Normandie Université

THÈSE

Pour obtenir le diplôme de doctorat

Spécialité Chimie

Préparée au sein de l'Université de Rouen Normandie

Réactivité chimique en solution dans une approche hybride DFT Conceptuelle et QTAIM

Présentée et soutenue par

Victoria CASTOR VILLEGAS

Thèse soutenue publiquement le 16/10/2025 devant le jury composé de		
M. Frédéric GUÉGAN	Maître de Conférences HDR (Université de Poitiers)	Rapporteur
M. Julien PILMÉ	Maître de Conférences HDR (Sorbonne Université)	Rapporteur
Mme. Isabelle CHATAIGNER	Professeur des Universités (Université de Rouen)	Examinatrice
M. Aurélien MONCOMBLE	Maître de Conférences HDR (Université de Lille)	Examinateur
M. José Manuel GUEVARA VELA	Research Fellow (Heriot-Watt University, Edinburgh, Scotland)	Examinateur
M. Laurent JOUBERT	Professeur des Universités (Université de Rouen)	Directeur
M. Vincent TOGNETTI	Maître de Conférences HDR (Université de Rouen)	Directeur

Thèse dirigée par LAURENT JOUBERT et VINCENT TOGNETTI , Institut CARMeN

UNIVERSITÉ DE ROUEN NORMANDIE

ED 508 NC Normande de Chimie

CARMeN

„New thesis,
Same old mistakes“.

“Dear Sir or Madam,
will you read my thesis?
it took me years to write,
will you take a look?”

«I’ve got nothing to say,
but it’s okay.»

Acknowledgements

I'd like to express all my gratitude to all those people who have helped me in one way or another to reach the point where i am today. They all —with or without the best intentions— have contributed to creating the person i am, and therefore, to the work i present today.

Je veux remercier l'Université de Rouen de m'avoir offert l'opportunité de réaliser mon doctorat, ainsi que le centre CRIANN pour le temps de calcul et l'infrastructure mise à disposition. J'exprime aussi ma gratitude à tous les membres du jury —rapporteurs, examinateurs et directeurs—, non seulement pour le temps consacré à évaluer ce travail, mais aussi pour l'intérêt et la confiance qu'ils ont portés à ma recherche.

Merci à Laurent Joubert et Vincent Tognetti pour leur encadrement et leur patience tout au long de cette thèse. À Julien Legros et Philippe Jubault, directeur et directeur-adjoint de l'Institut CARMeN, Frédéric Guégan, Julien Pilmé, Isabelle Chataigner et Aurélien Moncomble, merci pour le temps consacré à la relecture de ce manuscrit,

... y particularmente gracias a Toche (José Manuel Guevara Vela).

Software for Chemistry and Materials, bedankt! (et takk), it was a real pleasure to work with you. Being there during the long-awaited office move from the W&N to the OZW building, after years of hearing "we're moving in a few weeks". A bit of organised chaos gave rhythm to the days, het was gezellig.

À mes collègues de recherche d'hier et d'aujourd'hui,
never forgetting the one who is not physically with us,
but always in all research groups,
Алексáндра Элбакян (Alexandra Elbakyan).

Cinco años viviendo en otro continente, catorce (o más) horas de viaje de por medio u ocho horas de diferencia horaria no me han separado de mi núcleo familiar. Libia, Victor y Ana habéis sido un pilar en mi vida, sin ustedes no hubiera sido posible llegar hasta aquí…

it's been a long ride since my bachelor's days, isn't it?

Pau 🦕 LC and Andrea 🥕 FL,

idk how but you are not tired of me (yet).

And yeah… how could i forget the adventures that coloured this PhD? Skiing in Norway, forgetting my passport going to Barcelona, getting lost in Paris a thousand times with Trinidad, getting sick in Köln with Pau… i've lost count of my trips to Madrid, and of all the huge tiny things that made this time unforgettable: Amsterdam, Wien, Mexico…

et bien sûr, Rouen —tu… ma petite ville grise et pluvieuse,

tu es aussi réconfortante—

merci d'avoir été un chez moi quand j'en avais besoin.

非常感谢

✨

~VCastor, 2025

Contents

Acronyms

AI	*Artificial Inteligence*
BCP	*Bond Critical Point*
BOA	*Bond Order Analysis*
CCP	*Cage Critical Point*
CDFT	*Conceptual Density Functional Theory*
CP	*Critical Point*
DFT	*Density Functional Theory*
DFTB	*Density Functional Tight Binding*
FMO	*Frontier Molecular Orbital*
GGA	*Generalized-Gradient Approximation*
GTO	*Gaussian Type Orbital*
GUI	*Graphical User Interface*
HF	*Hartree-Fock*
HFRH	*Hartree-Fock-Roothaan-Hall*
HK	*Hohenberg-Kohn*
KS	*Kohn-Sham*
LCAO	*Linear Combination of Atomic Orbitals*
LDA	*Local Density Approximation*
ML	*Machine Learning*
NCP	*Nuclear Critical Point*
NNA	*Non-Nuclear Attractor*
QTAIM	*Quantum Theory of Atoms In Molecules*
RCP	*Ring Critical Point*
SCF	*Self Consistent Field*
STO	*Slater Type Orbital*
SVC	*Support Vector Clustering*
SVM	*Support Vector Machine*

Chapter

1 Introduction

Comprendre et prédire la réactivité chimique en solution constitue un défi majeur en chimie théorique. Cette réactivité dépend fondamentalement de deux composantes : la contribution électronique, gouvernée par la structure électronique des molécules, et la faisabilité thermodynamique, déterminée par l'énergie de Gibbs $G = E_{\text{electronic}} + \Delta H - TS$, qui combine l'énergie électronique, les corrections thermiques et l'entropie [1].

Dans ce contexte, l'analyse et l'interprétation des descripteurs quantiques s'avèrent indispensables pour la compréhension des systèmes chimiques. La densité électronique, les orbitales moléculaires, les descripteurs issus de la DFT conceptuelle (CDFT) [2], ainsi que les topologies dérivées de la théorie quantique des atomes dans les molécules (QTAIM) [3], constituent autant d'outils complémentaires permettant de décrypter la réactivité chimique. Toutefois, la synergie entre ces approches, et en particulier l'intégration des informations topologiques et conceptuelles, reste encore peu explorée à ce jour.

Ce travail propose une approche intégrée, alliant QTAIM et CDFT, afin d'accéder à une représentation multiscalaire de la réactivité. La QTAIM permet une partition rigoureuse de l'espace moléculaire en régions atomiques, à partir desquelles peuvent être définis des descripteurs localisés ou condensés tels que les charges, les moments dipolaires ou les polarisabilités. Ces grandeurs sont directement liées à la topologie de la densité électronique, et offrent une lecture spatialisée du comportement chimique.

La DFT conceptuelle, de son côté, propose un cadre théorique puissant pour l'analyse de la réactivité chimique à travers des descripteurs variés. Elle permet de définir, au moyen de la densité électronique, des indices globaux, locaux et non-locaux de réactivité, tels que le potentiel chimique electronique, entre autres. L'intégration de ces descripteurs au sein d'un cadre QTAIM offre la possibilité de concilier une analyse énergétique et structurelle, tout en conservant une interprétabilité chimique forte. Cette approche hybride a été implémentée dans le software Amsterdam Modeling Suite (AMS) [4], avec de nouveaux modules développés au cours de cette thèse.

La dimension thermodynamique est également prise en compte via une modélisation conformationnelle explicite. En effet, les propriétés mesurées expérimentalement correspondent souvent à une moyenne sur plusieurs conformères, particulièrement en phase condensée. Pour en rendre compte, notre méthodologie combine un échantillonnage rapide basé sur le DFTB [5] à une réoptimisation DFT [6] de haute précision, suivie d'une moyenne pondérée par la distribution de Boltzmann [7]. Ce protocole permet d'obtenir des descripteurs réalistes, intégrant les effets entropiques et conformationnels.

Dans certains cas, notamment pour l'étude de systèmes biologiques ou de réactions en milieu hétérogène, il est nécessaire d'avoir recours à des approches multi-échelles. Le formalisme mécanique quantique/mécanique moléculaire et en particulier le schéma de Polarisable Embedding [8], permet de tenir compte de la polarisation mutuelle entre une région traitée en mécanique quantique et un environnement classique. Ces méthodes nécessitent des descripteurs électrostatiques précis, comme les charges et les polarisabilités atomiques, pour lesquels la QTAIM constitue une base naturelle. Toutefois, les bases de données disponibles, notamment via la bibliothèque PyFrame [9], sont encore limitées en couverture chimique et en compatibilité avec les modèles avancés de solvatation.

En résumé, cette thèse s'inscrit dans une démarche de modélisation intégrée de la réactivité chimique en solution, en développant des outils robustes pour l'analyse électronique, la prise en compte des effets entropiques et conformationnels, et l'exploitation des méthodes de solvatation. En conjuguant la rigueur de QTAIM, la pertinence des descripteurs CDFT, et le potentiel de l'apprentissage automatique, nous proposons une boîte à outils numérique polyvalente, extensible, et physiquement interprétable, à même d'améliorer l'analyse et la prédiction de la réactivité dans des milieux complexes.

1.1 Objectifs

L'objectif principal de cette thèse est de développer une méthodologie computationnelle intégrée pour l'analyse et la prédiction de descripteurs chimiques en solution. Cette approche vise à concilier la rigueur des fondements théoriques (QTAIM et CDFT) avec des techniques modernes de modélisation moléculaire, dans un cadre compatible avec la chimie en phase condensée.

Les objectifs spécifiques de ce travail sont les suivants :

- Étendre l'analyse topologique à des éléments complexes de QTAIM, tels que les anneaux, cages et attracteurs non nucléaires, pour enrichir la description de la structure électronique.
- Développer et intégrer dans AMS des routines permettant le calcul de moments dipolaires et de polarisabilités atomiques dans le cadre de QTAIM à partir de la densité électronique issue de calculs DFT.
- Mettre en place un protocole combinant la vitesse DFTB avec la précision DFT pour l'échantillonnage conformationnel, et intégrer une moyenne thermodynamique via la distribution de Boltzmann afin de refléter les effets entropiques et statistiques.
- Mettre à profit des modèles d'apprentissage automatique, en utilisant comme variables d'entrée des descripteurs QTAIM et CDFT, pour prédire des propriétés chimiques telles que la nucléophilie.

L'ensemble de ces objectifs vise à produire des outils numériques robustes, interprétables, et transférables, capables de contribuer à la compréhension fine de la réactivité chimique en solution et à son intégration dans des modèles prédictifs à l'interface entre la théorie et l'expérience.

Chapter

2 Theoretical Foundations

In this chapter, we introduce the theoretical background that underpins this thesis, rather than embarking on a historical journey, traversing through the philosophies of the ancient Greeks or beginning with the standard model of particle physics —one of the most successful theories in modern science— we will provide a concise and targeted overview that will enable the reader to grasp the context of our work within the existing body of knowledge. Our aim is not to reconstruct the entire chemistry from scratch.

We assume the reader possesses a solid grounding in science, including familiarity with concepts from infinitesimal calculus, as well as terms originating in chemistry and physics, such as nucleophilicity and entropy.

2.1 Quantum Mechanics Foundations

"*I think I can safely say that nobody understands quantum mechanics*".

-Richard Feynman

In this section, we take as our starting point the well-known Schrödinger Equation [10], postulated by Erwin Schrödinger in 1925. This equation governs the quantum dynamics of a single particle of mass m evolving under the influence of a potential V.

Definition 2.1.1: Schrödinger Equation

$$i\hbar\frac{\partial}{\partial t}\left|\Psi(\mathbf{r},t)\right\rangle = \left[\frac{-\hbar^2}{2m}\nabla^2 + \widehat{V}(\mathbf{r},t)\right]\left|\Psi\right\rangle = \widehat{H}\left|\Psi(\mathbf{r},t)\right\rangle \tag{2.1}$$

In cases where the Hamiltonian does not depend explicitly on time, we may apply the method of separation of variables. This allows us to factor the wavefunction as $\Psi(\mathbf{r},t) = e^{-iEt/\hbar}\psi(\mathbf{r})$ and derive the time-independent Schrödinger equation:

Key Equation 1: Time-Independent Schrödinger Equation

$$\widehat{H}\left|\psi(\mathbf{r})\right\rangle = E\left|\psi(\mathbf{r})\right\rangle, \tag{2.2}$$

where E denotes the energy eigenvalue associated with the stationary state $|\psi\rangle$.

While the Schrödinger equation offers a fundamental description of non-relativistic quantum systems, a complete treatment must incorporate relativistic effects, using the Dirac equation [11]. Rather than delving into the complexities of relativistic quantum mechanics, we focus on the challenges posed by molecules with many electrons. These systems are particularly difficult to solve, as no analytic solutions are available using current state-of-the-art mathematical techniques.

The complexity and intractability of solving many-electron molecular systems necessitate the use of approximations. One of the earliest and most fundamental of these is the Born-Oppenheimer Approximation (BOA). Although the BOA is inherently contradictory to the principles of quantum mechanics, it remains a powerful tool due to the significant mass difference between nuclei and electrons.

Insight 2.1.1 (Born-Oppenheimer Approximation)
A full quantum mechanical description of a time-independent system is given by the time-independent Schrödinger equation:

$$\widehat{H}\,|\psi\rangle = E\,|\psi\rangle\,, \tag{2.3}$$

where the total energy E can be separated into nuclear and electronic contributions:

$$E = E_n + E_e. \tag{2.4}$$

The total Hamiltonian can then be expressed as:

$$\widehat{H} = \widehat{H}_n + \widehat{H}_e = \widehat{H}_{nn} + \widehat{H}_{ne} + \widehat{H}_{ee}. \tag{2.5}$$

Here, the n-n interactions are treated classically. The electronic Schrödinger equation thus takes the form:

$$\widehat{H}_e\,|\psi_e\rangle = E_e\,|\psi_e\rangle\,. \tag{2.6}$$

As a result, the total energy can be written as:

$$E = E_e + E_{nn}, \tag{2.7}$$

where E_{nn} corresponds to the classical Coulomb repulsion between nuclei. At this point, we "*just*" need to solve the electronic Schrödinger equation. The electronic Hamiltonian includes the kinetic energy of the electrons, the electron-nuclear attractions, and the electron-electron repulsions:

$$\widehat{H}_e = -\sum_i {}^1\!/_2\nabla_i^2 - \sum_{i,\alpha}\frac{Z_\alpha}{|\mathbf{R}_\alpha - \mathbf{r}_i|} + \sum_{i>j}\frac{1}{|\mathbf{r}_i - \mathbf{r}_j|}, \tag{2.8}$$

where the α index runs over nuclei, and the i, j indices run over electrons and it is writen in atomic units ($\hbar = m_e = e = 1$).

This approximation is generally robust, as discussed by Čurík and Sutcliffe in [12] and [13]. It allows us to simplify the problem by treating nuclei as stationary, quasi-classical particles relative to the much lighter and faster-moving electrons, which are the primary particles of interest in our study.

2.2 Hartree-Fock Formalism

One of the earliest and most influential methods for addressing the quantum mechanical complexity of many-electron systems is the Hartree-Fock approach, developed independently by Douglas Hartree and Vladimir Fock [14, 15]. Its core idea is to approximate the effect of electron-electron repulsion by assuming that each electron moves independently in the average field generated by all others. This Mean-Field Approximation greatly reduces computational effort by replacing explicit inter-electronic interactions with an effective potential.

To construct an approximate many-electron wavefunction, we begin with the *orbital approximation*, where the total wavefunction is written as a product of single-electron functions (orbitals). This leads to the so-called Hartree product:

$$\Psi = \prod_{i=1}^{N} \chi_i(\mathbf{r}_i). \tag{2.9}$$

However, the Hartree product fails to satisfy the antisymmetry requirement imposed by the Pauli exclusion principle: the total wavefunction of a system of fermions must change sign upon exchange of any two fermions. This ensures that no two electrons occupy the same quantum state. To impose antisymmetry, a correct form of the wavefunction is given by a Slater determinant, Equation 2.10 (or 2.11 for compactness):

Claim 2.2.1 Slater Determinant

$$|\Psi\rangle = \frac{1}{\sqrt{N!}} \begin{vmatrix} \chi_1(\mathbf{x}_1) & \chi_2(\mathbf{x}_1) & \cdots & \chi_N(\mathbf{x}_1) \\ \chi_1(\mathbf{x}_2) & \chi_2(\mathbf{x}_2) & \cdots & \chi_N(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \chi_1(\mathbf{x}_N) & \chi_2(\mathbf{x}_N) & \cdots & \chi_N(\mathbf{x}_N) \end{vmatrix}, \tag{2.10}$$

$$|\Psi\rangle = |\chi_1 \chi_2 \cdots \chi_N\rangle, \tag{2.11}$$

where χ_n denotes a spin-orbital, and $\mathbf{x}_n$ represents the combined spatial and spin coordinates of the n-th electron. The prefactor $1/\sqrt{N!}$ ensures normalisation.

The HF method approximates the electronic ground state by a wavefunction in the form of a Slater determinant. Within this ansatz, the total energy is expressed as the expectation value of the Hamiltonian:

$$E = \langle \Psi_{\text{Slater}} | \widehat{H} | \Psi_{\text{Slater}} \rangle. \tag{2.12}$$

According to the variational principle, the determinant that minimises this functional —while its constituent spin-orbitals remain orthonormal— gives the best Slater determinant approximation to the true ground state. The resulting optimisation problem can be written compactly as:

$$\min_{\{\chi_i\}} \left\langle \Psi_{\text{Slater}} \middle| \widehat{H} \middle| \Psi_{\text{Slater}} \right\rangle. \tag{2.13}$$

Enforcing the constraint that the spin-orbitals remain orthonormal: $\langle \chi_i | \chi_j \rangle = \delta_{ij}$, we obtain a set of coupled integro-differential equations, known as the Hartree-Fock equations:

Key Equation 2: Hartree-Fock Equations

$$\widehat{F} \chi_i(\mathbf{x}_i) = \varepsilon_i \chi_i(\mathbf{x}_i), \tag{2.14}$$

where $\widehat{F}$ is the Fock operator and ε_i is the energy associated with the i^{th} spin-orbital χ_i.

For the Fock operator, the repulsion between electrons is treated in an average sense, resulting in the following expression:

$$\widehat{F}(i) = \widehat{h}(i) + \sum_{b=1}^{N}[\hat{J}_b(i) - \hat{K}_b(i)], \tag{2.15}$$

where $\widehat{h}(i)$ is the one-electron operator, accounting for the kinetic energy of electron i and its attraction to the nuclei. The terms $\hat{J}_b$ and $\hat{K}_b$ denote the Coulomb and Exchange operators, respectively. The Coulomb operator describes the classical electrostatic repulsion between electron densities, while the Exchange operator arises purely from the antisymmetry of the wavefunction and has no classical analogue.

$$\begin{gathered}\widehat{h}(i) = -1/2\nabla_i^2 - \sum_{A=1}^{N}\frac{Z_A}{r_{iA}}, \\ \widehat{J}_b(i) = \langle\chi_b(j)|r_{ij}^{-1}|\chi_b(j)\rangle, \quad \hat{K}_b(i) = \langle\chi_b(j)|r_{ij}^{-1}|\chi_a(j)\rangle.\end{gathered} \tag{2.16}$$

The operators $\widehat{J}$ and $\hat{K}$ constitute the bi-electronic part of the Hamiltonian, and they are usually refered as the Coulomb and Exchange integrals, as follows:

$$J_{ij} \equiv \int \frac{\chi_i^\star(i)\chi_j^\star(j)\chi_i(i)\chi_j(j)}{r_{ij}}\mathrm{d}\tau_i\mathrm{d}\tau_j = \int \frac{\rho_i(i)\rho_j(j)}{r_{ij}}\mathrm{d}\tau_i\mathrm{d}\tau_j, \tag{2.17}$$

$$K_{ij} \equiv \int \frac{\chi_i^\star(i)\chi_j^\star(j)\chi_i(j)\chi_j(i)}{r_{ij}}\mathrm{d}\tau_i\mathrm{d}\tau_j. \tag{2.18}$$

Note:

The Coulomb and Exchange integrals can also be expressed in a more compact notation using Dirac bra-ket or Mulliken notation for two-electron integrals.

$$J_{ij} = \langle ij|ij\rangle = (ii|jj) \tag{2.19}$$

$$K_{ij} = \langle ij|ji\rangle = (ij|ji) \tag{2.20}$$

With the formalism developed above, we can now express the HF equations as:

$$\left(\hat{h}(i) + \sum_{b}^{N}[\hat{J}_b(i) - \hat{K}_b(i)]\right)\chi_a(i) = \varepsilon_a\chi_a(i) \qquad \forall a \in (1, N), \tag{2.21}$$

where N refers to the number of occupied spin-orbitals, and N represents the total number of spin-orbitals in the system.

The resulting HF equations are non-linear and integro-differential which defies analytic solution. To address this, their solution is approached through an iterative method known as the the self-consistent-field (SCF) procedure. Starting from an initial guess for the spin-orbitals, the Fock operator is constructed and its eigenfunctions are computed. These eigenfunctions (updated orbitals) are then used to reconstruct the Fock operator, the procedure continues until the orbitals from one cycle to the next become effectively indistinguishable.

To render this iterative loop tractable for real molecules, Roothaan introduced a matrix formulation in 1951 [16]. Each spin-orbital is expanded as a linear combination of predefined basis functions, transforming the equations into a matrix eigenvalue problem, significantly reducing the computational complexity of the SCF procedure.

The basis set is typically normalised but not necessarily orthogonal, allowing flexibility to describe molecular systems. Within this framework, each spin-orbitals is constructed through a Linear Combination of Atomic Orbitals (LCAO) using k predefined basis functions, as shown below:

Key Equation 3

$$\chi_i = \sum_{\nu}^{k} C_{\nu i}\phi_\nu, \tag{2.22}$$

where ϕ_ν represents the basis set functions, and $C_{\nu i}$ are the coefficients associated with each spin orbital.

In theory, if the basis set were complete, this approach would yield the exact solution. The choice of basis functions —whether Gaussian-, or Slater-type orbitals, or even plane waves— depends on the system under study. These options will be discussed in more detail in Section 2.3.

This new formulation is also called Roothaan-Hall, and can be written in a generalised eigenvalue problem,

Key Equation 4: Hartree-Fock-Roothaan-Hall Equation

$$\mathbf{FC} = \mathbf{SC}\varepsilon. \tag{2.23}$$

where $\mathbf{F}$ is the Fock matrix, $\mathbf{C}$ is the coefficient matrix, $\mathbf{S}$ is the overlap matrix, and ε is the orbital energy eigenvalues.

We still need to solve the eigenvalue problem, but now it is a linear algebra problem without integro-differential equations. We can also use the Löwdin Orthogonalisation to make the basis set orthogonal, which simplifies the problem even further, as the next following equation shows [17] :

Insight 2.2.1 (Löwdin Orthogonalisation)

Transforming the Fock matrix to an orthogonal basis set:

$$\begin{aligned} \mathbf{F}' &= \mathbf{X}^T\mathbf{F}\mathbf{X} \\ \mathbf{F}' &= {\mathbf{S}^{-\frac{1}{2}}}^T\mathbf{F}\mathbf{S}^{-\frac{1}{2}} \end{aligned} \tag{2.24}$$

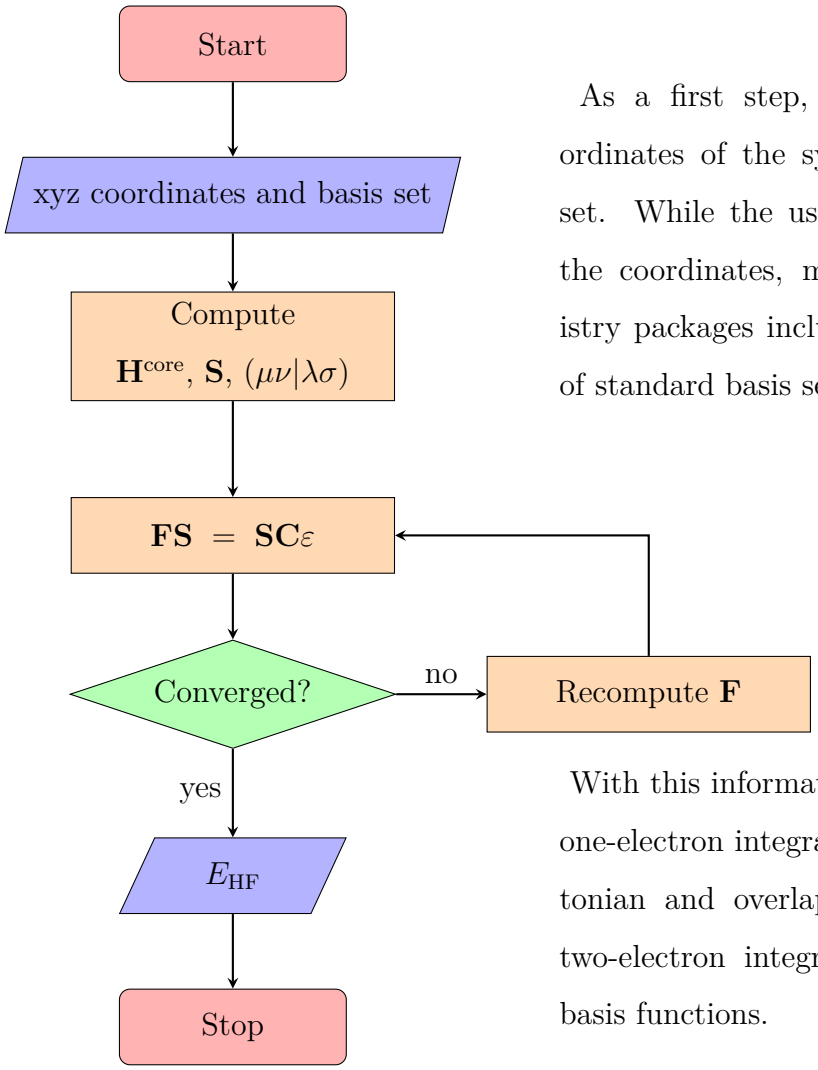
Then, the Roothaan-Hall equations become:

$$\mathbf{F}'\mathbf{C}' = \mathbf{C}'\varepsilon \tag{2.25}$$

The computational cost of the HFRH method is $\mathcal{O}(N^3)$ with or without the Löwdin orthogonalisation. However, solving the generalised eigenvalue problem $\mathbf{FC} = \mathbf{SC}\varepsilon$ involves more internal operations (*e. g.*, Cholesky decomposition of $\mathbf{S}$), while solving the transformed problem $\mathbf{F}'\mathbf{C}' = \mathbf{C}'\varepsilon$ is numerically more stable and faster, since $\mathbf{F}'$ is a symmetric matrix in an orthonormal basis.

In computational implementations, the HF problem can be solved without prior orthogonalisation, for example, using the `DSYGV` SUBROUTINE from LAPACK [18], which handles generalised symmetric eigenvalue problems. However, this routine internally reduces the problem to standard form and ultimately calls `DSYEV` —the same SUBROUTINE we would use after Löwdin orthogonalisation—. Performing the orthogonalisation explicitly avoids redundant internal steps, offers greater control, and can improve efficiency and numerical stability.

With all before mentioned, we can structure an algorithm to solve the HF problem for any given system, as we show in the next flowchart.



As a first step, we require the coordinates of the system and the basis set. While the user typically provides the coordinates, most quantum chemistry packages include internal libraries of standard basis sets.

With this information, we compute the one-electron integrals —the core Hamiltonian and overlap matrix— and the two-electron integrals over the chosen basis functions.

Once the integrals are computed, the iterative procedure is initiated. This continues until the convergence criterion is satisfied —usually defined in terms of energy difference and density matrix changes between iterations—.

However, it is important to emphasise that reaching convergence does not imply the absolute correctness of the solution. As shown by Cancès and Le Bris [19], the HF model possesses intrinsic mathematical limitations, and the SCF method may converge to local rather than global minima depending on the initial guess and system complexity. Moreover, Yang et al. [20] demonstrated that additional iterations beyond convergence do not yield improved accuracy, since the SCF solution is inherently limited by the chosen functional space and nonlinear nature of the eigenvalue problem.

Therefore, it is crucial to bear in mind that achieving full SCF convergence does not necessarily guarantee the correctness or accuracy of the solution. Instead, a converged SCF calculation should be regarded as a computational approximation to the true solution, rather than a systematically improvable and definitive result.

2.3 Basis Sets in Quantum Chemistry

"*There are probably as many basis set defined for polyatomic calculations as quantum chemists.*"

-Attila Szabo and Neil S. Ostlund

The choice of a basis set has a fundamental importance in quantum chemical calculations, as it directly influences both the accuracy of the results and the computational cost. Basis sets provide a finite, predefined collection of functions in which the molecular orbitals are expanded, and the mathematical properties of these functions determine how well they can describe features such as electron density, polarisation and correlation effects [21, 22]. A well-chosen basis set ensures that the quantum mechanical approximation has an accurate outcome while maintaining computational efficiency.

A key concept in basis set construction is the so-called "zeta" quality, which refers to the number of basis functions used per atomic orbital. A single-ζ (SZ) basis set uses one function per orbital and offers only a minimal description of the electronic structure. A double-ζ (DZ) basis set adds a second, independently optimised function per orbital, thereby increasing variational flexibility and allowing the orbitals to contract or expand depending on the chemical environment. Higher levels, such as triple-ζ (TZ) and quadruple-ζ (QZ) sets, include three and four functions per orbital, respectively, and

systematically improve the representation of the molecular wavefunction [23, 24]. These higher-ζ basis sets enable a better treatment of electron correlation by providing more accurate orbital shapes and greater angular momentum mixing. However, each increase in ζ quality incurs a substantial computational overhead, and thus the trade-off between accuracy and cost must be carefully considered.

It is also crucial to emphasise that basis set quality is not solely determined by the number of radial functions (ζ level), but also by the angular flexibility of the set. This means that functions with higher angular momentum (p, d, f, etc.) must be included to accurately describe anisotropic electron distributions and polarisation effects. For example, the unperturbed hydrogen atom requires only a single $1s$ orbital for its exact wavefunction. However, when exposed to an external electric field, the spherical symmetry is broken, and the electron density becomes displaced. To describe this polarised density, p-type functions must be incorporated. Similarly, for second-row elements like carbon, nitrogen, and oxygen, the inclusion of d-type polarisation functions significantly improves accuracy [25], particularly in chemical environments where the electron cloud is perturbed by the natural non-uniform distribution of electron density. These polarisation functions are essential for reproducing subtle features of the electron density in molecular systems.

Slater-Type Basis Sets

When solving the Schrödinger Equation for the hydrogen atom, the analytical solutions yield orbitals that exhibit the correct cusp behavior at the nucleus and an exponential decay at large distances. These features are naturally captured by Slater-Type Orbitals (STOs), which are typically expressed as:

$$\chi_{\mathrm{STO}}(\mathbf{r}) = N r^{n-1} e^{-\zeta r}, \tag{2.26}$$

where N is a normalisation constant, n is the principal quantum number, and ζ is an orbital exponent controlling the radial decay. The STOs provide a physical realism, especially an accurate representation of electron density near and far away the nucleus.

However, the evaluation of multicenter integrals with STOs is mathematically and computataionally cumbersome because the exponential form complicates the solution of two-electron integrals, that problem is the motivation for the development of different approximations, such as the Gaussian-Type Orbitals (GTO) and plane-wave basis sets.

Gaussian-type Basis Sets

To overcome the challenges associated with STOs, Gaussian-Type Orbitals (GTOs), first introduced by Boys [26], quickly gained popularity and have become the standard in many quantum chemistry applications. A typical Gaussian function is written as:

$$\chi_{\mathrm{GTO}}(\mathbf{r}) = N x^l y^m z^n e^{-\alpha r^2}, \tag{2.27}$$

with N as the normalisation constant, l, m, and n as the angular momentum components, and α setting the width of the Gaussian.

Thanks to the Gaussian Product Theorem [27], the product of two Gaussians yields another Gaussian centered along the interatomic axis, which allows four-center integrals to reduce systematically into a hierarchy of simpler integrals. This property permits an efficient evaluation of many of the integrals encountered in HF and post-HF methods.

Because a single Gaussian function does not reproduce the sharp nuclear cusp or the correct asymptotic decay observed with STOs, a linear combination of Gaussian functions is employed to approximate a single STO. For example, the STO-3G basis set uses three primitive Gaussian functions to emulate the radial behavior of a Slater orbital. Although this contracted Gaussian approach sacrifices some accuracy in reproducing fine details of the electron density (particularly near and far from the nucleus), it represents a well-balanced compromise between computational efficiency and accuracy.

Plane-Wave Basis Sets

In a different approach, plane wave basis sets are widely used in the study of periodic and extended systems, such as those found in solid-state physics. A plane wave basis function is given by:

$$\chi_{\mathrm{PW}}(\mathbf{r}) = \frac{1}{\sqrt{V}} e^{i\mathbf{k}\cdot\mathbf{r}}, \tag{2.28}$$

where V is the normalised volume of the unit cell and $\mathbf{k}$ is the reciprocal lattice vector.

Plane waves are especially advantageous to describe delocalized electronic states under periodic boundary conditions. Their systematic convergence is controlled by the kinetic energy cutoff, however, a large number of basis functions are required to accurately describe localized features like core electrons. Therefore, although plane-wave basis sets are well suited for periodic systems, they are less efficient when applied to isolated molecules.

2.3.1 Selecting a Basis Set

In summary, the choice between STOs, GTO, or plane waves involves a trade-off between physical accuracy and computational efficiency:

- **STOs**, high physical accuracy but are computationally demanding, particularly for multicenter integrals.
- **GTOs**, provide greater computational efficiency, though at the cost of reduced accuracy in representing localised features.
- **Plane waves**, well suited for periodic systems but perform poorly in describing localised phenomena.

Many quantum chemistry packages provide a wide range of basis sets. For example, AMS [28] offers several STO basis sets, while programs such as Gaussian16 [29] and Orca [30] include an extensive selection of GTO basis sets. In contrast, plane-wave basis sets are commonly used in codes like Quantum Espresso [31] and Vasp [32], which are specifically designed for periodic systems.

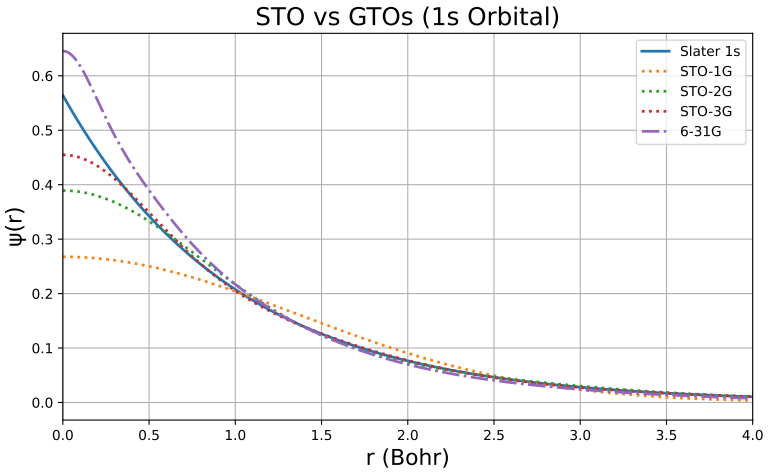
Majority of the quantum chemistry codes can also read basis sets provided by the user, the BasisSetExchange database [33] lists a wide range of basis sets.

However, it is important to note that since the mathematical properties of the STOs, GTOs and plane waves are different, the software designed to work with one type of basis set may not be able to handle the other types.

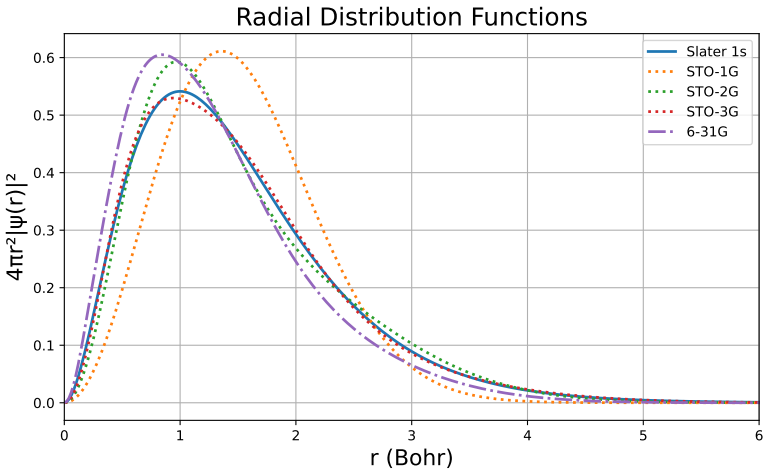
> *... the choice of a basis set is not nearly the black art,*
>
> *however,*
>
> *it may first appear.*
>
> Szabo and Ostlund [21]

The choice of basis set depends not only on its functional type —whether STOs, GTOs, or plane waves— but also on the number and nature of functions included. Higher ζ levels offer greater variational flexibility, while polarisation functions are essential for describing anisotropic effects. Ultimately, basis set selection reflects a trade-off between accuracy and computational cost.

For visual comparisons, Figure 2.1 illustrates the wavefunction and the radial distribution for the hydrogen 1s orbital using different basis sets.



(a) Comparison of a Slater function with Gaussian functions. Least squares fit of a 1s Slater function $\zeta = 1.0$.



(b) Comprasion of the corresponding radial distribution functions.

Figure 2.1. Comparison of the hydrogen 1s orbital using different basis sets: Slater-type orbitals (STO) and Gaussian-type orbitals (GTO)

2.4 Electronic Density

The electronic density, $\rho(\mathbf{r})$, is a fundamental concept in modern theoretical chemistry. It offers a compact yet informative description of a system's electronic structure and plays a central role in density-based methods such as Density Functional Theory (DFT), as well as in interpretative frameworks like QTAIM.

For a system comprising two electrons with spin-spatial coordinates $\mathbf{x}_1$ and $\mathbf{x}_2$, the electronic density is given by $|\Psi(\mathbf{x}_1, \mathbf{x}_2)|^2$ [34]. To calculate the probability of finding electron 1 in the volume element $\mathrm{d}\mathbf{x}_1$ and electron 2 in $\mathrm{d}\mathbf{x}_2$, the full probability density must be integrated over $\mathrm{d}\omega_1\mathrm{d}\mathbf{x}_2$, where $\mathrm{d}\mathbf{x}_n = \mathrm{d}\tau_n\mathrm{d}\omega_n$.

Note:

Since the electrons are indistinguishable, for a system with N electrons, the electronic density can be generalised as:

$$\rho(\mathbf{r}) = N \int |\Psi(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N)|^2 \mathrm{d}\omega_1 \mathrm{d}\mathbf{x}_2 \ldots \mathrm{d}\mathbf{x}_N, \tag{2.29}$$

such that:

$$\int \rho(\mathbf{r}) \mathrm{d}\tau = N. \tag{2.30}$$

In the case of a Slater Determinant (Hartree-Fock), the electronic density for a closed shell with spatial orbitals ψ_a can be written as:

$$\rho(\mathbf{r}) = 2 \sum_{a=1}^{N/2} |\psi_a(\mathbf{r})|^2. \tag{2.31}$$

For a system with N particles, it is possible to write an operator for the density $\hat{\rho}$ and then, get the expected value for the system wavefunction,

$$\hat{\rho} = \sum_{i}^{N} \hat{\delta}(\mathbf{r}_i - \mathbf{r}_0), \tag{2.32}$$

then $\rho(\mathbf{r})$ is a expected value of a quantum mechanics operator, which can be written as:

$$\begin{aligned} \rho(\mathbf{r}_0) &= \int \Psi^{\star}(\mathbf{x}_1, \ldots, \mathbf{x}_N) \sum_{i}^{N} \hat{\delta}(\mathbf{r}_i - \mathbf{r}_0) \Psi(\mathbf{x}_1, \ldots, \mathbf{x}_N) \mathrm{d}\mathbf{x}_1 \ldots \mathrm{d}\mathbf{x}_N \\ &= \langle \Psi | \, \hat{\rho} \, | \Psi \rangle \end{aligned} \tag{2.33}$$

The electronic density is not only central to theory but also accessible experimentally. Unlike many aspects of the electronic structure, $\rho(\mathbf{r})$ can be directly measured using techniques such as X-ray and neutron diffraction [35, 36]. This makes it a valuable bridge between theory and experiment, allowing direct comparison between computed and observed densities, a stringent test of the accuracy and reliability of computational methods.

Going forward, it is also possible to establish a density function for an electron couple, also called pair density, such that:

$$\rho_2(\mathbf{r}_1, \mathbf{r}_2) = N(N-1) \int |\Psi(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N)|^2 \mathrm{d}\omega_1 \mathrm{d}\omega_2 \mathrm{d}\mathbf{x}_3 \ldots \mathrm{d}\mathbf{x}_N, \tag{2.34}$$

where $\rho_2(\mathbf{r}_1, \mathbf{r}_2) N^{-1}(N-1)^{-1}$ determines the normalised probability of simultaneously finding two electrons per volume centered at positions $\mathbf{r}_1$ and $\mathbf{r}_2$. Two normalisation conventions are found in the literature: Löwdin [37] introduced the factor $N(N-1)$, corresponding to ordered electron pairs, while McWeeny [38] proposed an additional division by two to account for unordered pairs. This pair density is commonly expressed as a sum of an uncorrelated term and a correlation contribution, $\rho_2(\mathbf{r}_1, \mathbf{r}_2) = \rho(\mathbf{r}_1)\rho(\mathbf{r}_2) + \rho_2^{xc}(\mathbf{r}_1, \mathbf{r}_2)$.

The importance of the one- and two-electro densities stems from the fact that, within the BOA and in the absence of external fields, the total electronic energy of the system can be expressed entirely in terms of these densities.

Key Equation 5

$$
\begin{aligned}
E = &-\frac{1}{2}\int \nabla^2 \rho_1(\mathbf{r}_1, \mathbf{r}_1')\Big|_{\mathbf{r}_1' \to \mathbf{r}_1} \mathrm{d}\tau_1 - \sum_A \int \frac{Z_A \rho_1(\mathbf{r}_1)}{r_{1A}} \mathrm{d}\tau_1 + \sum_{A \neq B} \frac{Z_A Z_B}{r_{AB}} \\
&+ \frac{1}{2}\int\int \frac{\rho(\mathbf{r}_1)\rho(\mathbf{r}_2)}{r_{12}} \mathrm{d}\tau_1 \mathrm{d}\tau_2 + \frac{1}{2}\int\int \frac{\rho_2^{xc}(\mathbf{r}_1, \mathbf{r}_2)}{r_{12}} \mathrm{d}\tau_1 \mathrm{d}\tau_2 \\
&= T + V_{ne} + V_{nn} + V_{ee} + V_{xc}, \qquad (2.35)
\end{aligned}
$$

the first term represents the kinetic energy of the electrons, while the second and third correspond to the electron-nucleus and nucleus-nucleus interactions, respectively. The fourth term describes the classical electron-electron repulsion, treated as the Coulomb interaction between one-electron densities. The final term introduces exchange-correlation effects through the pair density.

The exchange-correlation term, V_{xc}, is introduced by subtracting the uncorrelated product $\rho(\mathbf{r}_1)\rho(\mathbf{r}_2)$ from the full pair density $\rho_2(\mathbf{r}_1, \mathbf{r}_2)$. This isolates the exchange-correlation contribution, capturing both the effects of electron indistinguishability and the correlated response of the electron distribution to the presence of another electron.

In practice, evaluating Equation 2.35 within an electronic structure method requires expressing the electron densities in terms of the molecular orbital basis functions ϕ_i,

$$\rho_1(\mathbf{r}) = \sum_{ij} D_{ij} \phi_i(\mathbf{r}) \phi_j(\mathbf{r}), \qquad (2.36)$$

$$\rho_2(\mathbf{r}_1, \mathbf{r}_2) = \sum_{ijkl} d_{ijkl} \phi_i(\mathbf{r}_1) \phi_j(\mathbf{r}_1) \phi_k(\mathbf{r}_2) \phi_l(\mathbf{r}_2), \qquad (2.37)$$

where D_{ij} and d_{ijkl} are the elements of the first- and second-order density matrices, respectively.

2.4.1 Density matrix

The density matrix offers a compact and versatile way to encode information about a quantum system, particularly when working with reduced representations of many-electron wavefunctions [37, 39]. Rather than relying on the full N-electron wavefunction Ψ, many physical properties can be derived from the reduced density matrices, which are central to modern electronic structure theory.

