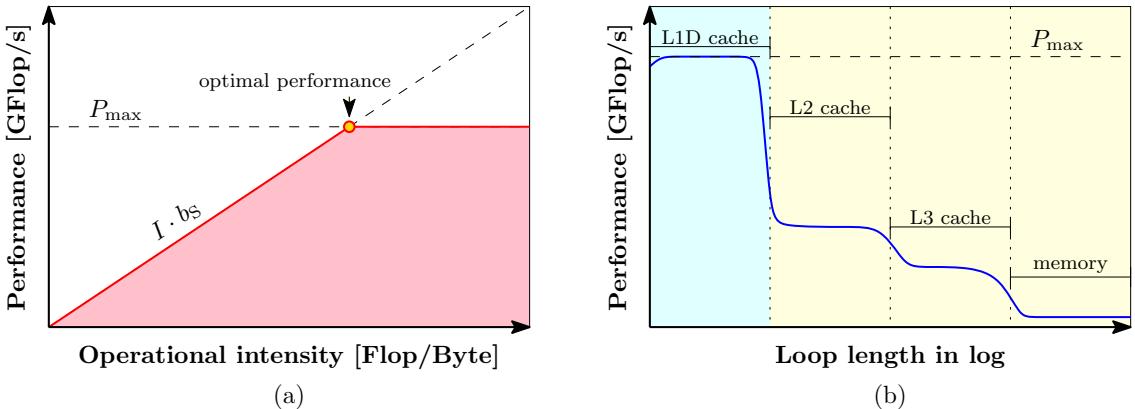


Figure A.2: The roofline model and performance pattern. (a) The roofline model. (b) Performance pattern of a simple loop with respect to the loop length in logarithm. The blue part is limited by operation execution, and the yellow part is limited by memory bottleneck.

The roofline model can give an idea of whether the diminuition of algorithm complexity is the most important optimization strategy, because it only counts the number of operations. In most of cases, avoiding slow data paths is the key to performance optimization.

As shown in figure A.3, the memory hardware has hierarchical architectures. The fastest ones are the registers included in the microprocessor, which are used for temporary storage of data, instructions and addresses required by the arithmetic logic unit (ALU) and the

control unit (CU) in CPU during execution of a program. The lowest is normally the input/output (I/O) process. The reading strategy of data (contiguous or not), as well as the size and initialized location of arrays, both play pivotal roles in the overall computing performance.

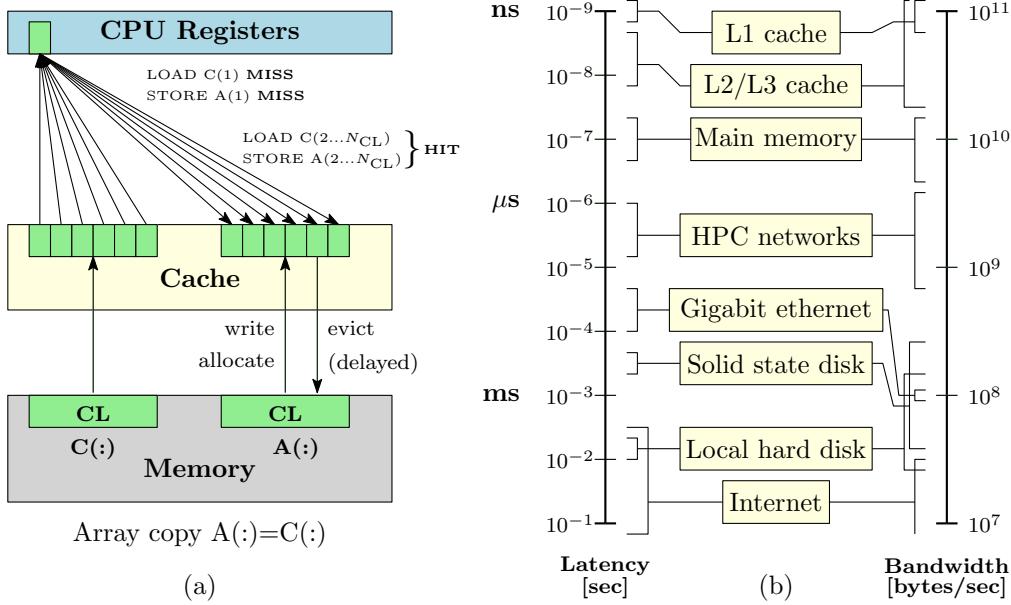


Figure A.3: Memory usage in hardware level [19]. (a) An example of array copy $A(:)=C(:)$. Caches are organized in cache lines (CL), only complete cache lines are transferred between memory hierarchy levels (except registers). HIT/MISS: Load or store instruction does/doesn't find the data in a cache level. (b) Computing latency and memory bandwidth vary by magnitude, from the fastest cache transfers to the lowest processes.

a.3 SCALABILITY OF PARALLELIZED CODE

For parallelized code, scalability is the key issue. Highly scalable codes can take advantage of numerous nodes of HPC centers, so that single core performance no longer matters.

The speed-up is defined as:

$$S(N) = \frac{t(1)}{t(N)} \quad (\text{A.4})$$

And the relative efficiency

$$E(N) = \frac{S(N)}{N} = \frac{t(1)}{Nt(N)} \quad (\text{A.5})$$

$S(N) \sim N$ or $E(N) \sim 100\%$ means the application scales. By contrast, $S(N) < N/2$ or $E(N) < 50\%$ means the application does not scale.

Amdahl's Law gives the theoretical speedup in latency of the execution of a task at fixed workload:

$$S(N) = \frac{1}{\alpha_s + \alpha_p/N} \quad (\text{A.6})$$

where α_s is the serial fraction and α_p the parallel fraction of the source code. Therefore the overall computing speed is limited by the unscalable part:

$$\lim_{N \rightarrow \infty} S(N) = \frac{1}{\alpha_s} \quad (\text{A.7})$$

making it the focus we wish to reduce.

a.4 PROFILING AND TRACING TOOLKITS

There are several types of software and toolkits for performance evaluation. They are of two categories: profiling and tracing. A trace is a collection of events or timestamps. A profile is a collection of timings. Profiling tools are usually more simple and rapid, but for subroutines that are called by a large number of times, the overhead in time measurement is not negligible.

The tool used in this thesis is mainly VTune, where application execution is interrupted every $\sim 100^{-6}$ s and information is stored (call stack, hardware counters, etc.). The execution time overhead is small. (To be detailed.)

B

EQUIVALENCE OF QUADRATURE-PROJECTION ORDER

b.1 GAUSSIAN QUADRATURE

Theorem:

Let $P_n(x)$ be a nonzero polynomial of degree n , $w(x)$ a positive weight function so that

$$\int_a^b x^k P_n(x) w(x) dx = 0, \quad (k = 0, \dots, n-1) \quad (\text{B.1})$$

If $\{x_i\}$ ($i = 1, \dots, n$) are the zeros of $P_n(x)$, then

$$\int_a^b f(x) w(x) dx \simeq \sum_{i=1}^n A_i f(x_i) \quad (\text{B.2})$$

with

$$A_i = \int_a^b l_{i-1}(x) w(x) dx \quad (\text{B.3})$$

is exact for all polynomials $f(x)$ of degree at most $2n - 1$, where $\{l_i\}$ are the usual Lagrange interpolating polynomials.

Proof:

Assume that $f(x)$ is a polynomial of degree at most $2n - 1$. Using long division

$$f(x) = P_n(x)p(x) + r(x) \quad (\text{B.4})$$

$p(x)$ and $r(x)$ are obtained as polynomials of degree at most $n - 1$.

By taking $\{x_i\}$ as the zeros of $P_n(x)$, we can easily find $f(x_i) = r(x_i)$, ($i = 1, \dots, n$), then

$$\begin{aligned} \int_a^b f(x) w(x) dx &= \int_a^b [P_n(x)p(x) + r(x)] w(x) dx \\ &\simeq \underbrace{\sum_{i=1}^n P_n(x_i)p(x_i)w_i}_{=0} + \sum_{i=1}^n A_i r(x_i) \end{aligned} \quad (\text{B.5})$$

is exact for $r(x)$ of degree at most $n - 1$ (c.f. Numerical Recipes [41] p.118), thus exact for $f(x)$ of degree at most $2n - 1$.

b.2 ANGULAR INTEGRATION IN GSHT

To expand a function onto GSHs, as in eq. (8.15), quadrature is needed. Assume that $F(\Omega)$ is a polynomial of $\cos \Theta$, $\cos \Phi$ and $\cos \Psi$ of order n . As $R_{\mu' \mu}^{m*}(\Omega)$ is also a polynomial of

order n , the total degree of integrand is $2n$. It should be noticed that the surface area element

$$d\Omega = \sin \Theta d\Theta d\Phi d\Psi = d \cos \Theta d\Phi d\Psi \quad (\text{B.6})$$

For $\cos \Theta$ integration, considering $w(x) = 1$ and $x = \cos \Theta$, Gauss-Legendre quadrature should be used. Thus $n + 1$ points on x should be taken, with $\{x_i\}$ given by Legendre polynomials $P_{n+1}(x)$.

For Φ and Ψ integration, taking $w(x) = (1 - x^2)^{-\frac{1}{2}}$, the abscissae are given by the $N = n + 1$ roots of the Chebyshev polynomial of the first kind

$$T_N(x) = \cos(N \cos x) \Rightarrow x_i = \cos \left[\frac{(2i-1)\pi}{2N} \right], \quad i \in 1, \dots, N \quad (\text{B.7})$$

with weight $w_i = \frac{\pi}{N}$, it corresponds to points in $\Phi \in [0, \pi]$ regularly distributed. However, for $\Phi \in [0, 2\pi]$, two times of function evaluation should be calculated

$$\begin{aligned} & \int_{-1}^1 f(\cos \Phi) \frac{1}{\sqrt{1 - \cos^2 \Phi}} d \cos \Phi \\ &= \begin{cases} \int_{\pi}^0 f(\cos \Phi) d\Phi = - \int_0^{\pi} f(\cos(\Phi)) d\Phi & \Phi \in [0, \pi] \\ \int_{-\pi}^0 f(\cos(\Phi)) d(\Phi) = \int_0^{\pi} f(\cos(-\Phi')) d\Phi' & \Phi' \in [0, \pi] \end{cases} \end{aligned} \quad (\text{B.8})$$

so that

$$\int_0^{2\pi} f(\cos \Phi) d\Phi = \int_{-\pi}^{\pi} f(\cos \Phi) d\Phi = \int_0^{\pi} [f(\cos(-\Phi)) - f(\cos \Phi)] d\Phi \quad (\text{B.9})$$

It corresponds to $2n + 2$ points in $\Phi \in [0, 2\pi]$ regularly distributed. However, it's not the minimal number of points necessary to do the exact integration. Suppose that $\Phi_2 \equiv \Phi/2$,

$$\int_0^{2\pi} f(\cos \Phi) d\Phi = \int_0^{\pi} f(\cos(2\Phi_2)) d\Phi_2 = \int_0^{\pi} [f(2 \cos^2 \Phi_2 - 1)] d\Phi_2 \quad (\text{B.10})$$

As $f(2 \cos^2 \Phi_2 - 1)$ is a polynomial of Φ of degree $2n$, it's a polynomial of Φ_2 of degree $4n$. Thus only $2n + 1$ points are needed.