In quantum mechanics, the expectation value of an observable associated with an operator $\widehat{Q}$ for a system described by Ψ is given by:

$$\langle \widehat{Q} \rangle = \int \Psi^\star(\mathbf{x}_1, \ldots, \mathbf{x}_n) \widehat{Q} \Psi(\mathbf{x}_1, \ldots, \mathbf{x}_n) \mathrm{d}\mathbf{x}_1 \ldots \mathrm{d}\mathbf{x}_n. \tag{2.38}$$

When the operator explicitly depends on only m variables ($m \leq n$), the expectation value can be simplified to:

$$\begin{aligned}
\langle \widehat{Q} \rangle &= \int \widehat{Q} \Psi(\mathbf{x}_1, \ldots, \mathbf{x}_m, \mathbf{x}_{m+1}, \ldots, \mathbf{x}_n) \Psi^\star(\mathbf{x}_1, \ldots, \mathbf{x}_m, \mathbf{x}_{m+1}, \ldots, \mathbf{x}_n) \mathrm{d}\mathbf{x}_1 \ldots \mathrm{d}\mathbf{x}_n \\
&= \int \left(\int \Psi^\star(\mathbf{x}_1, \ldots, \mathbf{x}_m, \mathbf{x}_{m+1}, \ldots, \mathbf{x}_n) \mathrm{d}\mathbf{x}_{m+1} \ldots \mathrm{d}\mathbf{x}_n \right) \widehat{Q} \Psi(\mathbf{x}_1, \ldots, \mathbf{x}_m, \mathbf{x}_{m+1}, \ldots, \mathbf{x}_n) \mathrm{d}\mathbf{x}_1 \ldots \mathrm{d}\mathbf{x}_m \\
&= \int \widehat{Q}(\mathbf{x}_1, \ldots, \mathbf{x}_m) F_m(\mathbf{x}_1, \ldots, \mathbf{x}_m) \mathrm{d}\mathbf{x}_1 \ldots \mathrm{d}\mathbf{x}_m,
\end{aligned} \tag{2.39}$$

where the function F_m is defined as the integral over the remaining variables:

$$F_m(\mathbf{x}_1, \ldots, \mathbf{x}_m) = \int \Psi(\mathbf{x}_1, \ldots, \mathbf{x}_m, \mathbf{x}_{m+1}, \ldots, \mathbf{x}_n) \Psi^\star(\mathbf{x}_1, \ldots, \mathbf{x}_m, \mathbf{x}_{m+1}, \ldots, \mathbf{x}_n) \mathrm{d}\mathbf{x}_{m+1} \ldots \mathrm{d}\mathbf{x}_n. \tag{2.40}$$

This object F_m is identified with the m-th order reduced density matrix Γ_m:

$$\Gamma_m(\mathbf{x}_1, \ldots, \mathbf{x}_m; \mathbf{x}'_1, \ldots, \mathbf{x}'_m) = \binom{n}{m} \int \Psi^\star(\mathbf{x}'_1, \ldots, \mathbf{x}'_n) \Psi(\mathbf{x}_1, \ldots, \mathbf{x}_n) \mathrm{d}\mathbf{x}_{m+1} \ldots \mathrm{d}\mathbf{x}_n. \tag{2.41}$$

Reduced density matrices form a hierarchy, as higher-order matrices can be contracted into lower-order ones:

$$\Gamma_m(\mathbf{x}_1,\ldots,\mathbf{x}_m;\mathbf{x}'_1,\ldots,\mathbf{x}'_m) = \frac{m+1}{n-m}\int \Gamma_{m+1}(\mathbf{1},\ldots,\mathbf{m}+\mathbf{1};\mathbf{1}',\ldots,\mathbf{m}+\mathbf{1}')\mathrm{d}\mathbf{x}_{m+1}. \tag{2.42}$$

In quantum chemistry, the operators of primary interest are typically mono- or bi-electronic, making the first- and second-order density matrices particularly relevant:

$$\Gamma_1(\mathbf{x}_1;\mathbf{x}'_1) = n\int \Psi^\star(\mathbf{1}',\mathbf{2}',\ldots,\mathbf{n}')\Psi(\mathbf{1},\mathbf{2},\ldots,\mathbf{n})\mathrm{d}\mathbf{x}_2\ldots\mathrm{d}\mathbf{x}_n, \tag{2.43}$$

$$\Gamma_2(\mathbf{x}_1,\mathbf{x}_2;\mathbf{x}'_1,\mathbf{x}'_2) = n(n-1)\int \Psi^\star(\mathbf{1}',\mathbf{2}',\ldots,\mathbf{n}')\Psi(\mathbf{1},\mathbf{2},\ldots,\mathbf{n})\mathrm{d}\mathbf{x}_3\ldots\mathrm{d}\mathbf{x}_n. \tag{2.44}$$

The first-order density matrix (Equation 2.43), often called the Fock-Dirac density matrix, becomes especially useful upon integration over spin coordinates, yielding the first-order reduced density matrix:

$$\rho_1(\mathbf{r}_1,\mathbf{r}'_1) = \int \Gamma_1(\mathbf{x}_1;\mathbf{x}'_1)\mathrm{d}\omega_1. \tag{2.45}$$

Unlike density functions, density matrix elements generally lack direct physical interpretation, with the notable exception of the diagonal elements of the first-order reduced density matrix, which represent the electronic density itself. Integrating these diagonal elements over all space yields the total number of electrons in the system, highlighting their importance. Consequently, many crucial properties —particularly energies— are conveniently expressed using the first-order reduced density matrix and the electron pair density [40].

2.5 Density Functional Theory

Having established the central importance of the electron density and its relationship to physical observables, we now turn to a theoretical framework where the electron density itself serves as the fundamental variable: Density Functional Theory (DFT).

The idea of using the electronic density to extract information about a system dates back to 1927, with the pioneering work of Llewellyn Hilleth Thomas and Enrico Fermi [41, 42]. Their approach involved approximating the kinetic energy, along with the nucleus-electron and electron-electron interactions, by modeling the system as a uniform electron gas.

Insight 2.5.1 (Thomas-Fermi Model)
Consider a hypothetical system with a constant electronic density:

$$T_{\mathrm{TF}}[\rho(\mathbf{r})] = \frac{3}{10}(3\pi^2)^{2/3}\int \rho(\mathbf{r})^{5/3}\mathrm{d}\tau \tag{2.46}$$

Combining this expression for the kinetic energy with classical electrostatic contributions leads to the total energy functional of the Thomas-Fermi model:

$$E_{\mathrm{TF}}[\rho(\mathbf{r})] = \frac{3}{10}(3\pi^2)^{2/3}\int \rho(\mathbf{r})^{5/3}\mathrm{d}\tau - Z\int \frac{\rho(\mathbf{r})}{r}\mathrm{d}\tau + \frac{1}{2}\int\int \frac{\rho(\mathbf{r}_1)\rho(\mathbf{r}_2)}{r_{12}}\mathrm{d}\tau_1\mathrm{d}\tau_2 \tag{2.47}$$

This model describes the electronic structure using a density only weakly confined by the nuclear potential. Although it neglects shell structure and quantum interference effects, it performs reasonably well for alkali and alkaline-earth metals, where valence electrons are loosely bound to the nuclei.

The Thomas-Fermi model is also noteworthy as a expression of the Copenhagen interpretation of Quantum Mechanics. This perspective laid the groundwork for the formal development of DFT, which advanced significantly with the foundational theorems of Hohenberg and Kohn [6], and the practical Kohn-Sham formalism introduced the following year [43].

2.5.1 Hohenberg-Kohn Theorems [6]

Theorem 1

The first Hohenberg-Kohn theorem asserts that the stationary ground-state electronic density, $\rho(\mathbf{r})$, uniquely determines the external potential $v_{\text{ext}}(\mathbf{r})$. Consequently, all properties of the system —including the Hamiltonian, the wavefunction, and the total energy— are uniquely determined by $\rho(\mathbf{r})$. Hence, two systems of N particles subject to distinct external potentials cannot exhibit the same ground-state density, unless the potentials differ only by a constant.

Theorem 2.5.1 HK's Theorem 1

The ground-state expectation value of any observable $\widehat{A}$ is a unique functional of the exact ground-state density:

$$\langle\psi|\widehat{A}|\psi\rangle = A[\rho_0(\mathbf{r})] \tag{2.48}$$

Corollary 2.5.1

The external potential, and thus all ground-state properties of the system, are uniquely determined by the ground-state electronic density. This includes the many-body wavefunction. In particular, the HK functional is defined as $F[\rho] = T[\rho] + U[\rho]$, a universal functional that is independent of the external potential.

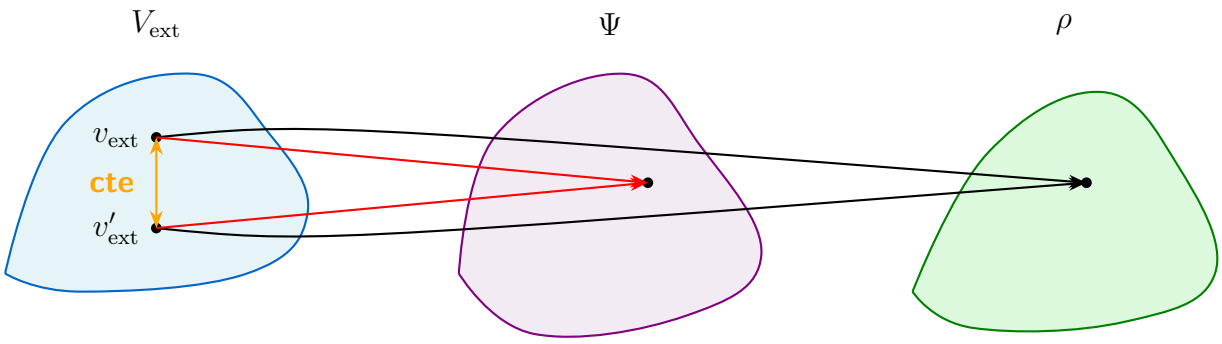


Figure 2.2. Hohenberg-Kohn Theorem 1. The red arrows point to the corresponding wavefunction, while the black arrows point directly to the density ρ.

Theorem 2

The second Hohenberg-Kohn theorem establishes a variational principle for the electron density. It states, for a given external potential $v_{ext}(\mathbf{r})$ and electron number N, the ground-state energy functional,

$$E_{v,N}[\rho] = F[\rho] + \int v_{ext}(\mathbf{r})\rho(\mathbf{r})\mathrm{d}\tau, \tag{2.49}$$

attains its minimum value when $\rho(\mathbf{r})$ is the exact ground-state density. That is, among all physically admissible densities, the true ground-state density yields the lowest energy.

Theorem 2.5.2 HK's Theorem 2

The ground-state energy is the minimum of the energy functional, providing a variational principle:

$$E_0 \leq E[\rho] = T[\rho] + V_{Ne}[\rho] + V_{ee}[\rho] \tag{2.50}$$

with equality only when ρ is the exact ground-state density.

2.5.2 Kohn-Sham Approximation [43]

Building on the Hohenberg-Kohn theorems, Kohn-Sham proposed a practical scheme to address the many-electron problem by introducing an auxiliary system of non-interacting particles that reproduces the exact ground-state electron density of the fully interacting system. Within this framework, the total energy of a system subject to an external potential $v_{\text{ext}}(\mathbf{r})$ is expressed as:

$$E[\rho(\mathbf{r})] = T_s[\rho(\mathbf{r})] + \int v_{ext}(\mathbf{r})\rho(\mathbf{r})\mathrm{d}\tau + \frac{1}{2}\int\int \frac{\rho(\mathbf{r})\rho(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|}\mathrm{d}\tau + E_{xc}[\rho(\mathbf{r})]. \tag{2.51}$$

In Equation 2.51, the first term $T_s[\rho(\mathbf{r})]$ represents the kinetic energy of the non-interacting reference electrons, the second term corresponds to the interaction of the electrons with the external potential (typically from the nuclei), the third is the classical Coulomb repulsion (Hartree energy), and the last term captures all remaining quantum many-body effects, which are encapsulated in the exchange-correlation functional $E_{xc}[\rho(\mathbf{r})]$.

To render the original interacting problem tractable, Kohn and Sham replaced it with an equivalent, auxiliary non-interacting system moving in an effective potential $v_{\text{eff}}(\mathbf{r})$, chosen such that it reproduces exactly the ground-state density of the interacting electrons. The resulting single-particle Kohn-Sham equation is:

$$\left(-1/2\nabla^2 + v_{eff}(\mathbf{r})\right)\varphi_i = \epsilon_i\varphi_i, \tag{2.52}$$

where the effective potential $v_{\text{eff}}(\mathbf{r})$ acting on the Kohn-Sham electrons is defined as: $v_{\text{eff}}(\mathbf{r}) = v_{\text{ext}}(\mathbf{r}) + v_{\text{H}}(\mathbf{r}) + v_{xc}(\mathbf{r})$; $v_{\text{H}}(\mathbf{r})$ is the classical Hartree potential arising from the electron density:

$$v_{\text{H}}(\mathbf{r}) = \int \frac{\rho(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|}\mathrm{d}\tau',$$

and $v_{xc}(\mathbf{r})$ is the exchange-correlation potential, defined as the functional derivative of $E_{xc}[\rho(\mathbf{r})]$ with respect to the density:

$$v_{xc}(\mathbf{r}) = \frac{\delta E_{xc}[\rho(\mathbf{r})]}{\delta\rho(\mathbf{r})}.$$

The exchange-correlation functional, $E_{xc}[\rho(\mathbf{r})]$, thus incorporates all many-body effects beyond the classical electrostatic interaction, including exchange effects due to the antisymmetry of the wavefunction and dynamic electron correlation. Formaly, it can be expressed as the difference between the true interacting energy and the components already accounted for in the Kohn-Sham formalism:

$$E_{xc}[\rho(\mathbf{r})] = T[\rho(\mathbf{r})] - T_s[\rho(\mathbf{r})] + V_{ee}[\rho(\mathbf{r})] - J[\rho(\mathbf{r})], \tag{2.53}$$

where $T[\rho(\mathbf{r})]$ is the kinetic energy of the interacting system, $T_s[\rho(\mathbf{r})]$ is the kinetic energy of the non-interacting system, $V_{ee}[\rho(\mathbf{r})]$ is the true electron-electron interaction energy, and $J[\rho(\mathbf{r})]$ is the classical Coulomb repulsion energy.

2.5.3 Exchange-Correlation Functionals

As the exact analytical form of the universal exchange-correlation functional remains unknown, DFT relies on approximate models to capture many-body exchange and correlation effects. Over the years, numerous functionals have been developed, varying in both complexity and accuracy.

To organise this diversity, functionals are often placed within the conceptual framework of Jacob's ladder (Figure 2.3), which symbolically depicts the climb toward chemical accuracy through successive levels of approximation. Each rung introduces additional physical ingredients —such as density gradients or explicit orbital dependence— with the goal of systematically improving the treatment of electron interactions [44]. In what follows, we briefly review the main levels of this hierarchy.

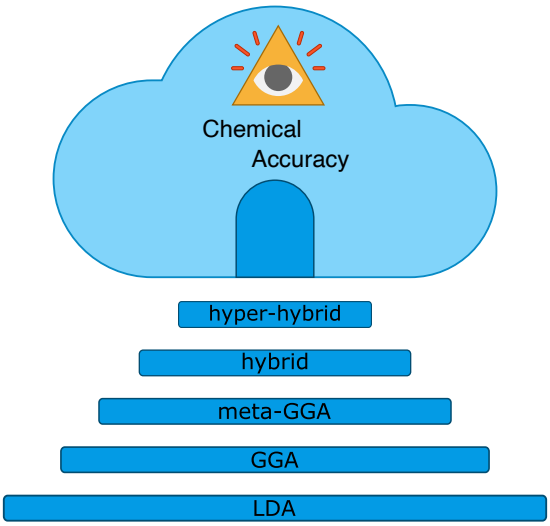


Figure 2.3. The rungs of the ladder represent different levels of approximation, with each level incorporating additional physical effects to improve accuracy.

Note:

Double-Hybrid It is important to note that, although double-hybrids are placed on the fifth rung of Jacob's ladder, they are not obtained by merely adding an extra density-dependent term to a functional. In practice, a double-hybrid combines a fraction of HF-like exchange with a second-order perturbative correlation contribution evaluated with KS orbitals (often "MP2-like"). The latter requires access to virtual orbitals and orbital energy differences and is typically evaluated in a post-SCF step [45].

- **Local Density Approximation (LDA).** In the LDA, the electron density is locally approximated as a homogeneous electron gas with density ρ. In this model, the electrons are immersed in a uniform background of positive charge, forming an electrically neutral system with infinite volume and an infinite number of non-interacting electrons [46].

  The corresponding exchange-correlation energy functional is written as:

  $E_{xc}^{LDA}[\rho] = \int \rho(\mathbf{r})\varepsilon_{xc}(\rho(\mathbf{r}))\mathrm{d}\tau$

  Taking the functional derivative with respect to the density yields the exchange-correlation potential:

  $v_{xc}^{LDA} = \frac{\delta E_{xc}^{LDA}}{\delta\rho} = \varepsilon_{xc}(\rho(\mathbf{r})) + \rho\frac{\mathrm{d}\varepsilon_{xc}(\rho(\mathbf{r}))}{\mathrm{d}\rho}$

  The exchange-correlation energy per particle, ε_{xc}, can be split into exchange and correlation contributions. The exchange part is given by:

  $\varepsilon_x = -\frac{3}{4}\left(\frac{3}{\pi}\right)^{1/3}\rho^{1/3}$

  Several variants of the LDA have been developed to improve accuracy. The X_α method [47] introduces an empirical parameter α to scale the exchange term. The Local Spin Density Approximation (LSDA) [47] extends the LDA to spin-polarised systems by treating the spin-up and spin-down densities, ρ_α and ρ_β, separately.

  While an analytical form for the correlation energy ε_{cor} is not available, several parameterisations based on Quantum Monte Carlo data have been proposed. The VWN functional [48] was derived from simulations by Vosko, Wilk, and Nusair. Another important development was made by Perdew and Zunger, who fitted an analytic expression to Quantum Monte Carlo data obtained by Ceperley and Alder [49], producing the widely used PL functional [50], which satisfies exact limits at both low and high densities.

- **Generalised Gradient Approximation (GGA).** The GGA improves upon the constant-density assumption of the LDA by incorporating the gradient of the electron density into the functional form. This allows GGA functionals to account for the inhomogeneity of the electron distribution more accurately.

  The general form of a GGA exchange-correlation functional is:

  $E_{xc}^{GGA}[\rho] = \int \rho(\mathbf{r})\varepsilon_{xc}(\rho(\mathbf{r}), \boldsymbol{\nabla}\rho(\mathbf{r}))\mathrm{d}\tau$

  As in the LDA, the exchange-correlation energy in the GGA framework is typically decomposed into separate exchange and correlation contributions. Various strategies have been developed to incorporate the density gradient. One influential approach, introduced by Becke, is based on the concept of the exchange hole. Among the functionals derived from this idea, the B88 exchange functional [51] is one of the most widely used.

  Some GGA functionals have been constructed as refinements of earlier GGAs. For example, the modified Perdew–Wang functional (mPW) [52] improves the long-range behaviour of the exchange term relative to its predecessors.

  The B88 exchange functional is often paired with the LYP correlation functional, developed by Lee, Yang, and Parr [53]. LYP is based on a transformation of the Colle-Salvetti correlation energy [54], originally derived for closed-shell systems, and incorporates the Weizsäcker kinetic energy term [55] to improve accuracy in regions of rapid density variation.

  Another notable GGA functional is P86 [56], which is built upon the idea of a natural separation between exchange and correlation. It recovers the gradient expansion for slowly varying densities and incorporates uniform-gas and inhomogeneity effects beyond the random phase approximation. Importantly, the gradient-dependent terms vanish for uniform densities, ensuring that the functional reduces to the local form in the homogeneous limit.

- **Meta-Generalised Gradient Approximation (meta-GGA).** The meta-GGA class of functionals extends the GGA by including higher-order derivatives of the density, such as the Laplacian $\nabla^2\rho(\mathbf{r})$ —in practice the kinetic energy density $\tau(\mathbf{r})$— which provides a measure of the localisation of the KS orbitals.

  The kinetic energy density is defined as:

  $\tau^L(\mathbf{r}) = -1/2 \sum_i^N \varphi_i^*(\mathbf{r})\nabla^2\varphi_i(\mathbf{r})$

  This leads to the general form of a meta-GGA exchange–correlation functional:

  $E_{xc}^{mGGA}[\rho] = \int \rho(\mathbf{r})\varepsilon_{xc}(\rho(\mathbf{r}), \boldsymbol{\nabla}\rho(\mathbf{r}), \tau(\mathbf{r}))\mathrm{d}\tau$

  The kinetic energy density and the Laplacian are connected through the external potential via the relation:

  $\tau(\mathbf{r}) = 1/2 \sum_i^N |\boldsymbol{\nabla}\varphi_i(\mathbf{r})|^2 - 1/4\nabla^2\rho(\mathbf{r})$

  Although computing the Laplacian is more demanding and early meta-GGA functionals offered only modest improvements over GGA, recent developments —notably the SCAN functional [57]— have shown significant gains in accuracy, especially for systems with varying degrees of localisation and inhomogeneity.

  Two well-known examples of meta-GGA functionals are: *i*) the VSXC functional [58], which is based on the density matrix expansion, and *ii*) the KCIS functional [59], which introduces a self-interaction-corrected correlation energy and a gap in the uniform electron gas.

- **Hybrid Functionals.** Hybrid functionals incorporate a fraction of exact exchange energy (HF-like level), using the KS orbitals. This combination enhances the accuracy of exchange interactions, particularly in systems where self-interaction errors are significant.

  The Hartree-Fock exchange energy is given by:
  $E_x^{HF} = -1/2 \sum_{i,j}^{N} \int \frac{\varphi_i^\star(\mathbf{r}_1)\varphi_j^\star(\mathbf{r}_1)\varphi_i(\mathbf{r}_2)\varphi_j(\mathbf{r}_2)}{r_{12}} \mathrm{d}\tau_1 \mathrm{d}\tau_2$
  One of the most widely used hybrid functionals is B3LYP, developed in 1994 by combining Becke's three-parameter exchange functional [60] with the LYP correlation functional [53]. Its form is given by:

  $E_{xc}^{B3LYP} = aE_x^{Slater} + (1-a)E_x^{HF} + bE_x^{Becke88} + cE_c^{LYP} + (1-c)E_c^{VWN},$

  where the parameters ($a = 0.80$, $b = 0.72$, and $c = 0.81$) were empirically determined to reproduce experimental values such as ionisation energies, atomisation enthalpies, proton affinities, and atomic energies.

  Other notable hybrid functionals include PBE0 [61], which blends 25 % exact HF exchange with 75 % DFT exchange from the PBE functional [62], resulting in a well-balanced formulation. Another widely used example is M06-2X [63], a meta-GGA hybrid designed to improve accuracy in modelling non-covalent interactions, thermochemistry, and reaction barrier heights.

Functionals in this work

This thesis primarily employs three exchange-correlation functionals: PBE [64], PBE0 [61], and M06-2X [63]. These were chosen for their proven balance between accuracy and computational efficiency, and their suitability for describing geometries, reaction energetics, and other electronic properties relevant to the systems studied.

PBE and PBE0

The Perdew-Burke-Ernzerhof (PBE) functional is a widely used GGA in DFT [64]. It was developed as a simplified and improved alternative to the earlier PW91 functional [65], which, despite its accuracy, exhibited some limietations: excessive complexity, non-transparent analytic forms, over-parameterisation, discontinuous parameter transitions, and incorrect scaling behaviour in certain density limits.

The correlation energy in PBE can be expressed as a sum of the uniform electron-gas correlation energy, $\varepsilon_C^{\text{unif}}$, and a gradient correction term, H:

$$E_C^{\text{PBE}}[\rho_\uparrow, \rho_\downarrow] = \int n(r) \left[\varepsilon_C^{\text{unif}}(r_s, \zeta) + H(r_s, \zeta, t)\right] \mathrm{d}\tau, \tag{2.54}$$

where r_s is the local Seitz radius, ζ the spin polarisation, and t a reduced density gradient. The gradient correction H is designed to satisfy conditions at slowly and rapidly varying density limits, as well as to avoid logarithmic singularities:

$$H = {}^{e^2}\!/_{a_0}\gamma\phi^3 \ln\left(1 + \frac{\beta}{\gamma}t^2 \frac{1 + At^2}{1 + At^2 + A^2t^4}\right). \tag{2.55}$$

The exchange energy functional in PBE, meanwhile, is constructed to fulfil conditions such as correct scaling behaviour, exact spin-scaling relations, and compliance with the Lieb-Oxford bound [66, 67]. Its enhancement factor $F_X(s)$ is given by:

$$F_X(s) = 1 + \kappa - \frac{\kappa}{1 + {}^{\mu s^2}\!/_{\kappa}}. \tag{2.56}$$

where $F_X(s)$ is the (dimensionless) exchange enhancement factor that multiplies the LDA exchange energy density, and s is another reduced density gradient,

$$s = \frac{|\boldsymbol{\nabla}\rho|}{2(3\pi^2)^{1/3}\rho^{4/3}}. \tag{2.57}$$

The hybrid variant PBE0 introduces Hartree-Fock exact exchange into the PBE formulation, improving predictions of electronic structure properties:

$$E_{XC}^{\text{PBE0}} = a_0 E_X^{\text{HF}} + (1 - a_0)E_X^{\text{PBE}} + E_C^{\text{PBE}}, \tag{2.58}$$

typically setting $a_0 = 0.25$ (25 % exact exchange).

M06-2X

The M06-2X functional [63] is a hybrid meta-GGA designed specifically to provide improved accuracy for non-covalent interactions, thermochemistry, and barrier heights. It incorporates 54 % exact HF exchange and includes additional dependence on the kinetic energy density, making it particularly effective in describing systems with pronounced electron delocalisation or dispersion interactions.

The meta-GGA formulation of M06-2X involves three primary density-dependent variables: the spin densities (ρ_σ), the reduced density gradients (s_σ), and the spin-dependent kinetic energy density (τ_σ).

The functional construction involves expressions adapted from the VSXC functional [58], using parameters fitted to empirical data. M06-2X defines auxiliary variables (z_σ) and functions (γ, h).

The exchange and correlation contributions to M06-2X are combined from VSXC and M05 functional forms. The total exchange-correlation energy is expressed as:

$$E_{XC}^{\mathrm{M06-2X}} = \frac{54}{100} E_X^{\mathrm{HF}} + \left(1 - \frac{54}{100}\right) E_X^{\mathrm{DFT}} + E_C^{\mathrm{DFT}}, \tag{2.59}$$

with the correlation energy split into opposite-spin ($\alpha\beta$) and parallel-spin ($\sigma\sigma$) components:

$$E_C^{\alpha\beta} = \int \varepsilon_{\alpha\beta}^{\mathrm{GEH}} \left[g_{\alpha\beta}(x_\alpha, x_\beta) + h(x_{\alpha\beta}, z_{\alpha\beta}) \right] \mathrm{d}\tau, \tag{2.60}$$

$$E_C^{\sigma\sigma} = \int \varepsilon_{\sigma\sigma}^{\mathrm{GEH}} \left[g_{\sigma\sigma}(x_\sigma) + h(x_\sigma, z_\sigma) \right] \mathrm{d}\tau. \tag{2.61}$$

This detailed construction allows M06-2X to reliably model chemically complex systems where accurate treatment of electron correlation is essential.

2.5.4 Computational implementation

All the theoretical considerations discussed so far are not only mathematically intriguing, but also offer deep insights into the physical nature of electronic systems. However, it is equally important to understand how these ideas are implemented computationally and whether their solutions are practically tractable.

Fortunately, the SCF algorithm, originally developed for HF theory, can be readily adapted for DFT calculations [43, 68]. By incorporating exchange-correlation effects into the Fock matrix, the SCF procedure becomes compatible with any chosen exchange-correlation functional.

The Fock matrix, which in the HF formalism depends on the core Hamiltonian, the Coulomb, and exchange integrals, expressed as:

$$\mathbf{F} = \mathbf{F}(\widehat{H}^{\text{core}},\, J,\, K), \tag{2.62}$$

can be modified at the DFT level by replacing the HF exchange term with the exchange-correlation contribution:

$$\mathbf{F} = \mathbf{F}(\widehat{H}^{\text{core}},\, J,\, \mathbf{F}^{\text{XC}}). \tag{2.63}$$

The exchange-correlation component of the Fock matrix for a given functional f takes the general form:

Key Equation 6

$$\mathbf{F}_{\mu\nu}^{\text{XC}\alpha} = \int \left[\frac{\partial f}{\partial \rho_\alpha} \phi_\mu \phi_\nu + \left(2 \frac{\partial f}{\partial \gamma_{\alpha\alpha}} \boldsymbol{\nabla} \rho_\alpha + \frac{\partial f}{\partial \gamma_{\alpha\beta}} \boldsymbol{\nabla} \rho_\beta \right) \cdot \boldsymbol{\nabla} \phi_\mu \phi_\nu \right] \mathrm{d}\tau. \tag{2.64}$$

This formulation allows exchange-correlation effects to be fully embedded within the SCF algorithm, as described earlier in Section 2.2.

2.6 Solvation Effects in Quantum Chemistry

Although all previous sections have modelled chemical systems in a non-interacting framework —completely isolated from the rest of the universe— this picture fails to reflect the actual conditions under which most chemical phenomena occur. With the exception of solid-state and astrochemistry, most chemical reactions and experimental structure determinations are carried out in solution. On the contrary, standard quantum chemical calculations often treat isolated molecular species in the gas phase. This mismatch between theoretical models and experimental conditions can lead to significant discrepancies in predicted reactivity, stability, and mechanism.

A classic example of this is the addition of bromine to an ethylenic hydrocarbon, a reaction whose mechanism is known to differ between gas-phase and solution-phase conditions. Even when the mechanism remains unchanged, the kinetic behaviour can vary dramatically: for instance, the rate constant for bromination may differ by a factor of 10^{10} when switching from carbon tetrachloride to water as solvent [69, 70].

Numerous strategies have been developed to incorporate solvation effects into quantum chemical calculations. Broadly, these approaches fall into two categories: *i*) *explicit* and *ii*) *implicit* solvation models. Explicit models treat solvent molecules individually, placing them around the solute to allow direct solute-solvent interactions to be sampled, typically via molecular dynamics or Monte Carlo simulations. In contrast, implicit models replace the discrete solvent with a continuous medium characterised by bulk properties such as the dielectric constant. These continuum approaches are generally less computationally demanding and are especially well-suited to routine quantum chemical workflows.

2.6.1 The Continuum Solvation Model

A basic continuum model relies on several idealised assumptions: *i*) the solute is treated at a uniform quantum mechanical level, *ii*) solute-solvent interactions are limited to electrostatic effects, *iii*) the system represents a very dilute solution, *iv*) the solvent is isotropic, *v*) only the electronic ground state of the solute is considered, and *vi*) dynamic effects (*e. g.*, solvent relaxation) are neglected [71].

An early attempt to estimate the electrostatic contribution to the free energy of solvation was due to Kirkwood [72], who proposed a model based on a multipole expansion of the solute charge distribution centred within a spherical cavity surrounded by a dielectric continuum representing the solvent.

A key concept in all continuum models is the cavity (illustrated in Figure 2.4) in which the solute is embedded. This cavity defines a void region within the continuous dielectric medium, intended to mimic the exclusion of solvent molecules from the solute volume. The shape and size of this cavity vary among different continuum models, but it should ideally exclude solvent penetration while containing the largest possible fraction of the solute's electron density [73]. The optimal cavity definition has been a topic of considerable discussion, and various schemes have been proposed, typically based on atom-specific radii or electron density isosurfaces.

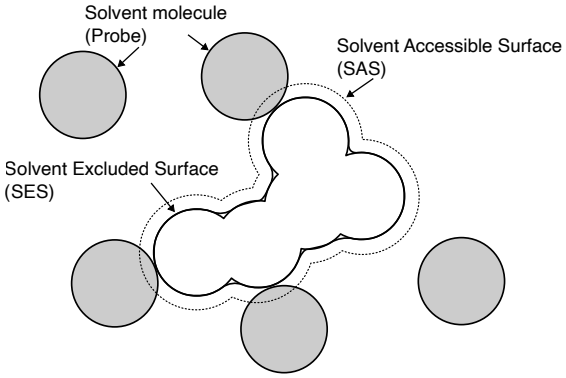


Figure 2.4. Solvent accessible surface (SAS) traced out by the center of the probe representing a solvent molecule. The solvent excluded surface (SES) is the topological boundary of the union of all possible probes that do not overlap with the molecule.

The cavity is often defined using concepts such as the solvent-accessible surface (SAS) and the solvent-excluding surface (SES), which aim to account for the physical space inaccessible to solvent molecules due to the three-dimensional shape of the solute. The SES, also known as the Connolly surface, is constructed by rolling a spherical probe (representing a solvent molecule) over the van der Waals surface of the solute.

This results in a composite surface comprising convex, toroidal, and concave (reentrant) regions, depending on whether the probe is in contact with one, two, or three atomic spheres, respectively. Analytical representations of the SES, first developed by Connolly [74], remain widely used, although they present computational challenges, such as the treatment of singularities and cusps.

Computational implementations, such as MSMS [75], have improved the stability and triangulation of these surfaces, making them suitable for applications in continuum solvation models. While detailed surface construction is rarely necessary for all solvation methods, the SES concept remains fundamental in defining realistic molecular cavities that exclude solvent penetration.

2.6.2 COSMO: Conductor-like Screening Model

The COSMO model is a dielectric continuum approach in which the solute is embedded within a molecule-shaped cavity surrounded by a dielectric medium characterised by a given dielectric constant ϵ. Electrostatic interactions are first computed under the assumption that the surrounding medium behaves as a perfect conductor ($\epsilon = \infty$), and are subsequently scaled to reflect the properties of the actual solvent.

Originally introduced by Klamt and Schüürmann [76], this method simplifies the solution of the electrostatic boundary problem. In the idealised case of a conductor, the total electrostatic potential vanishes at the surface of the cavity, which allows for an efficient determination of the induced surface charge distribution. To recover the correct behaviour for media with finite ϵ, the idealised surface charge density $\sigma^*(s)$ —obtained under the conductor assumption— is scaled according to: $\sigma(s) = f(\epsilon)\,\sigma^*(s)$, where s denotes a point on the cavity surface, and $f(\epsilon)$ is an empirical scaling function determined by comparison to accurate solute-solvent electrostatic energies, given by:

$$f(\epsilon) = \frac{\epsilon - 1}{\epsilon + k}, \tag{2.65}$$

where the empirical parameter k is typically set to a value between 0.5 and 1.0, depending on the implementation. In ADF, for example, the default atomic cavity radii are taken from the Van der Waals radii of the MM3 force field, scaled by a factor of 1.2 [77, 78].

It is important to note that the correction introduced by this factor is relatively small for solvents with a high dielectric constant, and the final solvation energy is generally insensitive to variations in the k parameter [71].

The inclusion of a linear parametrisation of non-electrostatic contributions is also possible as a function of surface area, introducing a correction that accounts for dispersion and cavity formation effects. Typically, only the surface area-dependent term is retained. This leads to the following expression:

$$E_{\text{non-elec}} = f(\epsilon)\left(Cav_0 + Cav_1 \times \text{area}\right). \tag{2.66}$$

2.6.3 The Quantum Mechanical/Molecular Mechanical (QM/MM) Models

In many chemical systems, the solvent acts primarily as a physical perturbation on a solute of interest (an individual molecule, a molecular complex, or a transition state) [69]. This separation of roles allows the chemically relevant subsystem to be treated at a higher level of theory, while the surrounding environment is described at a lower, less computationally demanding level.

Hybrid quantum mechanical/molecular mechanical (QM/MM) models take advantage of this partitioning. In these approaches, the solute is treated using quantum chemical methods, while the solvent is represented by a classical force field. The total Hamiltonian of the system is expressed as the sum of three contributions [79]:

$$\widehat{H}_{\text{QM/MM}} = \widehat{H}_{\text{QM}} + \widehat{H}_{\text{MM}} + \widehat{H}_{\text{QM/MM}}, \tag{2.67}$$

the first term is the quantum mechanical Hamiltonian of the solute, and the second is the classical Hamiltonian that describes the configuration of the solvent. Both are well-defined by the chosen quantum mechanical methodology and classical force field, respectively. The third term, which represents the interaction between the quantum solute and classical solvent, can be implemented in various ways.

One common approach includes: *i*) Van der Waals interactions between classical solvent atoms and quantum solute atoms; *ii*) long-range electrostatic interactions among the classical solvent molecules; and *iii*) the interaction between the electron density and nuclei of the solute and the classical solvent, which may also involve specific interactions such as hydrogen bonding [69].

Polarisable Embedding (PE) models extend this framework by introducing the self-energy of induced dipoles —representing the work required to induce the dipoles themselves— and the mutual repulsion between all induced dipoles. These contributions appear as the third and fourth terms in the following expression:

$$\begin{aligned} E^{\mathrm{QM/MM}} = {}& 1/2 \sum_i q_i V_i^{\mathrm{MM}} + \sum_i q_i V_i^{\mathrm{QM}} + 1/2 \sum_i \left(\alpha_i^{-1} \mu_i^2 + \sum_{j \neq i} \mu_i \mathscr{T}_{ij} \mu_j \right) \\ & - \sum_i \mu_i \left(E_i^{\mathrm{MM}} + E_i^{\mathrm{QM}} \right), \end{aligned} \tag{2.68}$$

where $\mathscr{T}_{ij}$ denotes the effective dipole-field interaction tensor [80, 81], which introduces distance dependent damping functions to avoid the divergence of the Coulomb interaction between two point dipoles when they get too close [82].

2.7 Quantum Theory of Atoms In Molecules

The electronic density is not only central to DFT, but also underpins the Quantum Theory of Atoms in Molecules (QTAIM). In contrast to DFT, which uses the density to compute a system's energy, QTAIM analyses its topological features, offering a real-space perspective on molecular structure and bonding [3].

By examining the topology of the electron density, QTAIM provides a rigorous quantum mechanical foundation for concepts such as atomic boundaries and chemical bonding. This framework enables the identification of atoms, functional groups, and molecular subunits within complex systems, including large assemblies and clusters.

2.7.1 Topological properties of the electronic density

In QTAIM, the analysis of electron density is based on the topology of the scalar field $\rho(\mathbf{r})$. Special attention is given to critical points (CPs), which correspond to fundamental structural features of a molecule: atoms, bonds, rings, and cages [3, 83, 84].

Critical points in a scalar field are defined as the locations where the gradient of the field vanishes:

$$\boldsymbol{\nabla}\rho(\mathbf{r_c}) = \frac{\partial\rho(\mathbf{r_c})}{\partial x}\hat{\mathbf{i}} + \frac{\partial\rho(\mathbf{r_c})}{\partial y}\hat{\mathbf{j}} + \frac{\partial\rho(\mathbf{r_c})}{\partial z}\hat{\mathbf{k}} = \mathbf{0}. \tag{2.69}$$

To classify a CP of the scalar field $\rho(\mathbf{r})$ —located at position $\mathbf{r_c}$— as a local minimum, maximum, or saddle point, we analyse the second derivatives of the field at that location. These are assembled into a symmetric matrix known as the Hessian.

$$\mathbf{H}_\rho(\mathbf{r_c}) = \begin{pmatrix} \frac{\partial^2\rho(\mathbf{r})}{\partial x^2} & \frac{\partial^2\rho(\mathbf{r})}{\partial x\partial y} & \frac{\partial^2\rho(\mathbf{r})}{\partial x\partial z} \\ \frac{\partial^2\rho(\mathbf{r})}{\partial y\partial x} & \frac{\partial^2\rho(\mathbf{r})}{\partial y^2} & \frac{\partial^2\rho(\mathbf{r})}{\partial y\partial z} \\ \frac{\partial^2\rho(\mathbf{r})}{\partial z\partial x} & \frac{\partial^2\rho(\mathbf{r})}{\partial z\partial y} & \frac{\partial^2\rho(\mathbf{r})}{\partial z^2} \end{pmatrix}_{\mathbf{r}=\mathbf{r_c}} . \tag{2.70}$$

The Hessian matrix is real and symmetric, and therefore diagonalisable. This diagonalisation corresponds to a rotation of the coordinate system, $(x, y, z) \to (x', y', z')$, such that the new axes align with the principal directions of curvature at the critical point.

$$\mathbf{H}_\rho(\mathbf{r_c}) = \begin{pmatrix} \frac{\partial^2\rho(\mathbf{r}')}{\partial x'^2} & 0 & 0 \\ 0 & \frac{\partial^2\rho(\mathbf{r}')}{\partial y'^2} & 0 \\ 0 & 0 & \frac{\partial^2\rho(\mathbf{r}')}{\partial z'^2} \end{pmatrix}_{\mathbf{r}'=\mathbf{r_c}} = \begin{pmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{pmatrix}_{\mathbf{r}'=\mathbf{r_c}} , \tag{2.71}$$

where λ_1, λ_2, and λ_3 are the eigenvalues of the Hessian matrix, representing the curvature of the electron density along the primed axes. Table 2.1 summarises the classification of critical points according to their characteristic curvature patterns and associated chemical interpretations.

Table 2.1. Topological description of the critical points (CP) used at the analysis of the $\rho(\mathbf{r})$ topology. The range (ω) represents the number of nonzero eigenvalues and the signature (σ) the algebraic sum of the eigenvalues signs.

Topological classification of critical points (CP)			
(ω, σ)	CP	Description	Interpretation
(3, −3)	NCP	All curvatures are negative; $\mathbf{r}_c$ is a local maximum of $\rho(\mathbf{r})$.	Position of a nucleus.
(3, −1)	BCP	Two negative curvatures and one positive.	Saddle point between two atomic basins.
(3, 1)	RCP	Two positive curvatures and one negative.	Saddle point surrounded by a closed loop of bond paths.
(3, 3)	CCP	All curvatures are positive; $\mathbf{r}_c$ is a local minimum of $\rho(\mathbf{r})$.	Local minimum fully enclosed by ring structures.

To move beyond the analysis of isolated points, we must consider the properties of regions in space. In QTAIM, the concept of an atom in a molecule is defined by the behaviour of the gradient vector field $\boldsymbol{\nabla}\rho(\mathbf{r})$, specifically its field lines. These lines, also called gradient paths, are trajectories $\sigma(t)$ that satisfy:

$$\sigma'(t) = \boldsymbol{\nabla}\rho\left(\sigma(t)\right). \tag{2.72}$$

Nuclei, due to their positive charge, act as attractors for the flux lines $\sigma(t)$. The region of space in which all flux lines converge to a given nucleus is known as an **atomic basin**, and corresponds to the quantum definition of an atom in QTAIM [3]. These atomic basins, or Bader atoms, are bounded by surfaces where the gradient vector field of the electron density satisfies the zero-flux condition:

$$\boldsymbol{\nabla}\rho(\mathbf{r}) \cdot \mathbf{n}(\mathbf{r}) = 0 \qquad \forall \mathbf{r} \in S(\Omega), \tag{2.73}$$

where Ω denotes the atomic basin, $S(\Omega)$ the surface bounding the basin, and $\mathbf{n}(\mathbf{r})$ the unit normal vector to the interatomic surface at point $\mathbf{r}$. An atom in QTAIM is thus defined as the union of a nucleus and its associated basin.

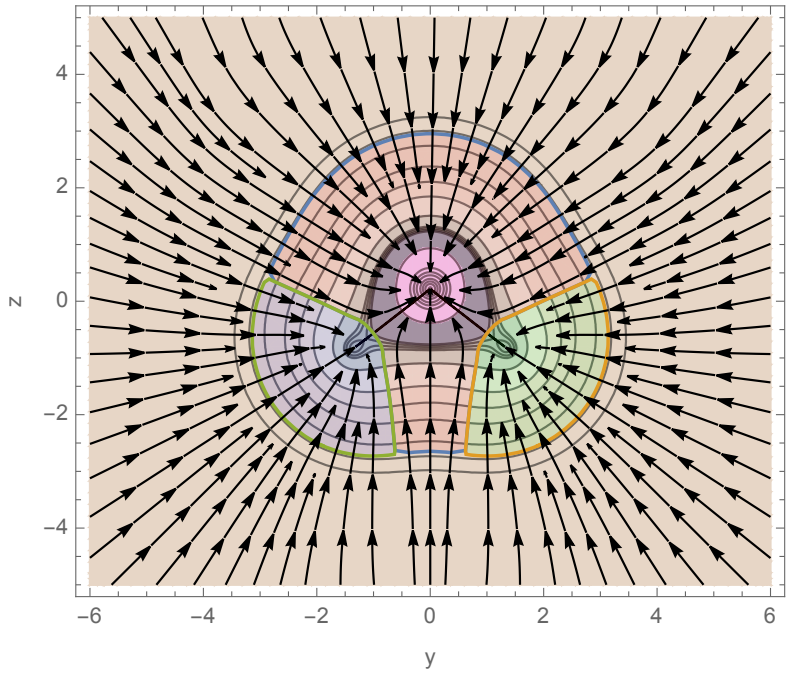


Figure 2.5. Atomic basins and gradient paths (black arrows) for the water molecule. The oxygen atom is shown in red, and the hydrogen atoms in green and blue. Blue, green, and orange contour lines represent zero-flux surfaces corresponding to the boundaries of the oxygen and hydrogen atomic basins. The orange background indicates regions excluded due to low electron density based on a threshold criterion.

In summary, the spatial partitioning into disjoint Bader regions [3] is governed by the gradient vector field of the electron density, $\boldsymbol{\nabla}\rho(\mathbf{r})$. This field defines a mapping $\mathbf{F} : \mathbb{R}^3 \to \mathbb{R}^3$, which can be analysed via its integral curves, or flux lines, given by the trajectories $\sigma(t) : \mathbb{R} \to \mathbb{R}^3$ defined in Equation 2.72.

As shown in Figure 2.5, the QTAIM partition provides well-defined spatial regions for each atom, enabling a detailed analysis of the system. The classification of the associated CP can then be carried out using the scheme summarised in Table 2.1.

2.7.2 Topological Constraints and the Poincaré-Hopf Theorem

The Poincaré-Hopf theorem provides a deep connection between analysis and topology. It relates the local behaviour of a vector field —through its CPs and their indices— to an invariant, the Euler characteristic $\chi(M)$ [85].

Theorem 2.7.1 Poincaré-Hopf Theorem

Let M be a compact, connected, and smooth manifold, & let V be a tangent vector field on M with isolated CPs. Then the sum of the indices of the isolated CPs is well defined and is equal to the Euler characteristic of M:

$$\sum_{c \in CP} \mathsf{index}(V, c) = \chi(M). \tag{2.74}$$

While the LHS of depends on a particular vector field, the RHS does not, *i. e.*, although the individual indices vary with the choice of vector field, their sum is determined solely by the topology of the manifold. The Euler characteristic does not depend on the smooth structure of M, only on its triangualation, the topological structure of M.

In the context of QTAIM, we are working with a partition of $\mathbb{R}^3$ induced by the gradient vector field of the electron density, $\boldsymbol{\nabla}\rho(\mathbf{r})$. The Poincaré-Hopf theorem can then be used to establish a constraint involving the number and types of CPs in the system, providing a consistency check on any topological analysis of ρ.

The alternating sum over the number of CPs of each index defines the Euler characteristic: $\chi(M) = \sum_{i=0}^{\max}(-1)^i M_i$, where M_i denotes the number of CPs of index i. For the case of a molecular system, the bounded region of $\mathbb{R}^3$ under consideration is homeomorphic to the closed 3-disk D^3, and thus $\chi(M) = 1$. However, for a periodic system, by their boundary conditions, the topological structure is that of a 3-torus T^3, where the Euler characteristic is zero, $\chi(T^3) = 0$.

$$\chi(M) = \begin{cases} n_{\mathrm{NCP}} - n_{\mathrm{BCP}} + n_{\mathrm{RCP}} - n_{\mathrm{CCP}} = 1 & \textit{molecular} \\ n_{\mathrm{NCP}} - n_{\mathrm{BCP}} + n_{\mathrm{RCP}} - n_{\mathrm{CCP}} = 0 & \textit{periodic} \end{cases} \tag{2.75}$$

2.7.3 Atomic properties in molecules

In QTAIM, the regions Ω correspond to atoms in the chemical sense, and it can be rigorously shown that the postulates of quantum mechanics apply within each of these atomic basins [3]. Enforcing the zero-flux condition on the gradient of the electron density yields a variationally well-defined framework for assigning properties to each subsystem independently [86]. Using the interatomic surfaces that define the boundaries of these regions, molecular properties can be expressed as the sum of the properties of the individual atomic basins:

$$A = \sum_{\Omega} a_{\Omega}, \tag{2.76}$$

where A denotes the total molecular property, and a_Ω is the corresponding contribution from the atomic basin Ω. This decomposition is based on the atomic variational principle, which states that if the operator $\hat{A}$ can be expressed as a sum of monoelectronic operators, $\hat{A} = \sum \hat{a}$, then its expectation value can be written as:

$$\begin{aligned} A(\Omega) &\equiv \langle \widehat{A} \rangle_{\Omega} \\ &= \int_{\Omega} \int \cdots \int \int \cdots \int [{}^{N}\!/_{2} \Psi_{el}^{\star} \hat{a} \Psi_{el} + (\hat{a}\Psi_{el})^{\star} \Psi_{el}] \, \mathrm{d}\omega_1 \ldots \mathrm{d}\omega_N \mathrm{d}\tau_2 \ldots \mathrm{d}\tau_N \mathrm{d}\tau_1. \end{aligned} \tag{2.77}$$

This expression shows that an atomic property can be obtained by integrating the expectation value of the corresponding operator over the atomic basin Ω. To facilitate this, the concept of a local operator density is introduced. The operator density $\rho_A(\mathbf{r})$ associated with $\hat{A}$ is defined as:

$$\rho_A(\mathbf{r}) = {}^{N}\!/_{2} \int \cdots \int \int \cdots \int [\Psi_{el}^{\star} \hat{a} \Psi_{el} + (\hat{a}\Psi_{el})^{\star} \Psi_{el}] \mathrm{d}\omega_1 \mathrm{d}\omega_2 \ldots \mathrm{d}\omega_N \mathrm{d}\tau_2 \ldots \mathrm{d}\tau_N, \tag{2.78}$$

where the integration is carried out over all coordinates except $\mathbf{r}$, which serves as the evaluation point of the density. The property associated with a given atomic basin is then obtained by integrating this density over the region:

$$A(\Omega) = \int_{\Omega} \rho_A(\mathbf{r}) \mathrm{d}\tau. \tag{2.79}$$

One of the simplest molecular properties that can be partitioned into atomic contributions is the atomic charge. Following the definition of atomic basins, the charge associated with a given atom is obtained as the difference between the nuclear charge and the number of electrons within its basin:

$$q(\Omega) = Z_\Omega - \int_\Omega \rho(\mathbf{r}) \mathrm{d}\tau. \tag{2.80}$$

While the decomposition of some properties such as charge is relatively straightforward, the partitioning of other quantities requires more elaborate treatments. In the following sections, we examine how QTAIM enables the decomposition of the molecular dipole moment and polarisability into well-defined atomic contributions.

2.7.4 Dipole Moment

The electric dipole moment is a vector quantity that characterises the spatial distribution of positive and negative charge within a molecular system. Classically, it is defined as the product of a point charge and its displacement vector, $\mathbf{p} = q\mathbf{d}$. In molecular systems, however, the total dipole moment $\boldsymbol{\mu}$ arises from the combined contributions of all nuclei and electrons. It is conventionally expressed as the sum of nuclear ($\boldsymbol{\mu}_c$) and electronic ($\boldsymbol{\mu}_p$) components: $\boldsymbol{\mu} = \boldsymbol{\mu}_c + \boldsymbol{\mu}_p$.

The nuclear contribution is straightforward to evaluate, involving the sum of the nuclear charges Z_A weighted by their position vectors $\mathbf{R}_A$ (Equation 2.81). The electronic component, by contrast, involves integrating the position operator $\hat{r}$ weighted by the electron density $\rho(\mathbf{r})$ over all space (Equation 2.82):

$$\boldsymbol{\mu}_c = \sum_A Z_A \mathbf{R}_A, \tag{2.81}$$

$$\boldsymbol{\mu}_p = -\int_{\mathbb{R}^3} \mathbf{r}\rho(\mathbf{r}) \mathrm{d}\tau. \tag{2.82}$$

In practical computational implementations, the electronic contribution to the dipole moment is typically expressed in terms of the one-electron density matrix $P_{\mu\nu}$ and the atomic orbital basis functions μ and ν:

$$\boldsymbol{\mu}_p = -\sum_{\mu\nu} P_{\mu\nu} \langle \mu | \mathbf{r} | \nu \rangle . \tag{2.83}$$

Insight 2.7.1 (Dipole moment decomposition)

Within QTAIM, both the nuclear and electronic components can be further decomposed into contributions from the atomic basins Ω_α. This results in an atom-based decomposition of the total molecular dipole moment:

$$\begin{aligned}
\boldsymbol{\mu} &= \boldsymbol{\mu}_c + \boldsymbol{\mu}_p \\
&= \sum_\alpha Z_\alpha \mathbf{R}_\alpha - \sum_\alpha \int_{\Omega_\alpha} \mathbf{r}\rho(\mathbf{r})\mathrm{d}\tau \\
&= \sum_\alpha \left(Z_\alpha \mathbf{R}_\alpha - \int_{\Omega_\alpha} (\mathbf{r} - \mathbf{R}_\alpha)\rho(\mathbf{r})\mathrm{d}\tau - \int_{\Omega_\alpha} \mathbf{R}_\alpha \rho(\mathbf{r})\mathrm{d}\tau \right) \\
&= \sum_\alpha \left(q_\alpha \mathbf{R}_\alpha - \int_{\Omega_\alpha} (\mathbf{r} - \mathbf{R}_\alpha)\rho(\mathbf{r})\mathrm{d}\tau \right)
\end{aligned} \tag{2.84}$$

Although the nuclear contribution to the dipole moment is origin-dependent, QTAIM allows a reformulation based on BCPs that offers additional physical insight [87]:

$$\boldsymbol{\mu}_c(\Omega) = \sum_\Lambda [\mathbf{R}_\Omega - \mathbf{R}_{\mathrm{BCP}}(\Omega|\Lambda)] Q(\Omega|\Lambda), \tag{2.85}$$

where, $\mathbf{R}_\Omega$ is the centroid of atomic basin Ω, $\mathbf{R}_{\mathrm{BCP}}(\Omega|\Lambda)$ denotes the position vector of the BCP between atoms Ω and Λ, and $Q(\Omega|\Lambda)$ is the bond charge shared between the two atoms. The computation of these bond charges is non-trivial, particularly in systems with complex bonding topologies. Their values are obtained by solving a linear system derived from charge conservation and symmetry constraints:

$$q(\Omega) = \sum_\Lambda Q(\Omega|\Lambda), \tag{2.86}$$

$$Q(\Omega|\Lambda) = -Q(\Lambda|\Omega), \tag{2.87}$$

$$0 = \sum_\Omega Q(\Omega|\Omega + 1), \tag{2.88}$$

where, $q(\Omega)$ denotes the QTAIM atomic charge, as defined in Equation 2.80. Equation 2.88 follows directly from the combination of Equations 2.86 and 2.87. Solving this system requires prior knowledge of both the QTAIM atomic charges and the full bonding topology of the molecule, including the presence of ring structures.

The minimal linear system required to solve the bond-charge decomposition problem involves N_{BCP} variables and a total of $N_{\text{atoms}} + N_{\text{RCP}}$ equations. A hypothetical topology is shown in Figure 2.6, and its corresponding matrix representation is given in Equation 2.89.

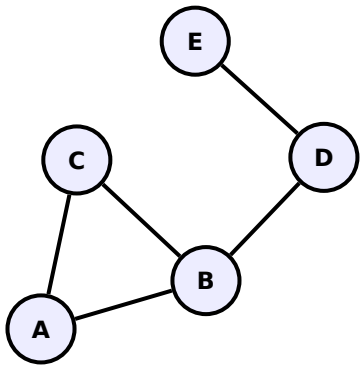


Figure 2.6. Graph of a system with 5 atoms and 1 ring.

$$\overbrace{\begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ -1 & 0 & 1 & 1 & 0 \\ 0 & -1 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & -1 \\ 1 & -1 & 1 & 0 & 0 \end{pmatrix}}^{\text{NBCP}} \begin{pmatrix} Q(A|B) \\ Q(A|C) \\ Q(B|C) \\ Q(B|D) \\ Q(D|E) \end{pmatrix} = \begin{pmatrix} q(A) \\ q(B) \\ q(C) \\ q(D) \\ q(E) \\ 0 \end{pmatrix} \begin{matrix} \left.\vphantom{\begin{matrix} a \\ a \\ a \\ a \\ a \end{matrix}}\right\} \text{N atoms} \\ \left.\vphantom{a}\right\} \text{NRCP} \end{matrix} \tag{2.89}$$

2.7.5 Polarisability

The molecular polarisability describes the response of a molecule's dipole moment to an external electric field $\mathbf{F}$. It plays a central role in determining macroscopic properties such as the dielectric constant of bulk media, as captured by the Clausius-Mossotti relation [88]. At the molecular level, the polarisability is a symmetric 3×3 tensor, defined as:

Key Equation 7: Polarisability tensor

$$\underline{\underline{\alpha}} = \left(\frac{\partial \mu_i}{\partial F_j}\right)_{F=0} \tag{2.90}$$

A scalar measure of the polarisability, independent of the coordinate system, is given by the isotropic mean polarisability. Owing to the invariance of the trace under orthogonal transformations:

$$\bar{\alpha} = 1/3\,\mathrm{Trace}(\underline{\underline{\alpha}}). \tag{2.91}$$

Polarisabilities can be computed either analytically (*e. g.*, via sum-over-states methods) or numerically. The numerical approach relies on a finite-difference approximation of the derivative, in which dipole moments are evaluated under small applied electric fields [89].

Figure 2.7 illustrates a molecular system placed in a simulation box and subjected to external electric fields applied along two Cartesian directions. A common approach to computing the polarisability involves comparing the dipole moment of the system in the presence of a small electric field with that of the unperturbed system. However, this method can lead to reduced numerical accuracy, as it mixes contributions from different terms in the Taylor expansion of the energy with respect to the field.

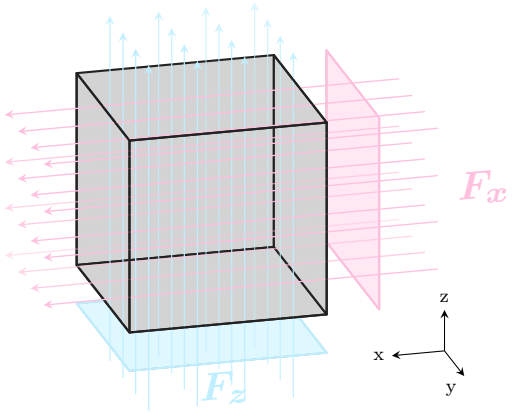


Figure 2.7. System in a box, subjected to an external electric field in x-axis and z-axis.

For a system subjected to an external electric field —denoted as $\mathbf{F}$— the total electronic energy $E(\mathbf{F})$ can be expanded as a Taylor series:

$$E(\mathbf{F}) = E^{(0)} + \sum_i \left(\frac{\partial E}{\partial F_i}\right)_{F=0} F_i + \frac{1}{2}\sum_i \left(\frac{\partial^2 E}{\partial F_i \partial F_j}\right)_{F=0} F_i F_j + \ldots \tag{2.92}$$

Direct comparison of the perturbed system $E(\mathbf{F})$ with the unperturbed system $E^{(0)}$ mixes contributions from first-order and second-order terms:

$$\Delta E(\mathbf{F}) = E(\mathbf{F}) - E^{(0)} \tag{2.93}$$

$$= -\sum_i \mu_i F_i - \frac{1}{2}\sum_{ij} \alpha_{ij} F_i F_j - \dots \tag{2.94}$$

To improve numerical accuracy and obtain a cleaner estimate of the polarisability tensor, a symmetric finite-difference scheme is typically employed. In this method, dipole moments are computed under equal-magnitude positive and negative electric field perturbations, usually around ± 0.005 a.u. [90]. This symmetry cancels out the odd-order terms in the Taylor expansion, resulting in a more accurate numerical derivative:

Note:

$$\alpha_{ij} = \begin{cases} \lim\limits_{\varepsilon\to 0} \dfrac{\mu_i^{+\varepsilon_j} - \mu_i^0}{\varepsilon_j} \\ \lim\limits_{\varepsilon\to 0} \dfrac{\mu_i^0 - \mu_i^{-\varepsilon_j}}{\varepsilon_j} \end{cases} \qquad \therefore \quad \lim_{\varepsilon\to 0} \frac{\mu_i^{+\varepsilon_j} - \mu_i^{-\varepsilon_j}}{2\varepsilon_j} \tag{2.95}$$

where, $\mu_i^{+\varepsilon_j}$ denotes the i-th component of the dipole moment when a small electric field ε_j is applied in the j-direction.

In practice, at least six calculations —corresponding to positive and negative field perturbations along the three Cartesian axes— are required to numerically construct the full polarisability tensor.

The atomic decomposition of molecular polarisability arises naturally from the QTAIM-based decomposition of the dipole moment. It is important to note that this decomposition is implemented numerically, as analytic expressions for atomic contributions are unavailable.

Due to numerical noise and finite numerical differentiation steps, the resulting polarisability tensors may exhibit slight asymmetry. This residual asymmetry, typically very small, can be corrected by applying a symmetrisation procedure as recommended by Nye [91]:

$$\underline{\underline{\alpha}}^{S} = 1/2(\underline{\underline{\alpha}} + \underline{\underline{\alpha}}^{\mathrm{T}}). \tag{2.96}$$

Symmetrisation ensures that the physically required symmetry properties of the tensor are maintained, and it reduces numerical errors resulting from neglected coupling between atomic basins during decomposition.

Finally, it is worth emphasising that computing polarisability tensors via finite differences incurs a significantly higher computational cost than a single-point dipole calculation.

2.7.6 Excited States

The study of electronic excited states plays an essential role to understand the spectroscopic and photochemical properties of molecular systems. Following the approach adopted in previous sections —where molecular properties such as dipole moments and polarisabilities were decomposed into atomic contributions via QTAIM— this subsection will similarly progress from global characterisation of electronic excitations toward an atomic-level interpretation facilitated by QTAIM partitioning.

We will restrict our focus here to vertical excitations, defined as electronic transitions occurring without nuclear displacement. This approximation aligns with the Franck – Condon principle [92, 93], assuming fixed nuclear geometry for both ground and excited states, and is widely employed in theoretical treatments of electronic excitations.

To characterise the nature of these excitations, it is insightful to examine the redistribution of the electron density between the excited $\rho_{\mathrm{EX}}(\mathbf{r})$ and ground states $\rho_{\mathrm{GS}}(\mathbf{r})$. This redistribution can be described quantitatively by the electron density difference:

$$\Delta\rho(\mathbf{r}) = \rho_{\mathrm{EX}}(\mathbf{r}) - \rho_{\mathrm{GS}}(\mathbf{r}). \tag{2.97}$$

The density difference $\Delta\rho(\mathbf{r})$ provides a spatial map of the electron redistribution upon excitation. Regions where $\Delta\rho(\mathbf{r}) > 0$ correspond to electron accumulation, while regions where $\Delta\rho(\mathbf{r}) < 0$ indicate electron depletion. To isolate these contributions, it is useful to define the positive and negative components of the density difference, denoted ρ_+ and ρ_-, respectively. These functions distinguish between areas where electronic charge has been gained (typically interpreted as particle-like) and lost (hole-like) [94, 95].

Definition 2.7.1: ρ_+ & ρ_-

$$\rho_+(\mathbf{r}) = \begin{cases} \Delta\rho(\mathbf{r}) & \text{if } \Delta\rho(\mathbf{r}) > 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.98}$$

$$\rho_-(\mathbf{r}) = \begin{cases} \Delta\rho(\mathbf{r}) & \text{if } \Delta\rho(\mathbf{r}) < 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.99}$$

A useful measure of charge separation can be obtained by computing the centroids of the ρ_+ and ρ_- distributions. These centroids, denoted $\mathbf{R}_+$ and $\mathbf{R}_-$, correspond to the average positions of the regions where electron density is accumulated and depleted, respectively.

$$\mathbf{R}_\pm = \frac{\int \mathbf{r}\rho_\pm(\mathbf{r})\mathrm{d}\tau}{\int \rho_\pm(\mathbf{r})\mathrm{d}\tau} = (x_\pm, y_\pm, z_\pm). \tag{2.100}$$

The spatial extent of charge transfer upon excitation can be estimated by the Euclidean distance between the centroids $\mathbf{R}_+$ and $\mathbf{R}_-$:

$$D_{\mathrm{CT}} = \|\mathbf{R}_+ - \mathbf{R}_-\|. \tag{2.101}$$

This distance characterises how far the electron density is displaced during the excitation. To quantify the magnitude of the charge transferred, we define q_{CT} as the integrated value of either ρ_+ or ρ_-:

$$q_{\mathrm{CT}} = \int \rho_+(\mathbf{r})\mathrm{d}\tau = -\int \rho_-(\mathbf{r})\mathrm{d}\tau. \tag{2.102}$$

Together, D_{CT} and q_{CT} provide a compact and physically meaningful expression for the change in dipole moment associated with the charge transfer process:

$$\|\Delta\boldsymbol{\mu}_{\mathrm{CT}}\| = D_{\mathrm{CT}} \cdot q_{\mathrm{CT}}. \tag{2.103}$$

A more detailed link between orbital transitions and real-space charge redistribution can be established by expressing ρ_+ and ρ_- in terms of natural transition orbitals (NTOs). Within this framework, ρ_+ and ρ_- correspond to the particle (electron) and hole densities, respectively, each constructed from weighted contributions of individual NTOs. This representation connects the orbital nature of the excitation with physically observable quantities such as the dipole moment change and the charge-transfer distance D_{CT}.

Insight 2.7.2 (Derivation of $\boldsymbol{\mu}_{\mathrm{CT}}$)
knowing $\rho^e = \sum \lambda(\phi^{IO})^2$ and $\rho^h = \sum \lambda(\phi^{OI})^2$,
where ϕ denotes the natural transition molecular orbitals,

all integrals run over $\mathrm{d}\tau$, the differentials are omitted for readability.
Jacobi's delta is used to indicate a particular differential within a integral, $\int_{\partial x} f(x,y) = \int f(x,y)\mathrm{d}x$.
The sign function returns -1, 0, or 1 depending on whether x is negative, zero, or positive, respectively.

$$\|\Delta\boldsymbol{\mu}_{\mathrm{CT}}\| = D_{\mathrm{CT}} \cdot q_{\mathrm{CT}} = D_{\mathrm{CT}} \int \rho_+ = -D_{\mathrm{CT}} \int \rho_- \tag{2.104}$$

$$= \|\mathbf{R}_+ - \mathbf{R}_-\| \int \rho_+ = \left\| \frac{\int \mathbf{r}\rho_+}{\int \rho_+} - \frac{\int \mathbf{r}\rho_-}{\int \rho_-} \right\| \int \rho_+ = \left\| \frac{\int \mathbf{r}\rho_+}{\int \rho_+} + \frac{\int \mathbf{r}\rho_-}{\int \rho_+} \right\| \int \rho_+$$

$$= \left\| \int \mathbf{r}\rho_+ + \int \mathbf{r}\rho_- \right\| \frac{\int \rho_+}{\|\int \rho_+\|} = \left\| \int \mathbf{r}\rho_+ + \int \mathbf{r}\rho_- \right\| \mathrm{sign}\left(\int \rho_+ \right) \tag{2.105}$$

with the notation $\rho_+ = \rho^e$ and $\rho_- = \rho^h$, we can write the most explicit form of the dipole moment change as

$$\|\Delta\boldsymbol{\mu}_{\mathrm{CT}}\| = \left\| \int \mathbf{r} \sum \lambda_i^2 |\psi_i^e|^2 + \int \mathbf{r} \sum \lambda_i^2 \left|\psi_i^h\right|^2 \right\| \mathrm{sign}\left(\int \sum \lambda_i^2 |\psi_i^e|^2 \right) \tag{2.106}$$

$$= \left\| \int_{\partial \mathbf{r}^e} \mathbf{r}^e \sum_i \lambda_i \left[\sum_p U_{pi}^e \phi_p(\mathbf{r}^e) \right]^2 - \int_{\partial \mathbf{r}^h} \mathbf{r}^h \sum_i \lambda_i \left[\sum_p V_{pi}^h \phi_p(\mathbf{r}^h) \right]^2 \right\| \cdots$$

$$\cdots \times \mathrm{sign}\left(\int_{\partial \mathbf{r}^e} \sum_i \lambda_i^2 \left[\sum_p U_{pi} \phi_p(\mathbf{r}^e) \right]^2 \right) \tag{2.107}$$

Various analyses of excited states rely on molecular orbitals and associated descriptors, such as the charge-transfer distance ($\Delta\mathbf{r}$) and the spatial overlap of a given excitation (Λ) [96, 97]. Equations 2.108 and 2.109, where $K_{ia} = X_{ia} + Y_{ia}$, account for both excitation (X_{ia}) and de-excitation (Y_{ia}) coefficients.

$$\Delta r = \frac{\sum_{ia} K_{ia}^2 |\langle \phi_a | r | \phi_a \rangle - \langle \phi_i | r | \phi_i \rangle|}{\sum_{ia} K_{ia}^2} \qquad \Lambda = \frac{\sum_{ia} K_{ia}^2 \langle |\phi_i| |\phi_a| \rangle}{\sum_{ia} K_{ia}^2} \tag{2.108}$$

$$\Delta r = \frac{\sum_{ia} K_{ia}^2 |\int r\phi_a^2 - \int r\phi_i^2|}{\sum_{ia} K_{ia}^2} \qquad \Lambda = \frac{\sum_{ia} K_{ia}^2 \int |\phi_i| |\phi_a|}{\sum_{ia} K_{ia}^2} \tag{2.109}$$

Since the integrals involved are defined over the entire spatial domain, real-space partitioning schemes —such as that used in QTAIM— can apply the same expressions by simply restricting the integration to individual atomic basins. Consequently, in a code that already supports $\Delta\mathbf{r}$ and Λ, such as ADF, extending the analysis to atomic contributions would require only modest additional effort.

By integrating the functions ρ_+ and ρ_- over each basin, we obtain atomic-level descriptors for electron depletion and accumulation, respectively:

$$q_{\pm}^{\mathrm{CT}}(\Omega) = \int_{\Omega} \rho_{\pm}(\mathbf{r})\mathrm{d}\mathbf{r}. \tag{2.110}$$

where Ω denotes the spatial domain associated with an atomic basin. These values indicate the amount of electron density gained or lost by each atom during the transition, enabling a more detailed interpretation of the excitation in chemically meaningful terms.

Conceptually, this atomic decomposition follows the same logic as standard population analyses, where global quantities —such as the total charge or the dipole moment— are partitioned into atomic contributions. The total integrals remain unchanged; only their domains are split according to the real-space partitioning.

2.8 Conceptual DFT

The topological analysis of the electron density in QTAIM represents just one facet of density-based approaches to chemical reactivity. A complementary perspective is offered by Conceptual Density Functional Theory (CDFT), which focuses on how variations in global and local electronic descriptors govern chemical behaviour. The foundations of CDFT can be traced to the work of Parr and co-workers, who identified the Lagrange multiplier in density functional theory —traditionally associated with the chemical potential— as the negative of electronegativity [98].

This idea was further developed by Parr and Pearson in 1983, who related the second derivative of the energy with respect to the number of electrons to the concept of chemical hardness [99]. The formal introduction of the Fukui function in 1984 extended the framework to local reactivity indices, enabling site-specific analyses of chemical reactivity [100]. These developments laid the groundwork for a density-based interpretation of chemical concepts, with the term "Conceptual DFT" formally introduced in 1995 [2].

Conceptual DFT offers a hierarchy of global and local descriptors —such as the chemical potential, chemical hardness, softness, Fukui functions, dual descriptors, and electrophilicity index— which allow us to predict and rationalise patterns of chemical reactivity.

2.8.1 Global Descriptors

The cornerstone of this approach is the identification of the chemical potential μ as the Lagrange multiplier associated with the constraint on electron number in the Euler-Lagrange variational formalism:

Insight 2.8.1 (Chemical Potential as Lagrange Multiplier)
The existence of the energy as a functional of the electron density, the constraint on the number of electrons, and the variational principle let us write the Euler-Lagrange equation as:

$$\frac{\delta}{\delta\rho(\mathbf{r})}\left(E[\rho]-\mu\left[\int\rho(\mathbf{r})\mathrm{d}\mathbf{r}-N\right]\right)=0 \tag{2.111}$$

implying $\mu=\frac{\delta E[\rho]}{\delta\rho(\mathbf{r})}$ as the Lagrange multiplier to enforce the electron-number constraint.

Solving it for the exact density, the chemical potential can be expressed as the partial derivative of the energy with respect to the number of electrons, quantifing the change in ground-state energy upon an infinitesimal change in electron number at constant external potential $v(\mathbf{r})$:

$$\mu=v(\mathbf{r})+\frac{\delta F[\rho]}{\delta\rho(\mathbf{r})}=\left(\frac{\partial E}{\partial N}\right)_{v(\mathbf{r})}. \tag{2.112}$$

Note:
Using the chain rule, we can show that:

$$\left(\frac{\partial E}{\partial N}\right)_{v(\mathbf{r})}=\int\left(\frac{\delta E}{\delta\rho}\right)_{v(\mathbf{r})}\left(\frac{\partial\rho}{\partial N}\right)_{v(\mathbf{r})}\mathrm{d}\mathbf{r} \tag{2.113}$$

$$=\mu\frac{\partial}{\partial N}\left[\int\rho\mathrm{d}\mathbf{r}\right]=\mu. \tag{2.114}$$

Accordingly, its sign and magnitude measure the system's propensity to exchange electrons with its surroundings: a more negative μ indicates a stronger tendency to accept electronic charge and vice versa.

From a thermodynamic perspective, the chemical potential governs the direction of charge transfer, favouring flow from regions of high μ to low μ. Importantly, μ can be connected to experimentally accessible quantities such as the vertical ionisation potential I and the vertical electron affinity A.

Using a finite difference approximation under the assumption of piecewise linearity of the energy with respect to electron number, we obtain:

$$\mu = \left(\frac{\partial E}{\partial N}\right)_{v(\mathbf{r})} \approx -\chi = \frac{I + A}{2}, \tag{2.115}$$

where χ is the electronegativity, traditionally defined as the tendency of an atom or molecule to attract electrons. In this formulation, electronegativity and chemical potential are essentially two sides of the same coin, differing only by a sign.

Within the Koopmans' theorem, the chemical potential can be approximated using frontier molecular orbital energies. Specifically, the ionisation potential I and electron affinity A are associated with the negative of the HOMO and LUMO energies, respectively [101].

Note:

Consequently, the chemical potential can be expressed as:

$$\mu \approx -\chi = {}^1\!/_2(\epsilon_{\text{HOMO}} + \epsilon_{\text{LUMO}}). \tag{2.116}$$

Building on the Hard and Soft Acids and Bases principle, Pearson introduced the concept of chemical hardness η [102], which measures a system's resistance to change in electron number. Formally, it corresponds to the second derivative of the energy with respect to the number of electrons [99], and the inverse of the hardness defines the softness S [103], a measure of the system's reactivity or its ability to adapt its electron density in response to external perturbations:

Definition 2.8.1: Softness and Hardness

$$\eta = \left(\frac{\partial^2 E}{\partial N^2}\right)_{v(\mathbf{r})} = \left(\frac{\partial \mu}{\partial N}\right)_{v(\mathbf{r})} \tag{2.117}$$

$$= I - A \approx -(\epsilon_{\text{HOMO}} - \epsilon_{\text{LUMO}}), \tag{2.118}$$

$$S = \frac{1}{\eta} = \left(\frac{\partial N}{\partial \mu}\right)_{v(\mathbf{r})} \tag{2.119}$$

This global descriptor balances the stabilizing energy a system gains upon acquiring charge (via μ) with its resistance to such charge transfer (via η), offering a compact measure of electrophilic power.

Hardness is generally associated with a system's resistance to deformation of its electron density under external perturbations. As such, hard systems tend to exhibit low polarisability and magnetisability, while soft systems respond more readily to changes in their environment. To further quantify a system's reactivity, Parr introduced the electrophilicity index [104], which combines the tendency of a system to acquire electrons (through the chemical potential) with its resistance to charge transfer (through the hardness):

$$\omega = \frac{\mu^2}{2\eta} = \frac{\chi^2}{2\eta}. \tag{2.120}$$

2.8.2 Local Descriptors

Many methods have been developed to predict if a chemical reaction will happen or not, and in case it happens, how it will proceed. One of the most popular method of prediction is frontier molecular orbital (FMO) theory [105], which relies on the shapes and symmetries of the highest occupied and lowest unoccupied molecular orbitals (HOMO and LUMO).

However, a major limitation of FMO theory is its reliance on a fixed orbital picture, which does not account for electron correlation or orbital relaxation. This limitation motivated the development of the Fukui function [100, 106, 107] within the context of DFT a reactivity descriptor that preserves the conceptual foundation of FMO while, in principle, incorporating both electron correlation [108] and orbital relaxation effects [107, 109, 110].

Definition 2.8.2: Fukui Function

$$f(\mathbf{r}) = \left[\frac{\delta\mu}{\delta v(\mathbf{r})}\right]_N = \left[\frac{\partial\rho(\mathbf{r})}{\partial N}\right]_{v(\mathbf{r})} \tag{2.121}$$

where μ is the electronic chemical potential, the negative of the electronegativity, $v(\mathbf{r})$ is the external potential due to the atomic nuclei, $\rho(\mathbf{r})$ is the electron density, and N is the number of electrons.

Because the electron density $\rho(\mathbf{r})$ exhibits discontinuities as a function of the number of electrons N [111, 112], its derivative with respect to N must be evaluated from the left and from the right.

This yields two distinct Fukui functions, each appropriate for describing different types of reactivity: one for nucleophilic attack (electron addition) and another for electrophilic attack (electron removal).

Insight 2.8.2 (Fukui functions $f^{\pm}(\mathbf{r})$)

$$f^{\pm}(\mathbf{r}) = \left[\frac{\partial \rho(\mathbf{r})}{\partial N}\right]^{\pm}_{v(\mathbf{r})} = \begin{cases} f^{+}(\mathbf{r}) & = \rho(N+1) - \rho(N) \\ f^{-}(\mathbf{r}) & = \rho(N) - \rho(N-1) \end{cases} \tag{2.122}$$

If we consider the density as a function of the KS orbitals and the occupation numbers,

$$\rho(\mathbf{r}) = \sum_{i=1}^{\infty} n_i |\phi_i(\mathbf{r})|^2 \quad \text{with} \quad n_i = \begin{cases} 1 & i \leq HOMO \\ 0 & i \geq LUMO \end{cases} \tag{2.123}$$

then, the Fukui functions became:

$$f^{+}(\mathbf{r}) = |\phi_{\mathrm{LUMO}}(\mathbf{r})|^2 + \sum_{\mathrm{LUMO}}^{\infty} \left(\frac{\partial |\phi_i(\mathbf{r})|^2}{\partial N}\right)^{+}_{v(\mathbf{r})} \tag{2.124}$$

$$f^{-}(\mathbf{r}) = |\phi_{\mathrm{HOMO}}(\mathbf{r})|^2 + \sum_{1}^{\mathrm{HOMO}} \left(\frac{\partial |\phi_i(r)|^2}{\partial N}\right)^{-}_{v(\mathbf{r})} \tag{2.125}$$

And therefore, neglecting the orbital relaxation terms, from linking to frontier molecular orbital theory,

$$f^{+}(\mathbf{r}) \approx |\phi_{\mathrm{LUMO}}|^2 \tag{2.126}$$

$$f^{-}(\mathbf{r}) \approx |\phi_{\mathrm{HOMO}}|^2 \tag{2.127}$$

Additionally, the dual descriptor, proposed by Morell [113], is defined as the mixed second derivative of the energy with respect to the external potential and the number of electrons, and equivalently, as the second derivative of the electron density with respect to the number of electrons:

$$f^{(2)}(\mathbf{r}) = \frac{\delta^3 E}{\partial N^2 \delta v(\mathbf{r})} = \left(\frac{\partial^2 \rho(\mathbf{r})}{\partial N^2}\right)_{v(\mathbf{r})} = \left(\frac{\partial f(\mathbf{r})}{\partial N}\right)_{v(\mathbf{r})} = \left(\frac{\delta \eta}{\delta v(\mathbf{r})}\right)_{N} \tag{2.128}$$

In practice, the dual descriptor is computed as the difference between the electrophilic and nucleophilic Fukui functions at a given point $\mathbf{r}$. Its sign provides insight into local reactivity: if $f^{(2)}(\mathbf{r}) > 0$, the site is more electrophilic than nucleophilic; conversely, if $f^{(2)}(\mathbf{r}) < 0$, the site is more nucleophilic than electrophilic.

The linear response function $\chi(\mathbf{r}, \mathbf{r}')$, —sometimes denoted as $\omega(\mathbf{r}, \mathbf{r}')$ and named polarisability kernel— [114] has been employed to probe local and non-local features of chemical reactivity. This function quantifies how the electron density at a point $\mathbf{r}$ responds to an external perturbation applied at another point $\mathbf{r}'$. Formally, it is defined as [115]:

$$\chi(\mathbf{r}, \mathbf{r}') = \left(\frac{\delta\rho(\mathbf{r})}{\delta v(\mathbf{r}')}\right)_N = \sum_k^{\infty} q_j \beta_k(\mathbf{r}) \beta_k(\mathbf{r}'). \tag{2.129}$$

This descriptor provides insight into a variety of chemically meaningful phenomena, including electron delocalisation, inductive and mesomeric effects, as well as aromaticity, thereby offering a more fundamental basis for understanding chemical reactivity [116].

The work of Langenaeker and Liu [114] was instrumental in this context, as they analysed the response of the electron density to perturbations in the external potential for atomic systems. Their study highlighted the potential of the linear response function to reveal the intrinsic reactivity patterns of atoms and molecules, going beyond orbital-based or purely local descriptors.

In the same way that global hardness provides insight into which species is more likely to undergo a chemical reaction, local hardness [103] indicates the most reactive site within a molecule. It is defined as the functional derivative of the chemical potential with respect to the electron density:

$$\eta(\mathbf{r}) = \left(\frac{\delta\mu}{\delta\rho(\mathbf{r})}\right)_{v(\mathbf{r})}. \tag{2.130}$$

Several definitions for local hardness have been proposed. Notably, the expressions introduced by Meneses [117] ($\eta_{\mathrm{M}}(\mathbf{r})$) and by Gál [118] ($\eta_{\mathrm{G}}(\mathbf{r})$) are among the most widely used:

$$\eta_{\mathrm{M}}(\mathbf{r}) = \left(\frac{f^{+}(\mathbf{r}) - f^{-}(\mathbf{r})}{2}\right) + \mu f^{(2)}(\mathbf{r}) \tag{2.131}$$

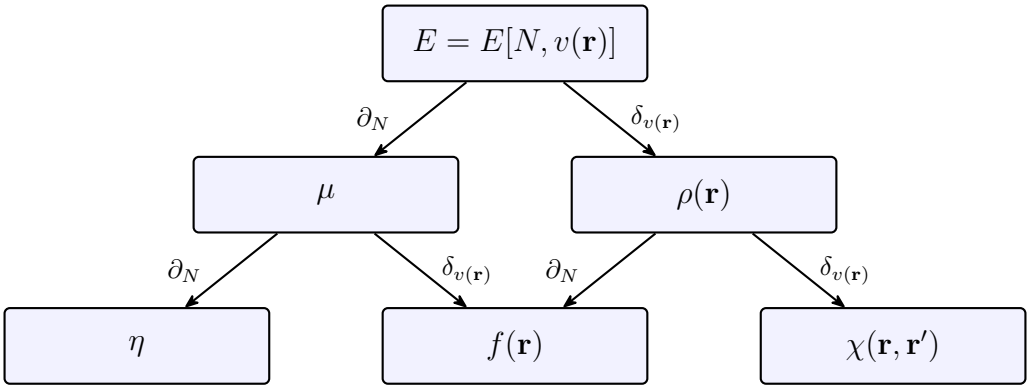
$$\eta_{\mathrm{G}}(\mathbf{r}) = I f^{-}(\mathbf{r}) + A f^{+}(\mathbf{r}) \tag{2.132}$$

Similarly, a local softness can be defined, though it is not merely the inverse of local hardness. It is more appropriately expressed as the product of the global softness and the Fukui functions:

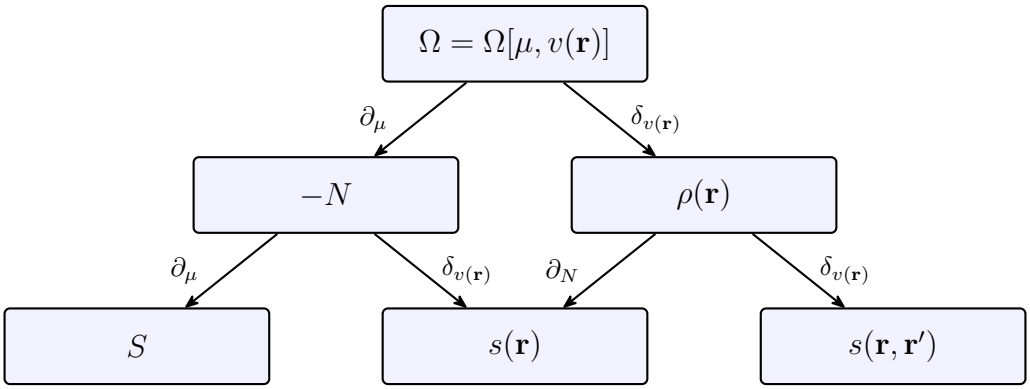
$$s^{\pm}(\mathbf{r}) = S f^{\pm}(\mathbf{r}). \tag{2.133}$$

By integrating the local softness over the entire space, one recovers the global softness, since the Fukui functions integrate to unity. At the scale of a single molecule, local softness conveys information similar to that provided by the Fukui functions. However, its principal advantage lies in the possibility of comparing reactive sites across different molecules, something the Fukui function alone cannot achieve.

It is important to remark that the previous discussion is based on the canonical ensemble, however, in a context of a finite temperature, the functional Ω, the grand potetial, is defined $\Omega = E - N\mu$ or $\Omega = E - \mu(N - N_0)$, where N_0 is the number of electrons in the reference state. At given temperature T, the following hierarchy of response functions was done [119]. In the Figure 2.8 we can see the energy derivatives and the comparison between the canonical and grand canonical ensembles.



(a) Energy derivatives and response functions in the Canonical Ensamble.



(b) Energy derivatives and response functions in the Grand Canonical Ensemble.

Figure 2.8. Comparison of energy derivatives n the Canonical and Grand Canonical ensembles. The arrows indicate with respect variable the derivative is taken.

2.9 Machine Learning

Moving beyond traditional theoretical physical chemistry, this section explores the rapidly evolving domain of Artificial Intelligence (AI), with a specific emphasis on Machine Learning (ML). Although frequently perceived as a recent innovation, AI has undergone steady and significant development over several decades [120]. Its applications now range from personalised recommendations and autonomous driving to enabling sophisticated language models capable of generating human-like text [121, 122]. A notable recent advancement is the introduction of attention mechanisms, exemplified by the Transformer Architecture, which has dramatically reshaped fields such as natural language processing and computer vision [123].

Machine Learning, a subfield of AI, focuses on algorithms capable of learning patterns directly from data without explicit programming. This paradigm broadly divides into: *i*) supervised learning —where models learn from labelled examples— and *ii*) unsupervised learning —which identifies inherent structures within unlabelled data—. The algorithms go from decision trees, and support vector machines (SVM) to artificial neural networks, each suited to distinct types of learning problems and datasets [124, 125] and they offer distinct trade-offs in interpretability, computational complexity, and predictive accuracy [126, 127].

In supervised learning, the objective is to predict outcomes based on previously observed input-output pairs. Typical supervised learning tasks encompass classification —assigning data points to discrete categories— and regression-predicting continuous numerical values. Conversely, unsupervised learning aims to uncover latent patterns or groupings within datasets lacking predefined labels. Methods such as k-means clustering [128] and hierarchical clustering [129] illustrate this approach, seeking natural groupings based purely on data similarity.

2.9.1 Preprocessing and Core Algorithms

Data preprocessing significantly influences the success of ML workflows. Typically, raw data require cleaning, normalisation, and transformation steps to handle inconsistencies, missing values, and outliers effectively. Feature selection, identifying the most informative attributes, not only enhances model performance but also simplifies model complexity, facilitating interpretability and reducing computational overhead [130]. Effective preprocessing thus ensures that subsequent learning algorithms receive optimal input, maximising predictive reliability and efficiency.

Neural networks, initially inspired by biological neurons [131], have evolved significantly into complex, multilayered architectures known as deep neural networks [121]. Deep learning techniques have notably driven breakthroughs in diverse domains, including image recognition, natural language processing, and predictive modelling across scientific disciplines [132, 133]. Their capacity to model intricate, nonlinear relationships within large-scale data makes neural networks particularly valuable in chemistry, facilitating sophisticated predictive models and augmenting traditional computational methodologies.

Decision trees provide intuitive and interpretable models by segmenting data into homogeneous subsets based on feature values [134]. Support Vector Machines, introduced by Cortes and Vapnik [135], offer powerful classification capabilities by determining optimal hyperplanes that maximise class separation margins. SVMs effectively handle linear and nonlinear problems through kernel functions, transforming inputs implicitly into higher-dimensional feature spaces [136]. This flexibility allows SVMs to robustly classify complex data distributions that may not be linearly separable in their original dimensionality.

2.9.2 Classifying Data with Machine Learning

Clustering is a common task in supervised and unsupervised ML, aiming to group data into subsets (clusters) such that data points within a cluster share similar characteristics. One of the most widely used clustering algorithms is k-means [128], which partitions a dataset into k clusters by iteratively assigning each point to the nearest cluster centroid and recomputing the centroids to minimise intra-cluster variance.

An alternative approach that overcomes some of these limitations is the Support Vector Clustering (SVC), derived from SVM, SVC maps the data into a high-dimensional feature space using a kernel function, —typically the radial basis function (RBF) kernel—, and constructs a minimal enclosing hypersphere [137]. The pre-image of this hypersphere in input space forms cluster boundaries, allowing the method to capture non-convex structures that are inaccessible to centroid-based techniques. The `scikit-learn` implementation of SVC provides practical access to this method, with straightforward interfaces for adjusting the kernel and its parameters, such as the parameter C [138].

Formaly, let $\phi : \mathbb{R}^n \to \mathcal{F}$ be a nonlinear mapping from the input space to a high-dimensional feature space $\mathcal{F}$. In this transformed space, SVC constructs the smallest enclosing sphere by solving the following optimisation problem:

$$\min_{\mathbf{a},R,\xi_i} \quad R^2 + C\sum_{i=1}^{N}\xi_i \quad \text{subject to} \quad \|\phi(\mathbf{x}_i) - \mathbf{a}\|^2 \leq R^2 + \xi_i, \quad \xi_i \geq 0. \tag{2.134}$$

where, $\mathbf{a}$ is the centre of the sphere, R its radius, and $C > 0$ is a regularisation parameter that controls the trade-off between the tightness of the boundary and the tolerance to outliers.

The use of the kernel trick enables this problem to be solved without explicit knowledge of ϕ, relying only on kernel evaluations $K(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle$.

The choice of kernel function $K(\mathbf{x}, \mathbf{x}')$ is central to the flexibility and performance of SVC. The kernel implicitly defines the geometry of the feature space and, consequently, the shape of the cluster boundaries. Some commonly used kernels include:

- **Linear kernel**: $K(\mathbf{x}, \mathbf{x}') = \mathbf{x}^\top \mathbf{x}'$ Equivalent to no transformation; effective when data are linearly separable.
- **Polynomial kernel**: $K(\mathbf{x}, \mathbf{x}') = (\gamma \mathbf{x}^\top \mathbf{x}' + r)^d$ Introduces interactions up to degree d; parameters $\gamma > 0$, $r \in \mathbb{R}$, and $d \in \mathbb{N}$ control flexibility and curvature.
- **Radial Basis Function (RBF) kernel**: $K(\mathbf{x}, \mathbf{x}') = \exp(-\gamma \|\mathbf{x} - \mathbf{x}'\|^2)$ Maps input to an infinite-dimensional feature space. Suitable for nonlinearly separable data; parameter $\gamma > 0$ controls the radius of influence of support vectors.
- **Sigmoid kernel**: $K(\mathbf{x}, \mathbf{x}') = \tanh\left(\gamma \mathbf{x}^\top \mathbf{x}' + r\right)$ Related to neural networks; behaves like a two-layer perceptron. Less commonly used due to non-positive definiteness for some parameters.

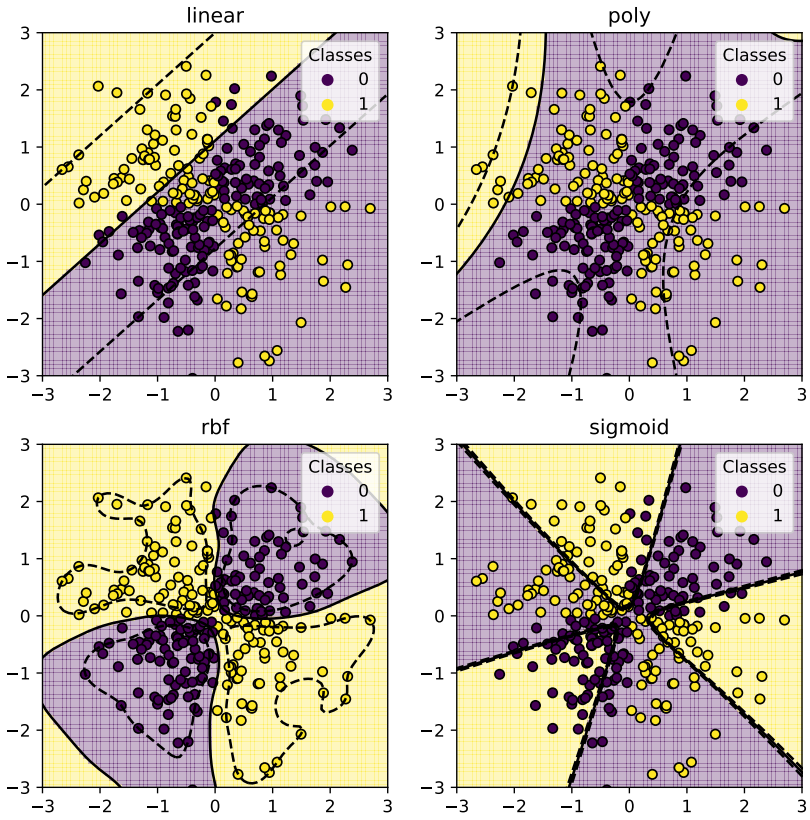


Figure 2.9. Visualisation of different kernel functions in SVC. The plot shows the decision boundaries formed by each kernel on a synthetic dataset with two classes. Data taken from the `scikit-learn` documentation [139].

The RBF kernel is the default and most widely used option in unsupervised SVC applications, as it allows capturing highly nonlinear and non-convex cluster structures with smooth boundaries. In the Figure 2.9, we illustrate the effect of different kernels, note that the plots do not evaluate the kernel's accuracy, they only provide a visual understanding.

2.9.3 Predicting Numerical Values with Machine Learning

In supervised machine learning, regression models aim to predict continuous numerical quantities from structured input features. This capability is particularly useful in theoretical and computational chemistry, where properties such as reactivity indices, energies, or solvation parameters can be modelled from molecular descriptors without requiring the full cost of ab initio calculations. A variety of regression algorithms exist, each making different assumptions about the data and offering different trade-offs between accuracy, interpretability, and computational efficiency.