C

ROTATIONAL INVARIANT EXPANSION

If a function $F(\mathbf{X}_1, \mathbf{X}_2)$, $\mathbf{X}_i \equiv (\mathbf{r}_i, \boldsymbol{\Omega}_i)$ have transitional and rotational invariance [4], it can be expanded as

$$F(\mathbf{X}_1, \mathbf{X}_2) = \sum_{mnl\mu\nu} F_{\mu\nu}^{mnl}(\|\mathbf{r}_{12}\|) \Phi_{\mu\nu}^{mnl}(\boldsymbol{\Omega}_1, \boldsymbol{\Omega}_2, \hat{\mathbf{r}}_{12}) \quad (\text{C.1})$$

where $\mathbf{r}_{12} \equiv \mathbf{r}_1 - \mathbf{r}_2$ according to the transitional invariance, and

$$\Phi_{\mu\nu}^{mnl}(\boldsymbol{\Omega}_1, \boldsymbol{\Omega}_2, \hat{\mathbf{r}}_{12}) = f^{mnl} \sum_{\mu'\nu'\lambda'} \begin{pmatrix} m & n & l \\ \mu' & \nu' & \lambda' \end{pmatrix} R_{\mu'\mu}^m(\boldsymbol{\Omega}_1) R_{\nu'\nu}^n(\boldsymbol{\Omega}_2) R_{\lambda'0}^l(\hat{\mathbf{r}}_{12}) \quad (\text{C.2})$$

where $R_{\mu'\mu}^m$ is the Wigner generalized spherical harmonics or Wigner D-symbol defined in the same convention as Messiah [36] (different than Edmonds [11]). f^{mnl} can be any arbitrary non-zero constant [14]. Here we define $f^{mnl} = f^m f^n = \sqrt{2m+1} \sqrt{2n+1}$ according to the definition of Belloni [1].

Two special cases are adopted in this thesis, which are the laboratory coordinate system with particle 1 at origin (fixed frame) and intermolecular coordinate system (local frame) shown in figure 7.1. Their formalism and symmetry properties are presented later.

C.1 ORTHOGONALITY OF Φ

The rotational invariant Φ in eq. (C.2) is orthogonal, proven as below

$$\begin{aligned} \langle \Phi | \Phi_2 \rangle &= \int d\boldsymbol{\Omega}_1 d\boldsymbol{\Omega}_2 d\hat{\mathbf{r}} \Phi_{\mu\nu}^{mnl}(\boldsymbol{\Omega}_1, \boldsymbol{\Omega}_2, \hat{\mathbf{r}}_{12}) \Phi_{\mu_2\nu_2}^{m_2n_2l_2*}(\boldsymbol{\Omega}_1, \boldsymbol{\Omega}_2, \hat{\mathbf{r}}_{12}) \\ &= f^m f^n f^{m_2} f^{n_2} \sum_{\mu'\nu'\lambda'} \begin{pmatrix} m & n & l \\ \mu' & \nu' & \lambda' \end{pmatrix} \sum_{\mu'_2\nu'_2\lambda'_2} \begin{pmatrix} m_2 & n_2 & l_2 \\ \mu'_2 & \nu'_2 & \lambda'_2 \end{pmatrix} \\ &\quad \times \left\{ \int d\boldsymbol{\Omega}_1 R_{\mu'\mu}^m(\boldsymbol{\Omega}_1) R_{\mu'_2\mu_2}^{m_2*}(\boldsymbol{\Omega}_1) \right. \\ &\quad \left[\int d\boldsymbol{\Omega}_2 R_{\nu'\nu}^n(\boldsymbol{\Omega}_2) R_{\nu'_2\nu_2}^{n_2*}(\boldsymbol{\Omega}_2) \left(\int d\hat{\mathbf{r}} R_{\lambda'0}^l(\hat{\mathbf{r}}_{12}) R_{\lambda'_20}^{l_2*}(\hat{\mathbf{r}}_{12}) \right) \right] \} \\ &= f^m f^n f^{m_2} f^{n_2} \sum_{\mu'\nu'\lambda'} \begin{pmatrix} m & n & l \\ \mu' & \nu' & \lambda' \end{pmatrix} \sum_{\mu'_2\nu'_2\lambda'_2} \begin{pmatrix} m_2 & n_2 & l_2 \\ \mu'_2 & \nu'_2 & \lambda'_2 \end{pmatrix} \\ &\quad \times \delta_{m,m_2} \delta_{n,n_2} \delta_{l,l_2} \delta_{\mu,\mu_2} \delta_{\nu,\nu_2} \delta_{\mu',\mu'_2} \delta_{\nu',\nu'_2} \delta_{\lambda',\lambda'_2} \\ &= (2l+1)^{-1} \sum_{\mu'\nu'\lambda'} \begin{pmatrix} m & n & l \\ \mu' & \nu' & \lambda' \end{pmatrix} \begin{pmatrix} m & n & l \\ \mu' & \nu' & \lambda' \end{pmatrix} \quad (\text{C.3}) \end{aligned}$$

and using the orthogonality of 3j-symbol [11]

$$\sum_{l\lambda'} (2l+1) \begin{pmatrix} m & n & l \\ \mu'_1 & \nu'_1 & \lambda' \end{pmatrix} \begin{pmatrix} m & n & l \\ \mu'_2 & \nu'_2 & \lambda' \end{pmatrix} = \delta_{\mu'_1\mu'_2} \delta_{\nu'_1\nu'_2} \quad (\text{C.4})$$

$$\sum_{\mu' \nu'} \begin{pmatrix} m & n & l \\ \mu' & \nu' & \lambda' \end{pmatrix} \begin{pmatrix} m & n & l_2 \\ \mu' & \nu' & \lambda'_2 \end{pmatrix} = (2l+1)^{-1} \delta_{ll_2} \delta_{\lambda'_1 \lambda'_2} \quad (\text{C.5})$$

it gives

$$\langle \Phi | \Phi_2 \rangle = (2l+1)^{-1} \quad (\text{C.6})$$

C.2 ROTATIONAL INVARIANCE OF Φ

In any coordinate system, the value of $\Phi_{\mu\nu}^{mn l}$ remains the same. Here is an example of demonstration with the fixed and local frame mentioned above, described in figure 7.1.

Let's use the definition in eq. (C.2):

$$\Phi_{\mu\nu}^{mn l}(\omega_1, \omega_2, 0) = f^{mn l} \sum_{\mu'' \nu'' \lambda''} \begin{pmatrix} m & n & l \\ \mu'' & \nu'' & \lambda'' \end{pmatrix} R_{\mu'' \mu}^m(\omega_1) R_{\nu'' \nu}^n(\omega_2) R_{\lambda'' 0}^l(0) \quad (\text{C.7})$$

$$\Phi_{\mu\nu}^{mn l}(0, \Omega, \hat{\mathbf{r}}) = f^{mn l} \sum_{\mu' \nu' \lambda'} \begin{pmatrix} m & n & l \\ \mu' & \nu' & \lambda' \end{pmatrix} R_{\mu' \mu}^m(0) R_{\nu' \nu}^n(\Omega) R_{\lambda' 0}^l(\hat{\mathbf{r}}) \quad (\text{C.8})$$

The spherical harmonics have property [11, 36]

$$R_{\mu' \mu}^m(0) = \sum_{\mu''} R_{\mu' \mu''}^m(\hat{\mathbf{r}}) R_{\mu'' \mu}^m(\omega_1) \quad (\text{C.9})$$

$$R_{\nu' \nu}^n(\Omega) = \sum_{\nu''} R_{\nu' \nu''}^n(\hat{\mathbf{r}}) R_{\nu'' \nu}^n(\omega_2) \quad (\text{C.10})$$

$$R_{\lambda' 0}^l(\hat{\mathbf{r}}) = \sum_{\lambda''} R_{\lambda' \lambda''}^l(\hat{\mathbf{r}}) R_{\lambda'' 0}^l(0) \quad (\text{C.11})$$

so

$$\begin{aligned} \Phi_{\mu\nu}^{mn l}(0, \Omega, \hat{\mathbf{r}}) &= f^{mn l} \sum_{\mu'' \nu'' \lambda''} R_{\mu'' \mu}^m(\omega_1) R_{\nu'' \nu}^n(\omega_2) R_{\lambda'' 0}^l(0) \times \\ &\quad \left[\sum_{\mu' \nu' \lambda'} \begin{pmatrix} m & n & l \\ \mu' & \nu' & \lambda' \end{pmatrix} R_{\mu' \mu''}^m(\hat{\mathbf{r}}) R_{\nu' \nu''}^n(\hat{\mathbf{r}}) R_{\lambda' \lambda''}^l(\hat{\mathbf{r}}) \right] \end{aligned} \quad (\text{C.12})$$

According to eq. (4.3.3) in Edmonds [11] or (A.91) in Gray & Gubbins [18]

$$\sum_{\mu' \nu' \lambda'} \begin{pmatrix} m & n & l \\ \mu' & \nu' & \lambda' \end{pmatrix} R_{\mu' \mu''}^{m*}(\hat{\mathbf{r}}) R_{\nu' \nu''}^{n*}(\hat{\mathbf{r}}) R_{\lambda' \lambda''}^{l*}(\hat{\mathbf{r}}) = \begin{pmatrix} m & n & l \\ \mu'' & \nu'' & \lambda'' \end{pmatrix} \quad (\text{C.13})$$

where we can also prove

$$\sum_{\mu' \nu' \lambda'} \begin{pmatrix} m & n & l \\ \mu' & \nu' & \lambda' \end{pmatrix} R_{\mu' \mu''}^m(\hat{\mathbf{r}}) R_{\nu' \nu''}^n(\hat{\mathbf{r}}) R_{\lambda' \lambda''}^l(\hat{\mathbf{r}}) = \begin{pmatrix} m & n & l \\ \mu'' & \nu'' & \lambda'' \end{pmatrix} \quad (\text{C.14})$$

$\Phi_{\mu\nu}^{mn l}$ remains identical in the two cases

$$\begin{aligned} \Phi_{\mu\nu}^{mn l}(0, \Omega, \hat{\mathbf{r}}) &= f^{mn l} \sum_{\mu'' \nu'' \lambda''} \begin{pmatrix} m & n & l \\ \mu'' & \nu'' & \lambda'' \end{pmatrix} R_{\mu'' \mu}^m(\omega_1) R_{\nu'' \nu}^n(\omega_2) R_{\lambda'' 0}^l(0) \\ &= \Phi_{\mu\nu}^{mn l}(\omega_1, \omega_2, 0) \end{aligned} \quad (\text{C.15})$$

Therefore, the projections $F_{\mu\nu}^{mn l}(r)$ also remains rotational invariant in this two coordinate system.