Ensemble methods, such as Random Forest Regression (RFR), are among the most widely used approaches. RFR builds an ensemble of decision trees [140], each trained on a bootstrap sample of the data and employing random subsets of features to reduce variance and prevent overfitting. The final prediction is obtained by averaging the outputs of the individual trees. Random Forests offer high accuracy with minimal tuning and handle nonlinear interactions and mixed-type features effectively, which makes them well suited to heterogeneous chemical datasets.

Artificial neural networks (ANNs), and in particular deep architectures, can approximate complex nonlinear functions and are capable of capturing subtle patterns in the input data [124]. A feedforward neural network comprises layers of nodes (neurons), each applying an affine transformation followed by a nonlinear activation function. When properly regularised and trained on sufficient data, ANNs can generalise well to unseen examples. However, they often require careful hyperparameter tuning and are sensitive to overfitting, particularly in low-data regimes, and commonly require large datasets to achieve optimal performance.

Alternative probabilistic regression models include Bayesian Ridge Regression (BRR) and Gaussian Process Regression (GPR). BRR extends classical linear regression by placing Gaussian priors on the coefficients, yielding a Bayesian posterior that reflects uncertainty over predictions [141]. In contrast, GPR is a nonparametric method that models the output as a realisation of a Gaussian process [142], fully defined by a mean function and covariance kernel. GPR provides both point estimates and predictive uncertainty, making it appealing in scientific applications where uncertainty quantification is essential.

2.10 Computational Framework

In this section, we describe the computational protocols and methodologies adopted in this study. All bespoke software was developed for Unix-like environments and has been validated under both GNU-Linux and macOS.

2.10.1 Amsterdam Modeling Suite

The majority of quantum chemical computations were performed using the Amsterdam Modeling Suite (AMS), a commercial software package developed by the Software for Chemistry and Materials (SCM) company [28]. Originally developed in the 70's within the theoretical chemistry department of the Vrije Universiteit in Amsterdam, AMS has evolved from its early FORTRAN foundations to include a broad set of modern features. These include support for Object-Oriented Programming (OOP), modular design, and an expanding set of functionalities via successive versions.

Some engines in AMS such as ADF, BAND, and DFTB remain written in modern FORTRAN. While the graphical user interface (GUI) is based on TCL, several high-performance modules have been implemented in C++. A significant extension is the PYTHON based scripting interface PLAMS (Python Library for Automating Molecular Simulations), which provides high-level access to workflow automation, job management, and data analysis across AMS engines.

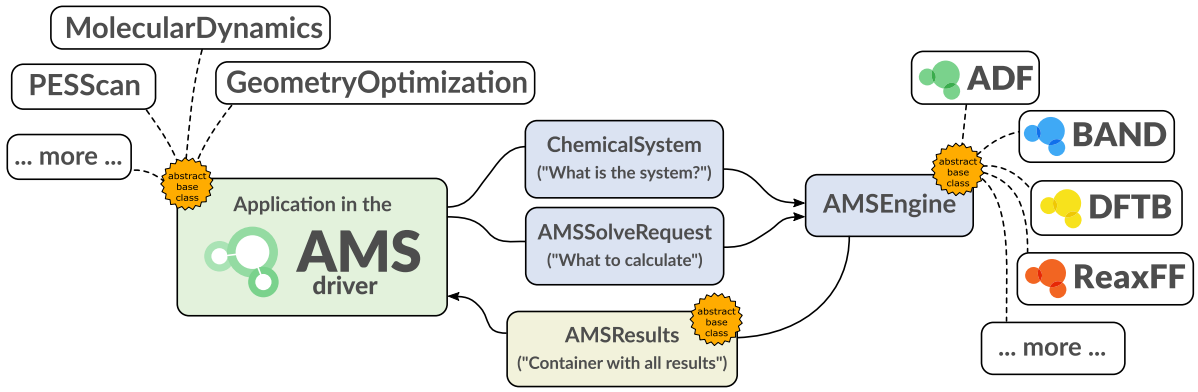


Figure 2.10. The AMS driver and its interaction with available engines. Diagram taken from the developer meeting slides of SCM, 2025.

In this work, we primarily employed ADF, one of the most mature and feature-rich engines within AMS. The AMS driver integrate also the QTAIM partition in BAND and DFTB. As illustrated in Figure 2.10, the AMS driver manages the coordination of jobs, delegating tasks to the appropriate engine according to the user-defined settings and task requirements.

Another important tool used in this work is the PLAMS, which provides a PYTHON library not only for the AMS driver, but also with many other tools and for sure given to the user the possibility to automatise a personal workflow. The current development of `amspython` runs on PYTHON 3.8 and it can interact with libraries such as NUMPY, SCIPY or PANDAS, (as shown in Figure 2.11) allowing the user to perform advanced data analysis, PLAMS also provides functions to read binary files created by AMS engines for direct access to molecular properties.

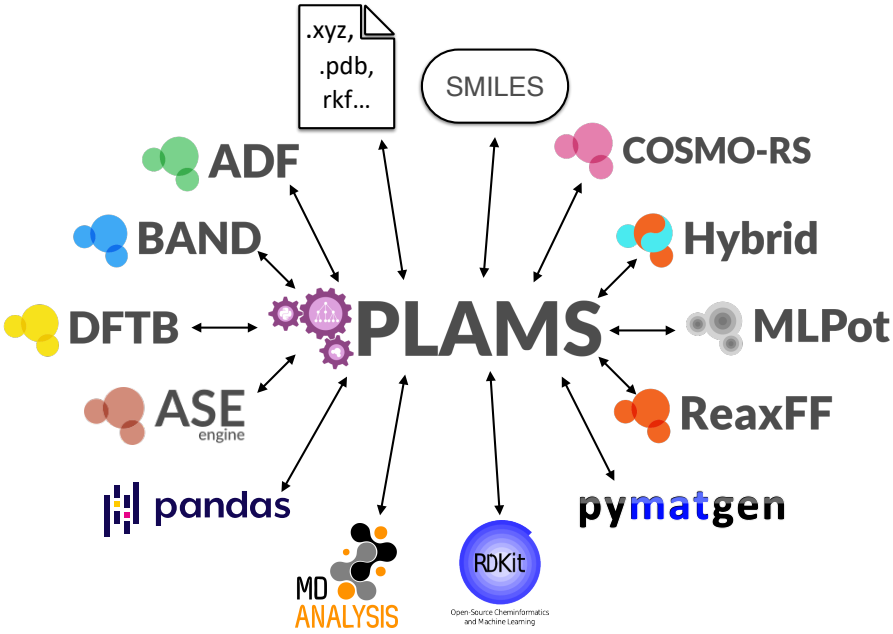


Figure 2.11. Interaction between PLAMS and different engines from AMS driver and other libraries. Diagram taken from the developer meeting slides of SCM, 2025.

While PLAMS is distributed as open-source software via /SCM-NV/PLAMS, the source code of AMS is not publicly available. Access is restricted to licensed users, and availability depends on a number of parameters including: *i*) selected modules, *ii*) total CPU core limit, *iii*) duration of licence, and *iv*) academic or commercial status of the institution. Source code access is restricted to developers.

Run File Structure

The AMS driver can be controlled via a flexible input file written in a shell-like >_ syntax. This run file acts not only as a traditional input, but as a lightweight script that manages the execution of the chosen engine. While calculations can also be launched through the GUI, scripting provides the advantage that it enables systematic and automated workflows, such as running multiple calculations with varying molecular geometries, basis sets, or external fields without user intervention.

```
#!/bin/sh
Task GeometryOptimization

Properties
    NormalModes true
End

System
    Atoms
        H 0.0 0.0 0.0
        H 0.8 0.0 0.0
    End
End

Engine DFTB
    Model GFN1-xTB
EndEngine
```

Figure 2.12. Example of an AMS run file. The blue highlighted line is the Engine Input, while the rest of the run file is interpreted by the AMS driver, independently of the chosen engine.

Figure 2.12 shows the structure of a typical AMS run file. The script defines the task (*e. g.* single point calculation or geometry optimisation), the properties to compute (such as normal modes), and the system under study, which includes molecular coordinates (by default in Å) and, optionally, a charge or external electric field, among other settings. The `Engine` block provides the engine-specific input and is closed with the keyword `EndEngine`, rather than `End`.

Since the interpreter of the run file is predefined as `sh`, it can be made executable and launched directly from the command line, like any standard shell script. The interpreter can also be changed to match the user's preferred shell (*e. g.* `bash`, `zsh`, `fish`, or any other), as long as the environment is properly set up. This setup is managed by the `adfbashrc.sh` script, which should be sourced in the appropriate shell initialisation file (such as `.bashrc`, `.zshrc`, etc.), ensuring that binaries, licence information, and scratch directories are properly configured.

Additional settings —such as the number of processes— can be specified either via command-line flags or environment variables. For example, the number of processes can be set in either of the following ways:

```
# 1. By a flag
"$AMSBIN/ams" -n 1 << eor
    # ams input
eor
# 2. By an environment variable
export NSCM=1
"$AMSBIN/ams" << eor
    # ams input
eor
```

A general procedure for executing a calculation with the AMS driver —independent of the engine used— is illustrated below. The user writes a run file, makes it executable, and launches it from the command line. Upon execution, a new directory named `"${AMS_JOBNAME}.results"` is automatically created to store auxiliary files:

```
vcastor@ada - bash
$ # Create a run file
$ vi run_file.run              # or any preferred text editor
$ chmod u+x run_file.run       # Make it executable
$ ./run_file.run > output.out # Execute and redirect output
$ # A new directory named ${AMS_JOBNAME}.results will be created
```

The resulting `*.results` directory contains a log file and two `*.rkf` files —one generated by the AMS driver and the other by the selected engine—. In addition, the directory may include several auxiliary files such as `TAPE10`, `TAPE13`, `TAPE15`, and `TAPE41`, as well as a separate `TAPE21` file for each element present in the system, and sometimes the binary `t12.rel` file, which is not a KF file, but an auxiliary file specific to relativistic calculations. Depending on the nature of the task binary scratch files may be created. If no job name is provided, the default name `ams.results` is used. The `*.rkf` files are binary containers developed by AMS to store structured information from the calculation. Human-readable output is printed directly to `stdout` and `stderr` during execution.

Figure 2.13 illustrates the file structure within the `*.results` directory for a single point calculation. In the case of a geometry optimisation, the directory will additionally contain an `.rkf` file for each optimisation step, named `GOStep`n`.rkf`, where n corresponds to the step number.

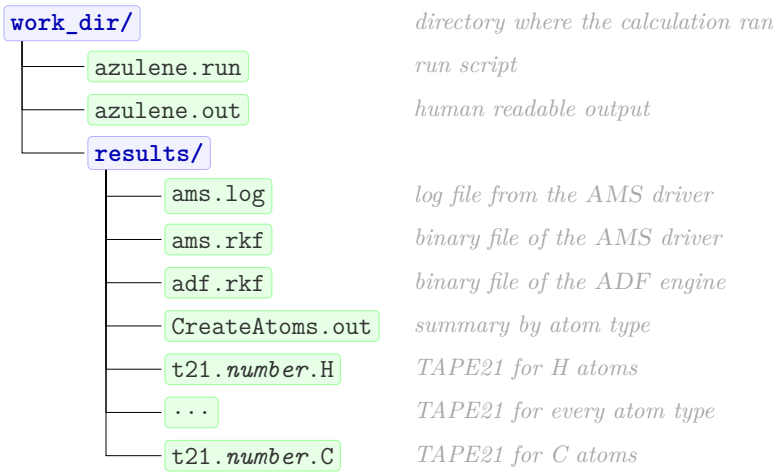


Figure 2.13. work directory structure for a Single Point Calculation with the ADF engine.

The following two examples demonstrate how a run script can be used to automate calculations by looping over different molecular geometries or computational settings. The first example illustrates how to set up a geometry scan using a scripted approach.

```
#!/bin/bash

# Edge lengths for each system
edge_lengths=(2.700 2.200 2.150 2.120 2.090 2.070 1.960 1.800)

# Function to calculate coordinates
calculate_x() {
  local length=$1
  echo "scale=6; $length/2" | bc -l
}
calculate_y() {
  local length=$1
  echo "scale=6; $length*sqrt(3)/2" | bc -l
}

# Loop to generate AMS input files and run calculations
for i in "${!edge_lengths[@]}"; do
  length="${edge_lengths[$i]}"
  y_coord=$(calculate_y "$length")
  x_coord=$(calculate_x "$length")

  # Variable with the job name; and execute the AMS driver [binary]
  AMS_JOBNAME="be_${i}" $AMSBIN/ams << eor

```

```
Task SinglePoint
System
  Atoms
    Be 0.000000 0.000000 0.000000
    Be $length 0.000000 0.000000
    Be $x_coord $y_coord 0.000000
  End
  Charge -2
End

Engine ADF
  # Engine specific input
EndEngine

eor

done
```

The same principle can be extended to alter not only molecular coordinates, but also keywords such as the basis set, external fields, numerical settings, or solvation models. This flexibility enables fully reproducible and automated protocols.

```
#!/bin/bash
# This ADF input has a loop to compute a system with an external electric field
# in different directions
# x, y, z, -x, -y and -z [V/Å]  (0.0100 a. u.)

declare -A EFvalues
nrows=7
ncolumns=3
name=( x y z mx my mz noE )

for ((i=0; i<nrows; i++)) do
  for ((j=0; j<ncolumns; j++)) do
    EFvalues[$i,$j]=0.00
    k=$(expr $i - 3)
    if [ $i -eq $j ]; then
      EFvalues[$i,$j]=0.5144
    elif [ $j -eq $k ]; then
      EFvalues[$i,$j]=-0.5144
    fi
  done
done

```

```
for ((j=0;j<nrows;j++)) do

AMS_JOBNAME=system_${name[$j]} $AMSBIN/ams << eor

Task SinglePoint
System
  Atoms
    O     -0.9952892833      1.0793836907     -0.8870331216
    C      0.3806786652      0.8563670307     -0.5242410141
    C      1.1343132702      2.1207737537     -1.0892343026
    # Molecular coordinates
  End
  ElectrostaticEmbedding
    ElectricField ${EFvalues[$j,0]} ${EFvalues[$j,1]} ${EFvalues[$j,2]}
  End
End

Engine ADF
  # Engine specific input
EndEngine

eor

done
```

It is important to note that when a calculation is launched from the GUI, the `TAPE10` file is automatically saved. In contrast, when using a run file, `TAPE10` must be explicitly requested within the Engine block; otherwise, it will not be retained after the calculation finishes. Although `TAPE10` primarily contains temporary data, it is required for certain types of analysis within the GUI—for instance, the visualisation of atomic basins in the QTAIM partition.

Additional options can be specified directly in the input file, such as disabling molecular symmetry or adjusting numerical quality thresholds. For further information, users are encouraged to consult the official AMS documentation and tutorials, available at: scm.com/doc/Documentation and scm.com/doc/Tutorials.

2.10.2 Numerical Stability

As with any numerical approach, it is essential to remain aware of finite-precision effects. A well-known example is the inexactness of floating-point arithmetic (*e. g.* $0.1+0.2 \neq 0.3$), which arises from the limitations of binary representation as defined by the IEEE 754 standard. While such discrepancies are typically small —on the order of 10^{-17}— and often negligible in practical workflows, they are not confined to artificial examples. Even robust, low-level numerical libraries such as LAPACK [18] can exhibit numerical instability in certain contexts.

Table 2.2. Floating-point representation of 0.1, 0.2, and 0.3 in IEEE 754 single precision format.

Decimal	Sign	Exponent (bias)	Mantissa (binary)
0.1	0	01111011 (123)	10011001100110011001101
0.2	0	01111100 (124)	10011001100110011001101
0.1 + 0.2	0	01111101 (125)	00110011001100110011010
0.3	0	01111101 (125)	00110011001100110011001

Dhillon [143] has shown that inverse iteration algorithms, as implemented in LAPACK, can fail to compute eigenvectors accurately when eigenvalues are nearly degenerate. This is primarily due to breakdowns in reorthogonalisation and heightened sensitivity to small perturbations. Demmel et al. [144] further explored the numerical stability of eigenvalue solvers, observing that Bisection/Inverse Iteration (BI) schemes, including SUBROUTINES such as `DGELSS`, may lose orthogonality when eigenvalues are clustered.

To mitigate such issues, it is good practice to incorporate validation procedures and redundancy checks, especially for tasks involving eigenvalue problems with closely spaced spectra.

It should be emphasised, however, that such numerical instabilities are edge cases and are rarely encountered in routine applications. For the majority of computational tasks carried out in this work, the numerical precision provided by the FORTRAN-based implementations and the LAPACK library proved entirely sufficient.

2.10.3 Code Management

All software, scripts, and data files used in this project were managed using version control systems, primarily git and SVN. Several contributions developed during this work have been integrated into the main (trunk) development branch of AMS, while others remain in feature branches awaiting final integration. In 2024, SCM transitioned from SVN to Gitea, a web-based platform for managing git repositories.

Supplementary scripts and workflows are publicly available at /vcastor/PhD-Scripts. The complete manuscript, including code, figures, and additional material, is archived at: /vcastor/PhDManuscript. Details on compilation, execution, and reproduction of the manuscript are provided in Appendix B.1.

For further technical queries,
please contact the author directly.

Chapter

3 Implementation of QTAIM in the AMS

In this chapter, we present the implementation of the Quantum Theory of Atoms in Molecules (QTAIM) partition within the AMS. We begin by outlining the legacy implementation that served as the foundation for this work, highlighting its capabilities and limitations.

The chapter concludes with a series of benchmark tests and validation results. These serve both to illustrate the accuracy and robustness of the implementation and to assess its computational performance across different molecular systems.

3.1 Original Implementation

The QTAIM partition was originally implemented in AMS by Rodríguez [145], enabling the computation of atomic properties without explicitly solving the zero-flux condition (Figure 3.1). This approach is a key reason why the QTAIM partition in AMS is remarkably fast.

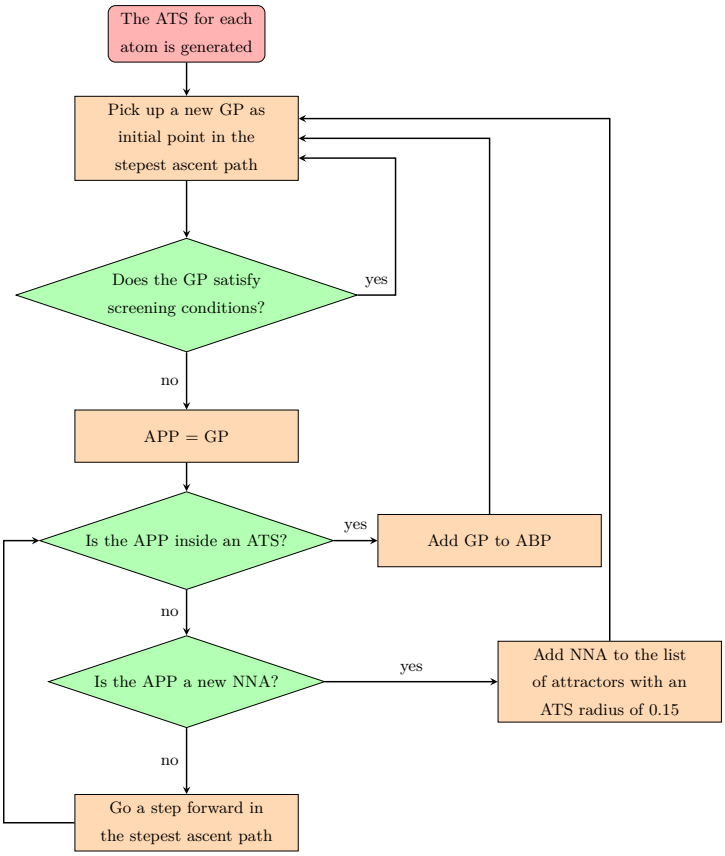


Figure 3.1. Flowchart of Rodríguez' original QTAIM partition algorithm in AMS. The next acronyms are used in the flowchart: ATS, atomic trust sphere; GP, grid point; APP, actual point in the path; NNA, non-nuclear attractor; and ABP, atomic basin property.

The analysis is performed by partitioning real space into a grid, either user-defined or automatically set by AMS with a default spacing of 0.5 bohr. A Newton-Raphson method is applied to locate the CPs of $\rho(\mathbf{r})$. Beyond this, the grid serves also to compute the atomic properties, with each point contributing to the numerical integration within its assigned atomic basin. Points close to the nuclei are directly assigned to the corresponding basin through the atomic trust sphere (ATS) with a radius of 0.23 bohr [145], accelerating the process.

Conversely, points far from the molecule typically have electron densities too low to be clearly assigned to any atom. Since their contribution to properties is negligible, they are excluded from the analysis. This screening condition is applied when the density and its gradient satisfy the conditions: *i)* $w_i\rho(\mathbf{r}_i) < 10^{-8}$ and *ii)* $|\boldsymbol{\nabla}\rho(\mathbf{r}_i)|^2 < 10^{-8}$, where w_i is the integration weight associated with grid point $\mathbf{r}_i$.

Although the implementation successfully identified CPs and atomic basins, it did not provide a complete topological analysis. The limitations of the original code are discussed in the following subsections, alongside the description of the new implementation. A summary of the updates introduced in the current version of the AMS code is provided in the next Table.

Table 3.1. Summary of updates introduced in the current version of the AMS code.

Comparison of Features in AMS			
Feature	**First Implementation**	**Current Implementation**	**Comments**
CPs	All detected	Code refactored and memory optimisation	Same results
NNAs	Detected	Detected and analysed	Bonds to NNA allowed; atomic properties are computed
Gradient paths	Implemented by Runge-Kutta 2^{nd} Order	Adaptive use of Runge-Kutta 2^{nd} or 4^{th} Order with dynamic step size	Improves numerical stability and precision
RCPs & CCPs	CPs detected	Atoms involved in rings/cages identified	Implemented gradient paths and graph theory
Atomic properties by the form $\int_{\Omega} \rho_A(\mathbf{r}) \mathrm{d}\tau$	Implemented	Refactored	Cleaner code structure; easier to read and maintain
Dipole moment	Electronic contribution only	Nuclear contribution added; total dipole compared with molecular value	Full dipole moment computed
Polarisability	Not implemented	Available via PLAMS; new AMS driver task planned	AMS integration ongoing
Excited states	Not implemented	Implemented	Pending final merge to trunk
Atomic basins in GUI	Implemented	New GUI implementation	Improved visualisation

3.2 Topological Analysis

The topological analysis, involving the identification of CPs and the evaluation of their properties, constitutes the first step in any QTAIM analysis. This procedure is carried out whenever the QTAIM block is included in the run script, and is essential for subsequent analyses. It is therefore mandatory for all `AnalysisLevel` settings, which may enclose atomic as well as non-local properties, with each stage computed only after the previous level, from topological to atomic to non-local properties.

```
QTAIM
  # [topological analysis / atomic properties / non-local properties]
  AnalysisLevel [Normal/Extended/Full]
End
```

3.2.1 Non-Nuclear Attractors (NNAs)

Nuclear attractors are added to the array of CPs before initiating the search for other CPs. Their properties are evaluated after applying a small displacement to the nuclear coordinates (10^{-8} bohr) to prevent numerical instabilities in the evaluation of the electron density at the nuclear positions.

A Newton-Raphson algorithm is then used to locate the remaining CPs of $\rho(\mathbf{r})$, which are classified according to their eigenvalue signatures. In earlier versions of the code, NNAs could be identified from this signature, but no dedicated treatment was implemented; such points were simply ignored in subsequent property calculations.

In the current implementation, NNAs are explicitly recognised, labelled, and fully incorporated into the analysis. All available properties are computed for them, and they are considered on the same footing as nuclear attractors, including their possible participation in bonds, rings, and cages.

Whenever one or more NNAs are detected, the global variable storing the `xyz` coordinates of the attractors is extended to include them, and the global counter for the number of attractors is updated accordingly. These variables are used throughout AMS code for the QTAIM partition, whereas in other parts of AMS the atom count continues to exclude NNAs.

Algorithm 1: NNA treatment.

```
1  ncp ← natoms ;                                  // Initialise number of critical points
2  nattractors ← natoms ;                          // Initialise number of attractors
   /* cpResults is an array to store the Results of every CP */
3  for iAtom ← 1 to natoms do
4  |  shiftedCoord = xyznuc(iAtom) + smallShift ;                          // 10^-8
5  |  CalcDensity(shiftedCoord, rho, gradRho, secDerRho);
6  |  cpResult(iAtom, coords) ← xyznuc(iAtom);
7  |  cpResult(iAtom, density) ← rho;
8  |  cpResult(iAtom, gradeent) ← gradRho;
9  |  cpResult(iAtom, hessian) ← secDerRho;
10 |  cpResult(iAtom, sisnature) ← 1.0 ;                  // Nuclear critical point
11 FindAllCriticalPoints();
   /* CharacterizeCriticalPoints */
   /* ncp is the number of critical points found */
12 for ipoint ← 1 to ncp do
13 |  hessian ← symmetricMatrix(cpResult(ipoint, idxHessian:idxHessian+5));
   |  /* LAPACK subroutine for eigenvalues of symmetric matrices */
14 |  eigenvalues ← dsyev(hessian);
15 |  signature ← DetermineCPSignature(eigenvalues);
16 |  if signature == 2 then
17 |  |  ncages ← ncages + 1 ;                  // Cage CP; All eigenvalues negative
18 |  if signature == 3 then
19 |  |  nbonds ← nbonds + 1 ;                  // Bond CP; Two eigenvalues negative
20 |  if signature == 4 then
21 |  |  nrings ← nrings + 1 ;                   // Ring CP; One eigenvalue negative
22 |  if signature == 1 then
23 |  |  nattractors ← nattractors + 1 ; // Attractor CP; All egienvalues positive
24 if nattractors > natoms then
25 |  UpdateGlobalVariablesForQTAIM();
```

3.2.2 Poincaré-Hopf relation

The original implementation already included an evaluation of the Poincarŧ-Hopf relation. In the current version of the code, its role has been extended beyond a simple check, serving as a robust criterion for verifying whether all CPs in the system have been correctly identified. The Poincaré-Hopf relation states that the sum of the indices of all CPs must be equal to either 0 or 1, depending on the topological nature of the system (as discussed in Section 2.7.2). This property arises from the topology of vector fields and provides an elegant and efficient global consistency check.

For closed systems, the expected sum is 1, whereas for periodic systems it is 0. Since AMS is capable of treating both classes of systems —ADF and DFTB for molecular calculations, and BAND for periodic ones— the implementation must account for both possibilities, as written in the Equation 3.1.

In practice, this criterion is used to determine whether the search for CPs should be repeated with a finer spatial grid. When the relation is not satisfied, the grid spacing can be reduced by factors of 2 or 3 to increase resolution, improving the likelihood of detecting missing CPs. Conversely, if the initial spacing is unnecessarily fine, it may be enlarged by a factor of 2 to decrease memory requirements and avoid potential overflow errors. This adaptive refinement strategy provides a balance between numerical accuracy and computational efficiency, ensuring reliable results across a wide range of systems.

$$\chi(M) = \begin{cases} (n_{\mathrm{NNACP}} + n_{\mathrm{NCP}}) - n_{\mathrm{BCP}} + n_{\mathrm{RCP}} - n_{\mathrm{CCP}} = 1 & \textit{molecular} \\ (n_{\mathrm{NNACP}} + n_{\mathrm{NCP}}) - n_{\mathrm{BCP}} + n_{\mathrm{RCP}} - n_{\mathrm{CCP}} = 0 & \textit{periodic} \end{cases} \tag{3.1}$$

Algorithm 2: AIMCriticalPoints Subroutine

```
   /* total energy subroutine calls AIMCriticalPoints */
   /* AIMCriticalPoints looks like this:  */
1  gridRefinements ← [1.0, 0.5, 0.66] ;                    // Grid refinement factors
2  gross ← 1.0 ;                                  // Inflation factor in case of overflow
3  poincaréHopfTarget ← [0 or 1] ;                        // Periodic or Molecular system
4  gridSpacing ← user-defined or default (0.5);
5  igrid ← 1;
6  belowOverflow ← true; everythingTried ← false; success ← false;
7  while belowOverflow and not success and igrid ≤ length(gridRefinements) do
       /* Update grid spacing using gridRefinements[igrid] and gross */
8      gridSpacing ← gridSpacing*gridRefinements[igrid]*gross;
9      if igrid > 1 then
10         Notify('Refining grid...')
11     if grid causes overflow then
12         Notify('Grid too small; trying gross grid');
13         gross ← 2.0;
14     FindCriticalPoints();
15     ClassifyCriticalPoints();
16     if periodic then
17         Generate equivalent CPs;                                          // BAND
18     success ← CheckPoincaréHopf();
19     if success then
20         break;
21     igrid ← igrid + 1;
```

3.2.3 Bonds, Rings and Cages

Once all CPs have been successfully identified, the next step is to determine the atoms involved in bonds, rings, and cages. In earlier versions of the code, only atoms in bonds could be identified, with no support for detecting ring or cage structures.

The identification of atoms involved in bonds, rings and cages is performed by following the gradient path with the corresponding CP as starting point. For bonds, the number of atoms and the directions to follow the gradient path are already known, two atoms and the directios are given the third eigenvector in both the positive and negative directions. In contrast, for rings and cages, neither the number of atoms nor the optimal directions for following the gradient path are known in advance.

To address this, we construct a spherical shell (icosphere) of points around the RCP or CCP and follow the gradient path from each point. The choice of the shell radius is critical: if it is too small, many points follow nearly identical paths and fail to reach all atoms; if too large, the integration path may overshoot the ring or cage. Further details on the generation of the icosphere are provided in Appendix A.1.

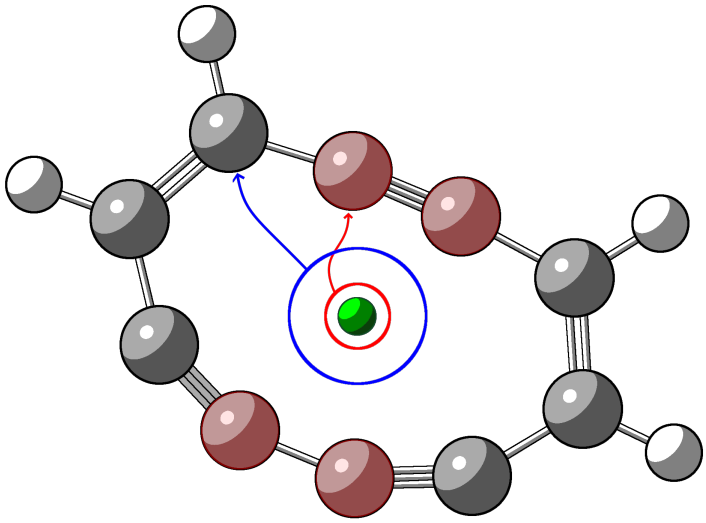


Figure 3.2. The RCP (green sphere) is enclosed by two icospheres: a red one (0.2 bohr) and a blue one (0.6 bohr). Gradient paths originating from the red icosphere reach only the atoms highlighted in red, whereas those from the blue icosphere connect to all atoms in the ring. For each case, one representative gradient path is shown.

The radius must be sufficiently small to capture compact structures such as cyclopropene. A value of 0.2 bohr has been found to work reliably in such cases. For more distorted rings, or those closed by non-covalent interactions, larger radii are required. We have therefore implemented three radii (0.2, 0.6, and 2.8 bohr) to accommodate a range of geometries. As illustrated in Figure 3.2, a radius of 0.2 bohr may fail to detect all atoms in very flat density regions, where gradient paths initiated from similar directions converge. A larger radius allows paths to explore the full structure more effectively.

The accuracy of the differential equation solver plays a decisive role in correctly following the gradient paths. By default, a second-order Runge-Kutta method is used. If the initial search fails to identify valid paths, the solver is automatically upgraded to a fourth-order Runge-Kutta method.

Finally, for each bond, ring, and cage CP, the set of atoms assigned to the corresponding structure is verified to contain only unique entries. In systems characterised by extremely flat electron densities, it is possible for multiple gradient paths to converge on the same attractor, *e. g.* a bond cannot be formed by the same atom twice.

3.2.4 Graph theory

In the previous section, we described the attempts to follow gradient paths from RCPs and CCPs. Here, we explain how the program determines whether recomputation of these paths is necessary. This is achieved through a graph theory analysis, it is used to validate if the topological features have been correctly identified.

For rings, we require that the number of atoms equals the number of bonds, and that each atom forms exactly two bonds with the other ring atoms. These simple criteria allow us to confirm the topology. If either condition is not met, the gradient paths are recomputed using a smaller step size and a higher-order Runge-Kutta method.

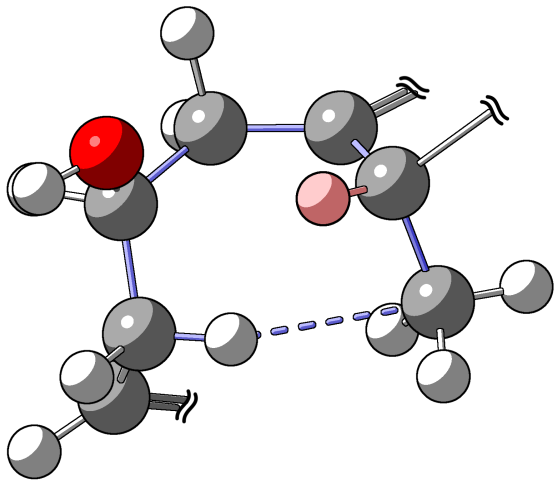


Figure 3.3. Extra H atom highlighted in pink detected in a ring. Parts of the molecule are omitted for clarity. The `xyz` coordinates are available in the Appendix A.4.

In one particular case the recomputation can be avoided. If exactly one extra atom is detected in a ring, it is likely an artefact. As shown in Figure 3.3, a hydrogen atom close to the ring plane may be erroneously assigned to the ring when following the gradient path. Computing the system's topology with AIMAll, we observe the misassignment of the H atom to the ring as an artefact of following the gradient path. Such atoms can be excluded from the final ring definition to prevent unnecessary recomputation.

If, after all tries, the topology remains inconsistent, a warning is issued in the output file. In the case of cages, even exotic topologies can be valid topologies. We therefore avoid rigid definitions, choosing instead to alert the user to atypical situations.

For rings, in case that the atoms does not form a valid cycle, we issue the warning: *i*) not all atoms in a ring have been successfully identified. While for cages, we issue the following warnings for potentially non-standard topologies:

i) a bond associated with a cage does not belong to at least two rings,

ii) not all atoms in a cage have at least three bonds,

iii) a cage is not a convex polyhedron,

iv) a cage and at least one ring are composed of the same atoms,

v) all atoms in a cage are also in a ring that contains additional atoms.

Case v is particularly common in highly deformed rings. In such situations, two RCPs may appear with a CCP between them, producing a cage that is essentially a distorted ring. Depending on the basis set or functional, these two RCPs may even collapse into a single RCP. This does not represent a failure of the topological analysis, but a warning is nonetheless issued to indicate that the topology is unusual.

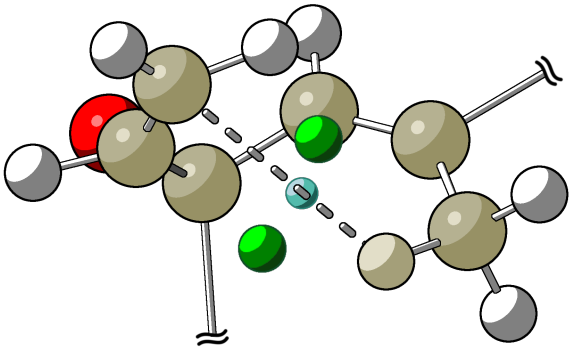


Figure 3.4. Cage formed by two rings. The two RCPs (green spheres) share the same atoms (highlighted in khaki), creating the CCP (blue sphere). Parts of the molecule are omitted for clarity. The atomic coordinates are available in the Appendix A.4.

Rings as cycles and cages as convex polyhedra

For rings, the analysis determines whether the set of bonds forms a closed cycle. This is assessed by examining the correspondence between vertices (atoms) and edges (bonds): in a valid cycle, the number of vertices must equal the number of edges. This condition provides a reliable and computationally efficient means of detecting topological inconsistencies, and is therefore used as a trigger for recomputing the topology when necessary. The procedure for this verification is summarised in Algorithm 3, which also illustrates how the algorithm detects the presence of extra atoms bonded to the ring, as exemplified in Figure 3.3.

Cages present a greater challenge, as their topology is inherently more complex and cannot be validated by a single criterion. As discussed previously, several warning checks have been implemented, one of which adopts a purely geometrical perspective: Euler's polyhedron formula. This relation links the number of faces (F), edges (E), and vertices (V) of a polyhedron through the expression $F - E + V = 2$. Compliance with this formula confirms that the cage exhibits at least a topologically consistent convex polyhedral structure. However, this condition alone does not guarantee physicochemical validity, and additional topological and geometrical criteria may be necessary to achieve a complete verification.

Algorithm 3: Algorithm to check the rings.

```
   /* Number of bonds between the atoms */
 1 nbonds = 0; found = false;
 2 for i ← 1 to ring size do
 3 |   atomA ← atomRing(i);
 4 |   for j ← i + 1 to ring size do
 5 |   |   atomB ← atomRing(j);
 6 |   |   if atomA and atomB are bonded then
 7 |   |   |   nbond + 1;
 8 if nbonds = ring size then
 9 |   info = true;
   /* Any extra atom bonded to the ring?  */
10 if info and first time called then
11 |   delatom(:) ← false ;                                  // Boolean array
12 |   for i ← 1 to ring size do
13 |   |   atomA ← atomRing(i);
14 |   |   nbond = 0;
   |   |   /* All against all */
15 |   |   for j ← 1 to ring size do
16 |   |   |   atomB ← atomRing(j);
17 |   |   |   if atomA and atomB are bonded then
18 |   |   |   |   nbond + 1;
   |   |   /* Just one bond means the atom is not in the cycle/ring */
19 |   |   if nbond < 2 then
20 |   |   |   delatom(i) = true;
   |   /* Delate the extra atom */
21 |   if any(delatom) then
22 |   |   j = 0;
23 |   |   for i ← 1 to ring size do
24 |   |   |   if not delatom(i) then
25 |   |   |   |   j = j + 1;
26 |   |   |   |   tmp(j) ← atomRing(i);
27 |   |   ring size ← j;
28 |   |   atomRing(:) ← tmp(:);
29 |   call me recursively;
```

3.2.5 Path-Following Strategy

The identification of atoms participating in bonds, rings, and cages is carried out in a sequential manner. First, the atoms forming bonds are determined; second, those involved in rings are identified; and finally, the atoms defining cages are established. This ordering is essential for the application of graph theory as a validation tool, since higher-order topological features rely on the accurate detection of lower-order ones.

Algorithm 4: Secuential Gradient Following for Bonds, Rings, and Cages

```
   /* Global variables for Bonds, Rings, and Cages */
 1 global.RunegeKuttaOrder = 2;
 2 stepSize = 0.1 ;                          // (borh) step size for gradient following
   /* Atoms in bonds */
 3 for cp ← 1 to bondCPs do
 4     firstPoints ← [cp.3rdEigenVector(), -cp.3rdEigenVector()];
 5     followGradient(firstPoints, RungeKuttaOrder, stepSize);
 6     if atoms in bond are the same or no atom(s) found then
 7         global.RunegeKuttaOrder = 4;
 8         stepSize = 0.7*stepSize;
 9         followGradient(firstPoints, RungeKuttaOrder, stepSize);
   /* Atoms in rings */
10 for cp ← 1 to ringCPs do
11     icosphereSize ← [0.2, 0.6, 2.8];
12     for icoSize in icosphereSize do
13         firstPoints ← createIcosphere(centred=cp, radius=icoSize);
14         followGradient(firstPoints, global.RungeKuttaOrder, stepSize=0.2);
15         graphTheoryCheck(atomsInRing);
16         if atomsInRing is a cycled path then
17             nextcp;
18         if extraAtom in ring then
19             removeExtraAtom(atomsInRing);
20             nextcp;
   /* Atoms in cages */
21 for cp ← 1 to cageCPs do
22     icosphereSize ← [0.2, 0.6, 2.8];
23     for icoSize in icosphereSize do
24         firstPoints ← createIcosphere(centred=cp, radius=icoSize);
25         followGradient(firstPoints, global.RungeKuttaOrder, stepSize=0.2);
26         graphTheoryCheck(atomsInCage) ;                    // Only Warning
```

3.3 Atomic Properties

Atomic properties are computed only when explicitly requested by the user by setting `AnalysisLevel` to `Extended` or `Full`. In these cases, the calculation of the topology is followed by the evaluation of the atomic properties:

```
QTAIM
  # [atomic properties / non-local properties]
  AnalysisLevel [Extended/Full]
End
```

3.3.1 Atomic Dipole Moment

The electronic contribution to the atomic dipole moment within QTAIM was already implemented in the AMS code. However, the nuclear contribution was not previously included. As shown in Equation 2.85, the nuclear contribution can be computed with the QTAIM topology and atomic charges.

In the rare event that the topology fails for a specific ring, the corresponding ring is excluded from the linear system used to compute the nuclear contribution. The dimensions of the linear system are given by the number of bonds (columns) and the number of atoms plus the number of rings (rows). Since the system is overdetermined, omitting a ring does not prevent a solution from being obtained.

In the even less common case where the linear system cannot be solved, a warning is issued in the output file and the nuclear contribution to the atomic dipole moment is set to zero (`0.d0`). The system is solved using the `DGELSS` SUBROUTINE from LAPACK, which handles both overdetermined and underdetermined systems. It computes the minimum norm solution to a real linear least squares problem: $\min ||\underline{\underline{A}}\mathbf{x}-\mathbf{b}||$, using the singular value decomposition of $\underline{\underline{A}}$. The `info` flag returned by `DGELSS` is checked to ensure successful resolution of the system, and an additional safeguard is applied: if any component of the solution lies outside the physically reasonable range of -19.9 to $+19.9$, the result is discarded.

A nuclear dipole contribution flagged as zero is internally interpreted as a failure. In practice, no physical system should yield a zero nuclear dipole contribution for all atoms, except in highly symmetric molecules. Even in such cases, numerical precision typically produces values fluctuating around positive and negative zero rather than exact double-precision zeros. The proof of this is provided in the Appendix A.3.

Because the calculation of the nuclear contribution draws upon both the topological data and the atomic charges, generated by `LocalProperties` and `AtomicProperties` respectively, it is invoked only once those two SUBROUTINES have terminated successfully. All three SUBROUTINES are dispatched from the total energy SUBROUTINE, as summarised in Algorithm 5.

Algorithm 5: Total energy subroutine for the ADF Engine

```
   /* Local properties */
1  ADFAIMCriticalPoints();
   /* The system is divided into blocks for parallelisation */
2  for iblock ← 1 to nblocks do
3      if skip_block(iblock) then
4          nextBlock
       /* Atomic properties */
5      if gQTAIM.atomic then
6          AssignAtomicBasinsForBlock() ;      // Map grid points to the attractors
7          UpdateAIMTerms() ;                  // Properties computed by integrating ρ
8  MergeValuesComputedInParallel();
   /* When the local and atomic properties are computed, the nuclear and
      (electronic + nuclear) dipole can be computed */
9  if gQTAIM.atomic then
10     AIMdipole() ;                                   // nuclear dipole moment
```

The `UpdateAIMTerms` SUBROUTINE evaluates all atomic observables that can be expressed as integrals of the electron density over a QTAIM basin; its control flow is given in Algorithm 6.

Algorithm 6: Atomic properties computed by the integral of the density

```
   /* Symmetry */
1  for no ← 1 to nogr do
       /* For each gridpoint in the block */
2      for i ← 1 nblocksize do
           /* -1 means that the point is not a part of an attractor basin */
3          if attLabel(i) > 0 then
               /* Weight from the gridpoint */
4              ρ += ρ(i)×weight(i);
5              μ_p += (coord_block(i) - coord_attractor(i))×ρ(i)×weight(i);
               /* The rest of the properties computed by integration of ρ */
               /* Quadrupole, laplacian, spin density, volume, Ts, Tsl, Tc */
```

Finally, `ADFdipole` assembles the atomic dipole moment by combining the electronic obtained above, computing the nuclear contributions and adding they two, as outlined in Algorithm 7.

Algorithm 7: Atomic properties computed by the integral of the density

```
   /* Set the topological connection in a matrix */
   /* no bond :  0; bond (A-B) : 1; bond (B-A) : -1 */
1  qbondmat ← topologicalConnection
   /* Expand the linear system in case of rings */
2  qbondmat ← overdetermineSystem(qbondmat, ringsInfo);
   /* Solve the linear system A*x = B */
3  chargeBonds ← lapack(qbondmat, atomicCharges);
4  check_values(chargeBonds) ;                    // Physically realistic values?
   /* Nuclear dipole moment contribution */
5  for i ← 1 to nAttractors do
6      for j ← 1 to nAttractors do
7          if bonded(i,j) then
8              μ_c(i) += (coord_attractors(i) - coord_bcp(i,j))× chargeBonds(i,j);
   /* Nuclear & Electronic dipole moment */
9  μ_t ← μ_c − μ_p
```

3.3.2 Polarisability

The calculation of atomic polarisabilities within QTAIM requires a numerical approach, these properties are obtained from finite-field calculations. This involves performing separate SCF calculations for each applied external electric field configuration. To manage this workflow, two parallel implementations were developed. The first, based on PLAMS, automates the sequence of calculations externally. The second, integrated directly into the AMS driver, was designed to provide seamless compatibility with the GUI.

In both cases, the required single-point calculations are executed in parallel to improve efficiency. The PLAMS implementation, however, is limited in that it cannot export the results in a format suitable for graphical visualisation. The AMS driver implementation overcomes this limitation by writing the computed polarisabilities not only to the standard output, but also to the binary files used by the AMS graphical interface, enabling direct visualisation of the atomic polarisability tensors.