C.3 TRANSFORM IN LOCAL FRAME

In the intermolecular (local) coordinate system, the 2 molecules are both positioned along z axis. Using the properties [11, 18, 36] of generalized spherical harmonics

$$R_{\mu'\mu}^m(\Theta, \Phi, \Psi) = \delta_{\mu'\mu} \quad \text{if } \Theta = \Phi = \Psi = 0 \quad (\text{C.16})$$

and 3j-symbol

$$\begin{pmatrix} m & n & l \\ \mu' & \nu' & \lambda' \end{pmatrix} \neq 0 \quad \text{only if } \mu' + \nu' + \lambda' = 0 \quad (\text{C.17})$$

$\Phi_{\mu\nu}^{mn}(\boldsymbol{\Omega}_1, \boldsymbol{\Omega}_2, \mathbf{r}_{12})$ in eq. (C.2) can be simplified to

$$\Phi_{\mu\nu}^{mn}(\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, 0) = \sum_{\chi} \begin{pmatrix} m & n & l \\ \chi & -\chi & 0 \end{pmatrix} f^m f^n R_{\chi\mu}^m(\boldsymbol{\omega}_1) R_{-\chi\nu}^n(\boldsymbol{\omega}_2) \quad (\text{C.18})$$

Thus eq. (C.1) becomes

$$\begin{aligned} F(\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, r) &= \sum_{mnl\mu\nu} F_{\mu\nu}^{mn}(r) \Phi_{\mu\nu}^{mn}(\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, 0) \\ &= \sum_{mnl\mu\nu} F_{\mu\nu}^{mn}(r) f^m f^n \sum_{\chi} \begin{pmatrix} m & n & l \\ \chi & -\chi & 0 \end{pmatrix} R_{\chi\mu}^m(\boldsymbol{\omega}_1) R_{-\chi\nu}^n(\boldsymbol{\omega}_2) \end{aligned} \quad (\text{C.19})$$

and the inverse equation

$$\begin{aligned} F_{\mu\nu}^{mn}(r) &= \int d\boldsymbol{\omega}_1 d\boldsymbol{\omega}_2 F(\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, r) \Phi_{\mu\nu}^{mn*}(\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, 0) \\ &= f^m f^n \sum_{\chi} \begin{pmatrix} m & n & l \\ \chi & -\chi & 0 \end{pmatrix} \times \\ &\quad \int d\boldsymbol{\omega}_1 R_{\chi\mu}^{m*}(\boldsymbol{\omega}_1) \int d\boldsymbol{\omega}_2 R_{-\chi\nu}^{n*}(\boldsymbol{\omega}_2) F(\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, r) \end{aligned} \quad (\text{C.20})$$

The function $F(\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, r)$ and the projections $F_{\mu\nu}^{mn}(r)$ can be transformed to each other by 2 simple steps.

C.3.1 Transform between $F_{\mu\nu}^{mn}(r)$ and $F_{\mu\nu,\chi}^{mn}(r)$

Suppose

$$F_{\mu\nu,\chi}^{mn}(r) = \sum_l \begin{pmatrix} m & n & l \\ \chi & -\chi & 0 \end{pmatrix} F_{\mu\nu}^{mn}(r) \quad (\text{C.21})$$

Using property of 3j-symbol [36]

$$\sum_{\chi} \begin{pmatrix} m & n & l' \\ \chi & -\chi & 0 \end{pmatrix} \begin{pmatrix} m & n & l \\ \chi & -\chi & 0 \end{pmatrix} = \frac{\delta_{l'l}}{2l+1} \quad (\text{C.22})$$

we have as the inverse transform

$$F_{\mu\nu}^{mn}(r) = (2l+1) \sum_{\chi} \begin{pmatrix} m & n & l \\ \chi & -\chi & 0 \end{pmatrix} F_{\mu\nu,\chi}^{mn}(r) \quad (\text{C.23})$$

Thus Eq. (C.19) becomes

$$\begin{aligned}
F(\omega_1, \omega_2, r) &= \sum_{mn\mu\nu} (2l+1) \sum_{\chi'} \begin{pmatrix} m & n & l \\ \chi' & -\chi' & 0 \end{pmatrix} F_{\mu\nu,\chi'}^{mn}(r) \times \\
&\quad \sum_{\chi} \begin{pmatrix} m & n & l \\ \chi & -\chi & 0 \end{pmatrix} f^m f^n R_{\chi\mu}^m(\omega_1) R_{-\chi\nu}^n(\omega_2) \\
&= \sum_{mn\mu\nu} \sum_{\chi'} \sum_{\chi} F_{\mu\nu,\chi'}^{mn}(r) f^m f^n R_{\chi\mu}^m(\omega_1) R_{-\chi\nu}^n(\omega_2) \times \\
&\quad \sum_l (2l+1) \begin{pmatrix} m & n & l \\ \chi' & -\chi' & 0 \end{pmatrix} \begin{pmatrix} m & n & l \\ \chi & -\chi & 0 \end{pmatrix} \tag{C.24}
\end{aligned}$$

As

$$\sum_l (2l+1) \begin{pmatrix} m & n & l \\ \chi' & -\chi' & 0 \end{pmatrix} \begin{pmatrix} m & n & l \\ \chi & -\chi & 0 \end{pmatrix} = \delta_{\chi'\chi} \tag{C.25}$$

we have

$$F(\omega_1, \omega_2, r) = \sum_{mn\mu\nu\chi} F_{\mu\nu,\chi}^{mn}(r) f^m f^n R_{\chi\mu}^m(\omega_1) R_{-\chi\nu}^n(\omega_2) \tag{C.26}$$

and

$$F_{\mu\nu,\chi}^{mn}(r) = \int d\omega_1 d\omega_2 F(\omega_1, \omega_2, r) f^m f^n R_{\chi\mu}^{m*}(\omega_1) R_{-\chi\nu}^{n*}(\omega_2) \tag{C.27}$$

Thus eq. (C.26, C.27) can be performed either by fast generalized spherical harmonic transform (FGSHT), or being developed into

$$F(\omega_1, \omega_2, r) = \sum_{mn\mu\nu\chi} F_{\mu\nu,\chi}^{mn}(r) f^m f^n r_{\chi\mu}^m(\theta_1) r_{-\chi\nu}^n(\theta_2) e^{-i\chi(\phi_{12}\equiv\phi_1-\phi_2)} e^{-i\mu\psi_1} e^{-i\nu\psi_2} \tag{C.28}$$

and be transformed with FFT-3D.

C.3.2 Rotational invariant transform with FFT-3D

Suppose

$$F_{\mu\nu,\chi}^m(r, \theta_2) = \sum_n F_{\mu\nu,\chi}^{mn}(r) f^n r_{-\chi\nu}^n(\theta_2) \tag{C.29}$$

then we have

$$F(\omega_1, \omega_2, r) = \sum_{m\mu\nu\chi} F_{\mu\nu,\chi}^m(r, \theta_2) f^m r_{\chi\mu}^m(\theta_1) e^{-i\chi\phi_{12}} e^{-i\mu\psi_1} e^{-i\nu\psi_2} \tag{C.30}$$

The inverse transform should be

$$F_{\mu\nu,\chi}^{mn}(r) = \frac{1}{2} \int d(\cos\theta_2) F_{\mu\nu,\chi}^m(r, \theta_2) r_{-\chi\nu}^n(\theta_2) \tag{C.31}$$

In the same way, suppose

$$F_{\mu\nu,\chi}(r, \theta_1, \theta_2) = \sum_m F_{\mu\nu,\chi}^m(r, \theta_2) r_{\chi\mu}^m(\theta_1) \tag{C.32}$$

and the inverse transform

$$F_{\mu\nu,\chi}^m(r, \theta_2) = \frac{1}{2} \int d(\cos\theta_1) F_{\mu\nu,\chi}(r, \theta_1, \theta_2) r_{\chi\mu}^m(\theta_1) \tag{C.33}$$

then we have

$$F(r, \omega_1, \omega_2) = \sum_{\mu\nu\chi} F_{\mu\nu,\chi}(r, \theta_1, \theta_2) e^{-i\chi\phi_{12}} e^{-i\mu\psi_1} e^{-i\nu\psi_2} \tag{C.34}$$

which can be treat as a normal FFT of 3 dimension.

C.4 TRANSFORM IN FIXED FRAME

The same way, in the laboratory coordinate system

$$\Phi_{\mu\nu}^{mn}(0, \Omega, \hat{\mathbf{r}}) = \sum_{\eta} \begin{pmatrix} m & n & l \\ \mu & \eta & -\mu-\eta \end{pmatrix} f^m f^n R_{\eta,\nu}^n(\Omega) R_{-\mu-\eta,0}^l(\hat{\mathbf{r}}) \quad (\text{C.35})$$

The rotational invariant doesn't take advantage of the χ transform as $\mu \neq 0$. The expansion on rotational invariants should be calculated directly.

c.4.1 Expansion of $F(\mathbf{r}, \Omega)$ on rotational invariants

The total equation of the forward transform is as shown

$$F_{\mu\nu}^{mn}(r) = f^m f^n \sum_{\eta} \begin{pmatrix} m & n & l \\ \mu & \eta & -\mu-\eta \end{pmatrix} \int d\hat{\mathbf{r}} R_{-\mu-\eta,0}^{l*}(\hat{\mathbf{r}}) \int d\Omega F(r, \hat{\mathbf{r}}, \Omega) R_{\eta,\nu}^{n*}(\Omega) \quad (\text{C.36})$$

Firstly, the FGSHT is performed

$$F_{\eta\nu}^n(\mathbf{r}) = \int d\Omega f^n F(\mathbf{r}, \Omega) R_{\eta,\nu}^{n*}(\Omega) \quad (\text{C.37})$$

Then the spherical harmonic transform by histogram should give

$$F_{\eta\nu,\lambda}^{nl}(r) = \int d\hat{\mathbf{r}} R_{\lambda 0}^{l*}(\hat{\mathbf{r}}) F_{\eta\nu}^n(r, \hat{\mathbf{r}}) \quad (\text{C.38})$$

As $F_{\eta\nu}^n(\mathbf{r})$ values are tabulated in the Cartesian grid, we cannot use a quadrature approach without interpolation, so histogram approach is used.

Histogram for a function f gives:

$$\bar{f}(r) = \int d\theta_r d\phi_r f(x, y, z) \quad (\text{C.39})$$

so if we want to compute

$$\bar{F}(r) = \int d\theta_r d\phi_r R_{\lambda 0}^{l*}(x, y, z) F(x, y, z) \quad (\text{C.40})$$

we just need to propose

$$f(x, y, z) = R_{\lambda 0}^{l*}(x, y, z) F(x, y, z) \quad (\text{C.41})$$

For complex numbers $F_{\eta\nu}^n(\mathbf{r})$, the real and imagine part can be calculated separately.

The rotational matrices $R_{\lambda 0}^{l*}(\mathbf{r})$ in Cartesian coordinate system can be pre-generate by recurrence as detailed in appendix D.

Finally, the combination of projections gives

$$F_{\mu\nu}^{mn}(r) = f^m \sum_{\eta} \begin{pmatrix} m & n & l \\ \mu & \eta & -\mu-\eta \end{pmatrix} F_{\eta\nu,-\mu-\eta}^{nl}(r) \quad (\text{C.42})$$

It should be noted that $F_{\mu\nu}^{mn}(r)$ is real?

C.4.2 Rebuilt of $F(\mathbf{r}, \Omega)$ from projections

and the rebuilt of $F(\mathbf{r}, \Omega)$ in a certain orientation is as simple as its definition

$$F(\mathbf{r}, \Omega) = \sum_{mnl\mu\nu} F_{\mu\nu}^{mnl}(r) f^m f^n \sum_{\eta} \begin{pmatrix} m & n & l \\ \mu & \eta & -\mu - \eta \end{pmatrix} R_{\eta\nu}^n(\Omega) R_{-\mu-\eta, 0}^l(\hat{\mathbf{r}}) \quad (\text{C.43})$$

C.5 SYMMETRY

In IEM and MDFT, the rotational invariants is used to describe the solvent. It possess symmetric rules, introduced by the indistinguishability of the two particles, symmetry properties of single particle and its real number property as a physical quantity. Here lists all the symmetric rules concerns the 2-molecule system.

C.5.1 Symmetric rules of $F(\omega_1, \omega_2)$ in intermolecular form

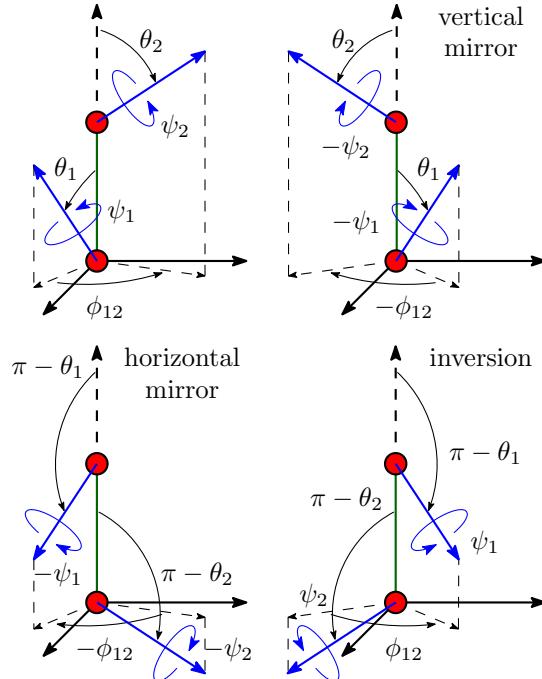


Figure C.1: Symmetry operations of a 2-molecule system

As shown in figure C.1, function in intermolecular coordinate system $F(\omega_1, \omega_2) \equiv F(\cos \theta_1, \cos \theta_2, \phi, \psi_1, \psi_2)$ possesses symmetry rules:

1. Symmetry of vertical mirror:

$$F(\theta_1, \theta_2, \phi, \psi_1, \psi_2) = F(\theta_1, \theta_2, -\phi, -\psi_1, -\psi_2) \quad (\text{C.44})$$

2. Symmetry of inversion:

$$F(\theta_1, \theta_2, \phi, \psi_1, \psi_2) = F(\pi - \theta_2, \pi - \theta_1, \phi, \psi_2, \psi_1) \quad (\text{C.45})$$

And an additive symmetric rule is possessed by particles having

1. [3.]Symmetry axe C_{2n}:

$$F(\theta_1, \theta_2, \phi, \psi_1, \psi_2) = F(\theta_1, \theta_2, \phi, \psi_1 + \pi, \psi_2 + \pi) \quad (\text{C.46})$$

C.5.2 Symmetric rules of rotational invariant projections

The definition of rotational invariant on χ -transform gives

$$c(\theta_1, \theta_2, \phi_{12}, \psi_1, \psi_2) = \frac{1}{2l+1} \sum_{mn\mu\nu\chi} c_{\mu\nu,\chi}^{mn}(r) d_{\chi\mu}^m(\theta_1) d_{\chi\nu}^n(\theta_2) e^{-i\chi\phi_{12}} e^{-i\mu\psi_1} e^{-i\nu\psi_2}$$

Thus

$$\begin{aligned} c(\theta_1, \theta_2, -\phi_{12}, -\psi_1, -\psi_2) &= \frac{1}{2l+1} \sum_{mn\mu\nu\chi} c_{\underline{\mu}\underline{\nu},\underline{\chi}}^{mn}(r) d_{\chi\mu}^m(\theta_1) d_{\chi\nu}^n(\theta_2) e^{-i\chi\phi_{12}} e^{-i\mu\psi_1} e^{-i\nu\psi_2} \\ &= \frac{1}{2l+1} \sum_{mn\mu\nu\chi} (-)^{\mu+\nu} c_{\underline{\mu}\underline{\nu},\underline{\chi}}^{mn}(r) d_{\chi\mu}^m(\theta_1) d_{\chi\nu}^n(\theta_2) e^{-i\chi\phi_{12}} e^{-i\mu\psi_1} e^{-i\nu\psi_2} \\ c(\pi - \theta_2, \pi - \theta_1, \phi_{12}, \psi_2, \psi_1) &= \frac{1}{2l+1} \sum_{mn\mu\nu\chi} c_{\mu\nu,\chi}^{mn}(r) d_{\chi\mu}^m(\pi - \theta_2) d_{\chi\nu}^n(\pi - \theta_1) e^{-i\chi\phi_{12}} e^{-i\mu\psi_2} e^{-i\nu\psi_1} \\ &= \frac{1}{2l+1} \sum_{mn\mu\nu\chi} (-)^{m+n+\mu+\nu} c_{\nu\mu,\chi}^{nm}(r) d_{\chi\mu}^m(\theta_1) d_{\chi\nu}^n(\theta_2) e^{-i\chi\phi_{12}} e^{-i\mu\psi_1} e^{-i\nu\psi_2} \end{aligned}$$

and μ, ν are even.

Thus

$$c_{\mu\nu,\chi}^{mn}(r) = c_{\underline{\mu}\underline{\nu},\underline{\chi}}^{mn}(r) = (-)^{m+n} c_{\nu\mu,\chi}^{nm}(r)$$

(Need proof.)

$$\hat{c}_{\mu\nu,\chi}^{mn}(\mathbf{k}) = (-)^{m+n} \hat{c}_{\mu\nu,\chi}^{*mn}(\mathbf{k})$$

In MDFT, rotational invariant projections is used to describe DCF. The original $c_{\mu\nu}^{mnl}(\mathbf{r})$ collected by IEM is real.

1. With r and l : As $c_{\mu\nu}^{mnl}(\mathbf{r})$ is real

$$F_{\underline{\mu}\underline{\nu}}^{mnl}(\mathbf{r}) = (-)^{m+n+l} F_{\mu\nu}^{mnl}(\mathbf{r}) \quad (\text{C.47})$$

As the two molecules are interchangeable,

$$F_{\nu\mu}^{nm l}(\mathbf{r}) = (-)^{m+n} F_{\mu\nu}^{mnl}(\mathbf{r}) \quad (\text{C.48})$$

2. [2.]With r and χ :

$$F_{\mu\nu,\chi}^{mn}(\mathbf{r}) = \sum_{\chi} \begin{pmatrix} m & n & l \\ \chi & -\chi & 0 \end{pmatrix} F_{\mu\nu}^{mnl}(\mathbf{r}) \quad (\text{C.49})$$

$$F_{\underline{\mu}\underline{\nu},\underline{\chi}}^{mn}(\mathbf{r}) = F_{\mu\nu,\chi}^{mn}(\mathbf{r}) \quad (\text{C.50})$$

$$F_{\nu\mu,\chi}^{nm}(\mathbf{r}) = (-)^{m+n} F_{\mu\nu,\chi}^{mn}(\mathbf{r}) \quad (\text{C.51})$$

3. [3.] With k and l :

$$\text{TF}(c_{\mu\nu}^{mnl}(\mathbf{r})) = \hat{c}_{\mu\nu}^{mnl}(\mathbf{k}) \quad (\text{C.52})$$

is real if l is even, is pure imaginary if l is odd.

4. [4.] With k and χ :

$$F_{\mu\nu,\chi}^{mn}(\mathbf{k}) = (-)^{m+n} F_{\mu\nu,\chi}^{*mn}(\mathbf{k}) \quad (\text{C.53})$$

Finally, if the solvent molecule possesses a symmetric axis C_2 , μ and ν is even.

For spherical solute, the Ω_1 -dependence vanishes so that $m = \mu = 0$, only the terms $F_{0\nu}^{0nl}(r)$ is not-zero.

D

CALCULATION OF ROTATION MATRIX ELEMENTS $R_{\mu\mu'}^m$ BY RECURRENCE

$\mathbf{R}^m(\Omega) \equiv \{R_{\mu'\chi}^m(\Omega)\}$ is the rotation matrix of dimension $(2m+1) \times (2m+1)$, defined in Messiah, etc. [11, 18, 36].