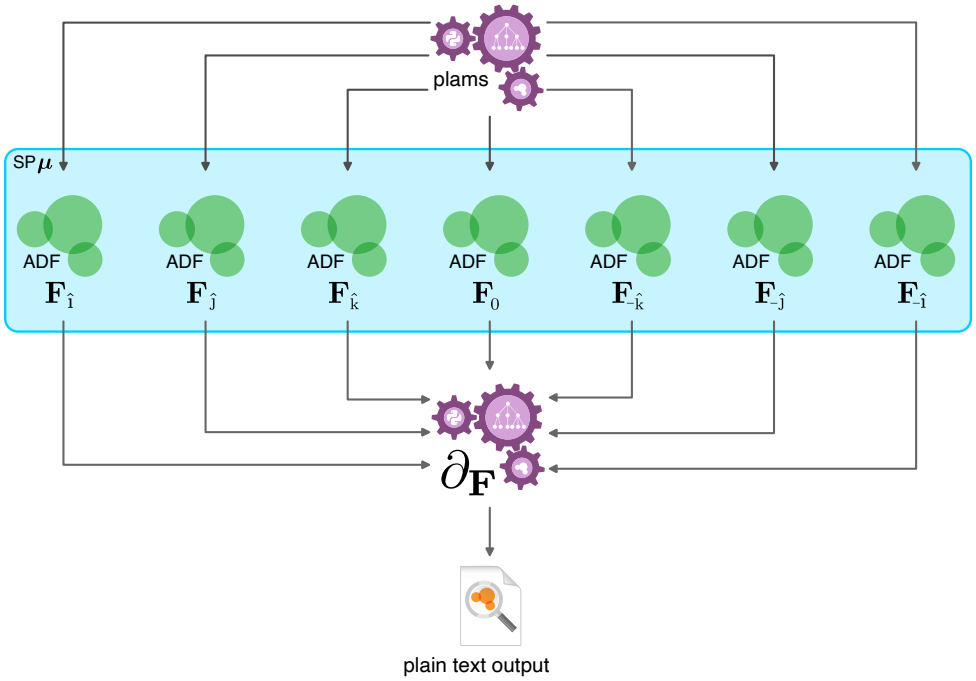


Figure 3.5. Workflow of the polarisability calculation by PLAMS.

To simplify the use of the PLAMS workflow, we encapsulated the procedure in a PYTHON class. As a result, the full set of calculations needed for the polarisability of a system can be launched easily as well as the numerical derivaties. The next summarised PYTHON code illustrates the implementation:

```
class DipoleMomentJob(MultiJob):
    """Dipole moment with an electric field in (x,y,z,−x,−y,−z) directions.

    Attributes:
        molecule:           The molecule to calculate the dipole moment for.
        common_settings: Common settings for all jobs.
        directions:         Directions of the electric field.
        eField:             Electric field values.
    """
    def ___init___(self, molecule, common_settings, directions, eField, **kwargs):
        """Initializes the DipoleMomentJob with molecule, settings, directions,
           and electric field."""
        MultiJob.___init___(self, children=OrderedDict(), **kwargs)
        # Code omitted for brevity
        # self.molecule = molecule; self.eField; self.setup_jobs()

    def setup_jobs(self):
        # Code omitted for brevity

    def collect_results(self) − Tuple[np.ndarray, np.ndarray]:
        """Collects and returns the dipole moments."""

# Inicialize and run the DipoleMomentJob's in the PLAMS workflow

dipole_job = DipoleMomentJob(molecule=mol,
                                    common_settings=adfsettings,
                                    directions=directions, eField=eField)
dipole_job.run()

# Collect the results
dipole, atom_charge = dipole_job.collect_results()
# Numerical derivatives
pol_tensor = np.zeros((natoms, 3, 3))
for j in range(natoms):
    pol_tensor[j,:,:] = (dipole[:3,j,:] − dipole[3:6,j,:])/(2.0∗eFieldmag)
```

The PLAMS recipe can operate either from a plain `xyz` geometry file or from the results of a prior ADF calculation. In both cases, the magnitude of the applied electric field defaults to 0.01 a.u. (input in V/Å), unless overridden by the user through the `eFieldmag` keyword.

When a previous ADF calculation is provided, the recipe automatically adopts the same level of theory as that calculation. If the atomic properties for the electric field-free system are already computed in the ADF output, the workflow bypasses their recomputation and proceeds directly to launch the six additional single-point calculations required for the finite-field analysis.

When an `xyz` file is used as the starting point, the workflow assigns a default level of theory of M06-2X/TZ2P. This ensures a consistent, reasonably accurate description for systems.

Example usage

A typical invocation of the recipe is illustrated below. The workflow is executed directly from the terminal, the calculation with the `xyz` file is performed with optional argument used to modify the electric field for the calculation (0.005 a.u.).

```
vcastor@aragorn - bash
$ ls -l
 total 7
 -rw-r-r-  1 user  group   10k  Jan 20 11:13 QTAIMpol.py
 -rw-r-r-  1 user  group   150  Aug 11 14:32 water.xyz
 -rwxr-r-  1 user  group   580  Aug 11 14:32 water.run
 -rw-r-r-  1 user  group   76k  Aug 11 14:32 water.out
 drw-r-r-  8 user  group   405  Aug 11 14:36 water.results
$ # from a previous calculation
$ $AMSBIN/plams QTAIMpol.py -v resultsdir=./water.results
$ # from scratch
$ $AMSBIN/plams QTAIMpol.py -v xyzfile=./water.xyz -v eFieldmag=0.26
$ ls -la *poldip
 -rw-r-r-  1 user  group  1.3k  Aug 11 15:39 water.poldip
```

3.3.3 Excited States

As shown in Equation 2.103, the change in dipole moment, $\Delta\boldsymbol{\mu}$, between the ground and an excited state can be evaluated. In ADF, molecular orbitals and their corresponding grid coordinates can be extracted for each block (grid region), enabling the computation of $\Delta\boldsymbol{\mu}$ within QTAIM.

Since the loop over all grid points, points already labelled with an attractor in the ground state, the decomposition of the excited-state dipole moment reduces to assigning each contribution to its corresponding atomic basin.

Although minor differences may exist between the ground- and excited-state topologies, adopting the ground-state partition significantly accelerates the calculation of the excited-state dipole moment while preserving accuracy. A condensed version of the FORTRAN implementation is provided below.

```
rhop = 0._kreal; rrhop = 0._kreal; rrhon = 0._kreal
! The system is divided into blocks for parallelisation
iblock_ : do iblock = 1, gDims%nblock
  if (skip_block(G, iblock)) cycle iblock_
  call get_block(G); call calc_bas_block(BAS, G, bas0) ! Parallel stuff
  if (BAS%local_num == 0) cycle iblock_

  allocate(moCoef(BAS%local_num, nocc+nvir), bas0tmp(G%npoints, BAS%local_num))

  ! get the occupied and virtual orbitals
  do i = 1, BAS%local_num
    iao                 = BAS%global_index(i)
    bas0tmp(:, i)       = bas0(:, iao)
    moCoef(i, 1:nocc) = occao(iao, :)
    moCoef(i, nocc+1:nocc+nvir) = virao(iao, :)
  enddo
  ! Optimised matrix multiplication
  ! (SCM BLAS [Basic Linear Algebra Subprogram] implementation)
  call SCMgemm(bas0tmp, moCoef, mo) ! mo = bas0tmp * moCoef

  ! Connect the grid to the QTAIM grid atom mapper
  call GetGridAtomMap(aimGridAtomMapper, attLabel)

  ! update the coordinates
  ! replicate G%w along dimension 2 to match shape of G%coord
  coordw = G%coord*spread(G%w, 2, 3)
  ! squere of the MO coefficients
  mo2 = mo*mo

  ! The integrals start here; k is the index for every point in the block
  do k = 1, blocksize; if (AttLabel(k) .gt. 0) then ! Skip points with no label
    atom = AttLabel(k)
    ! [[ \int r\rho_+ = \langle \varphi_a | \hat{r} | \varphi_a \rangle ]]
    do i = 1, nocc; do j = 1, 3
      rrhop(atom) = rrhop(atom) + sum(mo2(:,i)*coordw(:,j))
    enddo; enddo
    ! [[ \int r\rho_- = \langle \varphi_i | \hat{r} | \varphi_i \rangle ]]
    do i = 1, nvir; do j = 1, 3
      rrhon(atom) = rrhon(atom) + sum(mo2(:,i+nocc)*coordw(:,j))
    ! [[ \int \rho_+  = \langle \varphi_a | \varphi_a \rangle ]]
    enddo; enddo
    rhop(atom) = rhop(atom) + sum(mo2(:,1:nocc))
  endif; enddo
end do iblock_
```

3.4 Current Status of QTAIM in AMS

To encapsulate the QTAIM code and ensure its accessibility from the various engines within the suite, we have documented the interconnection of the main MODULES. This modular design not only facilitates integration into existing workflows but also simplifies future development. Figure 3.6 illustrates how QTAIM can be invoked within a typical AMS workflow, from user input through to the final output.

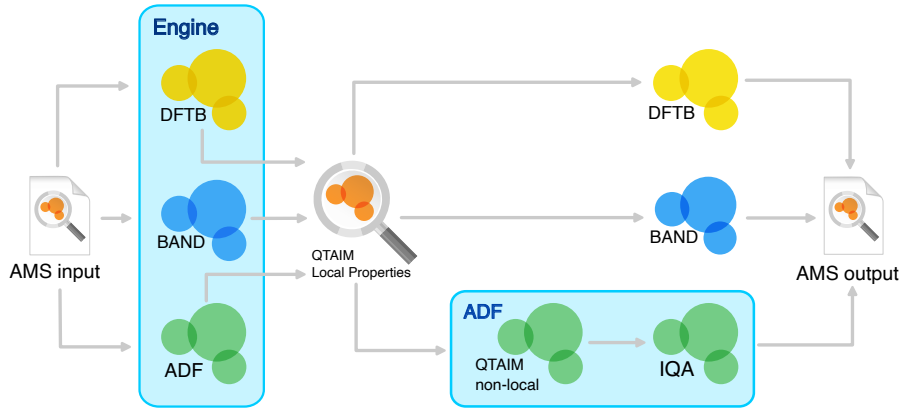


Figure 3.6. Workflow from AMS input to output for the engines supporting the QTAIM partition.

Figure 3.7 provides an overview of the principal SUBROUTINES and MODULES involved in the QTAIM implementation. The AMS driver can perform a single-point calculation with any of the three engines. In the subsequent post-analysis stage, the QTAIM partition is executed. In this workflow, the SUBROUTINE `DoSinglePoint` calls `DoPostSCF`, which first executes the total energy SUBROUTINE (`atoten`), followed by the Debye analysis (`adebye`). The main SUBROUTINE for producing QTAIM results (`AIMResults`) requires the outputs of these two subroutines to compare the dipole moments obtained from the Debye analysis and from the QTAIM partition.

The dipole moment calculation depends on both local properties, computed in `ADFAIMCriticalPoints`, and atomic properties, such as the atomic charges determined by `UpdateAIMTerms`. Consequently, the SUBROUTINE responsible for the dipole moment (`AIMDipoleMoment`) must be executed only after both `ADFAIMCriticalPoints` and `UpdateAIMTerms` have completed.

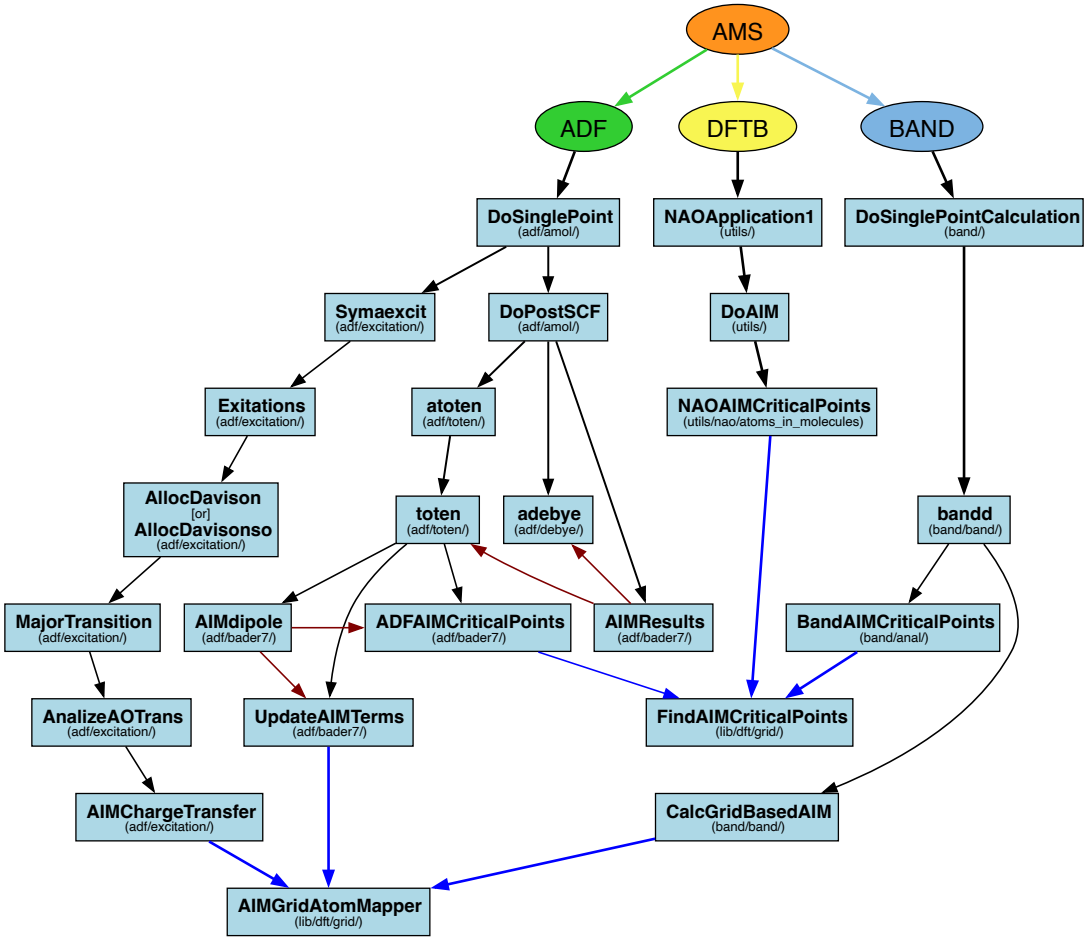


Figure 3.7. The blue arrows indicate the main dependencies; black arrows denote subroutine calls; and brown arrows indicate that the main subroutine of a module must wait for the execution of the targeted subroutines.

3.4.1 Descriptors Available in AMS

The descriptors from the topological analysis (`AnalysisLevel Normal`) are listed in Table 3.2. For BAND, the analysis is more limited, providing only the electron density and Laplacian for each basin.

Table 3.2. Descriptors available in ADF and DFTB for the Topological Analysis.

Descriptors Available in ADF and DFTB for CPs		
Descriptor	**Equation**	**Reference**
General Descriptors	$\rho, \nabla\rho, \mathbf{H}_\rho, \nabla^2\rho, \diamondsuit\rho, ellepticity$	
Metallicity	$\xi_m(\mathbf{r}_{\mathrm{cp}}) = \frac{36(3\pi^2)^{2/3}}{5}\frac{\rho^{5/3}}{\nabla^2\rho}$	[146]
Variation Rates	$\delta u = \frac{1}{4\pi}\int_\tau \sin\phi\sqrt{\lambda_1^2\sin^2\phi\cos^2\theta + \lambda_2^2\sin^2\phi\sin^2\theta + \lambda_3^2\cos^2\phi}\mathrm{d}\phi\mathrm{d}\theta$ $\delta s = \frac{A_s}{4\pi\rho^{4/3}}\int_\tau \sin\phi\sqrt{\lambda_1^2\sin^2\phi\cos^2\theta + \lambda_2^2\sin^2\phi\sin^2\theta + \lambda_3^2\cos^2\phi}\mathrm{d}\phi\mathrm{d}\theta$ $\delta t = \frac{A_t}{4\pi\rho^{7/6}}\int_\tau \sin\phi\sqrt{\lambda_1^2\sin^2\phi\cos^2\theta + \lambda_2^2\sin^2\phi\sin^2\theta + \lambda_3^2\cos^2\phi}\mathrm{d}\phi\mathrm{d}\theta$	[147, 148]
Inhomogeneites	$l_{s(\mathbf{r}_c)} = \frac{1}{\langle s'\rangle(\mathbf{r}_c)}$ $l_{e_x(\mathbf{r}_c)} = \sqrt{e_{x(\mathbf{r}_c)}/\langle\delta e_x\rangle_{\mathbf{r}_c}}$	[149]
Abramov's descriptors	$G(\mathbf{r}_{\mathrm{cp}}) = {}^3/_{10}(3\pi^2)^{2/3}\rho^{5/3} + {}^1/_6\nabla^2\rho$ $V(\mathbf{r}_{\mathrm{cp}}) = {}^1/_4\nabla^2\rho - 2G(\mathbf{r}_{\mathrm{cp}})$ $H = G(\mathbf{r}_{\mathrm{cp}}) + V(\mathbf{r}_{\mathrm{cp}})$	[150, 151]
Uniform Electron Gas	$e_x^{\mathrm{UEG}}(\mathbf{r}_{\mathrm{cp}}) = -\frac{3}{4}\left(\frac{3}{\pi}\right)^{1/3}\rho^{4/3}$ $e_c^{\mathrm{UEG}}(\mathbf{r}_{\mathrm{cp}}) = \frac{a_0+a_1\rho+a_2\rho^2}{b_0+b_1\rho+b_2\rho^2}$	[148]
Variation Rates	$\langle\delta e_x\rangle^{(2)} = \frac{1}{2\sqrt[3]{(9\pi)}}\rho^{1/3}\nabla^2\rho - \frac{\mu\diamondsuit\rho}{16(3\pi^5)^{1/3}\rho^{4/3}}$ $\langle\delta e_c\rangle^{(2)} = \frac{a_1b_0-a_0b_1+2(a_2b_0-a_0b_2)\rho+(a_2b_1-a_1b_2)\rho^2}{6(b_0+b_1\rho+b_2\rho^2)^2}\times\nabla^2\rho$ $+\frac{\pi^{1/3}\beta_c\diamondsuit\rho}{3^{4/3}16\rho^{4/3}}$ $\langle\delta e_T\rangle^{(2)} = \frac{\pi^{4/3}}{4\sqrt[3]{3}}\rho^{2/3}\nabla^2\rho + \frac{\diamondsuit\rho}{24\rho}$	[148]
Locally Compare Ratios	$Q_{xc}(\mathbf{r}_c) = \frac{\langle e_x\rangle^{(2)}_{\mathbf{r}_c}}{\langle e_c\rangle^{(2)}_{\mathbf{r}_c}}$ $P_{xc}(\mathbf{r}_c) = \frac{e_x(\mathbf{r}_c)}{e_c(\mathbf{r}_c)}$	[148]

The atomic descriptors (`AnalysisLevel Extended`) are summarised in Table 3.3. Finally, for `AnalysisLevel Full`, the localisation and delocalisation indices are computed. An example of the corresponding output file is provided in Appendix A.6.

Table 3.3. Descriptors available in ADF and for the Atomic Properties.

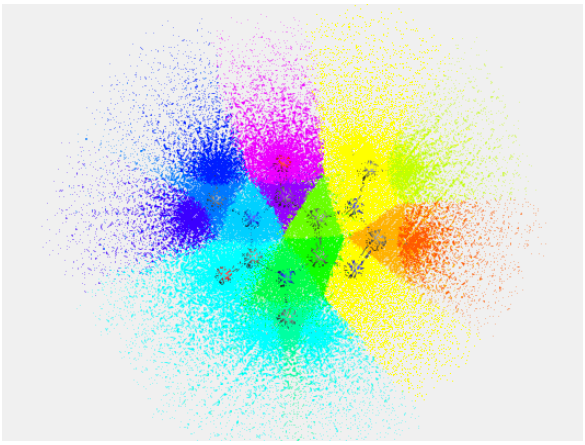
Descriptors Available in ADF for Atomic Basins		
Descriptor	**Equation**	**Reference**
Population	$\int \rho(\Omega)\mathrm{d}\tau$	[3]
Charge	$q(\Omega) = Z_\Omega - \int \rho(\Omega)\mathrm{d}\tau$	[3]
Spin density	$\rho(\uparrow) - \rho(\downarrow)$	[3]
Laplacian	$\int \nabla^2(\rho(\Omega))$	[3]
Volumen	$\int_t \rho(\Omega)\mathrm{d}\tau \qquad t \in \{0.002, 0.001, 0.0004\}$a.u.	[3]
Shanon Entropy	$H = \int \sigma \ln \sigma \mathrm{d}\tau$	[152]
Shanon Shape	$\frac{H}{\int \rho \mathrm{d}\tau} + \frac{\ln \int \rho \mathrm{d}\tau}{\int \rho \mathrm{d}\tau} \int_\Omega \rho \mathrm{d}\tau$	
Reyi Shape	$-\ln \int_\Omega \rho^2 \mathrm{d}\tau + \ln \int \rho^2 \mathrm{d}\tau$	
Dipole moment	$\boldsymbol{\mu}_c = q_\alpha R_\alpha \; ; \; \boldsymbol{\mu}_p = -\int (r - R_\alpha)\rho \mathrm{d}\tau$	[87]
Quadrupole moment	$Q_{ij} = \int \rho(\Omega)(3r_i r_j - r^2\delta_{ij})\mathrm{d}\tau$	
population decomposition over occupied orbitals	$\frac{\psi^2_{MO}}{\rho(\Omega)}$	

3.4.2 Graphical User Interface (GUI)

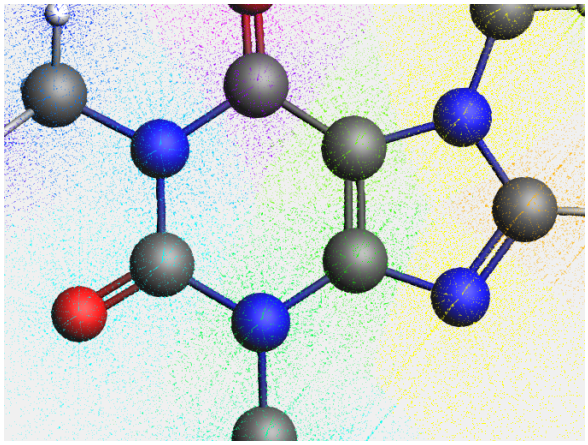
Although the QTAIM partition code had already been implemented in the FORTRAN backend, with results written in both human-readable output and binary files, the GUI offered only limited support for visualising the computed properties. In particular, the display of atomic basin shapes was not user-friendly.

The main difficulty in rendering atomic basin shapes arises from the feature that makes the QTAIM partition so fast in AMS: the absence of an explicit calculation of the zero-flux condition (as discussed in previous sections). The partitioning procedure assigns each grid point to a specific attractor (or to none), but the basin boundaries cannot be directly extracted from values computed.

In the previous GUI version, atomic basins were visualised by colouring the corresponding grid points. While functional, this approach often led to visual clutter, particularly when zooming in or out, as shown in Figure 3.8a.



(a) When zoomed out, the system is obscured by grid points, making it difficult to visualise the underlying molecular structure.



(b) When zoomed in, the grid points are spaced too far apart, and their association becomes difficult for the user.

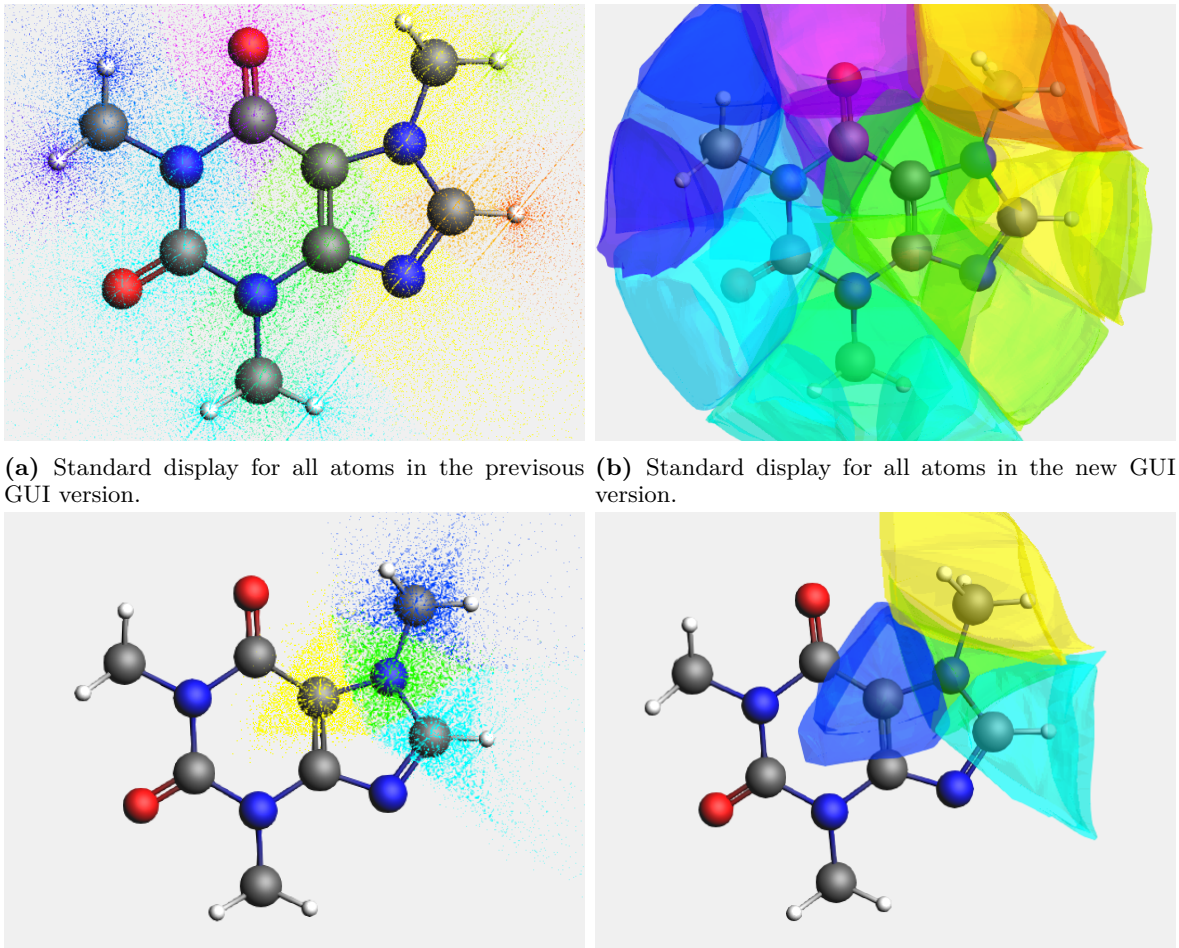
Figure 3.8. Caffeine molecule.

To address this limitation, we implemented a backwards-compatible GUI feature that reconstructs approximate basin surfaces using the labelled grid points. These shapes are rendered in VTK within the TCL-based GUI. A condensed version of the TCL code is provided below with comments explaining the main steps of the process.

```
# Read the TAPE10 ...
# A polygon for every atom
for {set iatom 0} {$iatom <= $natoms} {incr iatom} {
    # PolyData store the strucuture for every atom
    vtkPolyData $id.t10polydata.$iatom
    lappend VTK($id,ids) $id.t10polydata.$iatom
    $id.t10polydata.$iatom SetPoints $id.t10points
    $id.t10polydata.$iatom SetVerts $id.t10vert.$iatom

    # Sanity check
    if { [$id.t10polydata.$iatom GetNumberOfCells] < 4 } { continue }

    # Solid convex hull of the original points
    vtkDelaunay3D $id.del3d.$iatom
    # Code omitted for brevity ...

    # Exterior faces; from an unstructured grid to a surface
    vtkDataSetSurfaceFilter $id.surf.$iatom
    # Set the connection for the surface ...

    # More polygons; subdivision
    vtkLoopSubdivisionFilter $id.subdiv.$iatom

    # Set the smoothing parameters ...
    vtkSmoothPolyDataFilter $id.t10smooth1.$iatom

    # Normal vectors [Generated by vtk]
    vtkPolyDataNormals $id.norm.$iatom
    $id.norm.$iatom SetInputConnection [$id.t10smooth.$iatom GetOutputPort]

    # Set the values for opacity and visual appearance
    [$id.t10actor.$iatom GetProperty] SetOpacity 0.4
    # ...
}
```

It is worth noting that the implementation could be enhanced by increasing the number of polygons used to represent the surfaces. However, for large systems this would incur a significant performance cost. Another possible improvement is to replace the VTK-generated normal vectors with those derived from the gradient of the electron density. While this would not change the resolution, it would enable more realistic lighting calculations, resulting in smoother and more visually appealing surfaces. This latter feature has already been developed but is not yet included in the trunk branch. It will only be available for future AMS calculations, as the gradient at each grid point is not stored in the current binary files.

A comparison of the new and old GUI display for atomic basins is shown in the next Figures 3.9.



(a) Standard display for all atoms in the previsous GUI version.

(b) Standard display for all atoms in the new GUI version.

(c) Only four atoms selected in the previous GUI version.

(d) Only four atoms selected in the new GUI version.

Figure 3.9. Comparison of the new and old GUI display for atomic basins, for caffeine molecule.

3.5 Optimisation of the code

The Amsterdam Modeling Suite is a large and evolving code composed of multiple modules, each with its own features and interfaces. Over time it has transitioned from early FORTRAN dialects to modern standards and has developed a PYTHON library to coordinate the AMS engines. Despite extensive documentation and developer tooling, two avenues for improvement remain central to long-term sustainability: *i*) refactoring for clarity and maintainability, and *ii*) targeted optimisations for memory footprint and runtime.

3.5.1 Memory Optimisation

In previous versions, properties evaluated at CPs were stored in a single preallocated two-dimensional array. Its first dimension assumed an upper bound of 256 CPs per atom (multiplied by the number of atoms), while the second dimension listed the properties per CP (27 entries, one unused). This conservative approach wasted memory for realistic systems, where the actual number of CPs is markedly lower than the imposed maximum.

Estimating a Upper Bound

To reduce memory usage without compromising functionality, we analysed the theoretical upper limits for the number of bonds, rings, and cages in a system as a function of the number of atoms. Because the identification of NNAs occurs after memory allocation, a buffer must be included to account for their possible presence. We adopt a conservative limit of either ten additional attractors or 10% of the number of atoms, whichever is larger.

Then to estimate the number of bonds, we consider a hypothetical —albeit physically unrealistic— scenario in which every atom is bonded to every other atom: $\max(\mathrm{BCP}) = (n-1)n/2$, where n is the number of atractors. To estimate the maximum number of rings, we consider the upper limit based on polygonal connectivity, assuming no subsets: $\max(\mathrm{RCP}) = (n-3)n/2 + 1$. For cages, if we consider only convex polyhedra composed of triangular faces, we derive to $\max(\mathrm{CCP}) = 6n - 24$.

While such estimates exceed what occurs in chemically realistic systems, they provide conservative bounds for safe memory allocation.

$$\max(\mathrm{CP}) = n^2 + 4n - 23 \tag{3.2}$$

Dynamic Allocation Strategy

The original implementation allocated more memory than the theoretical upper limit discussed above. While this ensured correctness, it left room for improvement. Our aim was therefore to design a strategy that remains close to the theoretical bound for small systems, but grows more moderately with system size in order to reduce unnecessary overhead for large systems.

In constructing such a function, we also examined definitions based on piecewise regimes, where different growth laws are applied in distinct regions of system size. While these approaches can be tuned to follow the theoretical limit more closely in certain ranges, they inevitably introduce `if` statements into the code, which reduces readability. Since our goal was not to optimise memory allocation with mathematical precision, but rather to provide a practical and transparent strategy, we chose to avoid such constructions.

Several candidates were considered, as illustrated in Figure 3.10. Ultimately, we selected the simplest formulation, prioritising clarity. Although this choice slightly overallocates memory for small systems, the overhead is negligible in practice. More importantly, the gradual growth at larger system sizes ensures that the code remains both efficient and scalable.

```
integer, parameter :: numCPprops = 26 ! Number of properties per critical point

! Adjusted number of atoms for periodic systems and non-periodic systems
nVectors = 0_KINT
if (present(numLatticeVectors)) nVectors = numLatticeVectors
maxCP = numAtoms + 3**nVectors

! Maximum number of critical points
maxCP = max(maxCP + 10, maxCP + maxCP/10) ! 10 NNAs or 10 % NNAs
maxCP = int(50*sqrt(real(maxCP))*(1 - exp(-real(maxCP)/50)))
allocate(self%cpmat(maxCP, numCPprops), stat=ialloc)

! Check the allocation status; and error is fatal if first argument is 0
call chckmem(0, moduleName//'::cpmat', ialloc)
```

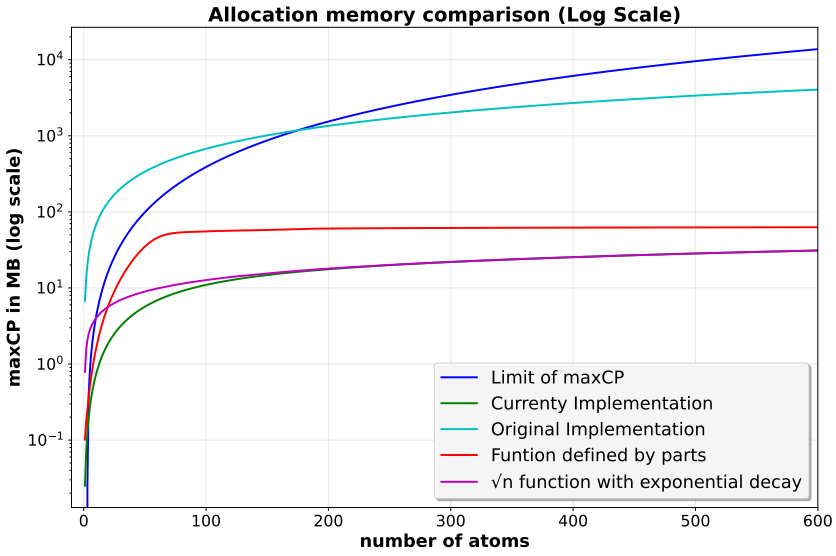


Figure 3.10. Comparison of alternative memory allocation strategies as function of system size.

3.5.2 Refactoring

Several parts of the code were originally written in a style that new developers might find difficult to follow and, more importantly, to maintain. For instance, the SUBROUTINE `UpdateAIMTerms` called two separate SUBROUTINES which computed almost identical quantities, differing only in the treatment of a few variables, as shown in the code excerpt below.

```
if (baderEnergy) then
   call elf(lblock,fragment%nspin,rho0,rho1,tau,elfu)
   call CalcAtomPropEnergy(gridAtomMap, xyzatm, &
                           coords,wghts,rho0,rho1,rho2,ntr,op,  &
                           tau,vXC,EpsXC,vNuc,vHartree,elfu, & ! Extra variables
                           fden,aimProperty,orbgrd,atomNbPts,inpatm)
else
   call CalcAtomProp(gridAtomMap, xyzatm, &
                     coords,wghts,rho0,rho1,rho2,ntr,op,  &
                     fden,aimProperty,orbgrd,atomNbPts,inpatm)
end if
```

Merging these two SUBROUTINES (`CalcAtomPropEnergy` and `CalcAtomProp`) eliminated duplicated logic, improved readability, and simplified the overall flow of the program.

```
if (baderEnergy) call elf(lblock, fragment%nspin, rho0, rho1, tau, elfu)
call CalcAtomProperties(gridAtomMap, xyzatm, coords, wghts, &
                        rho0, rho1, rho2, op, &
                        tau, vXC, EpsXC, vNuc, vHartree, elfu, & ! Ignore if not needed
                        fden, aimProperty, orbgrd, atomNbPts, inpatm)
```

The refactored SUBROUTINE has an extra `if` statement to use the extra variables, the SUBROUTINE is the one we describe in Algorithm 6.

Other refactoring steps focused on improving efficiency. The inclusion of precomputing factors outside of loops rather than repeatedly recalculating them was a common pattern throughout the codebase. We display an example in the code snippets below:

```
! Old code
! Loop over the number of Critical Points
! the name nna is not clear; can be understood as number of non-nuclear attractors]
DO i = 1, nna
  ! Factors over computed; every loop of i the same factors are computed
  DO ii=1,36
     newAbscissae(ii,1) = pi*abscissae(ii)+pi
     newAbscissae(ii,2) = pi/two*abscissae(ii)+pi/two
  END DO
  ! The scalars can be computed outside the integral
  integral = 0.0_KREAL
  DO ii=1,36
     DO jj=1,36
        integrand = SQRT(! Code omitted for brevity...
        integral  = integral + pi*pi/two*weights(ii)*weights(jj)*integrand
     END DO
  END DO
  ! ...
```

```
! New code
! Pre-compute the factors outside the integral loop
do i = 1, 36
   newAbscissae(i,1) =     pi*abscissae(i) + pi
   newAbscissae(i,2) = halfpi*abscissae(i) + halfpi
end do
do i = 1, ncp ! Loop over the number of Critical Points
   integral = 0.0_KREAL
   do itheta = 1, 36
      do iphi = 1, 36
         integrand = SQRT(! Code omitted for brevity...
         integral  = integral + weights(itheta)*weights(iphi)*integrand
      end do
   end do
   integral = pi*halfpi*integral ! Scale the integral outside the loop
   ! ...
```

Encapsulation was another key aspect: new MODULES were created to group related functionality. For example, a dedicated MODULE was introduced for handling plain-text output and binary file writing (see Appendix A.5), and another MODULE for general mathematical utilities. Likewise, functionality for following the gradient path was placed in a separate MODULE, ensuring that a single, consistent SUBROUTINE could be applied to bonds, rings, and cages.

Previously, the Runge-Kutta method was implemented directly inside the bond path SUBROUTINE and the SUBROUTINE dedicated to assigning grid points to basins, making it inaccessible to other parts of the code and leading to duplicated code. By encapsulating the method in a standalone SUBROUTINE, it became reusable across the QTAIM partition, enabling algorithms such as Algorithm 4 to be written without redundant implementations. In the same spirit, all Graph Theory and Geometry features were conceived from the outset as independent MODULES (Subsection 3.2.4 and Appendix A.1).

Finally, additional comments and explicit parameter definitions were introduced to improve readability and maintainability. In the main MODULE where all QTAIM variables are declared, comments now provide a clear map of array indices to the corresponding physical properties (*e. g.* density, dipole and quadrupole moments, Laplacian, spin density, basin volume). Similarly, in the MODULE responsible for computing properties at CPs, integer parameters were defined to label array indices such as coordinates, density, gradient, Hessian, signature, and eigenvectors. These additions reduce ambiguity, make the code self-documenting, and simplify future modifications by replacing hard-coded indices with descriptive constants.

3.6 Testing and Performance

To validate the new implementation, we performed extensive tests across a wide range of systems, aiming both to push the code to its limits and to ensure its robustness. These tests also allowed us to compare the performance of the current version with other programs and with the previous implementation.

3.6.1 Exotic and Complex Topologies

To assess the limits of the code, we selected systems that had previously been reported as problematic in earlier versions, as well as examples from the literature known to present topologically challenging cases.

Grid Refinement

In earlier versions, the default grid settings failed to satisfy the Poincaré-Hopf relation for systems such as H_2. The single BCP in this molecule requires a finer grid for reliable detection. This issue was not restricted to small systems; more complex molecules, such as ferrocene, $(\eta^5\text{-}C_5H_5)_2Fe$, also exhibited undetected CPs when using default grid parameters.

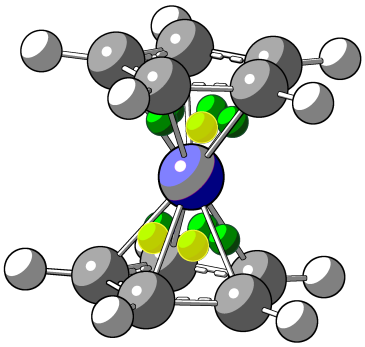
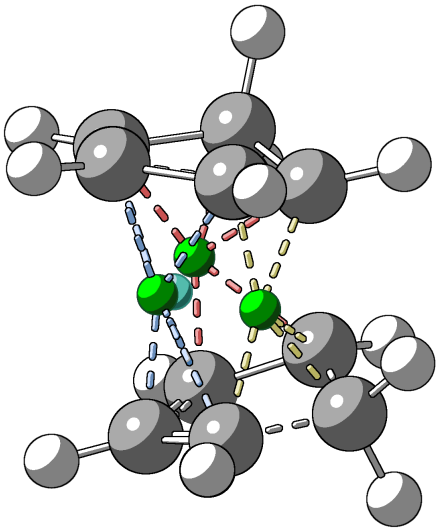


Figure 3.11. Ferrocene with its RCPs. The RCPs highlighted in yellow were not detected in the previous version of the code. Due to molecular symmetry, any of the RCPs could be among those missing.

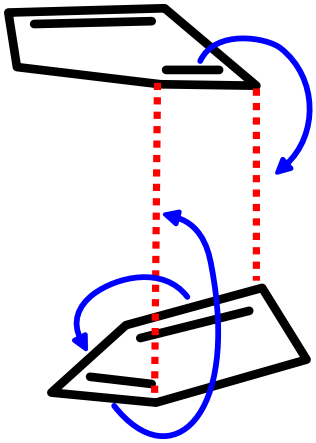
Gradient Path Algorithm

To further test the limits of the code, we examined two systems in which the electron density exhibits a relatively flat profile, making the algorithm used to follow the gradient path crucial for correctly determining how the atoms are bonded: *i*) the transition state of the Diels – Alder reaction of cyclopentadiene, and *ii*) the Be_3^{-2} system. It is worth noting that QTAIM analysis is not restricted to neutral or minimum-energy states.

In case i), illustrated in Figure 3.12a, the transition state features a complex topology, a Diels-Alder transition state between two molecules of cyclopentadiene, dicyclopentadiene as product. Here, fine grid refinement was not required to detect all CPs with BLYP/DZ, however, it was required for M06-2X/TZP. The use of fourth-order Runge-Kutta was essential for accurately tracing the gradient paths within the rings. While the two C-rings are clearly resolved as well as the cage, several additional RCPs appear in the topology, which can be attributed to the bond-forming and bond-breaking processes occurring at the transition state.



(a) The non-trivial RCPs (green spheres) are shown with their constituent atoms, with the associated paths highlighted in yellow, magenta, and cyan for clarity. The CCP (blue sphere) connects all carbon atoms. For simplicity, the two RCPs associated with the cyclopentadiene rings (considered as trivial) are omitted. Cartesian coordinates are provided in Appendix A.4.



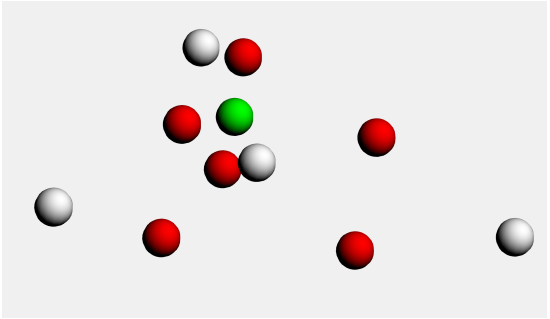
(b) Mechanism of reaction of the Diels-Alder transition state, in standard organic chemistry notation.

Figure 3.12. Diels-Alder transition state. The creation of bonds noted by dashed lines in Figure 3.12b is reflected in the ring highlighted in yellow in Figure 3.12a.

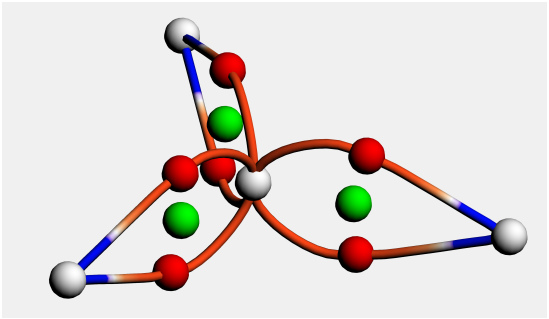
In case *ii*), we investigated the Be_3^{-2} system as a function of basis set, functional, and interatomic distance, as this system has previously been reported to exhibit complex topology [153]. The model consisted of an equilateral triangle of Be atoms, with the side length varied systematically. This analysis revealed that the number of CPs changes in function with both geometry and level of theory, including NNA. In the previous code version, some of the systems do not satisfy the Poincaré-Hopf relation, and in some of them the gradient path goes two times to the same atom or to no any.

In contrast, the improved algorithm in the current version consistently identifies all CPs and follows the gradient paths correctly. It is important to note that sometimes the refinement of the grid was required until the grid spacing was 0.013 bohr, which is a significant fine grid, we proposed to use 0.1 bohr for the new default grid spacing (used to be 0.5 bohr). This change will contemplate most of the systems.

To illustrate the performance, we display the case of the Be_3^{-2} system with 1.960 Å interatomic distance between the Be atoms, using the B3LYP functional and DZP basis set (Figure 3.13). For the previous version, the code finds the NNA, six BCPs, and only one RCPs, no any of the gradient paths goes to anywhere. The current version finds the same NNA, six BCPs, but also the three RCPs, as well following the paths from the BCPs to the atoms, properly.



(a) Every connection from the Be atoms to the NNA have two BCPs, but only one has the RCPs that should appear between the two BCP. The gradient paths simply do not appear, since they do not go to anywhere.



(b) All BCPs as well as the RCPs are found, and all the gradient paths go from the Be atoms to the NNA.

Figure 3.13. Comparison of the Be_3^{-2} system with 1.960 Å interatomic distance between the Be atoms, using B3LYP/DZP. Images generated directly from the AMS GUI. The BCPs are shown in red, the RCPs in green.

3.6.2 Numerical Comparison

Following the refactoring of the code, it was essential to verify that the new version reproduces the same physical properties as the previous one. While minor numerical differences are expected as a result of the refactoring, the underlying physical trends must remain unchanged.

For atomic properties already implemented in earlier versions, where the only modifications arose from code refactoring, we used the built-in tools of the AMS development environment to compare the results of the new implementation with those of the most recent stable and well-documented version. The differences in numerical values were on the order of 10^{-15}, well below the threshold of numerical noise, and were ignored by the comparison tools.

```
victoria@rameau - bash
$ # Compare the results of the new code with the previous version
$ cd $AMSHOME
$ for keyword in IQA Bader QTAIM; do
>   ./Utils/run_where $keyword
> done
$ # Summary files
$ ls diffs.*
```

In contrast to the properties affected only by refactoring, the atomic dipole moments have no direct counterpart in previous versions of the code, and must therefore be validated against external implementations. For this purpose, we computed 34 k molecular systems (∼468 k atoms) using ADF and compared the results with those obtained from Orca and AIMAll. The systems were taken from the database used to train NNAIMQ, a neural network model for predicting QTAIM charges [154], no any transition metal was included.

The comparison considered the three components of the dipole moment: electronic, nuclear, and total contributions. As shown in Figure 3.14, the interatomic (nuclear) contributions exhibit a difference of 0.004 ± 0.052 a.u., which has a larger dispersion compared with the intraatomic (electronic) terms, -0.020 ± 0.016 a.u., however, the average differences of the interatomic contribution is lower, this play between "less differences" and "higher dispersion" is mitigated when we analyse the total dipole moment, which exhibits a difference of 0.006 ± 0.045 a.u., balancing the two contributions. Moreover, the median value for the differences is 0.0019 a.u. for the total dipole moment, while for the interatomic and intraatomic contributions are 0.005 and -0.022 a.u.