In MDFT, evaluation of $R_{\mu'\chi}^m(\hat{\mathbf{k}})$ for each m, μ', χ and \mathbf{k} by its definition:

$$R_{\mu'\chi}^m(\hat{\mathbf{k}}) = r_{\mu'\chi}^m(\theta_k) e^{-i\mu'\phi_k} \quad (\text{D.1})$$

is too costing to be done in iterations, on the other hand, to stock directly the value of every element is heavy in terms of memory. An algorithm of $R_{\mu\mu'}^m(\hat{\mathbf{k}})$ evaluation by recurrence described by Choi *et al.* [8] suggests an acceptable cost during the computation, by generating the rotation matrix elements from those of lower order to avoid extra calculation.

d.1 CASE OF $m_{\max} \leq 1$

According to the definition in eq. (D.1), it's easy to find

$$R_{00}^0 = 1 \quad (\text{D.2})$$

For $m = 1$, $\mathbf{R}^1(\hat{\mathbf{k}})$ depends only on the 3×3 orthogonal matrix \mathbf{R} that defines the rotation from the basis vectors of laboratory frame to those of \mathbf{k} -frame

$$\mathbf{R} = \begin{bmatrix} R_{xx} & R_{yx} & R_{zx} \\ R_{xy} & R_{yy} & R_{zy} \\ R_{xz} & R_{yz} & R_{zz} \end{bmatrix} = \begin{bmatrix} \cos \theta_k \cos \phi_k & -\sin \phi_k & \sin \theta_k \cos \phi_k \\ \cos \theta_k \sin \phi_k & \cos \phi_k & \sin \theta_k \sin \phi_k \\ -\sin \theta_k & 0 & \cos \theta_k \end{bmatrix} \quad (\text{D.3})$$

The matrix \mathbf{R} can be calculated by the cross products of basis vectors as shown in figure 7.3

$$\begin{bmatrix} \mathbf{e}_1'' & \mathbf{e}_2' & \mathbf{e}_3'' \end{bmatrix} = \begin{bmatrix} \mathbf{e}_1 & \mathbf{e}_2 & \mathbf{e}_3 \end{bmatrix} \mathbf{R} = \mathbf{R} \quad (\text{D.4})$$

The rotation matrix \mathbf{R}^m can be separated into the real \mathbf{F}^m and imaginary \mathbf{G}^m part, which can be given by the relations

$$R_{\chi\chi'}^m = F_{\chi\chi'}^m + iG_{\chi\chi'}^m \quad (\text{D.5})$$

$$\begin{bmatrix} F_{11}^1 & F_{10}^1 & F_{11}^1 \\ F_{01}^1 & F_{00}^1 & F_{01}^1 \\ F_{11}^1 & F_{10}^1 & F_{11}^1 \end{bmatrix} = \begin{bmatrix} (R_{yy} + R_{xx})/2 & R_{xz}/\sqrt{2} & (R_{yy} - R_{xx})/2 \\ R_{zx}/\sqrt{2} & R_{zz} & -R_{xz}/\sqrt{2} \\ (R_{yy} - R_{xx})/2 & -R_{xz}/\sqrt{2} & (R_{yy} + R_{xx})/2 \end{bmatrix} \quad (\text{D.6})$$

$$\begin{bmatrix} G_{11}^1 & G_{10}^1 & G_{11}^1 \\ G_{01}^1 & G_{00}^1 & G_{01}^1 \\ G_{11}^1 & G_{10}^1 & G_{11}^1 \end{bmatrix} = \begin{bmatrix} (R_{yx} - R_{xy})/2 & R_{yz}/\sqrt{2} & -(R_{yx} + R_{xy})/2 \\ -R_{zy}/\sqrt{2} & 0 & -R_{zy}/\sqrt{2} \\ (R_{yx} + R_{xy})/2 & R_{yz}/\sqrt{2} & (R_{xy} - R_{yx})/2 \end{bmatrix} \quad (\text{D.7})$$

d.2 CASE OF $m_{\max} > 1$

D.2.1 Recurrence relation for $-m + 1 \leq \chi' \leq m - 1$

The recurrence relation for $-m \leq \chi \leq m$, $-m + 1 \leq \chi' \leq m - 1$ between matrix elements is

$$R_{\chi\chi'}^m = a_{\chi\chi'}^m R_{00}^1 R_{\chi\chi'}^{m-1} + b_{\chi\chi'}^m R_{10}^1 R_{\chi-1,\chi'}^{m-1} + b_{-\chi,\chi'}^m R_{-1,0}^1 R_{\chi+1,\chi'}^{m-1} \quad (\text{D.8})$$

where

$$\begin{aligned} a_{\chi\chi'}^m &= \left[\frac{(m+\chi)(m-\chi)}{(m+\chi')(m-\chi')} \right]^{\frac{1}{2}} \quad (-m+1 \leq \chi \leq m-1) \\ b_{\chi\chi'}^m &= \left[\frac{(m+\chi)(m+\chi-1)}{2(m+\chi')(m-\chi')} \right]^{\frac{1}{2}} \quad (-m+2 \leq \chi \leq m-2) \end{aligned} \quad (\text{D.9})$$

To separate the real and imaginary part, suppose

$$H_{\chi\chi'}^m(i, j) = F_{ij}^1 F_{\chi\chi'}^{m-1} - G_{ij}^1 G_{\chi\chi'}^{m-1} \quad (\text{D.10})$$

$$K_{\chi\chi'}^m(i, j) = F_{ij}^1 G_{\chi\chi'}^{m-1} + G_{ij}^1 F_{\chi\chi'}^{m-1} \quad (\text{D.11})$$

therefore

$$F_{\chi\chi'}^m = a_{\chi\chi'}^m H_{\chi\chi'}^m(0, 0) + b_{\chi\chi'}^m H_{\chi-1,\chi'}^m(1, 0) + b_{-\chi,\chi'}^m H_{\chi+1,\chi'}^m(-1, 0) \quad (\text{D.12})$$

$$G_{\chi\chi'}^m = a_{\chi\chi'}^m K_{\chi\chi'}^m(0, 0) + b_{\chi\chi'}^m K_{\chi-1,\chi'}^m(1, 0) + b_{-\chi,\chi'}^m K_{\chi+1,\chi'}^m(-1, 0) \quad (\text{D.13})$$

In the case of $\chi = \pm m$, certain terms in eq. (D.8) is out of definition. They are supposed to be zero. Another way is to suppose that

$$\begin{aligned} a_{\chi\chi'}^m &= 0 \quad \text{for } \chi = \pm m \\ b_{\chi\chi'}^m &= 0 \quad \text{for } \chi = \pm m \text{ and } \chi = \mp(m-1) \end{aligned} \quad (\text{D.14})$$

D.2.2 Recurrence relation for $-m + 2 \leq \chi' \leq m$

For the case $\chi' = \pm m$ which are not covered in eq. (D.8), another recurrence relation supposes that

$$R_{\chi\chi'}^m = c_{\chi\chi'}^m R_{0,1}^1 R_{\chi,\chi'-1}^{m-1} + d_{\chi\chi'}^m R_{1,1}^1 R_{\chi-1,\chi'-1}^{m-1} + d_{-\chi,\chi'}^m R_{-1,1}^1 R_{\chi+1,\chi'-1}^{m-1} \quad (\text{D.15})$$

$$F_{\chi\chi'}^m = c_{\chi\chi'}^m H_{\chi,\chi'-1}^m(0, 1) + d_{\chi\chi'}^m H_{\chi-1,\chi'-1}^m(1, 1) + d_{-\chi,\chi'}^m H_{\chi+1,\chi'-1}^m(-1, 1) \quad (\text{D.16})$$

$$G_{\chi\chi'}^m = c_{\chi\chi'}^m K_{\chi,\chi'-1}^m(0, 1) + d_{\chi\chi'}^m K_{\chi-1,\chi'-1}^m(1, 1) + d_{-\chi,\chi'}^m K_{\chi+1,\chi'-1}^m(-1, 1) \quad (\text{D.17})$$

with

$$\begin{aligned} c_{\chi\chi'}^m &= \left[\frac{2(m+\chi)(m-\chi)}{(m+\chi')(m+\chi'-1)} \right]^{\frac{1}{2}} \quad (-m+1 \leq \chi \leq m-1) \\ d_{\chi\chi'}^m &= \left[\frac{(m+\chi)(m+\chi-1)}{(m+\chi')(m+\chi'-1)} \right]^{\frac{1}{2}} \quad (-m+2 \leq \chi \leq m-2) \end{aligned} \quad (\text{D.18})$$

and

$$\begin{aligned} c_{\chi\chi'}^m &= 0 \quad \text{for } \chi = \pm m \\ d_{\chi\chi'}^m &= 0 \quad \text{for } \chi = \pm m \text{ and } \chi = \mp(m-1) \end{aligned} \quad (\text{D.19})$$

which is available for $-m + 2 \leq \chi' \leq m$.

D.2.3 Symmetries

The symmetries of $R_{\chi\chi'}^m$ allow us to calculate only a half of elements:

$$R_{-m,-m'}^l = (-1)^{m+m'} R_{mm'}^{l*} \quad (\text{D.20})$$

which gives

$$F_{-m,-m'}^l = (-1)^{m+m'} F_{mm'}^l \quad (\text{D.21})$$

$$G_{-m,-m'}^l = -(-1)^{m+m'} G_{mm'}^l \quad (\text{D.22})$$

E

PROPERTIES OF WIGNER 3J-SYMBOL AND GSH

The properties of Wigner 3j-symbol and Wigner generalized spherical harmonics (GSH, Wigner D-symbol) play a huge role in the reduction of molecular Ornstein-Zernike equation as well as finding the relation between rotational invariant projections. Their main properties presented in Messiah [36], Gray & Gubbins [18] and Edmonds [11] are listed here.

e.1 PROPERTIES OF WIGNER 3J-SYMBOL

Wigner 3j-symbols are equivalent to Clebsch-Gordon (CG) coefficients with phase factor:

$$\begin{pmatrix} m & n & l \\ \mu & \nu & -\lambda \end{pmatrix} = \frac{(-)^{m-n+\lambda}}{\sqrt{2l+1}} \langle mn\mu\nu | l\lambda \rangle \quad (\text{E.1})$$

and can be calculated with the Racah formula [36].

Reality

The 3j-symbols are real.

$$\begin{pmatrix} m & n & l \\ \mu & \nu & \lambda \end{pmatrix} = \begin{pmatrix} m & n & l \\ \mu & \nu & \lambda \end{pmatrix}^* \quad (\text{E.2})$$

Selection rules

$$\begin{pmatrix} m & n & l \\ \mu & \nu & \lambda \end{pmatrix} = 0 \text{ if } \begin{cases} \mu + \nu + \lambda = 0 \\ |m - n| < l < m + n \\ \text{(triangular inequalities)} \end{cases} \text{ are not meet.} \quad (\text{E.3})$$

Permutation

1. Even permutation

$$\begin{pmatrix} m & n & l \\ \mu & \nu & \lambda \end{pmatrix} = \begin{pmatrix} n & l & m \\ \nu & \lambda & \mu \end{pmatrix} = \begin{pmatrix} l & m & n \\ \lambda & \mu & \nu \end{pmatrix} \quad (\text{E.4})$$

2. Odd permutation

$$\begin{aligned} (-)^{m+n+l} \begin{pmatrix} m & n & l \\ \mu & \nu & \lambda \end{pmatrix} &= \begin{pmatrix} n & m & l \\ \nu & \mu & \lambda \end{pmatrix} \\ &= \begin{pmatrix} m & l & n \\ \mu & \lambda & \nu \end{pmatrix} = \begin{pmatrix} l & n & m \\ \lambda & \nu & \mu \end{pmatrix} \end{aligned} \quad (\text{E.5})$$

3. Simultaneous change of signs of μ , ν and λ

$$\begin{pmatrix} m & n & l \\ \mu & \nu & \lambda \end{pmatrix} = (-)^{m+n+l} \begin{pmatrix} m & n & l \\ -\mu & -\nu & -\lambda \end{pmatrix} \quad (\text{E.6})$$

Orthogonality

$$\sum_{l=|m-n|}^{m+n} \sum_{\lambda=-l}^l (2l+1) \begin{pmatrix} m & n & l \\ \mu & \nu & \lambda \end{pmatrix} \begin{pmatrix} m & n & l \\ \mu' & \nu' & \lambda \end{pmatrix} = \delta_{\mu\mu'} \delta_{\nu\nu'} \quad (\text{E.7})$$

$$\sum_{\mu=-m}^m \sum_{\nu=-n}^n \begin{pmatrix} m & n & l \\ \mu & \nu & \lambda \end{pmatrix} \begin{pmatrix} m & n & l' \\ \mu & \nu & \lambda' \end{pmatrix} = (2l+1)^{-1} \delta_{ll'} \delta_{\lambda\lambda'} \quad (\text{E.8})$$

e.2 PROPERTIES OF GSH

There are many different definitions of GSH in the lectures, here we adopt the definition in Messiah

$$R_{\mu'\mu}^m(\phi\theta\psi) = e^{-i\mu'\phi} r_{\mu'\mu}^m(\theta) e^{-i\mu\psi} \quad (\text{E.9})$$

where $r_{\mu\mu'}^m$ is the generalized Legendre polynomial (GLP), which is real, and can be evaluated using the Wigner formula

$$\begin{aligned} r_{\mu'\mu}^m(\theta) &= [(m+\mu')! (m-\mu')! (m+\mu)! (m-\mu)!]^{\frac{1}{2}} \times \\ &\sum_i \frac{(-)^i (\cos \theta/2)^{2m+\mu'-\mu-2i} (\sin \theta/2)^{2i-\mu'+\mu}}{(m+\mu'-i)! (m-\mu-i)! i! (i-\mu'+\mu)!} \end{aligned} \quad (\text{E.10})$$

Symmetries of $r_{\mu'\mu}^m(\theta)$

$$r_{\mu\mu'}^m(\theta) = (-)^{\mu'-\mu} r_{\mu'\mu}^m(\theta) \quad (\text{E.11})$$

$$\underline{r}_{\mu'\mu}^m(\theta) = (-)^{\mu'-\mu} r_{\mu'\mu}^m(\theta) \quad (\text{E.12})$$

$$r_{\mu'\mu}^m(\theta) = r_{\mu\mu'}^m(-\theta) \quad (\text{E.13})$$

$$r_{\mu'\mu}^m(\theta + \pi) = (-)^{m+\mu} r_{\mu'\mu}^m(\theta) \quad (\text{E.14})$$

where $\underline{\mu} \equiv -\mu$.

Symmetries of $R_{\mu'\mu}^m(\phi\theta\psi)$

$$R_{\mu'\mu}^m(\phi\theta\psi) = (-)^{\mu'-\mu} R_{\mu'\mu}^{m*}(\phi\theta\psi) \quad (\text{E.15})$$

$$R_{\mu'\mu}^m(\phi\theta\psi) = (-)^{\mu'-\mu} R_{\mu\mu'}^{m*}(\phi\theta\psi) \quad (\text{E.16})$$

$$R_{\mu'\mu}^m(\phi\theta\psi) = (-)^{m+\mu'} R_{\mu'\mu}^m(-\phi, \theta + \pi, \psi) = (-)^{m+\mu} R_{\mu'\mu}^m(\phi, \theta + \pi, -\psi) \quad (\text{E.17})$$

Unitarity and orthogonality

$$\sum_{\mu'} R_{\mu'\mu}^m(\phi\theta\psi) R_{\mu'\mu''}^{m*}(\phi\theta\psi) = \delta_{\mu\mu''} \quad (\text{E.18})$$

$$\sum_{\mu} R_{\mu'\mu}^m(\phi\theta\psi) R_{\mu''\mu}^{m*}(\phi\theta\psi) = \delta_{\mu'\mu''} \quad (\text{E.19})$$

$$\sum_{m\mu'\mu} R_{\mu'\mu}^m(\phi\theta\psi) R_{\mu'\mu'}^{m*}(\phi'\theta'\psi') = \delta_{\phi\phi'} \delta_{\theta\theta'} \delta_{\psi\psi'} \quad (\text{E.20})$$

$$\frac{1}{8\pi^2} \int d\cos\theta d\phi d\psi R_{\mu'\mu}^m(\phi\theta\psi) R_{\mu'\nu}^{n*}(\phi\theta\psi) = \frac{\delta_{mn}\delta_{\mu'\nu'}\delta_{\mu\nu}}{2n+1} \quad (\text{E.21})$$

$r_{\mu'\mu}^m(\theta)$ in terms of $\cos \theta$ and $\sin \theta$

1. If $(-)^{\mu'+\mu} = +1$, $r_{\mu'\mu}^m(\theta)$ is polynomial of degree m in $\cos \theta$.
2. If $(-)^{\mu'+\mu} = -1$, $r_{\mu'\mu}^m(\theta)/\sin \theta$ is polynomial of degree $(m-1)$ in $\cos \theta$.

Rotation and product

$$R_{\mu'\mu}^m(\omega) = \sum_{\chi} R_{\mu'\chi}^m(\omega_2) R_{\chi\mu}^m(\omega_1) \quad (\text{E.22})$$

where ω is the result of the successive application of ω_1 and ω_2 in order.

$$\begin{aligned} R_{\chi\mu}^m(\omega) &= \sum_{\mu'} R_{\mu'\chi}^{m*}(\hat{k}) R_{\mu'\mu}^m(\Omega) \\ R_{\mu'\mu}^m(\Omega) &= \sum_{\chi} R_{\chi\mu'}^{m*}(\hat{k}^{-1}) R_{\chi\mu}^m(\omega) = \sum_{\chi} R_{\mu'\chi}^m(\hat{k}) R_{\chi\mu}^m(\omega) \end{aligned} \quad (\text{E.23})$$

Composition relation for GSHs

$$\sum_{\mu'\nu'\lambda'} \begin{pmatrix} m & n & l \\ \mu' & \nu' & \lambda' \end{pmatrix} R_{\mu'\mu}^m(\phi\theta\psi) R_{\nu'\nu}^n(\phi\theta\psi) R_{\lambda'\lambda}^l(\phi\theta\psi) = \begin{pmatrix} m & n & l \\ \mu & \nu & \lambda \end{pmatrix} \quad (\text{E.24})$$

ERROR EVALUATION OF INTERPOLATION STRATEGIES FOR DCF IN LOCAL FRAME

The error introduced by the two interpolation orders for a DCF of order $n_{\max} = 1$ (for which the exact DCF can be computed directly; see details later) is shown in figure F.1.

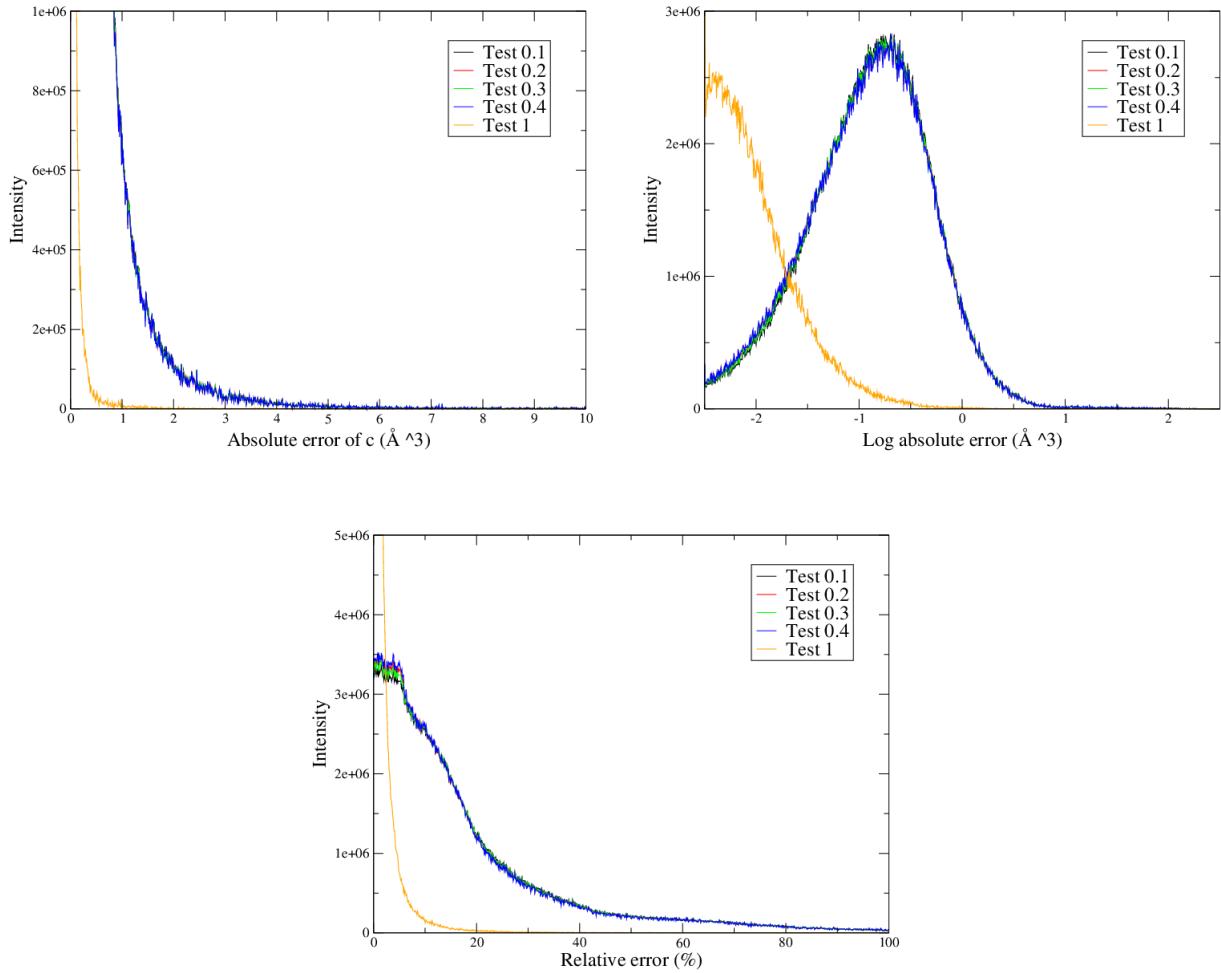


Figure F.1: Error of finding $\hat{c}(\mathbf{k}, \Omega_1, \Omega_2)$ by interpolation compared to direct calculation: Test 0.1-0.4 is zero order interpolation with ϕ tabulated as in figure 7.4. Test 1 is linear interpolation.