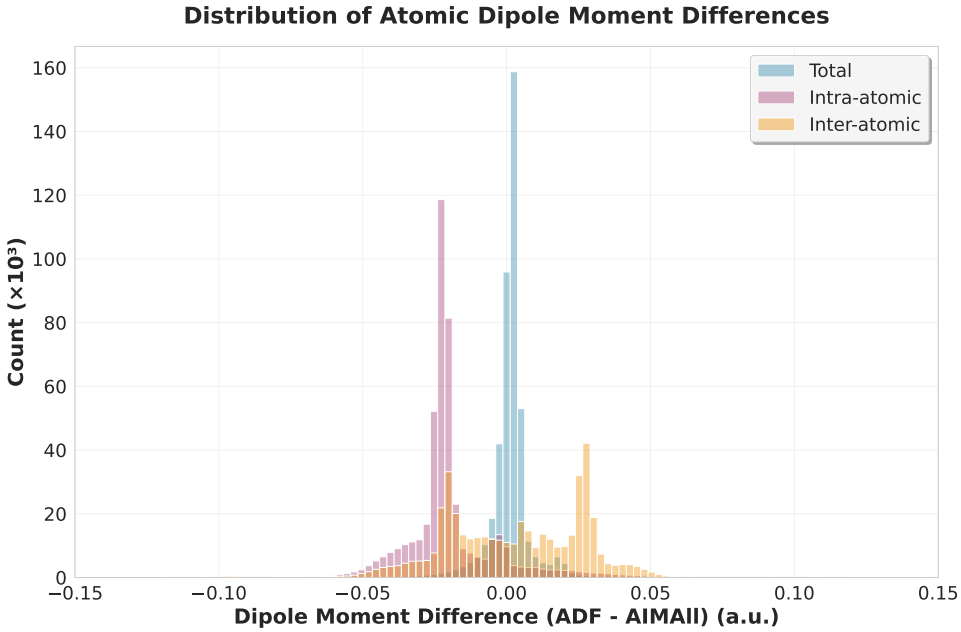


Figure 3.14. Histogram of differences in dipole moment components between ADF and AIMAll: electronic, nuclear, and their sum.

These minor discrepancies are expected given the different numerical approaches used by each program. Orca and AIMAll employ GTOs, whereas ADF uses STOs. All calculations were carried out with the M06-2X functional, using the aug-cc-pVTZ basis set in Orca/AIMAll and the TZ2P basis in ADF. The difference in basis set type can introduce small variations in the electron density distribution, and consequently in the atomic properties derived from it.

Our implementation of atomic polarisabilities was also numerically compared, as illustrated in Figure 3.15, which shows the distribution of differences between ADF and AIMAll for a test set of 150 systems (∼4 k atoms), randomly selected from the dataset used for dipole moment comparisons. The reduced sample size is justified by the high computational cost of polarisability calculations and by the fact that the main source of numerical error originates from the atomic dipole moments, which have already been validated.

Because the polarisability is obtained from numerical derivatives of the dipole moment with respect to the applied electric field, a similar level of numerical noise is expected. The $\bar{\alpha}$ values can be directly compared between the two codes, with an average difference of -0.15 ± 1.30 a.u. and a median of 0.02 a.u. By contrast, the individual λ_i require additional care, to eliminate ambiguities from the arbitrary ordering of eigenvalues, they must be sorted prior to any meaningful comparison.

As anticipated, the magnitudes of the eigenvalues vary considerably. When ordered from smallest to largest, the average differences between codes are -0.0073, -0.0597, and -0.3958, with corresponding standard deviations of 2.6279, 1.8753, and 1.2294 a.u.

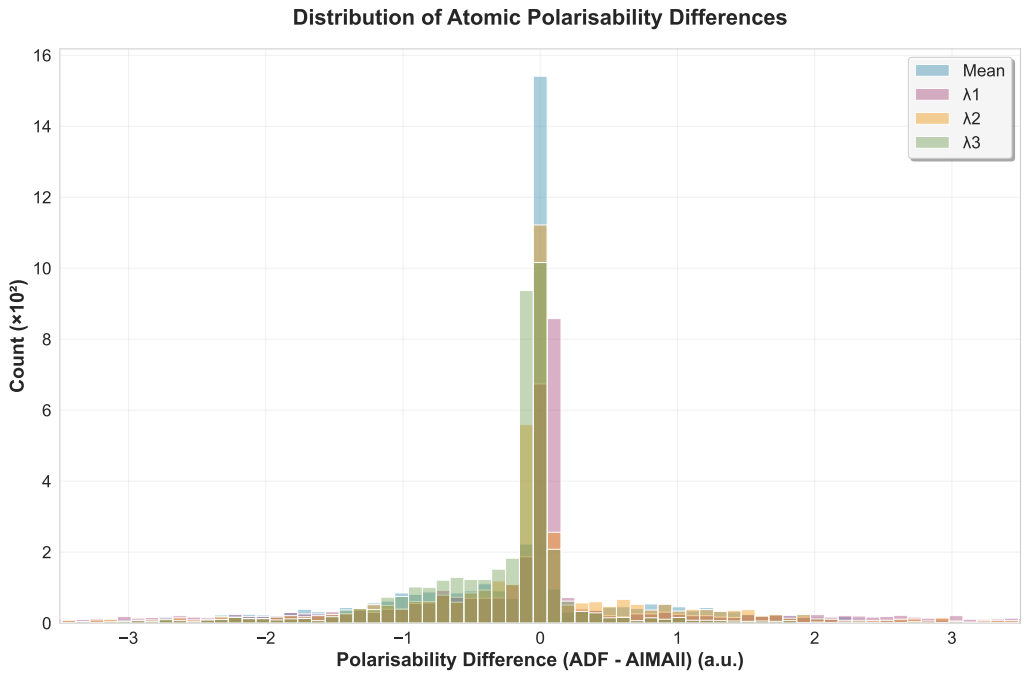


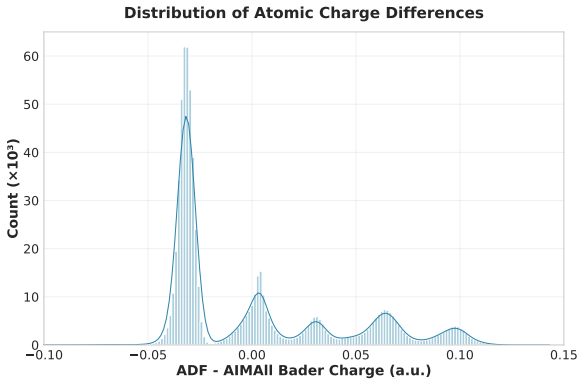
Figure 3.15. Histogram of differences in atomic polarisabilities between ADF and AIMAll.

Given access to a large number of calculations performed with both STOs (via ADF) and GTOs (via Orca and AIMAll), we observed differences in the computed atomic properties. While these discrepancies do not affect the physical interpretation of the results, they provide valuable insight into how the choice of basis set influences the representation of the electron density.

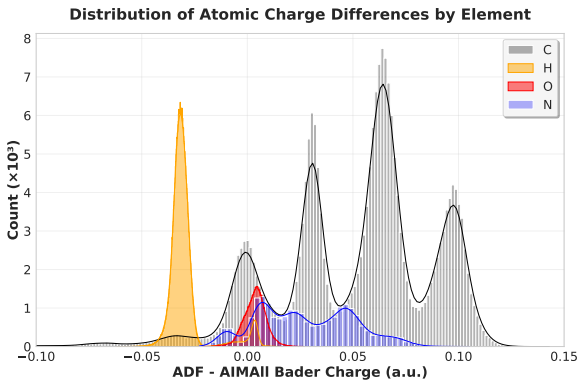
In contrast to the dipole moments and polarisabilities, the differences in atomic charges between implementations are more pronounced in terms of dispersion. As shown in Figure 3.16a, while the average for the differences is 0.000 ± 0.043, the overall distribution appears to be a superposition of several normal distributions, To investigate this further, we analysed the data by atom type. Figure 3.16b shows that the distributions for individual elements remain multimodal, although clearer patterns emerge. For example, hydrogen and carbon display two and four distinct peaks, respectively. Average differences for hydrogen, carbon, and oxygen are condensed in Table 3.4.

Table 3.4. Average differences in atomic charges between ADF and AIMAll.

Charge differences for CHON atoms			
Atom	**Average difference**	**Standard Deviation**	**Median**
C	0.0510	0.0362	0.0596
H	−0.0293	0.0094	−0.0313
O	0.0036	0.0052	0.0039
N	0.0277	0.0224	0.0255



(a) Histogram of atomic charge differences across all atoms.



(b) Histogram of charge differences split by atom type.

Figure 3.16. Histograms of atomic charge differences.

This observation prompted a more detailed analysis of the carbon data, grouping atoms according to the number of bonded neighbours. The resulting histogram is shown in Figure 3.17. However, no clear correlation was identified. Even when splitting the data by hybridisation state (approximated by the number of bonds), the distributions still appeared as a superposition of several normal distributions. Furthermore, as illustrated in Figure 3.18, no consistent trend was found with respect to basis set size. Moving from double- to triple- or quadruple-ζ basis sets produced changes in opposite directions: both the double- and quadruple-ζ sets showed a shift in the same direction, whereas the triple-ζ set exhibited a shift in the opposite direction, suggesting a non-monotonic relationship between basis set size and the observed differences.

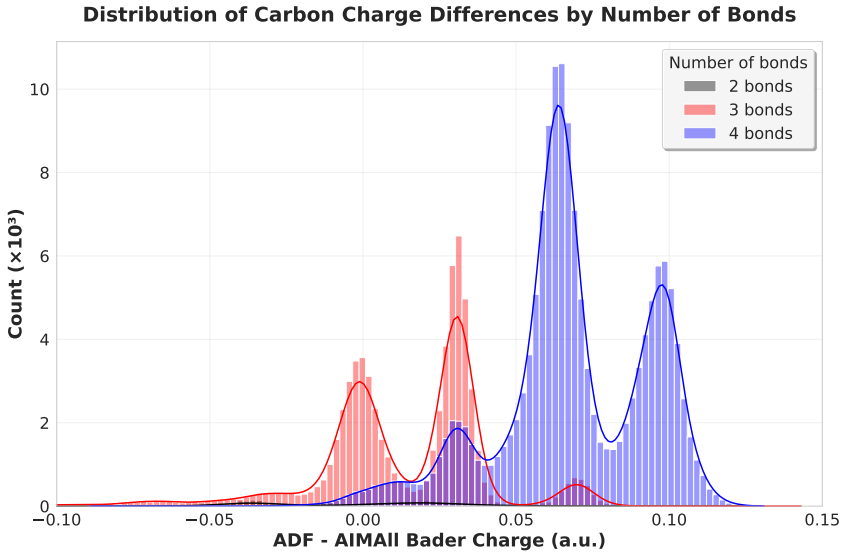


Figure 3.17. Histogram of charge differences for carbon atoms, grouped by number of bonded neighbours.

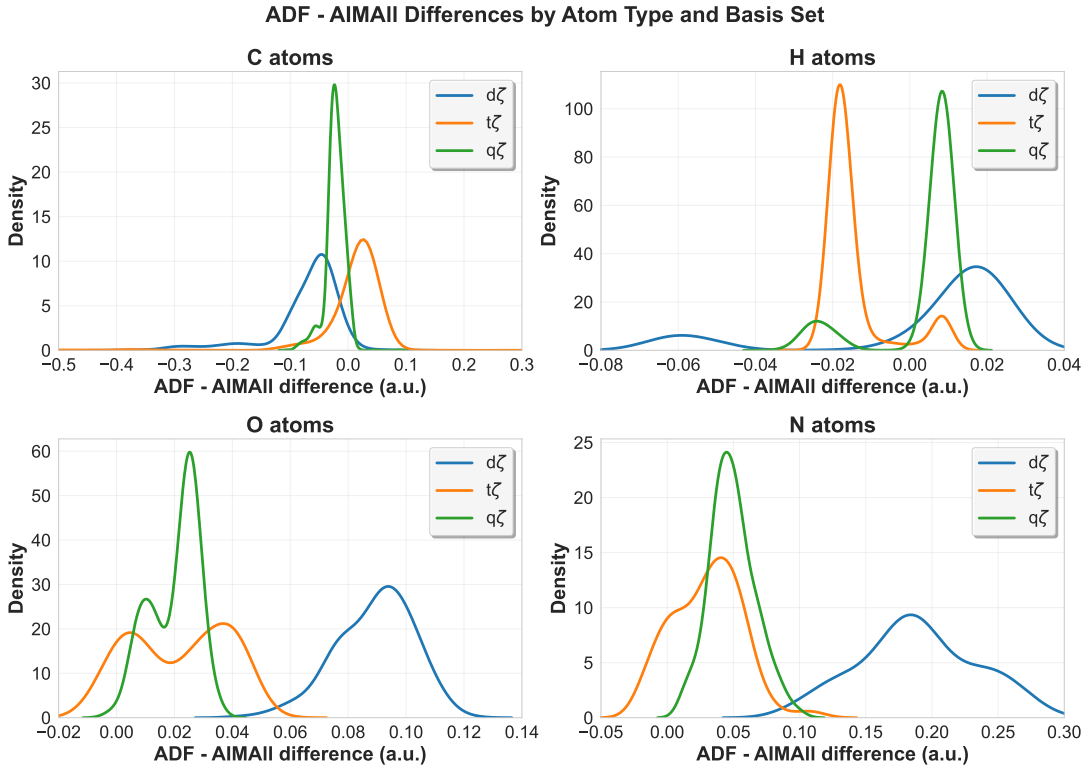


Figure 3.18. Histograms of charge differences for carbon atoms, grouped by basis set type.

Chapter

4 Nucleophilicity Prediction with QTAIM and Conceptual DFT

In this chapter, we investigate nucleophilicity prediction by combining QTAIM and Conceptual DFT. Our approach integrates these frameworks with elements of statistical mechanics to offer a broader perspective on chemical reactivity.

The computational tools developed here are closely related to those described in the following chapter. This reflects our intention to provide a robust and automated workflow that connects QTAIM and CDFT analyses in a practical and reproducible way to analyse chemical reactivity. While QTAIM is fundamentally temperature-independent, Conceptual DFT, particularly in its grand canonical formulation, allows us to consider temperature effects explicitly —although, of course, temperature remains a macroscopic parameter, rather than an external perturbation applied directly to a quantum system—. Both approaches ultimately rely on output from DFT calculations, where perturbatoins such as solvent effects can be incorporated. These, in turn, influence the resulting QTAIM and CDFT properties.

To introduce further chemical realism, it is important to consider the presence of multiple conformers in solution. Our workflow addresses this by sampling geometries for each system as a function of temperature, thus capturing the diversity of conformational behaviour encountered experimentally.

To manage this complexity, we have developed a PYTHON workflow that leverages PLAMS for connectivity to AMS engines. As illustrated in Figure 4.1, the workflow is initiated with plain text input files defining the molecules of interest in SMILES format, along with calculation parameters such as basis set, DFT functional, DFTB method, solvent, temperature, and energy thresholds. All computed data are stored automatically in a SQLite database, facilitating systematic analysis and reproducibility. The same infrastructure underpins our investigations of nucleophilicity, as well as the entropy calculations described in Chapter 5.

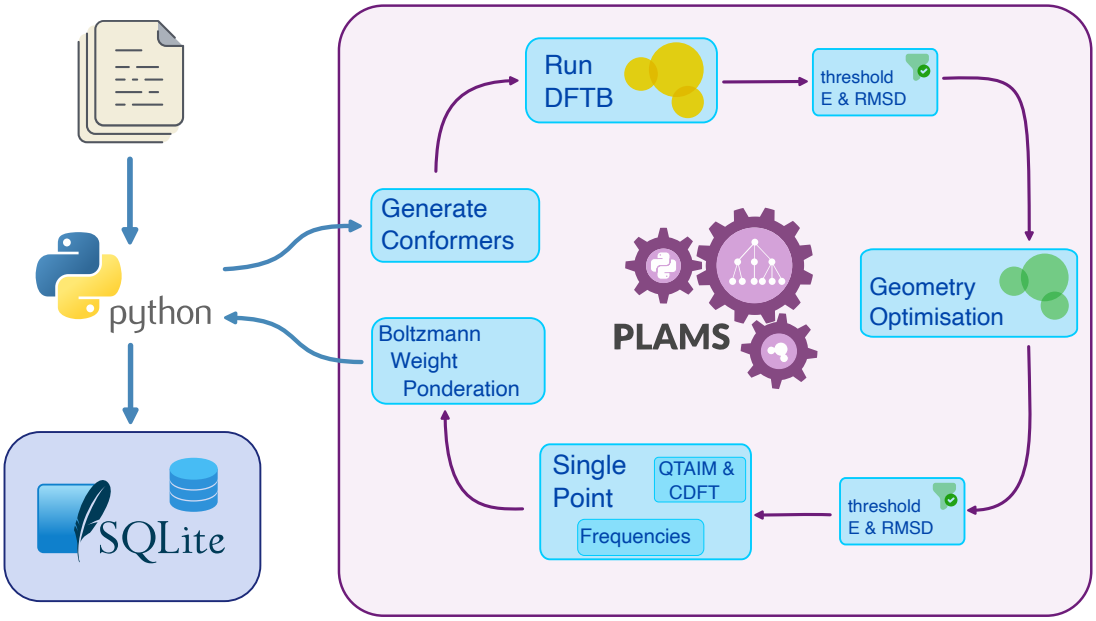


Figure 4.1. Automated workflow developed for high-throughput calculations. The user provides input files containing the systems of interest and calculation parameters. All computed data are stored in a SQLite database for subsequent analysis.

The nucleophilicity project presented in this chapter is ongoing, and a journal submission is in preparation. The latest version of the manuscript, prior to submission, is included in the following pages.

On the prediction of Mayr's nitrogen nucleophilicity parameter by a mixed conceptual density functional theory and atoms-in-molecules approach

Victoria Castor-Villegas, Vincent Tognetti,* Laurent Joubert*

Institut CARMEN UMR6064 (ex lab. COBRA), Université de Rouen, INSA Rouen, CNRS, 1 Rue Tesnière, Mont-Saint-Aignan, 76821 Cedex France.

Abstract

In this work, we propose a robust theoretical framework to predict experimental nucleophilicity values of nitrogen-containing molecules by combining quantum chemical descriptors derived from conceptual density functional theory and the quantum theory of atoms-in-molecules with machine learning algorithms. We assess the performances of several models —including support vector regression, Gaussian process regression, and gradient boosting decision trees— on Mayr's reactivity database, a widely recognised reference for experimental reactivity data based on kinetics measurements, acting on both global and atomic properties. Our results demonstrate strong predictive performance, highlighting the effectiveness of integrating quantum chemical descriptors with machine learning techniques for modelling nucleophilicity. Beyond numerical predictions, our approach also offers insight into the underlying chemical families, providing a richer characterisation of nucleophilic behaviour and enlightening the importance of neighbouring atoms to account for the reactivity of a given atomic site.

Keywords: nucleophilicity, conceptual density functional theory, quantum theory of atoms-in-molecules, machine learning, chemical reactivity.

Corresponding authors: vincent.tognetti@univ-rouen.fr, laurent.joubert@univ-rouen.fr

1 Introduction

The concept of nucleophilicity, first introduced by Ingold [1], constitutes a cornerstone for interpreting reaction mechanisms and guiding synthetic strategies in organic chemistry. It characterises the intrinsic propensity of a molecule to donate electrons in chemical reactions, directly influencing reaction kinetics. Despite its fundamental importance, accurately quantifying nucleophilicity remains challenging due to its intricate dependence on molecular structure, electronic effects, solvation, and reaction conditions.

Fundamentally, reactivity is a kinetic phenomenon defined by the tendency of a species to undergo a chemical reaction, with higher reactivity corresponding to a larger rate constant for a specified elementary reaction [2]. Quantitative reactivity scales, such as those developed by Mayr and co-workers, provide robust frameworks by correlating experimentally determined kinetic parameters to each individual reactants, thus facilitating comparative assessments of nucleophiles and electrophiles [3, 4]. More specifically, Mayr's double-scale approach, given by

$$\log(k_{20^\circ C}) = S_N(N_{Mayr} + E_{Mayr}), \tag{1}$$

quantitatively relates nucleophilicity N_{Mayr} and electrophilicity E_{Mayr} indices to reaction rates measured at 20°C by various spectroscopic experiments (while S_N is the nucleophile-specific sensitivity parameter, which will not be studied here). This strategy broadened the applicability of kinetic studies significantly, ensuring measurable reactions within practical rate limits (10^{-5} to 10^{8} $\mathrm{M^{-1}s^{-1}}$), but it is actually far from being straightforward, requiring refined techniques and non-negligible experimental time. For such reasons it quickly captured the attention of computational chemists [4–6].

Indeed, quantum chemical methods offer two notable advantages: firstly, it may accelerate research through computational efficiency and ease of automation, surpassing experimental methods in terms of speed and resource economy; secondly, it may provide deeper mechanistic insights by uncovering electronic and structural details that are often inaccessible experimentally [7]. Theoretical predictions frequently make use of the Eyring equation, which relates calculated activation free energies in solution ($\Delta G^{\dagger}_{\mathrm{sol}}$) to experimentally determined rate constants k [8]:

$$k(T) = \kappa \frac{k_B T}{h c^{\circ}_{\mathrm{sol}}} \exp\left(-\frac{\Delta G^{\dagger}_{\mathrm{sol}}}{RT}\right), \tag{2}$$

where key physical constants and standard-state conditions are explicitly defined and where κ is the transmission coefficient. Nonetheless, achieving accurate predictions of $\log(k)$ using quantum chemical methods remains a formidable challenge, described metaphorically as the "jigsaw puzzle of quantum chemistry", primarily due to computational approximations inherent in electronic structure methods and solvation models, complicating the attainment of chemical accuracy (commonly defined within 1 kcal $\mathrm{mol^{-1}}$) [7, 9, 10].

From the methodological point of view, substantial research has been dedicated to evaluating various exchange-correlation (XC) functionals for predicting kinetic parameters relevant to nucleophilicity. Hybrid functionals such as B3LYP, M06-2X, and ωB97X-D have frequently been employed due to their balanced treatment of electron exchange and correlation, significantly impacting the accuracy of computed reaction barriers and energetics [11–16]. These functionals generally offer improved performance over standard Generalised Gradient Approximation (GGA) functionals, especially for reaction energies and barrier heights, due to the inclusion of exact exchange. Double-hybrid functionals, which combine exact exchange and perturbation theory correlation (e.g., B2PLYP), represent a further step towards achieving higher accuracy by explicitly accounting for electron correlation effects more comprehensively, though at a higher computational cost [17].

Besides this electronic perspective, thermal and entropic contributions are other crucial aspects influencing kinetic predictions. Typically, theoretical calculations approximate these contributions at standard conditions of 298.15 K (25°C), even though Mayr's reactivity scale defines kinetic parameters at 20°C. This slight discrepancy introduces additional complexity and potential inaccuracies [12–16]. While many researchers adhere to this standard approximation due to computational convenience, Wang et al. notably employed the exact reference temperature of 20°C, aligning directly with Mayr's experimental framework, thereby improving the direct comparability between theoretical predictions and experimental data [11].

Solvation effects constitute another significant source of complexity (Mayr has regularly emphasized that N_{Mayr} and S_N are solvent-dependent), critically affecting the reaction environment and thus reaction kinetics. For instance, the N_{Mayr} value for the 1-methyl-imidazole has beend found equal to 11.9 in acetonitrile and to 9.9 in water solution. To address this point, theoretical studies have predominantly employed implicit solvation models, such as Polarizable Continuum Models (PCM) [18], due to their computational efficiency and reasonable accuracy for capturing bulk solvent effects [11, 13–17].

Nonetheless, recent studies highlight the limitations of purely implicit treatments for two main reasons: *i)* the accurate treatment of entropy usually computed from ideal gas formulas (see, for instance our recent work on the topic [19]), *ii)* in the presence of specific solvent-solute interactions (for instance hydrogen bonding) that significantly influence reaction pathways. Such limitations have prompted increasing interest in explicit solvation models or hybrid explicit-implicit approaches, which can better capture these specific interactions, albeit at increased computational expense [14–16].

Moreover, accurately accounting for conformational complexity remains essential yet challenging. Reactants and transition states often exhibit multiple accessible conformations, each associated with distinct free energies that collectively influence the reaction barrier and overall rate constants. Numerous recent studies have thoroughly explored the conformational space using robust molecular mechanics force fields such as MMFF94x and OPLS3, identifying the lowest-energy conformers to represent reaction intermediates and transition

states in subsequent quantum chemical calculations [12, 14–16].

However, selecting only the global minimum conformation neglects energetically close conformations that could significantly contribute at equilibrium. A more rigorous approach involves population-weighted averaging of multiple conformations, considering Boltzmann distributions to yield more accurate free energy profiles and kinetic predictions [7]. Another method is to perform molecular dynamics (MD) simulations [20] that obviously considerably increase the computational time. MD is also the method of choice to estimate the transmission coefficient, but many trajectories are then necessary to evaluate the recrossing even probability with a low statistical uncertainty.

From all the previous points, it clearly appears that accurately computing reaction rates in solution in order to theoretically predict Mayr's parameters remains an elusive tasks owing to the inherent complexity that governs even so "simple" chemical reactions such as nucleophilic additions. One thus then wonders whether this should be achieved using another paradigm, namely Machine learning (ML), which is designed for dealing with complexity issues. ML techniques have actually emerged as powerful complements to conventional quantum chemical approaches, further enhancing predictive accuracy and providing novel interpretative insights into reactivity trends.

Nonlinear ML methods, including neural networks (NNs), Gaussian process Regression (GPR), and decision-tree-based algorithms like Gradient Boosting Decision Trees (GBDT), have demonstrated considerable potential when trained on comprehensive datasets comprising diverse chemical descriptors and experimentally validated kinetic parameters [21, 22]. While earlier approaches relying on Support Vector Machines (SVMs) yielded comparatively poor results due to limitations in capturing highly nonlinear and complex relationships, recent advances in feature selection and model optimisation have significantly improved the reliability and interpretability of ML-driven kinetic predictions [23]. The field is so active that many papers are regularly published on this topic [24–26], so that we do not consider here to make an extensive review on the use of ML to investigate chemical reactivity.

In the present paper, less ambitiously, we intend to show that the general method that we developed and applied (within linear and non-linear ML approaches) to predict electrophilicity using quantum descriptors [27–29] can be efficiently used to predict nucleophilicity across structurally and electronically diverse nucleophile classes. The peculiarity of our approach is that it is based on the consistent combination of two theories that shares the same basic ingredient, namely Conceptual Density Functional Theory (C-DFT) [30, 31] and the Quantum Theory of Atoms-in-Molecules (QTAIM) [32, 33], both based on the electron density function $\rho(\vec{r})$. A special care will also be devoted to the chemical interpretation of the features involved in the most promising models.

For instance, in a seminal paper, Orlandi and co-workers have critically assessed various molecular descriptors, emphasising that no single parameter —such as the energy of the Highest Occupied Molecular Orbital (HOMO)— can comprehensively capture nucleophilicity. They advocated for multidimensional regression models integrating electronic, steric, and solvation descriptors,

wherein each effect is explicitly represented by tailored descriptors to achieve reliable predictive accuracy [34, 35]. Importantly, they also calculated the protonated products, as molecular models of the results of a nucleophilic attack, thereby probing how post-attack stabilisation influences the observed rate constants. Nonetheless, consistently quantifying steric effects across chemically distinct nucleophiles, such as olefins versus N-heterocyclic carbenes or amines, remains problematic due to intrinsic structural differences [34].

Moreover, correlations between nucleophilicity and thermodynamic parameters such as experimental pK_a values further highlighted the intricate interplay between kinetic and thermodynamic properties. These correlations are nuanced and significantly modulated by solvent effects and steric hindrance, complicating attempts to derive straightforward quantitative relationships [36–38]. Such complexities underscore the need for comprehensive, multifaceted computational approaches to reliably predict nucleophilicity indices.

To address the inherent complexities and uncertainties in nucleophilicity predictions, recent advancements have introduced uncertainty quantification (UQ) methodologies into Mayr's reactivity. Proppe et al. demonstrated that integrating UQ allows computational chemists to report theoretically calculated reaction rates ($\log k$) in a format directly analogous to experimental measurements, namely as expectation values accompanied by corresponding deviations or uncertainties [39].

Such an approach significantly enhances the transparency and reliability of theoretical predictions, explicitly acknowledging the limitations and approximations intrinsic to quantum chemical methods. Moreover, incorporating uncertainty estimates not only strengthened confidence in predicted kinetic parameters but also opened novel avenues for systematically benchmarking different computational approaches against experimental datasets, even in cases where experimental data is sparse or currently unavailable [40, 41].

The recent work by Liu and colleagues further represents a substantial advancement in the field by compiling the most extensive dataset to date for predicting Mayr's nucleophilicity (N_{Mayr}) and electrophilicity (E_{Mayr}) parameters using machine learning methods. Their dataset comprises 1115 nucleophilicity parameters alongside 285 electrophilicity parameters, encompassing chemically diverse compounds with varied nucleophilic reaction centres [23].

Recognising the critical role of solvent environments in influencing nucleophilicity, their comprehensive approach incorporated multiple sophisticated solvent descriptors. Specifically, they included solvent parameters such as Reichardt's, Laurence, Kamlet-Abboud-Taft, Catalán and Hansen parameters [42–48]. These diverse features collectively capture various physicochemical solvent characteristics, significantly improving the predictive capabilities and interpretability of ML models.

In their methodological approach, Liu et al. also systematically evaluated a broad range of molecular descriptors using Random Forest (RF) algorithms to identify the most informative descriptors for accurate predictions of nucleophilicity indices. However, their analysis also highlighted critical limitations associated with certain ML approaches, such as SVR, which provided relatively

poor predictive performance compared to other methods. This emphasises the importance of carefully selecting appropriate ML techniques and underscores the necessity of rigorous descriptor selection processes to ensure optimal model accuracy and interpretability [23].

The benefits of an accurate and efficient model are even larger. Indeed, despite significant methodological advances, the experimental determination of nucleophilicity indices remains challenging for specific classes of highly reactive or structurally unstable nucleophiles —such as tertiary enamines— which highlights a crucial area where computational predictions can offer substantial benefits, particularly in scenarios where direct kinetic measurements are experimentally impractical or impossible [49–52].

In order to adress some of this issues, our paper will be thus divided as follows. In the next section, we provide an overview of the quantum chemical descriptors, the ML techniques we used, and the computational details. Then the database will be presented, before reporting the results and discussing them.

2 Theory

2.1 Quantum Chemical Descriptors

The C-DFT descriptors can be classified into three categories: *global* (one value for the whole molecule), *local* (functions $a(\vec{r})$ that depend only on one position in real-space, allowing for discussing regioselectivity), *non-local* (functions $b(\vec{r},\vec{r}',...)$ that depend on more than one 3D-space variables, also referred as kernels) ones. All of them can be also categorised as being either *basic* or *composite*, the first class corresponding to descriptors that are directly expressed as derivatives of the electronic energy E_e with respect to the two variables describing the system in the chosen ensemble representation [53]. Conversely, composite descriptors are obtained by combination of basic descriptors.

For instance, in the canonical ensemble, E_e is expressed as a function of the number of electrons N and as a functional of the external potential, $v(\vec{r})$, generated by the nuclei. The electronic chemical potential μ and molecular hardness η are basic global descriptors defined by

$$\mu = \left(\frac{\partial E_e[N, v(\vec{r})]}{\partial N}\right)_{v(\vec{r})}, \tag{3}$$

$$\eta = \left(\frac{\partial^2 E_e[N, v(\vec{r})]}{\partial N^2}\right)_{v(\vec{r})} \tag{4}$$

while Chattaraj's nucleophilicity index [54] which is a composite global descriptor derived from these quantities is defined as:

$$N_C = \frac{2\eta}{\mu^2}. \tag{5}$$

Still in the canonical ensemble, important local descriptors are the Fukui functions, which quantify the change in electron density upon addition or removal of an electron, thereby pinpointing the most nucleophilic and electrophilic sites in a molecule.

$$f^{\pm}(\vec{r}) = \frac{\partial}{\partial N}\left(\frac{\delta E_e}{\delta v(\vec{r})}\right) \begin{matrix} \partial N > 0, \\ \partial N < 0 \end{matrix} \tag{6}$$

Originally developed to remedy the fixed-orbital picture of FMO theory [55], these functions admit a simple approximation in terms of the HOMO and LUMO when orbital relaxation is neglected:

$$f^{+}(\vec{r}) \approx |\phi_{\mathrm{LUMO}}(\vec{r})|^2, f^{-}(\vec{r}) \approx |\phi_{\mathrm{HOMO}}(\vec{r})|^2. \tag{7}$$

Additionally, the non-local linear response kernel $\chi(\vec{r}, \vec{r}')$, extensively studied in C-DFT in the last years, describes how the electron density at position $\vec{r}$ responds to an external perturbation at other position $\vec{r}'$, revealing various chemical phenomena, including electron delocalization, inductive and mesomeric effects, and aromaticity reactivity[56]:

$$\chi(\vec{r}, \vec{r}') = \left(\frac{\delta^2 E_e[N, v(\vec{r})]}{\delta v(\vec{r}) \delta v(\vec{r}')}\right)_N \tag{8}$$

Then, in order to condense the information embodied in local or non-local descriptors, QTAIM [32, 33] has been used. Condensation can be seen as information coarse-graining, going from an infinite amount of data to a finite one (here, related to the number of atoms). It also allows for translating the information embodied by such descriptors into the usual chemical language based on "sites". In a nutshell, QTAIM partitions, based on the field lines of the electron density gradient vector, the 3D-real space in non-overlapping domains that each corresponds (in the absence of non-nuclear attractors) to a topological atomic basin Ω_A. *Atomic* and *di-atomic values* features can thus be computed for any local $a(\vec{r})$ or non-local $b(\vec{r}, \vec{r}')$ functions by:

$$a(A) = \int_{\Omega_A} a(\vec{r}) d^3r \tag{9}$$

$$b(A, B) = \int_{\Omega_A} \int_{\Omega_B} b(\vec{r}, \vec{r}') d^3r d^3r' \tag{10}$$

Whereas the canonical ensemble treats the particle number N as the privileged extensive variable the grand-canonical ensemble replaces N by its conjugate, the chemical potential μ, through a Legendre transformation. This operation generates the grand potential $\Omega(\mu, v(\vec{r}))$ and recasts all derived quantities. The transformation is conceptually identical to the shift from the internal energy $U(S, V)$ to the enthalpy $H(S, P)$. In this ensemble, the local softness functions, defined by

$$s^{\pm}(\vec{r}) = f^{\pm}(\vec{r})/\eta, \tag{11}$$

are basic descriptors while they are composite ones in the canonical representation.

Finally, since our interest spans reactivity at defined temperatures, conformational sampling must be considered. For a property P, the following Boltzmann-weighted average $\overline{P}$ across the stable conformers k will be computed:

$$\overline{P} = \frac{\sum_k e^{-\frac{E_k^{SCF}}{RT}} P_k}{\sum_k e^{-\frac{E_k^{SCF}}{RT}}} \tag{12}$$

2.2 Machine Learning

When applied to chemical reactions, ML mainly serves two complementary purposes: *i*) classification to reveal latent structure in the molecular-descriptor space, and *ii*) regression to predict reactivity metrics such as the nucleophilicity parameter N_{Mayr} or the family. The following summary emphasises the algorithms actually employed in this study.

All descriptor matrices were centred and scaled to unit variance. Missing values, present in $< 2\,\%$ of the entries, were neglected (non-neutral species were excluded; see Section 3). Highly correlated features ($> 95\,\%$) were pruned to reduce redundancy, and the remaining variables were ranked via mutual information against the target [57]. The final feature set thus balances information content with model parsimony.

2.2.1 Classification Algorithms

Support Vector Machine (SVM) classifier. For supervised classification we use the C-SVC formulation of support vector machines [58], as implemented in `scikit-learn`'s `svm.SVC`, which wraps LIBSVM [59]. Given labelled data $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ with $y_i \in \{-1, +1\}$, the soft-margin primal problem is:

$$\min_{\mathbf{w},\, b,\, \boldsymbol{\xi}} \ \tfrac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n} \xi_i \quad \text{such that} \quad y_i\big(\mathbf{w}^\top \phi(\mathbf{x}_i) + b\big) \geq 1 - \xi_i,\ \xi_i \geq 0. \tag{13}$$

Kernelisation yields the dual

$$\max_{\boldsymbol{\alpha}} \ \sum_{i=1}^{n} \alpha_i - \tfrac{1}{2}\sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \quad \text{such that} \quad 0 \leq \alpha_i \leq C,\ \sum_{i=1}^{n} \alpha_i y_i = 0. \tag{14}$$

The decision function is

$$f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) + b. \tag{15}$$

We tune C and kernel hyperparameters by cross-validation.
Note: Support Vector *Clustering* [60] is an unsupervised method and is not used here.

Choice of kernel. The kernel $K(\mathbf{x}, \mathbf{x}')$ determines the geometry of the feature space and therefore the shape of the decision boundary. We consider:

- **Linear kernel**: $K(\mathbf{x}, \mathbf{x}') = \mathbf{x}^\top \mathbf{x}'$ Equivalent to no transformation; effective when data are linearly separable.
- **Radial Basis Function (RBF) kernel**: $K(\mathbf{x}, \mathbf{x}') = \exp\big(-\gamma \|\mathbf{x} - \mathbf{x}'\|^2\big)$ Maps input to an infinite-dimensional feature space. Suitable for nonlinearly separable data; parameter $\gamma > 0$ controls the radius of influence of support vectors.

2.2.2 Regression Algorithms

Random Forest Regressor (RFR). A RF is an ensemble of T decision-tree regressors $\{h_t(\vec{x})\}_{t=1}^{T}$ grown on independent bootstrap samples of the training set [61]. At each split, a random subset of $m < p$ features is considered, decorrelating trees and reducing variance. The final prediction is the arithmetic mean

$$\hat{y}(\vec{x}) = \frac{1}{T} \sum_{t=1}^{T} h_t(\vec{x}). \tag{16}$$

Individual trees minimise the squared-error impurity $\mathcal{I}(S) = \sum_{(\vec{x}_i, y_i) \in S} (y_i - \bar{y}_S)^2$ within every leaf S, ensuring piecewise-constant approximations that capture nonlinear interactions without explicit functional assumptions.

Support Vector Regression (SVR). Employing the same kernel $K(\mathbf{x}, \mathbf{x}') = \langle \varphi(\mathbf{x}), \varphi(\mathbf{x}') \rangle$ used above for *classification*, ε-SVR seeks a function $f(\mathbf{x}) = \mathbf{w}^\top \varphi(\mathbf{x}) + b$ that deviates from targets by at most ε while penalising slack variables ξ_i, ξ_i^* outside the tube [62]:

$$\min_{\vec{w}, b, \xi_i, \xi_i^*} \ \frac{1}{2}\|\vec{w}\|^2 + C \sum_{i=1}^{n} (\xi_i + \xi_i^*) \quad \text{subject to} \quad \big|y_i - f(\vec{x}_i)\big| \le \varepsilon + \xi_i, \ \xi_i, \xi_i^* \ge 0. \tag{17}$$

The representer theorem yields $f(\vec{x}) = \sum_{i=1}^{n} (\alpha_i - \alpha_i^*) K(\mathbf{x}_i, \mathbf{x}) + b$, where α_i, α_i^* are Lagrange multipliers obtained from the dual quadratic programme.

Feed-Forward Neural Network (FFNN). A fully connected network defines a hierarchical composition of affine maps and element-wise nonlinearities σ:

$$\hat{y}(\vec{x}) = \vec{W}^{(L)} \sigma\big(\vec{W}^{(L-1)} \sigma(\cdots \sigma(\vec{W}^{(1)} \vec{x} + \vec{b}^{(1)}) + \vec{b}^{(L-1)})\big) + \vec{b}^{(L)}, \tag{18}$$

where $\{\vec{W}^{(\ell)}, \vec{b}^{(\ell)}\}_{\ell=1}^{L}$ are learned parameters. Training minimises the mean-squared error $\mathcal{L} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}(\vec{x}_i))^2$ via stochastic gradient descent with back-propagation, optionally regularised by dropout or ℓ_2 weight decay to curb overfitting.

Gaussian Process Regression (GPR). We place a Gaussian process prior $f \sim \mathcal{GP}\big(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')\big)$ with $m \equiv 0$ [63]. For training inputs $X = [\mathbf{x}_1, \ldots, \mathbf{x}_n]^\top$ and targets $\mathbf{y} \in \mathbb{R}^n$ observed with i.i.d. Gaussian noise variance σ^2, define the kernel matrix $\mathbf{K} \in \mathbb{R}^{n \times n}$ by $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$ and $\mathbf{k}_* = [\, k(\mathbf{x}_1, \mathbf{x}_*), \ldots, k(\mathbf{x}_n, \mathbf{x}_*)\,]^\top$. Then the predictive distribution at $\mathbf{x}_*$ is Gaussian with

$$\mu_* = \mathbf{k}_*^\top \big(\mathbf{K} + \sigma^2 \mathbf{I}\big)^{-1} \mathbf{y}, \tag{19}$$

$$\sigma_*^2 = k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}_*^\top \big(\mathbf{K} + \sigma^2 \mathbf{I}\big)^{-1} \mathbf{k}_*. \tag{20}$$

Kernel hyperparameters and σ^2 are estimated by maximising the log marginal likelihood. We report point predictions μ_* with credible intervals $\pm 2\sigma_*$.

All algorithms were implemented using `scikit-learn` [64] (`v` 1.5) with default settings unless stated otherwise. Source code and hyperparameter grids are provided in the Supporting Information.

2.3 Computational Details

All DFT calculations were performed with the ADF software [65, 66] using the PBE exchange-correlation functional [67] with the Grimme's dispersion correction with Becke-Johnson damping function (D3-BJDAMP) [68] and a triple-ζ basis set (TZP), with no relativistic effects added. We used the default settings for the numerical quality and integration grid, but we did not use the frozen core approximation.

The choice for a dispersion-corrected Generalized Gradient Approximation (GGA) functional was mainly governed by its significant trade-off between accuracy and computational time. Even if it is certainly outperformed by more modern approaches such as double hybrids, it is more suited for a large number of calculations (due to the large dataset considered here and to the inclusion of conformational effects). Besides, evaluating the C-DFT and QTAIM features at the best level of theory is not necessary since their computation is intended only to feed the ML models, which are able, in principle, to take into account and to indirectly correct the initial GGA deficiencies.

The automation of the computational workflow was achieved by a Python script, inspired by recipes already available in the Python Library for Automating Molecular Simulations (PLAMS) [69]. The pipeline is made of the following steps:

i) The RDKit [70] utility generates the 3D coordinates from the sole Simplified Molecular Input Line Entry System (SMILES) code and generates a set of molecular geometries to explore the conformational space;

ii) The obtained geometries are optimised at the Density-Functional based tight-binding (DFTB) level of theory, and the conformers are selected based on the lowest energy conformers, with a threshold of 5.0 kcal/mol in terms of relative SCF energies and 3.0 Å for the structural Root-mean-Square Deviation (RMSD);

iii) The selected conformers are then optimised at the final DFT level of theory, followed by a second conformer refinement, now with a threshold of 2.5

kcal/mol for energy and 1.5 Å for RMSD;

iv) The retained conformers are then used to calculate the QTAIM and C-DFT descriptors by means of single-point calculations;

v) All descriptors are then Boltzmann-averaged to afford the mean values according to eq. 12. For the sake of comparison, we also saved the numerical values for all descriptors corresponding to the most stable conformer.

Importantly, all descriptors used in this work were computed solely for the reactant structures, without introducing any modified or protonated or other activated forms to mimic the products of nucleophilic attack. This intrinsic analysis thus focuses exclusively on the electronic and topological information intrinsically present in the reactant molecule, as captured through QTAIM and C-DFT. By avoiding structural modifications or reaction intermediates, the models aim to predict reactivity from the chemical ground state, with the only additional layer being the consideration of conformers accessible at 20°C under solvation conditions.

Solvent effects were described using the implicit Conductor-like Screening Model (COSMO) [71, 72], with default parameters applied in most cases. In a few instances involving solvent mixtures —such as MeOH/MeCN (45/55 or 91/9), or EtOH/MeCN (91/9)— the relevant parameters (solvent radii and relative permittivity) were explicitly specified, withvalues computed as weighted averages based on the composition of the pure solvents.

Descriptors used for the analysis were provided by the ADF software with the *Analysis Level* set to *Full*, which enabled the calculation of global and local descriptors based on C-DFT. The computed descriptors include the electronic chemical potential (μ), electronegativity (χ), hardness (η), softness ($S = 1/\eta$), electrophilicity index ($\omega = 1/N_C$), nucleofuge, electrofuge, electrodonating power, and electroaccepting power [73–75], all of them defined as global descriptors.

For the canonical atomic descriptors, we have the Fukui functions ($f^{\pm}$, f^0), and dual descriptor (DD) $f^{(2)}$ [76] in the canonical ensemble. In the grand canonical ensemble, $s^{\pm}$ and s^0. We also computed composite Fukui functions such as $\mu^+ f^+$, $\mu^- f^-$, μf^+, composite DDs such as $\omega f^{(2)}$, $S f^{(2)}$, $S^2 f^{(2)}$, supplemented by condensed local electrophilicity and nucleophilicity. For non-local descriptors, we considered the condensed linear response $\chi(A, B)$. For the sake of simplicity, $\chi(A)$ denotes the value when A and B refer to the same atom. Additionally, we computed the recently revivified P and $P f^{(2)}$ [53], not provided in the ADF output, where $P = \gamma^{-1}$.

Finally, atomic charges, denoted q, were evaluated using both the QTAIM and Hirshfeld partitioning schemes.

The ML models were implemented using `scikit-learn` in Python. The data were split into 80 % for training and 20 % for validation with the corresponding functions in `scikit-learn`, for all cases the splitting was done only one time with a random state for the split. Model optimization was performed using the minimization routine in `scipy`, with the ML model as the objective function and the L-BFGS-B method.

3 The Molecular Database

For the present work, we employed the full set of Mayr's reported nucleophilicity values for nitrogen-containing compounds for neutral molecules (indeed, modelling cations or anions in solution is not straightforward since it may require considering the counterion or the use of an explicit solvent layer). After this filtering process, a total of 244 systems containing nucleophilic nitrogen atoms were retained from the families (named following Mayr's typology): *i*) aliphatic amines, *ii*) aromatic amines, *iii*) amidines and imines, *iv*) amino acids, *v*) azoles and azoles anions, *vi*) guanidines, *vii*) hydrazines, hydroxylamines, etc., *viii*) imidazolines and related compounds, *ix*) isothioureas, *x*) pyridines, quinolines, etc. *xi*) other N-centered nucleophiles.