Absolute error is the histogram that counts the number of times that the calculated DCF gives the corresponding absolute error E_a^i with a resolution of 0.01, in range of [0, 10]:

$$E_a^i = |c_k^i - c_k| \quad (\text{F.1})$$

where c_k^i is any element of $\hat{c}(\mathbf{k}, \Omega_1, \Omega_2)$ of unity \AA^3 calculated as described and c_k is the one calculated directly as reference.

Log absolute error is treated the same way as E_a^i , with E_l^i defined as

$$E_l^i = \log |c_k^i - c_k| \quad (\text{F.2})$$

Relative error is defined as

$$E_r^i = |c_k^i - c_k| / |c_k| \quad (\text{F.3})$$

with resolution of 0.1%, in range of [0, 1].

In all three figures, the 4 curves given by zero order interpolation don't divers a lot compared with the linear interpolation one. The result of MDFT also shows that zero order interpolation gives large energy error with a DCF of $n_{\max} = 1$, and has convergency problems in certain cases. As a conclusion, the linear interpolation scheme is absolutely needed. On the other hand, as seen in eq. (7.16), it is computationally much more expensive than the simple histogram scheme as it requires $2^5 = 32$ times of operations.

ORIGINAL DATA OF MDFT IMPLEMENTATION

In this appendix, some original implementation data which is thought redundant to put in the main context are put here. These details may contributes to further researches and gives a global image of the parameter and version sensibility of code MDFT.

g.1 SERIES OF CHARGED CH₄ CENTER

These tests are done with the sequential code.

G.1.1 *Branch “naive” result*

with Lebedev quadrature

G.1.2 *Branch “standard” result*

BIBLIOGRAPHY

- [1] Luc Belloni and Ioulia Chikina. “Efficient full Newton - Raphson technique for the solution of molecular integral equations - example of the SPC/E water-like system.” In: *Molecular Physics* 112.9-10 (2014), pp. 1246–1256. DOI: 10.1080/00268976.2014.885612. eprint: <http://dx.doi.org/10.1080/00268976.2014.885612>. URL: <http://dx.doi.org/10.1080/00268976.2014.885612> (cit. on p. 99).
- [2] H. J. C. Berendsen, J. R. Grigera, and T. P. Straatsma. “The missing term in effective pair potentials.” In: *The Journal of Physical Chemistry* 91.24 (1987), pp. 6269–6271. DOI: 10.1021/j100308a038. eprint: <http://dx.doi.org/10.1021/j100308a038>. URL: <http://dx.doi.org/10.1021/j100308a038> (cit. on p. 15).
- [3] L. Blum. “Invariant Expansion. II. The Ornstein Zernike-Equation for Nonspherical Molecules and an Extended Solution to the Mean Spherical Model.” In: *The Journal of Chemical Physics* 57.5 (1972), pp. 1862–1869. DOI: <http://dx.doi.org/10.1063/1.1678503>. URL: <http://scitation.aip.org/content/aip/journal/jcp/57/5/10.1063/1.1678503> (cit. on pp. 2, 39, 45, 47, 48).
- [4] L. Blum and A. J. Torruella. “Invariant Expansion for Two-Body Correlations: Thermodynamic Functions, Scattering, and the Ornstein-Zernike Equation.” In: *The Journal of Chemical Physics* 56.1 (1972), pp. 303–310. DOI: <http://dx.doi.org/10.1063/1.1676864>. URL: <http://scitation.aip.org/content/aip/journal/jcp/56/1/10.1063/1.1676864> (cit. on pp. 2, 39, 45, 47, 99).
- [5] Richard H. Byrd, Peihuang Lu, Jorge Nocedal, and Ciyou Zhu. “A Limited Memory Algorithm for Bound Constrained Optimization.” In: *SIAM J. Sci. Comput.* 16.5 (Sept. 1995), pp. 1190–1208. DOI: 10.1137/0916069. URL: <http://dx.doi.org/10.1137/0916069> (cit. on p. 33).
- [6] David Chandler and Hans C. Andersen. “Optimized Cluster Expansions for Classical Fluids. II. Theory of Molecular Liquids.” In: *The Journal of Chemical Physics* 57.5 (1972), pp. 1930–1937. DOI: <http://dx.doi.org/10.1063/1.1678513>. URL: <http://scitation.aip.org/content/aip/journal/jcp/57/5/10.1063/1.1678513> (cit. on p. 2).
- [7] Martin Chaplin. *Water Model*. URL: http://www1.lsbu.ac.uk/water/water_models.html (cit. on p. 15).
- [8] Cheol Ho Choi, Joseph Ivanic, Mark S. Gordon, and Klaus Ruedenberg. “Rapid and stable determination of rotation matrices between spherical harmonics by direct recursion.” In: *The Journal of Chemical Physics* 111.19 (1999), pp. 8825–8831. DOI: <http://dx.doi.org/10.1063/1.480229>. URL: <http://scitation.aip.org/content/aip/journal/jcp/111/19/10.1063/1.480229> (cit. on p. 107).
- [9] Christopher J. Cramer and Donald G. Truhlar. “Implicit Solvation Models: Equilibria, Structure, Spectra, and Dynamics.” In: *Chemical Reviews* 99.8 (1999). PMID: 11849023, pp. 2161–2200. DOI: 10.1021/cr960149m. eprint: <http://dx.doi.org/10.1021/cr960149m>. URL: <http://dx.doi.org/10.1021/cr960149m> (cit. on p. 1).

- [10] Liem X. Dang, Julia E. Rice, and Peter A. Kollman. “The effect of water models on the interaction of the sodium–chloride ion pair in water: Molecular dynamics simulations.” In: *The Journal of Chemical Physics* 93.10 (1990), pp. 7528–7529. DOI: <http://dx.doi.org/10.1063/1.459714>. URL: <http://scitation.aip.org/content/aip/journal/jcp/93/10/10.1063/1.459714> (cit. on p. 3).
- [11] A. R. Edmonds. *Angular Momentum in Quantum Mechanics*. 1960 (cit. on pp. 99–101, 107, 111).
- [12] R. Evans. “The nature of the liquid-vapour interface and other topics in the statistical mechanics of non-uniform, classical fluids.” In: *Advances in Physics* 28.2 (1979), pp. 143–200. DOI: 10.1080/00018737900101365. eprint: <http://dx.doi.org/10.1080/00018737900101365>. URL: <http://dx.doi.org/10.1080/00018737900101365> (cit. on p. 2).
- [13] Sebastiao Formosinho and Monica Barroso, eds. *Proton-Coupled Electron Transfer. A Carrefour of Chemical Reactivity Traditions*. RSC Catalysis Series. The Royal Society of Chemistry, 2012, P001–157. DOI: 10.1039/9781849733168. URL: <http://dx.doi.org/10.1039/9781849733168> (cit. on p. 1).
- [14] P. H. Fries and G. N. Patey. “The solution of the hypernetted-chain approximation for fluids of nonspherical particles. A general method with application to dipolar hard spheres.” In: *The Journal of Chemical Physics* 82.1 (Jan. 1985), pp. 429–440. DOI: [doi:10.1063/1.448764](https://doi.org/10.1063/1.448764). URL: http://jcp.aip.org/resource/1/jcpa6/v82/i1/p429_s1 (visited on 01/15/2013) (cit. on pp. 2, 39, 47, 99).
- [15] M. Frigo and S.G. Johnson. “The Design and Implementation of FFTW3.” In: *Proceedings of the IEEE* 93.2 (2005), pp. 216–231. DOI: 10.1109/JPROC.2004.840301 (cit. on pp. 51, 53, 67).
- [16] Peter Gács and László Lovász. *Complexity of Algorithms*. Lecture Notes. 1999 (cit. on p. 93).
- [17] Gaussian 09 keyword: SCRF. 2014. URL: http://www.gaussian.com/g_tech/g_ur/k_scrf.htm (cit. on pp. 1, 12).
- [18] C.G. Gray and K.E. Gubbins. *Theory of Molecular Fluids: I: Fundamentals*. International Series of Monographs on Chemistry. OUP Oxford, 1984. URL: <http://books.google.de/books?id=3mz2RcnnMGwC> (cit. on pp. 1, 2, 7, 12, 13, 15, 51, 100, 101, 107, 111).
- [19] Georg Hager and Gerhard Wellein. *Formation lecture: Node-Level Performance Engineering*. Dec. 2015. URL: <http://moodle.rrze.uni-erlangen.de/course/view.php?id=274> (cit. on p. 95).
- [20] J. P. Hansen. “Basic Statistical Theory of Liquids.” In: *The Physics and Chemistry of Aqueous Ionic Solutions*. Ed. by M.-C Bellissent-Funel and G. W. Neilson. Dordrecht: Springer Netherlands, 1987, pp. 1–59. DOI: 10.1007/978-94-009-3911-0_1. URL: http://dx.doi.org/10.1007/978-94-009-3911-0_1 (cit. on p. 2).
- [21] Jean-Pierre Hansen and Ian R. McDonald. *Theory of Simple Liquids*. Ed. by Jean-Pierre Hansen and Ian R. McDonald. Fourth Edition. Oxford: Academic Press, 2013 (cit. on pp. 2, 12, 28–30).
- [22] Robert B. Hermann. “Theory of hydrophobic bonding. II. Correlation of hydrocarbon solubility in water with solvent cavity surface area.” In: *The Journal of Physical Chemistry* 76.19 (1972), pp. 2754–2759. DOI: 10.1021/j100663a023. eprint: <http://dx.doi.org/10.1021/j100663a023>. URL: <http://dx.doi.org/10.1021/j100663a023> (cit. on p. 8).