The dataset displays a broad chemical diversity, encompassing nearly all common organic functional groups. Notably, several nucleophiles contain multiple nitrogen atoms, as illustrated in Figure 1. In trimethylhydrazine, for instance, the two nitrogen atoms are chemically distinct, and both nucleophilic sites have been experimentally characterised, with separate N_{Mayr} values reported in the database. Conversely, in the guanidine derivative, only the non-cyclic nitrogen has a reported nucleophilicity value; the two cyclic nitrogen atoms, being equivalent in this context, are considered non-reactive. For azoles, a single N_{Mayr} value is given, which represents the reactivity of both nitrogen atoms involved in the nucleophilic attack.

This supports the interpretation of Mayr's scale as a site-specific rather than a purely molecular descriptor. Throughout this work, whenever a nitrogen atom within a multi-centre system lacks an associated N_{Mayr} value, it is treated as non-nucleophilic by default.

From the numerical point of view, the range for nucleophilicity values for the reactive systems spans a large range from 5 to 23, with a notably unimodal Gaussian-like distribution with a maximum frequency near 14 with a standard deviation of 3.2. Besides, we also considered the 1310 nucleophilic systems reported in Mayr's database to analyse if there is any particular trend related with the nuclei. As shown in Figure 2, there is a significant overlap between them, precluding any simple and direct correlation between the atomic number and the nucleophilicity.

There is thus no intrinsic atomic hierarchy: a nitrogen atom can be more or less nucleophilic than a carbon or a hydrogen, depending on its environment. Clearly, the other atoms like H, C, O shows nucleophilicity distributions that are much more spread and of a clear multimodal character (for instance one peak around 5 and another one around 18 for oxygen). A more detailed analysis of these trends is however outside the scope of this paper.

It should be noticed that we take all systems without considering the "star classification system" for any splitting of the data (at variance with some of our previous works). Let us recall that this classification is linked to the number of the reference electrophiles/nucleophiles used for the determination of the reactivity parameter. Hence, it constitutes a kind of evaluation of the reliability/accuracy of the numerical values, which compensates for the lack of

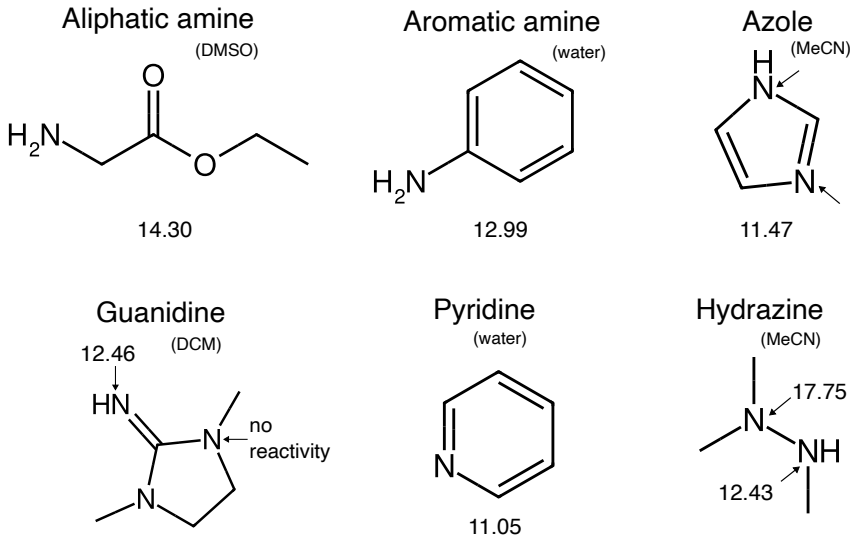


Figure 1: Representative examples of nitrogen-containing nucleophiles. For trimethylhydrazine, distinct N_{Mayr} values are reported for each nitrogen atom. In the guanidine example, only the non-cyclic nitrogen is considered reactive; the two equivalent cyclic nitrogens are treated as non-nucleophilic. In the azole example, a single N_{Mayr} value reflects the concerted involvement of both nitrogen atoms. Experimental solvents are indicated in parentheses.

measured numerical uncertainties. One has thus to keep in mind that one-star systems should be used only to give a good idea about the relative reactivities of strong nucleophiles towards weak electrophiles, and also that two-star systems could be subject to re-evaluation.

4 Results and Discussion

4.1 Discriminating between reactive and unreactive nitrogen atoms

As any chemical feature, reactivity can be discussed from a qualitative and a quantitative perspectives. In ML, the first one can be tackled from the classification point of view, in particular to distinguish between reactive and non-reactive nitrogen atoms in the full dataset. We thus investigated Support Vector Classification (SVC) models (linear and RBF kernels) based on Boltzmann-averaged QTAIM and C-DFT descriptors.

More precisely, in order to identify the most relevant features for classification, the classification procedure exhaustively assessed all pairwise combinations across the full set of computed descriptors. In addition, three-dimensional clas-

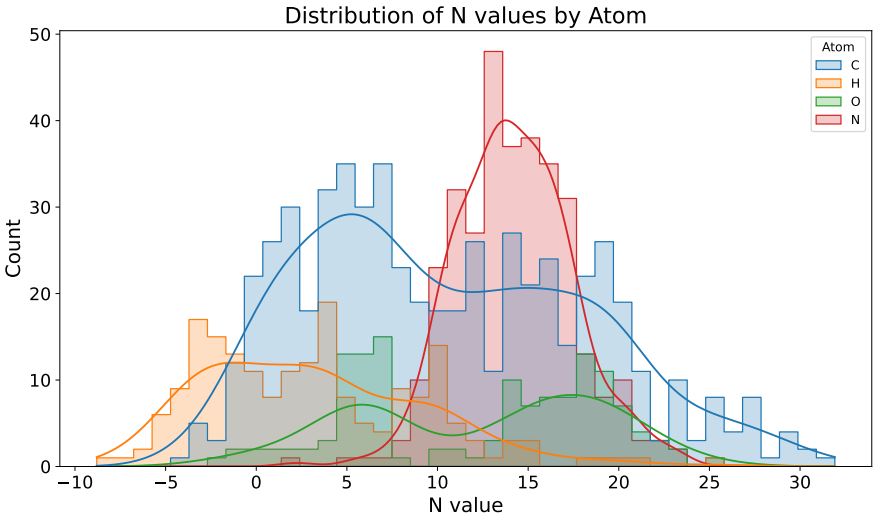


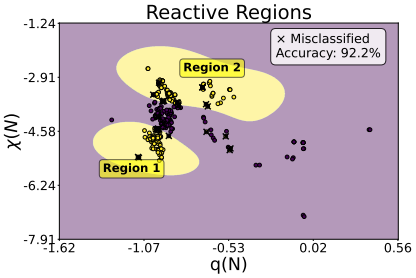
Figure 2: Histogram of the N_{Mayr} values for the systems in Mayr's database.

sification was explored by including a third descriptor, but this approach led to signs of overfitting, with no improvement in classification performance compared to the two-dimensional case.

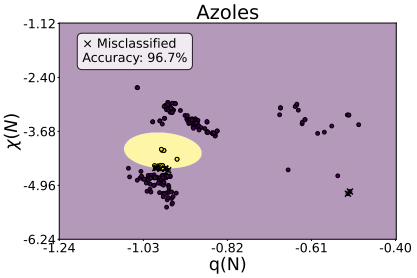
The optimal partition —maximising classification accuracy— is shown in Figure 3a, achieving an overall accuracy of 92.2 % for the full dataset (i.e., no separation between training and test sets was applied in this exploratory classification). The background colours indicate the decision regions obtained from the classification: the purple region corresponds to the non-reactive class, while the yellow region corresponds to the reactive class. Data points are coloured according to their true class labels according to Mayr's experiments: purple for non-reactive sites and yellow for reactive sites. This allows visual comparison between the predictions and the actual values. For instance, a yellow point in the yellow zone corresponds to a site that is experimentally reactive and that is predicted also reactive by the model. Conversely, a purple point in the yellow zone corresponds to a site that is not experimentally reactive but that was erroneously predictive as reactive. Such misclassified sites are represented by a cross.

It is also valuable to use the same approach by restricting it to each of the three dominant chemical families since they are sufficiently populated to allow meaningful statistical evaluation, namely azoles (33 compounds), pyridines and quinolines (56 compounds), and aliphatic amines (104 compounds). The predictive performance further improved, reaching 99.2 % for azoles, 96.3 % for pyridines and quinolines, and 98.9 % for aliphatic amines. These results were obtained using the RBF kernel.

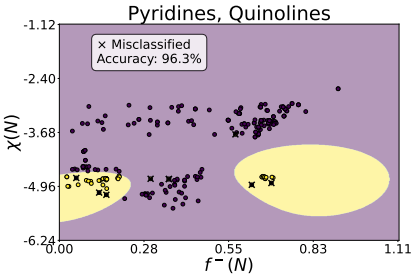
Among the most effective descriptor combinations, we found those involv-



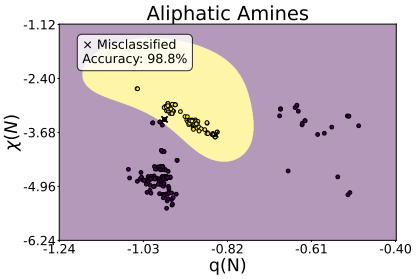
(a) Clasification of the systems in the Mayr's database, with a classification accuracy of 92.2 %.



(b) Azole family, 96.7 %



(c) Pyridines and quinolines, 96.3 %



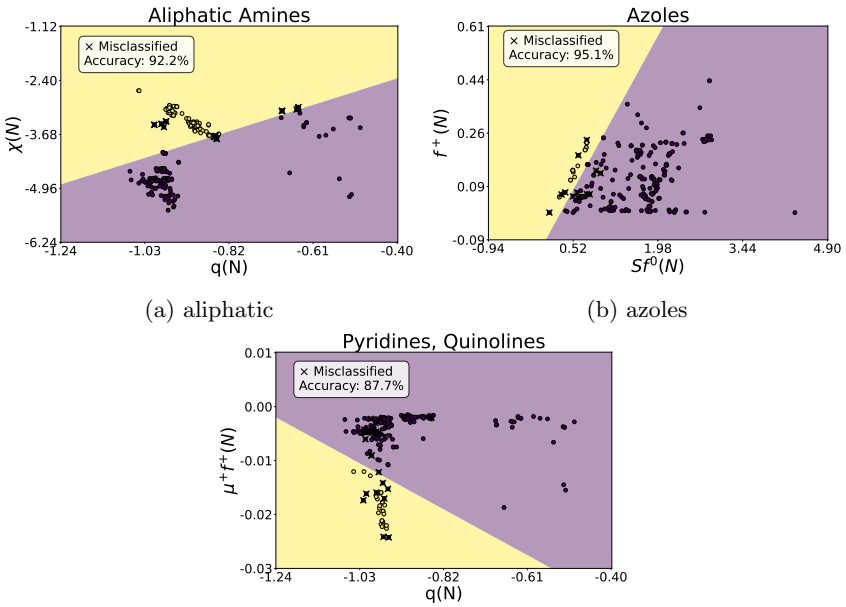
(d) Aliphatic amines, 98.9 %

Figure 3: (a) Classification of all systems, (b) Classification of the Azole family, (c) Classification of the Pyridines and Quinoline families, (d) Classification of the Aliphatic Amines family. Atomic units for all descriptors.

ing the electrostatic potential, condensed linear response descriptors, f^2, $\mu^- f^-$, and f^-. However, the final classification non-linear models reported in Figure 3 are based on the $q(N), \chi(N)$ combination (except for the pyridine family that involves the $q(N)$ and $f^-(N)$). Interestingly, the highest overall accuracy provides also a clear chemical interpretability since it mixes one feature describing charge control ($q(N)$, useful for hard species) and one related to orbital control ($\chi(N)$, well suited to soft species, since high values of χ corresponds to high polarisabilities), this non-linear analysis revealing two well-defined regions associated with reactive and non-reactive nitrogen atoms.

We also tested the linear kernel, but only achieved an overall accuracy of about 95 %, 87 %, 92 %, highlighting the significant non-linear character of the boundaries that separate the chemical families. A visual inspection of the decision regions (Figure 4) further illustrates that the misclassified points are artefacts of the linear assumption rather than genuine overlap between families.

The azole family deserve a particular comment. Indeed, the optimal classification performance was actually obtained when $\chi(N)$ was used in conjunction



(a) aliphatic

(b) azoles

(c) pyridines

Figure 4: Classification of reactive and non-reactive nitrogen sites using the SVC algorithm with the linear kernel.

with $f^+(N)$ that is a descriptor describing electrophilicity and not nucleophilicity, making this model a priori not chemically interpretable. However, one should also notice that the azoles form a compact cluster near zero for this descriptor. This is computationally not satisfying since these values may be more prone to numerical errors. Pleasingly, the already mentioned $q(N), \chi(N)$ combination yielded strong predictive results, and as it is more interpretable, this is the one that we keep for our figure. Indeed, it fails to correctly classify only two compounds: benzotriazole and 1-methylbenzotriazole, which are the only triazoles in the dataset, deviating significantly from the overall trend.

On the other hand, the pyridine and quinoline family remarkably exhibited a clear bifurcation into two distinct clusters, primarily differentiated by the $f^-(N)$ descriptor. One cluster was centred near zero, while the other grouped around a value of approximately 0.74. This separation suggests fundamentally different reactivity mechanisms within this family, despite structural similarity among its members.

Lastly, it should be noted that, although the grand canonical ensemble provides the natural framework for defining C-DFT descriptors in terms of the chemical potential and is often argued to be the appropriate choice, particularly for open or electrochemical systems where particle exchange is relevant [77], some other studies have shown that canonical and grand canonical formu-

lations yield nearly equivalent results in practice [78]. In our case, we find that a canonical description is sufficient to obtain reliable predictive accuracy for the systems under study.

While the use of QTAIM charges combined with the condensed linear response already provides a good classification accuracy of about 92 %, we observed that the description can be further improved by accounting for the contributions of neighbouring atoms. In this approach, the atomic charge and $\chi(A, B)$ are weighted by an exponential decay factor that takes the influence of surrounding atoms into account. For any atomic property P and atom A, it can be achieved by:

$$P_w(A) = P(A) + \sum_{B \neq A} e^{-\alpha_P R_{AB}} q(B), \tag{21}$$

where R_{AB} is the internuclear distance. For instance in the case $P = q$, a negatively charged atom surrounded by several positively charged neighbours will consequently "appear" less negatively charged for another charge approaching it. The previous equation is hence the simplest way to incorporate screening effects. exhibit a reduced effective reactivity. Similarly, for the linear response kernel (and more generally for any kernel):

$$\chi_w(A) = \chi(A) + \sum_{B \neq A} e^{-\alpha_\chi R_{AB}} \chi(A, B), \tag{22}$$

Considering now the $q_w(N), \chi_w(N)$ combination, we optimized the α_q and α_χ decay parameters (a very high value would imply that the neighbouring atoms have only negligible contributions) to maximise the classification accuracy using the SVC approach with the RBF kernel. We obtained a symmetric solution with $\alpha_q = \alpha_\chi = 1.0\mathring{A}^{-1}$. This choice achieved an improved accuracy of 97.5 %. Moreover, the two distinct regions of predicted reactivity identified in the previous analysis were merged into a single unified region that captures the reactivity for all N atoms, as shown in Figure 5.

Noteworthy, it can be added that the influence of the neighbouring atoms for the linear-response kernel can be taken into account using the eigenvalues of the di-atomic QTAIM condensed kernel, following the methodology recently described by Grincourt et al. [79]. Indeed, these eigenvalues can be mapped to each atom by looking at the highest absolute coefficient in the linear expansion giving the corresponding eigenvectors. We are currently working on that point.

4.2 Predicting Mayr's nucleophilicity values

After showing that our approach was successful in determining if a nitrogen atom is reactive or not, we now attempt to predict Mayr's nucleophilicity. We first considered linear regression models based on descriptors such as atomic charges and HOMO energies, which met with limited success. Indeed, these models yielded a mean absolute error (MAE) of approximately 2.5 and exhibited considerable scatter, lacking any consistent trend or correlation. This result

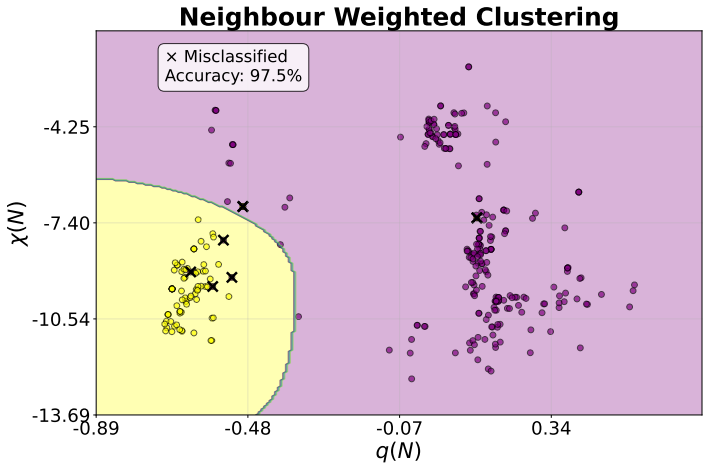


Figure 5: Classification of reactive and non-reactive regions using weighted atomic descriptors for the whole dataset. Atomic units.

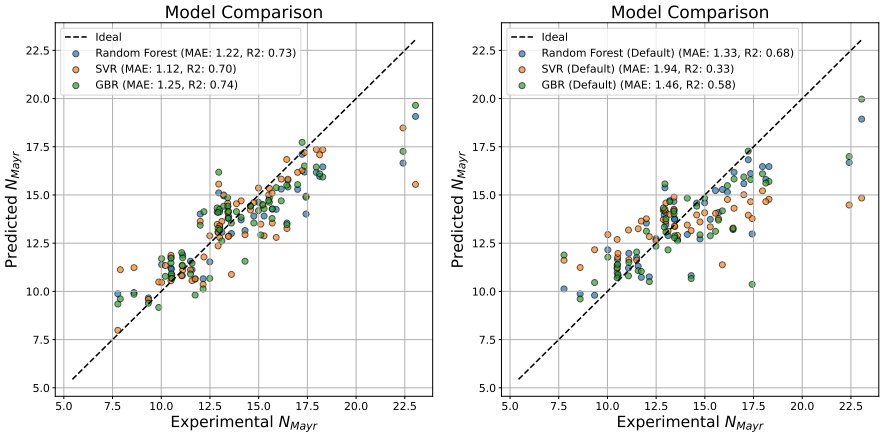
underscores the inadequacy of simple linear relationships in capturing the complexity of nucleophilic behaviour.

To address the limitations observed with traditional regression techniques, advanced ML approaches —including Random Forests, Support Vector Regression (SVR), Gradient Boosted Regression (GBR), and neural networks— were explored. These models significantly improved predictive performance, reducing the mean absolute error (MAE) to approximately 1.2, although some dispersion remained. Nonetheless, the ability of these ML methods to capture underlying chemical trends underscores their value, particularly when applied within chemically meaningful clusters identified earlier.

Figure 6 compares models trained with tuned hyperparameters to their counterparts trained with default settings. Both strategies produce similar error profiles and overall MAE, with hyperparameter optimisation yielding only modest gains (most visibly for SVR). This behaviour suggests that the descriptor space is already informative enough for robust prediction, even without extensive tuning.

A key advantage of the RF models is their feature-importance analysis (Figure 7a), which we used to design compact predictors. For the "small" models trained with default hyperparameters, we retained only the seven most important descriptors identified by RF. In contrast, the tuned models were trained on the full descriptor set to maximise any potential gain from optimisation. To maintain methodological continuity with the classification stage, the reduced set prioritised the $\chi(N)$ which had underpinned our classification analysis and was the descriptor to cut the set, delating also the Hirshfeld charge since it has basically the same weight as the QTAIM charge.

Complementary correlation analysis (Figure 7b) highlighted redundancies



(a) Different ML models (b) Default values of ML libraries

Figure 6: ML

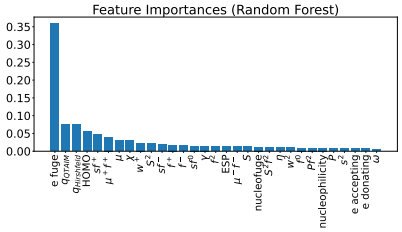
among several descriptors and clarified their relationships with N_{Mayr}. These insights guided the removal of low-contribution or strongly collinear features, streamlining the models without degrading accuracy and thereby reducing computational cost. An interestingly correlation was observed between the descriptors N_{Mayr} and S_N.

Finally, across this dataset, models trained with default hyperparameters were often competitive with their tuned counterparts. While careful optimisation can yield incremental improvements —and clearly outperforms any simple analytical equation— the marginal gains beyond defaults are limited. Taken together, the results support a practical workflow: use RF feature importances to define a compact, seven-descriptor model (preferentially including $\chi(N)$ for consistency with classification), and rely on default settings for rapid, robust predictions of N_{Mayr}.

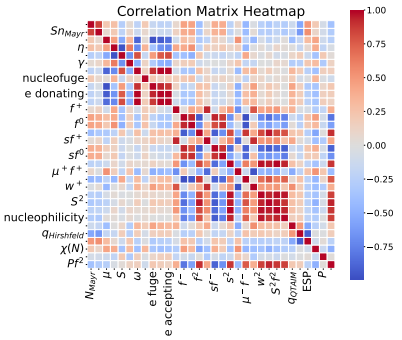
4.3 Conformers

The difference between using only the most stable conformer and including all conformers accessible at 20°C in solvation proved to be negligible. Variations in the predicted results were comparable to the impact of adding or removing a molecule from the dataset, supporting the robustness of the method with respect to conformational diversity.

It is also important to note that, in the context of classification analysis, the use of all accessible conformers produced results nearly identical to those obtained using only the most stable conformer. This outcome reflects a physically meaningful interpretation. Our classification analysis reveals whether a reaction is thermodynamically favoured —that is, whether it occurs or not— but



(a) Importance of descriptors



(b) The order of the descriptors is the same for both axes, but the names are alternated for readability. ESP is the electrostatic potential at the nucleus.

Figure 7: Impotance of the descriptors and correlation between them.

does not capture kinetic aspects of reactivity, while the N_{Mayr} value is directly related to the reaction rate constant and therefore encodes kinetic information.

5 Conclusions

Although the QTAIM and Conceptual-DFT descriptors do not yet yield a fully satisfactory analytic expression for the nucleophilicity index N_{Mayr} of the nitrogen atoms, our ML model successfully reproduces the global trend of N_{Mayr} across the entire data set. The MAE obtained for N_{Mayr} is larger than that previously reported for the electrophilicity index E_{Mayr}, a difference that is consistent with the intrinsically broader energetic landscape associated with nucleophilic reactivity.

Nevertheless, the present analysis demonstrates that QTAIM charges and the condensed linear response already encode sufficient information to discriminate, between nitrogen atoms that engage in nucleophilic attack and those that remain inert. Even more importantly, this interpretation is reinforced when the influence of the neighbour environment is explicitly taken into account, underscoring the local, yet context-dependent, nature of nucleophilicity.

Acknowledgments

This work was partially supported by Normandie Université, the Région Normandie, the Centre National de la Recherche Scientifique, Université de Rouen Normandie, INSA Rouen Normandie, Université Caen Normandie, ENSICAEN, Labex SynOrg (ANR-11-LABX-0029), the graduate school for research XL-

Chem (ANR-18-EURE-0020 XL CHEM) and Innovation Chimie Carnot (I2C). The Centre Régional Informatique et d'Applications Numériques de Normandie (CRIANN) is warmly acknowledged for providing high-performance computational resources.

References

[1] C. K. Ingold. Significance of tautomerism and of the reactions of aromatic compounds in the electronic theory of organic reactions. *Journal of the Chemical Society (Resumed)*, page 1120, 1933.

[2] P. Muller. Glossary of terms used in physical organic chemistry (iupac recommendations 1994). *Pure and Applied Chemistry*, 66(5):1077–1184, January 1994.

[3] Herbert Mayr and Armin R. Ofial. Do general nucleophilicity scales exist? *Journal of Physical Organic Chemistry*, 21(7-8):584–595, May 2008.

[4] Herbert Mayr. Reactivity scales for quantifying polar organic reactivity: the benzhydrylium methodology. *Tetrahedron*, 71(32):5095–5111, August 2015.

[5] Patricia Pérez, Alejandro Toro-Labbé, Arie Aizman, and Renato Contreras. Comparison between experimental and theoretical scales of electrophilicity in benzhydryl cations. *The Journal of Organic Chemistry*, 67(14):4747–4752, May 2002.

[6] Claus Schindele, K. N. Houk, and Herbert Mayr. Relationships between carbocation stabilities and electrophilic reactivity parameters, e: Quantum mechanical studies of benzhydryl cation structures and stabilities. *Journal of the American Chemical Society*, 124(37):11208–11214, August 2002.

[7] Maike Vahl and Jonny Proppe. The computational road to reactivity scales. *Physical Chemistry Chemical Physics*, 25(4):2717–2728, 2023.

[8] Henry Eyring. The activated complex in chemical reactions. *The Journal of Chemical Physics*, 3(2):107–115, February 1935.

[9] Jeremy N. Harvey, Fahmi Himo, Feliu Maseras, and Lionel Perrin. Scope and challenge of computational methods for studying mechanism and reactivity in homogeneous catalysis. *ACS Catalysis*, 9(8):6803–6813, June 2019.

[10] Jonny Proppe, Tamara Husch, Gregor N. Simm, and Markus Reiher. Uncertainty quantification for quantum chemical models of complex reaction networks. *Faraday Discussions*, 195:497–520, 2016.

[11] Chen Wang, Yao Fu, Qing-Xiang Guo, and Lei Liu. First-principles prediction of nucleophilicity parameters for π nucleophiles: Implications for mechanistic origin of mayr's equation. *Chemistry - A European Journal*, 16(8):2586–2598, February 2010.

[12] Lian-Gang Zhuo, Wei Liao, and Zhi-Xiang Yu. A frontier molecular orbital theory approach to understanding the mayr equation and to quantifying nucleophilicity and electrophilicity by using homo and lumo energies. *Asian Journal of Organic Chemistry*, 1(4):336–345, November 2012.

[13] Harish Jangra, Quan Chen, Elina Fuks, Ivo Zenz, Peter Mayer, Armin R. Ofial, Hendrik Zipse, and Herbert Mayr. Nucleophilicity and electrophilicity parameters for predicting absolute rate constants of highly asynchronous 1, 3-dipolar cycloadditions of aryldiazomethanes. *Journal of the American Chemical Society*, 140(48):16758–16772, November 2018.

[14] Robert J. Mayer, Martin Breugst, Nathalie Hampel, Armin R. Ofial, and Herbert Mayr. Ambident reactivity of phenolate anions revisited: A quantitative approach to phenolate reactivities. *The Journal of Organic Chemistry*, 84(14):8837–8858, June 2019.

[15] Zhen Li, Robert J. Mayer, Armin R. Ofial, and Herbert Mayr. From carbodiimides to carbon dioxide: Quantification of the electrophilic reactivities of heteroallenes. *Journal of the American Chemical Society*, 142(18):8383–8402, April 2020.

[16] Jingjing Zhang, Quan Chen, Robert J. Mayer, Jin-Dong Yang, Armin R. Ofial, Jin-Pei Cheng, and Herbert Mayr. Predicting absolute rate constants for huisgen reactions of unsaturated iminium ions with diazoalkanes. *Angewandte Chemie International Edition*, 59(30):12527–12533, May 2020.

[17] Dominik S. Allgäuer, Harish Jangra, Haruyasu Asahara, Zhen Li, Quan Chen, Hendrik Zipse, Armin R. Ofial, and Herbert Mayr. Quantification and theoretical analysis of the electrophilicities of michael acceptors. *Journal of the American Chemical Society*, 139(38):13318–13329, September 2017.

[18] Jacopo Tomasi, Benedetta Mennucci, and Roberto Cammi. Quantum mechanical continuum solvation models. *Chemical Reviews*, 105(8):2999–3094, July 2005.

[19] Victoria Castor-Villegas, Vincent Tognetti, and Laurent Joubert. On the prediction by density functional theory of entropies in solution within implicit solvation models. *Journal of Molecular Modeling*, 31(1), December 2024.

[20] Guillaume Hoffmann, Vincent Tognetti, and Laurent Joubert. On the influence of dynamical effects on reactivity descriptors. *Chemical Physics Letters*, 724:24–28, June 2019.

[21] Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*, volume 4. Springer, 2006.

[22] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, November 2005.

[23] Yidi Liu, Qi Yang, Junjie Cheng, Long Zhang, Sanzhong Luo, and Jin-Pei Cheng. Prediction of nucleophilicity and electrophilicity based on a machine-learning approach. *ChemPhysChem*, 24(14), June 2023.

[24] Nicolai Ree, Jan M. Wollschläger, Andreas H. Göller, and Jan H. Jensen. Atom-based machine learning for estimating nucleophilicity and electrophilicity with applications to retrosynthesis and chemical stability. *Chemical Science*, 16(13):5676–5687, 2025.

[25] Nicolai Ree, Andreas H. Göller, and Jan H. Jensen. Automated quantum chemistry for estimating nucleophilicity and electrophilicity with applications to retrosynthesis and covalent inhibitors. *Digital Discovery*, 3(2):347–354, 2024.

[26] Sebastián A. Cuesta, Martín Moreno, Romina A. López, José R. Mora, José Luis Paz, and Edgar A. Márquez. Electropredictor: An application to predict mayr's electrophilicity e through implementation of an ensemble model based on machine learning algorithms. *Journal of Chemical Information and Modeling*, 63(2):507–521, January 2023.

[27] Guillaume Hoffmann, Vincent Tognetti, and Laurent Joubert. Can molecular and atomic descriptors predict the electrophilicity of michael acceptors? *Journal of Molecular Modeling*, 24(10), September 2018.

[28] Aël Cador, Vincent Tognetti, Laurent Joubert, and Paul L. A. Popelier. Aza-michael addition in explicit solvent: A relative energy gradient–interacting quantum atoms study. *ChemPhysChem*, 24(24), November 2023.

[29] Guillaume Hoffmann, Muhammet Balcilar, Vincent Tognetti, Pierre Héroux, Benoît Gaüzère, Sébastien Adam, and Laurent Joubert. Predicting experimental electrophilicities from quantum and topological descriptors: A machine learning approach. *Journal of Computational Chemistry*, 41(24):2124–2136, July 2020.

[30] H. Chermette. Chemical reactivity indexes in density functional theory. *Journal of Computational Chemistry*, 20(1):129–154, January 1999.

[31] P. Geerlings, F. De Proft, and W. Langenaeker. Conceptual density functional theory. *Chemical Reviews*, 103(5):1793–1874, April 2003.

[32] Paul Popelier, F. Aicken, and S. O'Brien. *Atoms in molecules*, volume 1. 01 2000.

[33] R. F. W. Bader. *Atoms in Molecules: A Quantum Theory (International Series of Monographs on Chemistry)*. Oxford University Press, USA, 1994.

[34] Manuel Orlandi, Margarita Escudero-Casao, and Giulia Licini. Nucleophilicity prediction via multivariate linear regression analysis. *The Journal of Organic Chemistry*, 86(4):3555–3564, February 2021.

[35] Matthew S. Sigman, Kaid C. Harper, Elizabeth N. Bess, and Anat Milo. The development of multidimensional analysis tools for asymmetric catalysis and beyond. *Accounts of Chemical Research*, 49(6):1292–1301, May 2016.

[36] Stefan T. A. Berger, Armin R. Ofial, and Herbert Mayr. Inverse solvent effects in carbocation carbanion combination reactions: The unique behavior of trifluoromethylsulfonyl stabilized carbanions. *Journal of the American Chemical Society*, 129(31):9753–9761, July 2007.

[37] Roland Appel, Robert Loos, and Herbert Mayr. Nucleophilicity parameters for phosphoryl-stabilized carbanions and phosphorus ylides: Implications for wittig and related olefination reactions. *Journal of the American Chemical Society*, 131(2):704–714, December 2008.

[38] Roland Lucius and Herbert Mayr. Constant selectivity relationships of addition reactions of carbanions. *Angewandte Chemie International Edition*, 39(11):1995–1997, June 2000.

[39] Johannes Proppe. Uncertainty quantification of reactivity scales. `https://gitlab.com/jproppe/mayruq`, 2025. Last accessed 17 July 2025.

[40] Ricardo A. Mata and Martin A. Suhm. Benchmarking quantum chemical methods: Are we heading in the right direction? *Angewandte Chemie International Edition*, 56(37):11011–11018, April 2017.

[41] Gregor N. Simm, Jonny Proppe, and Markus Reiher. Error assessment of computational models in chemistry. *CHIMIA*, 71(4):202, April 2017.

[42] Christian Reichardt. Pyridinium-n-phenolate betaine dyes as empirical indicators of solvent polarity: Some new findings. *Pure and Applied Chemistry*, 80(7):1415–1432, 2008.

[43] Christian Laurence, Julien Legros, Agisilaos Chantzis, Aurélien Planchat, and Denis Jacquemin. A database of dispersion-induction di, electrostatic es, and hydrogen bonding α1 and β1 solvent parameters and some applications to the multiparameter correlation analysis of solvent effects. *The Journal of Physical Chemistry B*, 119(7):3174–3184, January 2015.

[44] Mortimer J. Kamlet, Jose Luis M. Abboud, Michael H. Abraham, and R. W. Taft. Linear solvation energy relationships. 23. a comprehensive collection of the solvatochromic parameters, .pi.*, .alpha., and .beta., and some methods for simplifying the generalized solvatochromic equation. *The Journal of Organic Chemistry*, 48(17):2877–2887, August 1983.

[45] Javier Catalán, Vicenta López, Pilar Pérez, Rosa Martin-Villamil, and José-Gonzalo Rodríguez. Progress towards a generalized solvent polarity scale: The solvatochromism of 2-(dimethylamino)-7-nitrofluorene and its homomorph 2-fluoro-7-nitrofluorene. *Liebigs Annalen*, 1995(2):241–252, February 1995.

[46] Javier Catalán, Cristina Díaz, Vicenta López, Pilar Pérez, José-Luis G. De Paz, and José Gonzalo Rodríguez. A generalized solvent basicity scale: The solvatochromism of 5-nitroindoline and its homomorph 1-methyl-5-nitroindoline. *Liebigs Annalen*, 1996(11):1785–1794, November 1996.

[47] Javier Catalán and Cristina Díaz. A generalized solvent acidity scale: The solvatochromism of o-tert-butylstilbazolium betaine dye and its homomorph o, o'-di-tert-butylstilbazolium betaine dye. *Liebigs Annalen*, 1997(9):1941–1949, September 1997.

[48] Charles M. Hansen. *Hansen Solubility Parameters: A User's Handbook, Second Edition*. CRC Press, June 2007.

[49] Tanja Kanzian, Sami Lakhdar, and Herbert Mayr. Kinetic evidence for the formation of oxazolidinones in the stereogenic step of proline-catalyzed reactions. *Angewandte Chemie International Edition*, 49(49):9526–9529, November 2010.

[50] Sami Lakhdar, Biplab Maji, and Herbert Mayr. Imidazolidinone-derived enamines: Nucleophiles with low reactivity. *Angewandte Chemie International Edition*, 51(23):5739–5742, April 2012.

[51] Hannes Erdmann, Feng An, Peter Mayer, Armin R. Ofial, Sami Lakhdar, and Herbert Mayr. Structures and reactivities of 2-trityl- and 2-(triphenylsilyl)pyrrolidine-derived enamines: Evidence for negative hyperconjugation with the trityl group. *Journal of the American Chemical Society*, 136(40):14263–14269, September 2014.

[52] Daria S. Timofeeva, Robert J. Mayer, Peter Mayer, Armin R. Ofial, and Herbert Mayr. Which factors control the nucleophilic reactivities of enamines? *Chemistry - A European Journal*, 24(22):5901–5910, March 2018.

[53] Guillaume Hoffmann, Frédéric Guégan, Vanessa Labet, Laurent Joubert, Henry Chermette, Christophe Morell, and Vincent Tognetti. Expanding horizons in conceptual density functional theory: Novel ensembles and descriptors to decipher reactivity patterns. *Journal of Computational Chemistry*, 45(20):1716–1726, April 2024.

[54] P. K. Chattaraj and B. Maiti. Reactivity dynamics in atom-field interactions: A quantum fluid density functional study. *The Journal of Physical Chemistry A*, 105(1):169–183, December 2000.

[55] Robert G. Parr and Weitao Yang. Density functional approach to the frontier-electron theory of chemical reactivity. *Journal of the American Chemical Society*, 106(14):4049–4050, July 1984.

[56] Paul Geerlings, Stijn Fias, Zino Boisdenghien, and Frank De Proft. Conceptual dft: chemistry from the linear response function. *Chemical Society Reviews*, 43(14):4989, 2014.

[57] Isabelle Guyon and André Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, 2003.

[58] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, September 1995.

[59] Chih-Chung Chang and Chih-Jen Lin. Libsvm: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3):1–27, April 2011.

[60] Asa Ben-Hur, David Horn, Hava T. Siegelmann, and Vladimir Vapnik. Support vector clustering. *Journal of Machine Learning Research*, 2:125–137, 2001.

[61] Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, October 2001.

[62] Bernhard Schölkopf and Alexander J. Smola. Learning with kernels. In *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, 2002.

[63] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[64] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[65] G. te Velde, F. M. Bickelhaupt, E. J. Baerends, C. Fonseca Guerra, S. J. A. van Gisbergen, J. G. Snijders, and T. Ziegler. Chemistry with ADF. *J. Comput. Chem.*, 22(9):931–967, 2001.

[66] Evert Jan Baerends, Nestor F. Aguirre, Nick D. Austin, Jochen Autschbach, F. Matthias Bickelhaupt, Rosa Bulo, Chiara Cappelli, Adri C. T. van Duin, Franco Egidi, Célia Fonseca Guerra, Arno Förster, Mirko Franchini, Theodorus P. M. Goumans, Thomas Heine, Matti Hellström, Christoph R. Jacob, Lasse Jensen, Mykhaylo Krykunov, Erik van Lenthe, Artur Michalak, Mariusz M. Mitoraj, Johannes Neugebauer, Valentin Paul Nicu, Pier Philipsen, Harry Ramanantoanina, Robert Rüger,

Georg Schreckenbach, Mauro Stener, Marcel Swart, Jos M. Thijssen, Tomáš Trnka, Lucas Visscher, Alexei Yakovlev, and Stan van Gisbergen. The amsterdam modeling suite. *The Journal of Chemical Physics*, 162(16):162501, 04 2025.

[67] John P. Perdew, Kieron Burke, and Matthias Ernzerhof. Generalized gradient approximation made simple. *Phys. Rev. Lett.*, 77(18):3865–3868, October 1996.

[68] Stefan Grimme, Stephan Ehrlich, and Lars Goerigk. Effect of the damping function in dispersion corrected density functional theory. *J. Comput. Chem.*, 32(7):1456–1465, March 2011.

[69] Theoretical Chemistry SCM. Plams interfaces: Conformers. `https://www.scm.com/doc/plams/interfaces/conformers.html`, 2023. Software for Chemistry & Materials, Amsterdam, The Netherlands.

[70] Theoretical Chemistry SCM. Plams components: RDKit. `https://www.scm.com/doc/plams/components/mol_rdkit.html`, 2023. Software for Chemistry & Materials, Amsterdam, The Netherlands.

[71] Andreas Klamt. Conductor-like screening model for real solvents: A new approach to the quantitative calculation of solvation phenomena. *The Journal of Physical Chemistry*, 99(7):2224–2235, February 1995.

[72] Cory C. Pye and Tom Ziegler. An implementation of the conductor-like screening model of solvation within the amsterdam density functional package. *Theoretical Chemistry Accounts: Theory, Computation, and Modeling (Theoretica Chimica Acta)*, 101(6):396–408, May 1999.

[73] Ralph G. Pearson. *Chemical Hardness*. Wiley, October 1997.

[74] W Yang and R G Parr. Hardness, softness, and the fukui function in the electronic theory of metals and catalysis. *Proceedings of the National Academy of Sciences*, 82(20):6723–6726, October 1985.

[75] Robert G. Parr, László v. Szentpály, and Shubin Liu. Electrophilicity index. *Journal of the American Chemical Society*, 121(9):1922–1924, February 1999.

[76] Paul W. Ayers and Mel Levy. *Perspective on "Density functional approach to the frontier-electron theory of chemical reactivity"*, page 353–360. Springer Berlin Heidelberg, 2000.

[77] Ravishankar Sundararaman, William A. Goddard, and Tomas A. Arias. Grand canonical electronic density-functional theory: Algorithms and applications to electrochemistry. *The Journal of Chemical Physics*, 146(11), March 2017.

[78] Daniel de las Heras and Matthias Schmidt. Full canonical information from grand-potential density-functional theory. *Physical Review Letters*, 113(23), December 2014.

[79] R. Grincourt, G. Hoffmann, F. Guégan, V. Tognetti, L. Joubert, H. Chermette, A. Toro Labbé, and C. Morell. Title of the article. *J. Chem. Phys.*, 2025. in press.

Chapter

5 Entropies in Solvation

In this section, we present an already published paper in which we adress the entropies in solvation.

Les entropies représentent une contribution fondamentale à l'énergie libre de Gibbs, porteuse d'informations chimiques essentielles, notamment dans l'étude des mécanismes réactionnels. Toutefois, leur évaluation en solution reste une tâche loin d'être triviale. Dans ce travail, nous nous concentrons sur cette évaluation dans le cadre des modèles de solvatation implicites. Pour cela, des corrections successives —de complexité croissante— sont proposées. Elles ne font appel qu'à des grandeurs accessibles via tout programme standard de chimie quantique ainsi qu'à des propriétés macroscopiques du solvant. Ces modèles sont évalués par comparaison à plus d'une centaine de valeurs expérimentales d'entropie mesurées en phase liquide. Il en ressort qu'une amélioration significative par rapport à l'approximation classique du gaz perfect peut être obtenue à un coût computationnel quasi négligeable, menant à un modèle prédictif à la fois robuste et transférable.

Journal of Molecular Modeling (2025) 31:7
https://doi.org/10.1007/s00894-024-06225-3

ORIGINAL PAPER

On the prediction by density functional theory of entropies in solution within implicit solvation models

Victoria Castor-Villegas[1] · **Vincent Tognetti**[1] · **Laurent Joubert**[1]

Received: 31 August 2024 / Accepted: 14 November 2024 / Published online: 4 December 2024

Abstract

Context Entropies are fundamental contributions to Gibbs energies that carry important chemical information, in particular when investigating reaction mechanisms. However, evaluating them in solution is far from being straightforward. In this paper, we focus on its evaluation within the framework of implicit solvation models. To this aim, successive corrections (with increased complexity) involving only contributions available from any standard quantum chemistry code and macroscopic solvent properties are built and assessed by comparison to more than one hundred experimental entropy values measured in a liquid phase. It turns out that significant improvement with respect to the standard ideal gas approximation can be achieved at an almost negligible computational cost, affording a robust and transferable predictive model.

Methods DFT calculations with the ADF software at the PBE or PBE0/TZ2P level of theory with COSMO solvent model. Python scripts for regressions.

Keywords Entropy · Gibbs energy · Solvent phase · Continuum models · Density functional theory

Introduction

Quantum chemistry is often used to determine the possible mechanisms of a chemical reaction, often within the potential energy surface (PES) paradigm. For reactions performed at constant temperature T and pressure (this is the most common case in organic synthesis or in homogeneous catalysis), the relevant thermodynamic state function is the Gibbs energy G defined by

$$G = H - TS \tag{1}$$

where H and S respectively denote the enthalpy and entropy of the studied system. This last one thus plays a crucial role in thermodynamic calculations and in understanding the behavior of chemical systems, in particular when the number of molecules Δn varies along the reaction. Indeed, while the variations of Gibbs energies and enthalpies are quite close to each other for isomerization processes or conformational changes (that correspond to $\Delta n = 0$), chemical additions (inducing negative values for Δn) and eliminations ($\Delta n > 0$) can exhibit consequent entropy effects. In some cases, entropy can even reveal the main driving force or resistance of the reaction and may also induce kinetic selectivity. [1]

In fact, as recalled by [2], "a useful rule of thumb" is that "for any reaction step in which two molecules combine to form a single one, the standard Gibbs energy change at room temperature will be roughly $10\,\mathrm{kcal\,mol^{-1}}$ less favourable (or more unfavourable) than the electronic energy change," a value that should be compared to the $10\,\mathrm{kcal\,mol^{-1}}$ to $30\,\mathrm{kcal\,mol^{-1}}$ values typically encountered for activation barriers. Besides, this effect is even enhanced for multicomponent reactions involving more than two partners, a synthetic approach that is notably popular in drug design. We refer the interested reader to the enlightening didactical paper by [3] for more examples and an insightful analysis of important chemical trends.

In this study, we will focus on reactions occurring in solvents. The still most widespread approach is to approximate the entropy in solution by its gas phase expression (see more details in next section), while it has been recognized for many

✉ Vincent Tognetti
vincent.tognetti@univ-rouen.fr

✉ Laurent Joubert
laurent.joubert@univ-rouen.fr

Victoria Castor-Villegas
victoria.castor-villegas@univ-rouen.fr

[1] Normandy Univ., COBRA UMR 6014 & FR 3038, Université de Rouen, INSA Rouen, CNRS, 1 rue Tesnière, 76821 Mont St Aignan Cedex, France

decades that entropy values in solution and in the gas phase can significantly differ (see for instance the work by [4] based on experimental results). Several correcting schemes have been proposed in the last years (we emphasize that we do not consider to extensively review them in this paper), such as the seminal works by [5, 6] and [7], this last one being the basis for the Ariai and Gellrich's methodology [8]. Of particular significance is the remarkable automated approach by [9] and [10], which is well suited to accurately evaluate entropies of large molecules in solution [11].

It can be noticed that some of these methods are not implemented in standard quantum chemistry software or that they require complex computational pipelines. Crudely speaking, entropy can be estimated either using static or dynamic approaches. For instance, [12] have assessed the use of ab initio molecular dynamics applied to organometallic complexes, while [13] have shown how metadynamics can be efficiently used to evaluate entropies at all temperatures by a single simulation. Although these approaches are both physically grounded and chemically attractive, they can reveal computationally intractable if one wants to rely on a pure quantum chemistry level without resorting to any classical mechanics methods.

Besides, a second important factor that affects the calculation time is how the solvent is described. While explicit approaches in which a given number of solvent molecules are described at the atomic scale allow for a precise description of the interactions between the solute and the solvent, they are considerably more expensive than implicit ones, epitomized by the various flavors of the polarizable continuum model (PCM) [14]. These last ones disregard the atomistic structure of the environment since the molecule is placed into a cavity inside a continuum characterized by its relative permittivity ϵ_r, but the mutual polarization between the solute electron density and the infinite continuum dielectric medium is however taken into account by solving the appropriate Poisson electrostatic equation.