- [23] Fumio Hirata, ed. *Molecular Theory of Solvation*. en. Vol. 24. Understanding Chemical Reactivity. Dordrecht: Kluwer Academic Publishers, 2004. URL: <http://link.springer.com/10.1007/1-4020-2590-4> (visited on 08/10/2016) (cit. on pp. 2, 11, 23).
- [24] Laura D. Hughes, David S. Palmer, Florian Nigsch, and John B. O. Mitchell. “Why are some properties more difficult to predict than others? A study of QSPR models of solubility, melting point, and Log P.” In: *Journal of Chemical Information and Modeling* 48.1 (Jan. 2008), pp. 220–232. DOI: 10.1021/ci700307p (cit. on p. 1).
- [25] Philippe Hunenberger and Maria Reif. “Chapter 2 Fundamental Experimental Problems.” In: *Single-Ion Solvation: Experimental and Theoretical Approaches to Elusive Thermodynamic Quantities*. The Royal Society of Chemistry, 2011, pp. 8–38. DOI: 10.1039/9781849732222-00008. URL: <http://dx.doi.org/10.1039/9781849732222-00008> (cit. on p. 57).
- [26] Frank Jensen. *Introduction to Computational Chemistry*. John Wiley & Sons, 2006 (cit. on pp. 1, 7, 12).
- [27] William L. Jorgensen and Julian Tirado-Rives. “Free energies of hydration for organic molecules from Monte Carlo simulations.” In: *Perspectives in Drug Discovery and Design* 3.1 (1995), pp. 123–138. DOI: 10.1007/BF02174470. URL: <http://dx.doi.org/10.1007/BF02174470> (cit. on p. 2).
- [28] M. A. Kastenholz and Philippe H. Hünenberger. “Computation of methodology-independent ionic solvation free energies from molecular simulations. I. The electrostatic potential in molecular liquids.” In: *The Journal of Chemical Physics* 124.12, 124106 (2006). DOI: <http://dx.doi.org/10.1063/1.2172593>. URL: <http://scitation.aip.org/content/aip/journal/jcp/124/12/10.1063/1.2172593> (cit. on pp. 57, 58).
- [29] Mika A. Kastenholz and Philippe H. Hünenberger. “Computation of methodology-independent ionic solvation free energies from molecular simulations. II. The hydration free energy of the sodium cation.” In: *The Journal of Chemical Physics* 124.22, 224501 (2006). DOI: <http://dx.doi.org/10.1063/1.2201698>. URL: <http://scitation.aip.org/content/aip/journal/jcp/124/22/10.1063/1.2201698> (cit. on p. 57).
- [30] Peter G. Kusalik and Igor M. Svishchev. “The Spatial Structure in Liquid Water.” In: *Science* 265.5176 (1994), pp. 1219–1221. DOI: 10.1126/science.265.5176.1219. eprint: <http://science.sciencemag.org/content/265/5176/1219.full.pdf>. URL: <http://science.sciencemag.org/content/265/5176/1219> (cit. on p. 16).
- [31] B. Lee and F.M. Richards. “The interpretation of protein structures: Estimation of static accessibility.” In: *Journal of Molecular Biology* 55.3 (1971), 379 –IN4. DOI: [http://dx.doi.org/10.1016/0022-2836\(71\)90324-X](http://dx.doi.org/10.1016/0022-2836(71)90324-X). URL: <http://www.sciencedirect.com/science/article/pii/002228367190324X> (cit. on p. 8).
- [32] Massimo Marchi, Daniel Borgis, Nicolas Levy, and Pietro Ballone. “A dielectric continuum molecular dynamics method.” In: *The Journal of Chemical Physics* 114.10 (2001), pp. 4377–4385. DOI: <http://dx.doi.org/10.1063/1.1348028>. URL: <http://scitation.aip.org/content/aip/journal/jcp/114/10/10.1063/1.1348028> (cit. on pp. 11, 35).

- [33] Lorenzo Maschio, Bernard Kirtman, Roberto Orlando, and Michel Rèrat. “Ab initio analytical infrared intensities for periodic systems through a coupled perturbed Hartree-Fock/Kohn-Sham method.” In: *The Journal of Chemical Physics* 137.20, 204113 (2012). DOI: <http://dx.doi.org/10.1063/1.4767438>. URL: <http://scitation.aip.org/content/aip/journal/jcp/137/20/10.1063/1.4767438> (cit. on p. 58).
- [34] A. D. Mcnaught and A. Wilkinson. *IUPAC Compendium of Chemical Terminology*, 2nd ed. (the "Gold Book"). WileyBlackwell; 2nd Revised edition edition (cit. on p. 1).
- [35] N. David Mermin. “Thermal Properties of the Inhomogeneous Electron Gas.” In: *Physical Review* 137.5A (Mar. 1965), A1441–A1443. DOI: 10.1103/PhysRev.137.A1441. URL: <http://link.aps.org/doi/10.1103/PhysRev.137.A1441> (visited on 02/10/2014) (cit. on p. 2).
- [36] Albert Messiah. *Quantum Mechanics*. Quantum Mechanics vol. 2. North-Holland, 1981. URL: https://books.google.fr/books?id=VR93vUk8d__C (cit. on pp. 51, 99–101, 107, 111).
- [37] B. R. A. Nijboer and Th. W. Ruijgrok. “On the energy per particle in three- and two-dimensional Wigner lattices.” In: *Journal of Statistical Physics* 53.1 (), pp. 361–382. DOI: 10.1007/BF01011562. URL: <http://dx.doi.org/10.1007/BF01011562> (cit. on p. 57).
- [38] Trenton H. Parsell, Meng-Yin Yang, and A. S. Borovik. “C-H Bond Cleavage with Reductants: Re-Investigating the Reactivity of Monomeric MnIII/IV-Oxo Complexes and the Role of Oxo Ligand Basicity.” In: *Journal of the American Chemical Society* 131.8 (2009). PMID: 19196005, pp. 2762–2763. DOI: 10.1021/ja8100825. eprint: <http://dx.doi.org/10.1021/ja8100825>. URL: <http://dx.doi.org/10.1021/ja8100825> (cit. on p. 1).
- [39] German L. Perlovich, Tatyana V. Volkova, and Annette Bauer-Brandl. “Towards an understanding of the molecular mechanism of solvation of drug molecules: A thermodynamic approach by crystal lattice energy, sublimation, and solubility exemplified by paracetamol, acetanilide, and phenacetin.” In: *Journal of Pharmaceutical Sciences* 95.10 (), pp. 2158–2169. DOI: 10.1002/jps.20674. URL: <http://dx.doi.org/10.1002/jps.20674> (cit. on p. 1).
- [40] GermanL Perlovich and Annette Bauer-Brandl. “Solvation of Drugs as a Key for Understanding Partitioning and Passive Transport Exemplified by NSAIDs.” In: *Chemical Processes with Participation of Biological and Related Compounds*. CRC Press, 2008, pp. 291–325. DOI: doi:10.1201/b12241-9. URL: <http://dx.doi.org/10.1201/b12241-9> (cit. on p. 1).
- [41] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes 3rd Edition: The Art of Scientific Computing*. 3rd ed. New York, NY, USA: Cambridge University Press, 2007 (cit. on pp. 50, 67, 74, 97).
- [42] Joël Puibasset and Luc Belloni. “Bridge function for the dipolar fluid from simulation.” In: *The Journal of Chemical Physics* 136.15, 154503 (2012), pp. –. DOI: <http://dx.doi.org/10.1063/1.4703899>. URL: <http://scitation.aip.org/content/aip/journal/jcp/136/15/10.1063/1.4703899> (cit. on p. 37).
- [43] R E Raab and O L De Lange. *Multipole theory in electromagnetism: classical, quantum, and symmetry aspects, with applications*. International series of monographs on physics. Oxford: Clarendon Press, 2005. URL: <https://cds.cern.ch/record/859649> (cit. on p. 58).

- [44] V.R. Saunders, C. Freyria-Fava, R. Dovesi, L. Salasco, and C. Roetti. “On the electrostatic potential in crystalline systems where the charge density is expanded in Gaussian functions.” In: *Molecular Physics* 77.4 (1992), pp. 629–665. DOI: 10.1080/00268979200102671. eprint: <http://dx.doi.org/10.1080/00268979200102671> (cit. on p. 58).
- [45] Michiel Sprik and Michael L. Klein. “A polarizable model for water using distributed charge sites.” In: *The Journal of Chemical Physics* 89.12 (1988), pp. 7556–7560. DOI: <http://dx.doi.org/10.1063/1.455722>. URL: <http://scitation.aip.org/content/aip/journal/jcp/89/12/10.1063/1.455722> (cit. on p. 3).
- [46] Jacopo Tomasi and Maurizio Persico. “Molecular Interactions in Solution: An Overview of Methods Based on Continuous Distributions of the Solvent.” In: *Chemical Reviews* 94.7 (1994), pp. 2027–2094. DOI: 10.1021/cr00031a013. eprint: <http://dx.doi.org/10.1021/cr00031a013>. URL: <http://dx.doi.org/10.1021/cr00031a013> (cit. on p. 1).
- [47] Samuel Williams, Andrew Waterman, and David Patterson. “Roofline: An Insightful Visual Performance Model for Multicore Architectures.” In: *Commun. ACM* 52.4 (Apr. 2009), pp. 65–76. DOI: 10.1145/1498765.1498785. URL: <http://doi.acm.org/10.1145/1498765.1498785> (cit. on p. 94).
- [48] Shuangliang Zhao, Rosa Ramirez, Rodolphe Vuilleumier, and Daniel Borgis. “Molecular density functional theory of solvation: From polar solvents to water.” In: *The Journal of Chemical Physics* 134.19, 194102 (2011). DOI: <http://dx.doi.org/10.1063/1.3589142>. URL: <http://scitation.aip.org/content/aip/journal/jcp/134/19/10.1063/1.3589142> (cit. on pp. 36, 39).
- [49] Ciyou Zhu, Richard H. Byrd, Peihuang Lu, and Jorge Nocedal. “Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization.” In: *ACM Transactions on Mathematical Software* 23.4 (Dec. 1997), pp. 550–560. DOI: 10.1145/279232.279236. URL: <https://acces-distant.upmc.fr/http/portal.acm.org/citation.cfm?doid=279232.279236> (visited on 06/09/2011) (cit. on p. 33).