Owing to their simplicity, their versatility (all common solvents have been parameterized), and their high accuracy/time ratio, PCMs are ubiquitous in the modelling of chemical reactions and of physico-chemical properties of molecules in solution. In this contribution, we thus only consider static PCM approaches. Furthermore, still targeting at a minimal computational cost, only properties already computed by standard density functional theory (DFT) codes will be employed, so that this strategy can be straightforwardly implemented from the output of any DFT software.

Henceforth, this paper will be divided into the following four next sections: a theoretical one that presents the main equations to evaluate the various entropy components and entropy corrections, followed by the description of the experimental database on which our several models (each incorporating different mathematical complexities and physical considerations) will be trained and of the computational protocol, before reporting their performances and discussing their meaning and usefulness.

Theory

The mathematical expressions for entropy in the gas phase are based on standard statistical physics formulas (see for instance Ochterski's paper [15]) within the ideal gas (IG) hypothesis. Even if these expressions are standard, we decided to report them here, so that this paper remains self-explanatory. More specifically, the IG entropy (thus incorrect for molecules in solution) of a system can be divided into four contributions: vibrational (S_v), rotational (S_r), translational (S_t), and electronic entropies (S_e) according to the following:

$$S_{IG} = S_v + S_r + S_t + S_e \tag{2}$$

If we assume that the electronic excited states are energetically far in energy from the ground state, the electronic component is simply the following:

$$S_e = R \ln(d) \tag{3}$$

where d denotes the electronic degeneracy equal to the (electronic) spin multiplicity and R the molar gas constant. For non-radical species, this term exactly vanishes, so that we will not consider it any longer. The translational contribution is evaluated from the celebrated Sackur-Tetrode formula:

$$S_t = R(\ln(q_t) + 5/2) \tag{4}$$

where q_t represents the translational thermodynamic partition function

$$q_t = \left(\frac{2\pi m k_B T}{h^2}\right)^{3/2} V \tag{5}$$

and h and k_B are the Planck and Boltzmann's constants, respectively, and V is the volume. Similarly, the rotational entropy reads as follows (model of the quantum rigid rotor):

$$S_r = R(\ln(q_r) + 5/2) \tag{6}$$

where the rotational partition function q_r involves the three moments of inertia J_x, J_y, and J_z (along each Cartesian axis generically denoted α, which can be all three conveniently collected in the $\vec{J}$ vector) of the molecule. More precisely, once the three rotational temperatures are defined by $\Theta_\alpha^r = h^2/(8\pi^2 k_B J_\alpha)$, we have

$$q_r = \frac{\sqrt{\pi}}{\sigma_r}\left(\frac{T^{3/2}}{\sqrt{\Theta_x^r \Theta_y^r \Theta_z^r}}\right) \tag{7}$$

where σ_r is the rotation symmetry number (equal to 1 for non-symmetric molecules).

Finally, for each of the $3N-6$ vibrational modes (for non-linear molecules) of frequency ν_i, we define the characteristic vibrational temperature by the following:

$$\Theta_i^v = h\nu_i/k_B \tag{8}$$

The vibrational entropy can be evaluated by a more intricate, but closed formula, in the case we assumed that pure harmonic vibrations (i.e., the energy levels are those from the quantum oscillator model):

$$S_v^h = \sum_i \left(\frac{\Theta_i^v/T}{e^{\Theta_i^v/T}-1} - \ln\left(1-e^{\Theta_i^v/T}\right)\right) \tag{9}$$

Obviously, the harmonic approximation is sometimes too crude. Full anharmonic calculations (for instance using second-order perturbation theory (VPT2) or vibrational mean-field (VSCF) approaches) are available, but are in general feasible only for systems of small or moderate size. A popular cheap alternative is the simple scaling of harmonic frequencies, but it does not really cure the problems of the harmonic approximation, in particular for the low-lying vibrational frequency modes.

However, as shown by Eq. 9, these modes are those who contribute the most to the vibrational entropy (S_v^h actually diverges for $\nu_i \rightarrow 0$). In 2012, [16] proposed to correct this incorrect behavior (leading to the so-called rigid-rotor-harmonic-oscillator (RRHO) approximation) by building a smooth interpolation (using a damping function w_{HG} proposed by [17]) between a free rigid rotor whose effective inertia moment depends on the vibrational frequency (and is thus not the molecular one) and the harmonic oscillator at a negligible computational cost. More precisely, the entropy for each normal mode is computed by the following:

$$S_v^G(\nu_i) = w_{HG}(\nu_i)S_r(\nu_i) + (1-w_{HG}(\nu_i))S_v^h(\nu_i) \tag{10}$$

which is not diverging anymore for vanishing ν_i values. It should be emphasized that this RRHO approach should not be confounded with the more refined *hindered* rotor approach to model the low-frequency vibrations [18, 19] in which the potential entering the vibrational Schrödinger equation is fitted by a Fourier series (which thus required many points on the PES). Finally, the total vibrational entropy obtained by the simple (but efficient) Grimme's treatment, S_v^G, is obtained by summation on all frequencies, allowing us to define Grimme's correction according to the following:

$$\Delta S_v^G = S_v^G - S_v^h \tag{11}$$

Having recalled the main equation for gas phase entropies, we now come back to our main purpose that is to predict accurate values for entropies in solution. These should not be confused with solvation entropies that are defined by the following:

$$\begin{aligned}\Delta S^{solvation} = & S\left(\vec{J}^{solv}, \left\{\Theta_i^{r,solv}\right\}, ...\right) \\ & -S\left(\vec{J}^{gas}, \left\{\Theta_i^{r,gas}\right\}, ...\right)\end{aligned} \tag{12}$$

In this last equation, we have made explicitly some relevant variables in the two contributions in the right-hand side, which are the entropy in solution and the entropy in the gas phase. As geometries in the gas phase and in solvent often differ (for instance, some structures are more folded in the gas phase), the inertia moments will also have different values. Besides, as solvent and gas phase PESs are not the same, the vibrational temperatures $\Theta_i^{r,solv}$ and $\Theta_i^{r,gas}$ are not equal. So, even if we use the same formulas to evaluate S^{solv} and S^{gas}, values will differ if we use the molecular properties computed specifically in the two phases. This is in a way reminiscent of the fact that, in DFT, the Kohn-Sham exact exchange value differs from the Hartree-Fock exchange value since, even if the very same orbital functional is used, it is applied on different orbitals.

It can be added that using these gas phase formulas with gas phase quantities will only lead to approximate gas phase entropy values due to the various approximations we have already mentioned. However, it is possible that solvation entropies would be not so bad due to spurious error compensation in Eq. 12.

We now come back to our main target: building an accurate model of entropies in solution using the formulas of gas phase entropy applied on quantities evaluated in solution. To this aim, it has been well established that translational and rotational degrees of freedom are reduced in solution [20] (from a physical point of view, the available volume is decreased due to the solvent pressure, so that, in virtue of Eq. 5, S_t decreases).

As a consequence, [21] proposed to fully remove S_t and S_r in their calculations. In a less drastic way, [22] proposed to scale S_t and S_r by a two-third factor, a simple correction that has been regularly used, mainly in the context of organometallic catalysis [23]. From thermodynamic measurements, [24] proposed a very close factor value (0.65), also close to the one (0.60) we used [25] in a recent paper in organic synthesis. All these corrections correspond to the following one-parameter model (in a general way, the number of parameters in our various models will be indicated by an integer value in subscript position):

$$S_1^l = \alpha\,(S_t + S_r) + S_v^h \tag{13}$$

An easy generalization of this linear model (hence the l superscript) implies four parameters:

$$S_4^l = \alpha S_t + \beta S_r + \gamma S_v^h + \delta \Delta S_v^G \tag{14}$$

These two first models can be seen as a first-order Taylor expansion of the entropy in solution using the ideal gas-type entropy components as basic variables. Pushing toward second-order would involve S_t^2, S_r^2, $(S_v^h)^2$ squared terms, as well as $S_t S_r$, $S_t S_v$, and $S_r S_v$ crossed ones. Preliminary tests have shown that the inclusion of S_t^2 and S_r^2 squared terms do not significantly improve the model. As we also seek the most parsimonious models (in terms of parameters) in order to prevent it from too much overfitting, only $S_t S_v$, $S_r S_v$, and $(S_v^h)^2$ (this one becoming dominant when the molecular size increases) will be further considered.

However, as entropy increases with the molecular size, the mixed variables would become largely dominant for extended molecules and will bias the regression process. It is thus preferable to work with some "normalized" quantities, which would have all the same dimension (that of an entropy). Our own non-linear (*nl*, or second-order) extensions of S_1^l, where S_t and S_r are still grouped and where no action is made on S_v, is accordingly as follows:

$$S_2^{nl} = \alpha (S_t + S_r) + S_v^h + \beta \left(\frac{S_t S_v^h}{S_t + S_r + S_v^h} + \frac{S_r S_v^h}{S_t + S_r + S_v^h} \right) \tag{15}$$

involving two scaling factors. If we now allow to correct the vibrational part, the natural second-order extension of Eq. 14 includes seven parameters:

$$S_7^{nl} = \alpha S_t + \beta S_r + \gamma S_v^h + \delta \Delta S_v^G + \zeta \frac{S_t S_v^h}{S_t + S_r + S_v^h} + \eta \frac{S_r S_v^h}{S_t + S_r + S_v^h} + \theta \frac{(S_v^h)^2}{S_t + S_r + S_v^h} \tag{16}$$

It should also be noticed that using this normalization of the second-order terms, all corresponding parameters are *unitless*, so that their magnitude can be safely controlled.

A refinement of these schemes is to correct the first-order vibrational entropy using the main solvent properties, namely the relative permittivity ϵ_r and the radius R_{solv} used in implicit solvent model computations (see Computational details below). Still, in the quest for the simplest models, the last (experimental) parameter has been retained by us for the vibrational correction:

$$S_3^{nlv} = \alpha (S_t + S_r) + f_\gamma (R_{solv}) S_v^h + \beta \left(\frac{S_t S_v^h}{S_t + S_r + S_v^h} + \frac{S_r S_v^h}{S_t + S_r + S_v^h} \right) \tag{17}$$

where γ is a parameter controlling the vibrational weight, which can be further extended by the following:

$$S_4^{nlv} = \alpha (S_t + S_r) + f_\gamma (R_{solv}) S_v^h + \beta \left(\frac{S_t S_v^h}{S_t + S_r + S_v^h} + \frac{S_r S_v^h}{S_t + S_r + S_v^h} \right) + g_\delta (\epsilon_r) \tag{18}$$

In these two last equations, f_γ and g_δ are tailored functions whose expressions will be based (and justified) on the results obtained from the previous models (note that we do not include ΔS_v^G, and that S_r and S_r are grouped here for reasons that will be reported in the results section).

The molecular database

Experimental values for entropy in solution are extracted from the National Institute of Standards and Technology (NIST) Chemistry WebBook [26], labelled S_0^{liq} in the database, corresponding to the entropy of the pure liquid in standard thermodynamical conditions. When several values are available for a given chemical system, the most recent one was systematically chosen (unfortunately, the database does not provide error margins). All solvents reported in the online ADF [27] and [28] manuals were considered, excluding solvent mixtures and, obviously, solvents for which no experimental data were available, to which we added selected small molecules still from the NIST website.

In total, we collected 106 molecules of small or moderate size (from 3 to 56 atoms), including all common solvents used in organic synthesis, covering the full type spectrum (polar, apolar, protic, aprotic, with high or low dielectric constant), and the main chemical functions and families (alkanes, alkenes, alkynes, ethers, alcohols, amines, nitriles, carboxylic acids, esters, ketones, nitro and halogenated compounds, thiols...) in both aliphatic and aromatic series. The full list can be found in the supplementary material.

More precisely, Fig. 1 shows the distributions of values for the relative permittivity (left panel) and for the experimental standard entropies (right panel) along the whole dataset. It can be seen that ϵ_r values spread from almost 1 to 80 (water) with an important concentration in the [1,20] range where the distribution is quite uniform. The shape for the entropy distribution strongly differs, close to a Gaussian one with a major peak around 220 J mol^{-1} K^{-1}.

It should be noticed that the considered experimental range is restricted by the fact that, in our approach, the system should be in a pure liquid phase in standard conditions (for instance, for several interesting chemical systems, only entropies in the solid state have been measured). The entropy value distribution is a little bit biased toward small values: we thus conducted a full search in the NIST database for

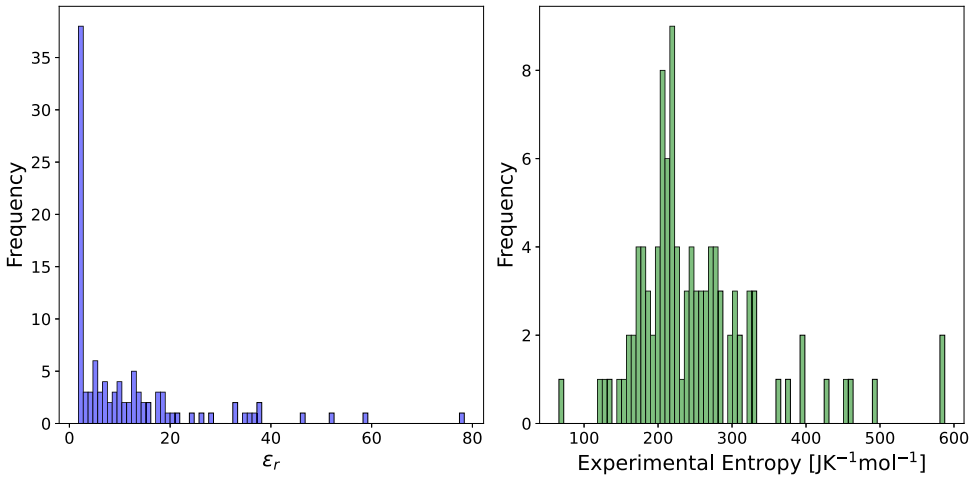


Fig. 1 Distribution of relative permittivity (left panel) and experimental standard entropy (right panel) values across the dataset

all molecules including from 14 to 19 carbon atoms, and we incorporated all those for which full experimental data was available (unfortunately, in some cases where liquid entropies values have been measured, we were not able to find the corresponding ϵ_r values, precluding the use of these points in our dataset).

Computational details

As mentioned in the previous section, experimental values correspond to the pure liquid at 298.15 K. This means that the solute and the solvent correspond to the same molecule. Within all this work, the COSMO implicit model [29], as implemented in the ADF software [30] (with default parameters), was exclusively considered (for instance, the experimental value of entropy for liquid dichloromethane (DCM) will be retrieved by performing a DFT calculation on one DCM molecule immersed inside a DCM continuum). Such a computation requires the knowledge of the relative permittivity and of the solvent radius. This last one was evaluated according to the following:

$$R_{solv} = 0.735 \left(\frac{M|_{\mathrm{g/mol}}}{d_l|_{\mathrm{g/mL}}} \right)^{1/3} \text{Å} \tag{19}$$

where M and d_l represent the molar mass and liquid density, respectively, whose values are taken either from the NIST or the [31] website.

All DFT calculations were performed with the ADF software [30] using either the PBE [32] or PBE0 [33] exchange-correlation functionals with Grimme's BJDAMP dispersion correction [34], this comparison between a pure gradient generalized approximation (GGA) and a hybrid functional allowing for discussing the possible transferability of the optimized models. The TZ2P triple-ζ basis set was used, with default numerical quality and integration grid settings. No relativistic effects were included, and no frozen core approximation was applied.

From a theoretical viewpoint, experimental values actually correspond to an average on all stable conformations. A conformational search was thus carried out for all systems by adapting an available recipe [35] in the Python Library for Automating Molecular Simulation (PLAMS) library, in which for each chemical system specified by its Simplified Molecular Input Line Entry System (SMILES) notation, the computational workflow automatically performs the following sequence of steps within the very same run:

i) The RDKit [36] generator is used to extract a three-dimensional structure from the SMILES code and to generate a set of randomized molecular geometries for extensively exploring the PES, in an almost instantaneous way.

ii) These geometries are then fully optimized at the Density-Functional based Tight-Binding (DFTB) level of theory that is a computationally cheap but reliable semi-empirical method.

iii) An energy threshold of 5.0 kcal mol^{-1} $\approx$ 20.9 kJ mol^{-1} and a structural root mean square deviation (RMSD) threshold equal to 3.0 Å are then applied to filter the most stable representative geometries.

iv) These last ones are fully reoptimized at the chosen DFT level (so that both energies and geometries are refined at this more time-consuming step).

v) A final filtering with threshold of 2.5 kcal mol^{-1} $\approx$ 10.5 kJ mol^{-1} (let us recall that RT is about 0.59 kcal mol^{-1} $\approx$ 2.48 kJ mol^{-1} at room temperature) and 1.5 Å for energy and RMSD are respectively used.

vi) Vibrational frequencies and associated physico-chemical properties are calculated on all of the retained structures (single point calculations). The absence of any imaginary values for frequencies was checked to ensure that genuine energy minima were obtained. Grimme's vibrational correction was computed using the default values as implemented in ADF.

A weight w_i is subsequently associated with each retained conformer according to Boltzmann's distribution, here based on the self-consistent-field (SCF) energy (thermal and entropy effects—which are not yet known since the models have not been built—are in general not important when dealing with conformational preferences, as recalled in the introduction):

$$w_i = Ae^{-\frac{E_i^{SCF}}{RT}} \tag{20}$$

where A is the normalization constant. For any property P, an average value is then simply obtained by (with obvious notations)

$$\overline{P} = \sum_i w_i P_i \tag{21}$$

In machine learning (ML), model parameters are those that minimize the so-called loss function, which is often built using a norm of the difference between the vector ($\overrightarrow{Y}_{ref}$) gathering the reference values and the one ($\overrightarrow{Y}_{pred}$) collecting the predicted values. Popular norms are those belonging to the p-norm family (here for a vector space of dimension n):

$$\left\| \overrightarrow{u} \right\|_p = \left(\sum_{i=1}^{n} u_i^p \right)^{1/p} \tag{22}$$

p=1 corresponds to the so-called taxicab norm associated with the mean absolute error (MAE) that is the most common performance metrics in theoretical chemistry, and p=2 to the Euclidean norm linked to the root mean square error, while p=∞ returns the maximal value. Whereas norms are equivalent in any space vector of finite dimension, the learnt models will differ from a norm to the other, since each norm emphasizes one given aspect of the value distribution. For instance, $\|.\|_2$ is more sensitive to high values, while it is insensitive to small ones. In order to reduce the learning bias induced by a particular norm, one can mix them. In this study, we consider the following interpolation for n data,

$$\left\| \overrightarrow{u} \right\|_{mix} = (1-\lambda) \left\| \overrightarrow{u} \right\|_\infty + \lambda \left\| \overrightarrow{u} \right\|_1 /n \tag{23}$$

which reduces to $\lambda \left\| \overrightarrow{u} \right\|_1 /n$ for λ=1 and to $\left\| \overrightarrow{u} \right\|_\infty$ for λ=0.

Assessing the performances of a ML model requires a proper splitting of data. Traditionally, they are split into three non-overlapping sets (to reduce overfitting, avoiding that final tests are applied on already seen data). The *training* set is made of data used to fit machine learning models under construction, the *validation* set is used to tune and control the model at this stage (for instance by determining the hyperparameters), and the *test* set is employed to provide an unbiased final evaluation of the model. Here, we will evaluate this final entropy prediction performance by assessing the mean absolute error (MAE) values according to the following:

$$MAE = \frac{1}{N_{mol}} \sum_{i=1}^{N_{mol}} \left| S_i^{exp} - S_i^{pred} \right| \tag{24}$$

All regression models were optimized using Python libraries. More specifically, ML parameters were determined using the *minimize* function available in the *scipy.optimize* Python library using the sequential least squares programming (SLSQP) algorithm. Initial guesses for the parameters to optimize were randomly generated using a uniform distribution of the [0.0,1.0] range. Similarly, bounds imposed for the optimized parameters also corresponded to the same range. This prevents to get too much high values that will have no physical interpretation, which could bring numerical instability, or that could be too much sensitive to data noise. Besides, in order to get rid of the initial random guess bias, this minimization protocol was repeated 500 times.

Such constraints actually bear some similarity with regularization procedures such as in Tikhonov (ridge) and lasso regressions in which penalty terms are added to the loss function to add parameter control. For instance, if we come back to Eq. 14, fitting without any constraint in the regression procedure may lead to negative values for α, β, or γ that are not physically motivated (the translation, rotational, and vibrational contributions should be positive), or to values that are above 1.0 for S_r and S_t, while we know that the rotational and vibrational entropies are reduced in the condensed phase.

All of the other mathematical manipulations are handled using standard *numpy* routines, while graphs were generated using *pyplot*.

Results and discussion

As explained in the previous sections, our feature set consists in the rotational, translational, and vibrational entropies from ADF calculations, taking into account the conformer populations at 298.15 K calculated using the Boltzmann distribution. This is illustrated by Fig. 2 in the case of butanone for which three conformers are retained, with populations equal to 72%, 15%, and 13% when modelled by immersion into the COSMO butanone implicit medium.

According to Eqs. 4 and 5, they feature as expected the same value for the translation entropy. Actually, they really differ from their vibrational entropy that ranges from 58.0 to 61.8 J mol^{-1} K^{-1}. In order to have more insight into the importance of a correct PES sampling, we define for each

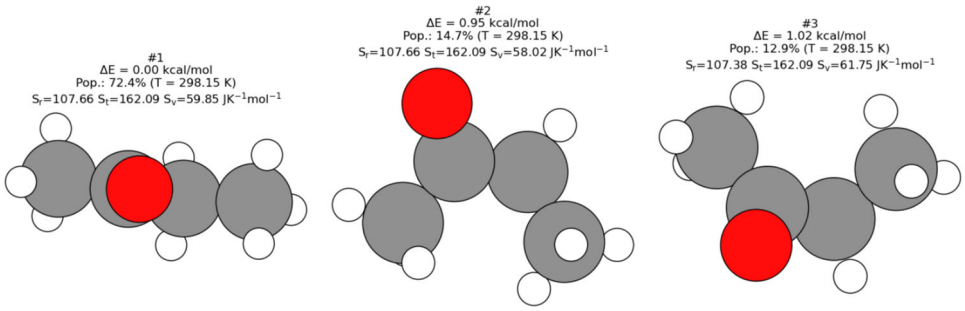


Fig. 2 Conformers selected based on potential energy surface exploration of butanone, with their respective Boltzmann weights at 298.15 K. These three conformers were identified and chosen using the method detailed in the "Computational details" section

chemical system the vibrational conformational span on all retained conformations by the following:

$$\Delta S_v^{conf} = \underset{\{conf\}}{max}\,(S_v^h) - \underset{\{conf\}}{min}\,(S_v^h) \tag{25}$$

We now complement these first results by having a quick look at the role of Grimme's anharmonic correction, as we show in Fig. 3. Three different behaviors can be identified. The first one is when $\overline{\Delta S}_v^G$ can be fully negligible. This is the case for molecules that do not exhibit low vibrational wavenumbers (typically lower than 100 cm^{-1}), as epitomized by the water molecule with its two a1 normal modes around 1600 cm^{-1} and 3700 cm^{-1}, and its b1 one at 3800 cm^{-1}. However, in the major cases, ΔS_v^G is negative, spanning the whole [-20, 0] $J K^{-1}$ mol^{-1} range, so that this correction tends to counter the entropy overestimation induced by the IG approximation, as expected. More surprisingly, positive values (between 10 and 15 $J K^{-1}$ mol^{-1}) were found in three cases, namely, artifacts probably due to the simplicity of this approach. Thus, it turns out that, in absolute value, $T\overline{\Delta S}_v^G$ can reach up to 6 kJ mol^{-1}, even for moderate-size systems, a magnitude that is far from negligible when chemical accuracy is sought.

After having focused on the vibrational component, it is also valuable to compare it with the two other ones. Figure 3 shows the Boltzmann-averaged values, denoted $\overline{S}_t$, $\overline{S}_r$, $\overline{S}_v^h$, for the whole dataset.

It appears that the translational one is always higher than the rotational one and that both predominate for systems exhibiting a total entropy lower than 300 J mol^{-1} K^{-1}). Conversely, for the molecules with the highest entropy values, $\overline{S}_v^h$ clearly becomes the major contribution. It can be also noticed that the slope for $\overline{S}_v^h$ is much higher, suggesting that it is much more sensitive to the system size or composition.

Predictive models can now be built. However, the existence of various conformers makes the task not so straightforward. Indeed, as explained in "The molecular database," section the experimental value corresponds to an average value. As discussed before, the number of retained conformers may vary from one system to the other. Two approaches (note that this chemical complexity may be addressed by more refined techniques such as multi-instance machine learning [37] that are outside the scope of this paper) are possible: either all components in the model are Boltzmann-averaged and then these values are used for fitting or each component for each conformer are corrected, and then the Boltzmann weighting is performed. Mathematically, in the presence of non-linear terms, both approaches are actually not equivalent.

The first one is certainly the simplest to implement since there is exactly one average value for each component of a

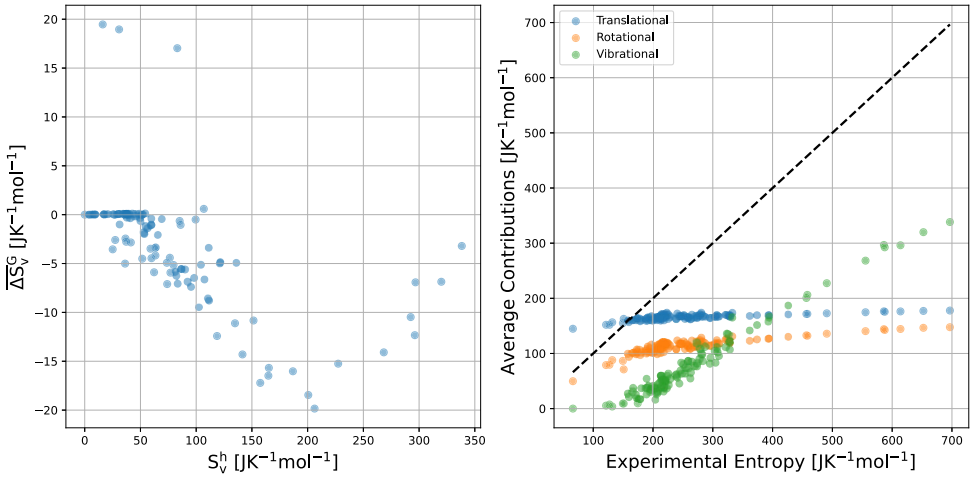


Fig. 3 *Left panel*: Comparison of harmonic contributions (x-axis) and anharmonic corrections (y-axis) in the system under study. *Right panel*: Plot of the average entropy contribution across conformers, with the experimental entropy on the x-axis and the Boltzmann-weighted average contribution on the y-axis

given system, but it would provide a model only made to evaluate average entropies. It is thus in principle not suited if one is interested in the entropy value for a specific conformation (which is useful when investigating, for instance, competitive reaction pathways). Moreover, its application requires that an exhaustive PES sampling has been performed, something that a user could not or should not want to carry out for more complex systems. The second approach, however, will not only learn how to correct entropy, but also include learning of weight correction, entangling both thermodynamical and electronic effects in an intricate way, making the task more complicated to achieve.

Obviously, both fitting methodologies reduce to the same one when only one conformer is retained, removing the ambiguity of the whole protocol. We thus decided to split our dataset into two parts. On the one hand, the *training* set will be made only of systems for which only one conformer was retained after the conformational search: it encompasses 54 molecules at the PBE level of theory. On the other hand, the remaining systems define the *validation* and *test* sets. The entropy of each conformation is then computed using the model optimized on the training set. Then, for a given system, the average entropy is calculated using the Boltzmann average and can then be compared to the experimental value.

It should added that, in a way, this splitting will give a kind of upper bound for the MAE values. Indeed, only one retained conformer usually corresponds to cases of small and rigid molecules, while multiple conformers are encountered for larger and flexible structures (not used at all for the full training). This means that the training and the test sets might be chemically quite different. Good performances on the test set for a given model would thus suggest its high versatility and would suggest that it can be trustfully applied on a large variety of systems.

Then, in order to split validation and test sets, we numbered (from 1 to 52) the systems with multiple retained conformers in increasing experimental entropy values. Systems with an odd label constitute the validation set, while the test set collects systems with even labels. By doing so, we ensure that both sets cover the full molecular range.

As explained in the "Computational details" section, training is performed by minimizing a given loss function. In the following, we will restrict our analysis to the special one expressed by Eq. 23. First, we trained the four first models, namely S_1^l (Eq. 13), S_4^l (Eq. 14), S_2^{nl} (Eq. 15), and S_7^{nl} (Eq. 16), for various values of the λ hyperparameter, by minimizing this loss function on the training set. All these models are then evaluated on the validation set by calculating the corresponding $MAE_{valid}(\lambda)$ value.

We have found that for any λ value within the [0.0,1.0] range, the optimal α parameter determined on the training set for S_1^l (Eq. 13, only one scaling coefficient) is almost constant, belonging to the [0.60, 0.62] range (a value that is close to the popular 2/3 correction), so that $MAE_{valid}(\lambda)$ is also almost constant for this one-parameter linear model, close to 37.3 J mol^{-1} K^{-1}.

The situation is much more contrasted for the non-linear extension, S_2^l that includes a normalized second-order term, still grouping translation and rotational contributions (Eq. 15). In the [0.0,0.4] λ range, the trained models converge to the same solution: α=0.578 and β=0.403, leading to a significant improvement with $MAE_{valid}(\lambda)$ equal to 19.5 J mol^{-1} K^{-1}. Conversely, for higher λ values, the optimal model changes. For instance, for λ=0.8, α=0.591 and β=0.111, deteriorating the performances on the validation set with MAE_{valid} equal to 30.8 J mol^{-1} K^{-1}.

At first sight, this could seem a counter-intuitive result. Indeed, as λ increases, the weight of $\|.\|_1$ in the loss function also increases. The optimal model on the training set is thus close to the one that minimizes the MAE on the training set. But, it appears that, concomitantly, it is less accurate, in terms of MAE, for the validation set. This can actually be rationalized by the fact that, as already mentioned, the training and the validation sets importantly differ. In a way, introducing the λ parameter allows for mitigating these differences and the data heterogeneity. It also underlines the importance of choosing relevant loss functions and the high impact on this choice in the final performances.

After reviewing these models (plotted in Fig. 4) with only one and two parameters, we can look at the four-parameter one represented by Eq. 14 (linear model in which the translational, rotational, and vibrational contributions are separately scaled). Strikingly, its performance of the validation set is disappointing with $MAE_{valid}(\lambda)$ values within the [33,41] J mol^{-1} K^{-1} range, being significantly higher than the one-parameter linear model. The best model corresponds to α=0.422, and $\beta = \gamma = \delta$=1.0, this last value being the maximal bound for optimization and leading to a suboptimal model. One could then wonder whether adding second-order terms and fully decoupling translational and rotational effects would be more efficient, leading to Eq. 16. We found that $MAE_{valid}(\lambda)$ then exhibits a high λ-dependence, varying from 23.7 J mol^{-1} K^{-1} (for $\lambda = 0$) to 47.2 J mol^{-1} K^{-1} ($\lambda = 1$), still being larger than the minimal value for S_2^l.

This last model, in spite of its simplicity (that has the advantage of preventing overfitting), can thus be considered the more satisfying one. This is the reason why we built Eq. 17 (S_3^{nlv} model) from Eq. 14 and not from Eq. 16. As shown in the left panel in Fig. 4, Eq. 14 leads to an underestimation of entropy for the highest values. As discussed earlier (see also Fig. 3), this corresponds to cases for which the vibrational entropy dominates, and also to the largest molecules. Intuitively, one thus would like to slightly increase the S_v^h weight with increasing R_{solv} values. Once more, we would like to do

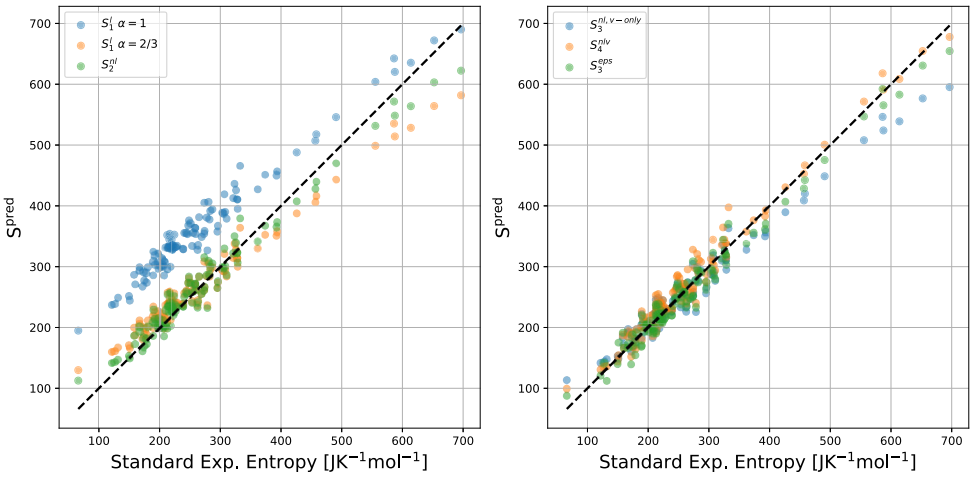


Fig. 4 Plots of the various models for entropies in solution at 298.15 K. x-axis, experimental values; y-axis, predicted values, all in J mol^{-1} K^{-1}. The black line represents a perfect correlation between experimental and predicted values

it in a controlled way and prevent the correction to take too much large values. Such a control is already at work for the mixed terms. Indeed, as all entropy components are positive, it is straightforward to show that $\frac{S_t S_v^h}{S_t+S_r+S_v^h}$, for instance, is already bounded by S_t, which becomes a minor contribution for the largest systems. Arbitrarily, we hence chose to bound the scaling f function in Eq. 17 to 20%:

$$f_\gamma(R_{solv}) = 1 + 0.2\,\widetilde{f}_\gamma(R_{solv}) \tag{26}$$

with

$$\begin{aligned} 0 \le \gamma \le 1 \\ \widetilde{f}_\gamma(0) = 0 \\ \lim_{R_{solv}\to\infty} \widetilde{f}_\gamma(R_{solv}) = 1 \end{aligned} \tag{27}$$

Simple functions obeying these conditions are sigmoids. We tested several standard flavors of them such as the logistic, arctan, and tanh functions. The best performance on the validation set was obtained using the Gudermannian function:

$$gd(x) = 2arctan(tanh(x)) \tag{28}$$

We thus optimized the following vibrational weight correction:

$$f_\gamma(R_{solv}) = 1 + 0.4\,arctan(tanh(\gamma R_{solv})) \tag{29}$$

with parameter γ constrained, as before, to belong to the [0.0,1.0] range, describing how "quick" the (unoptimized) asymptotic limit is reached. For λ=0, the optimized γ value was found equal to 0.0, so that this S_3^{nlv} model exactly reduces to S_2^{nl}, and this remains the case on the whole [0.0,0.50] λ range. Then, $MAE_{valid}(\lambda)$ decreases, reaching its minimum (17.9 J mol^{-1} K^{-1}) around λ=0.60 (with α=0.582, β=0.238, γ=0.131), before significantly increasing for higher λ values. The best model represents in fact a noticeable improvement with respect to the S_2^{nl} model. The right panel of Fig. 4 shows the obtained model, with an overall MAE of 16.6 J mol^{-1} K^{-1}.

Nevertheless, it appears that the highest values remain underestimated, while, on the opposite, the water molecule (lowest experimental value) is importantly overestimated. The first family gathers molecules with relative permittivity values around 2.0, while water distinguishes itself by its very high ϵ_r value (78.5). In order to cure these two deficiencies of the S_3^{nlv} model, an additional function of ϵ_r can be introduced (see Eq. 16). Ideally, it should give a positive value near 2.0, a negative one for water, and should be almost vanishing for the other systems. We thus proposed the following ansatz:

$$g_{delta}(\epsilon_r) = \delta ln(A/\epsilon_r) \tag{30}$$

The A value determines the transition between negative (for high ϵ_r) to positive (low ϵ_r) corrections. A simple choice is to choose it as the median ϵ_r value, that is to say, 5.82. Only δ is then still to be determined. Here, we decided to fix its value (it must be noticed that we did not reoptimize the other parameters) so that the standard deviation between MAE_{pred}, MAE_{valid}, and MAE_{test} is the smallest possible. This was reached with $\delta = 5.4$. It turned out that the corresponding S_3^{nlv} model improves over S_2^{nlv}, with a MAE on the full database equal to 15.4 J mol^{-1} K^{-1}. This corresponds to an energy at room temperature (multiplying by 298.15 K) equal to 4.6 kJ mol^{-1} $\approx$1.1 kcal mol^{-1}. In other words, this four-parameter non-linear model reaches the so-called "chemical accuracy."

Continuing to focus on the idea that the main contribution to the entropy trend comes from the vibrational component, we propose the following three-parameter model:

$$S_3^{nl,v-only} = (S_v^h)^{1+\alpha} - \beta\epsilon_r + \gamma \tag{31}$$

We constrained β to the range $0.0 \le \beta \le 1.0$ and γ to the range $0.0 \le \gamma \le 200.0$. Minimizing the mean absolute error

Table 1 Mean absolute error (MAE) values for the various trained models with respect to experimental entropy values (in J mol^{-1} K^{-1})

Model	MAE_{train}	MAE_{valid}	MAE_{pred}	MAE_{tot}
S_1^l, α=1.0	111.0	82.6	82.2	97.0
S_1^l, α=2/3	21.7	27.9	27.5	24.6
S_2^{nl}	16.0	19.4	19.3	17.7
S_3^{nlv}	15.2	17.9	18.3	16.6
S_4^{nlv}	15.4	15.3	15.5	15.4
$S_3^{nl,v-only}$	15.0	17.3	16.5	15.9

(MAE_{valid}) yielded $\lambda = 0.5$ as the optimal value, with $\alpha = 0.045$, $\beta = 0.904$, and $\gamma = 166$. The performance of this model is shown in the right panel of Fig. 4 and summarized in Table 1.

Despite its promising accuracy, the physical interpretation of the $S_3^{nl,v-only}$ model remains unclear due to several factors.

First, the model does not explicitly account for the rotational or translational contributions to entropy. While these may not be numerically significant, they are important in understanding the overall entropy behavior. Second, the application of a fractional exponent to the vibrational entropy introduces concerns regarding its physical justification. Though this approach works well for the systems in our dataset where vibrational contributions are predominant and avoids underestimation, it carries the risk of a significant "acceleration" in entropy for systems with extremely large vibrational effects, potentially leading to overestimation. Finally, the constant term γ implies a non-zero entropy even in the absence of any elements, which prompts further questions: does γ represent an intrinsic entropy of solvation, or is it capturing an average contribution from rotational and translational entropy? This leaves open questions about broader applicability and scalability.

Finally, we briefly discuss the transferability of our models by applying them to data obtained using the PBE0 exchange-correlation hybrid functional, instead of GGA PBE. For this purpose, all geometries were fully reoptimized with PBE0, followed by frequency calculations. We then applied our final S_4^{nlv} (Eq. 18) and $S_3^{nl,v-only}$ (Eq. 31) models using the parameter values initially derived with PBE (i.e., no refitting). Pleasingly, the respective MAEs for the entire dataset remained very close to the previous ones: 15.7 and 16.8 J mol^{-1} K^{-1}, respectively. These results increase our confidence that the models can be reliably used with any exchange-correlation functional.

Conclusions

In this paper, we developed and evaluated various models for entropies in solution within implicit solvation approaches with increasing complexity and physical considerations, involving only a few parameters and only features that are available from any quantum chemical calculations. Our models were subsequently trained on publicly available experimental values using different metrics, careful parameter control, and data splitting. The best-performing models were found to be transferable and close to the chemical accuracy, at a negligible computational cost.

Supplementary information Data used to build, validate and test the predictive models.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s00894-024-06225-3.

Acknowledgements The Centre Régional Informatique et d'Applications Numériques de Normandie (CRIANN) is acknowledged for providing access to computational resources.

Author contribution All authors have contributed to the conceptualization, scientific research, writing, and review of the paper.

Funding This work has been partially supported by University of Rouen Normandy, INSA Rouen Normandy, the "Centre National de la Recherche Scientifique" (CNRS), the European Regional Development Fund (ERDF), Labex SynOrg (ANR-11-LABX-0029), Carnot Institut I2C, the graduate school for research XL-Chem (ANR-18-EURE-0020 XL CHEM), and the "Région Normandie."

Data availability No datasets were generated or analyzed during the current study.

Declarations

Conflict of interest The authors declare no competing interests.

Dedication It is a pleasure to dedicate this paper to our colleague and friend Prof. Alejandro Toro-Labbé, in recognition of his inspiring works, among others, on chemical reactivity and conceptual DFT.

References

1. Tantillo DJ (2022) Portable models for entropy effects on kinetic selectivity. J Am Chem Soc 144(31):13996–14004. https://doi.org/10.1021/jacs.2c04683
2. Harvey JN, Himo F, Maseras F et al (2019) Scope and challenge of computational methods for studying mechanism and reactivity in homogeneous catalysis. ACS Catal 9(8):6803–6813. https://doi.org/10.1021/acscatal.9b01537
3. Watson L, Eisenstein O (2002) Entropy explained: the origin of some simple trends. J Chem Educ 79(10):1269. https://doi.org/10.1021/ed079p1269
4. Abraham MH (1981) Relationship between solution entropies and gas phase entropies of nonelectrolytes. J Am Chem Soc 103(22):6742–6744. https://doi.org/10.1021/ja00412a036
5. Leung BO, Reid DL, Armstrong DA et al (2004) Entropies in solution from entropies in the gas phase. J Phys Chem A 108(14):2720–2725. https://doi.org/10.1021/jp030265a

6. Liu SC, Zhu XR, Liu DY et al (2023) DFT calculations in solution systems: solvation energy, dispersion energy and entropy. Phys Chem Chem Phys 25(2):913–931. https://doi.org/10.1039/d2cp04720a
7. Garza AJ (2019) Solvation entropy made simple. J Chem Theory Comput 15(5):3204–3214. https://doi.org/10.1021/acs.jctc.9b00214
8. Ariai J, Gellrich U (2023) The entropic penalty for associative reactions and their physical treatment during routine computations. Phys Chem Chem Phys 25(20):14005–14015. https://doi.org/10.1039/d3cp00970j
9. Pracht P, Grimme S (2021) Calculation of absolute molecular entropies and heat capacities made simple. Chem Sci 12(19):6551–6568. https://doi.org/10.1039/d1sc00621e
10. Gorges J, Grimme S, Hansen A et al (2022) Towards understanding solvation effects on the conformational entropy of non-rigid molecules. Phys Chem Chem Phys 24(20):12249–12259. https://doi.org/10.1039/d1cp05805c
11. Conquest OJ, Roman T, Marianov A et al (2021) Calculating entropies of large molecules in aqueous phase. J Chem Theory Comput 17(12):7753–7771. https://doi.org/10.1021/acs.jctc.1c00848
12. Besora M, Vidossich P, Lledós A et al (2018) Calculation of reaction free energies in solution: a comparison of current approaches. J Phys Chem A 122(5):1392–1399. https://doi.org/10.1021/acs.jpca.7b11580
13. Michel C, Laio A, Milet A (2009) Tracing the entropy along a reactive pathway: the energy as a generalized reaction coordinate. J Chem Theory Comput 5(9):2193–2196. https://doi.org/10.1021/ct900177h
14. Lipparini F, Mennucci B (2016) Perspective: polarizable continuum models for quantum-mechanical descriptions. J Chem Phys 144(16). https://doi.org/10.1063/1.4947236
15. Gaussian Inc. (2024) Thermochemistry in Gaussian. Accessed 06 Aug 2024
16. Grimme S (2012) Supramolecular binding thermodynamics by dispersion-corrected density functional theory. Chem Eur J 18(32):9955–9964. https://doi.org/10.1002/chem.201200497
17. Chai JD, Head-Gordon M (2008) Long-range corrected hybrid density functionals with damped atom-atom dispersion corrections. Phys Chem Chem Phys 10(44):6615. https://doi.org/10.1039/b810189b
18. Dzib E, Merino G (2021) The hindered rotor theory: a review. Wiley Interdiscip Rev Comput Mol Sci 12(3). https://doi.org/10.1002/wcms.1583
19. Vansteenkiste P, Van Speybroeck V, Marin GB et al (2003) Ab initio calculation of entropy and heat capacity of gas-phase n-alkanes using internal rotations. J Phys Chem A 107(17):3139–3145. https://doi.org/10.1021/jp027132u
20. Ardura D, López R, Sordo TL (2005) Relative Gibbs energies in solution through continuum models: effect of the loss of translational degrees of freedom in bimolecular reactions on Gibbs energy barriers. J Phys Chem B 109(49):23618–23623. https://doi.org/10.1021/jp0540499
21. Sumimoto M, Iwane N, Takahama T et al (2004) Theoretical study of trans-metalation process in palladium-catalyzed borylation of iodobenzene with diboron. J Am Chem Soc 126(33):10457–10471. https://doi.org/10.1021/ja040020r
22. Cooper J, Ziegler T (2002) A density functional study of SN2 substitution at square-planar platinum(II) complexes. Inorg Chem 41(25):6614–6622. https://doi.org/10.1021/ic020294k
23. Tobisch S (2005) Organolanthanide-mediated ring-opening Ziegler polymerization (ROZP) of methylenecycloalkanes: a theoretical mechanistic investigation of alternative mechanisms for chain initiation of the samarocene-promoted rozp of 2-phenyl-1-methylenecyclopropane. Chem Eur J 11(10):3113–3126. https://doi.org/10.1002/chem.200401102
24. Chen P, Dougan BA, Zhang X et al (2013) Reactions of a tungsten alkylidyne complex with mono-dentate phosphines: thermodynamic and theoretical studies. Polyhedron 58:30–38. https://doi.org/10.1016/j.poly.2012.07.042
25. Jouanno LA, Di Mascio V, Tognetti V et al (2014) Metal-free decarboxylative hetero-diels-alder synthesis of 3-hydroxypyridines: a rapid access ton-fused bicyclic hydroxypiperidine scaffolds. J Org Chem 79(3):1303–1319. https://doi.org/10.1021/jo402729a
26. Linstrom P (1997) NIST Chemistry Webbook, NIST Standard Reference Database 69. https://doi.org/10.18434/T4D303
27. SCM TC (2023) COSMO: conductor-like screening model. https://www.scm.com/doc/ADF/Input/COSMO.html, software for Chemistry & Materials, Amsterdam, The Netherlands
28. Gaussian I (2023) SCRF: self-consistent reaction field. https://gaussian.com/scrf/, gaussian, Inc., Wallingford CT
29. Klamt A (1995) Conductor-like screening model for real solvents: a new approach to the quantitative calculation of solvation phenomena. J Phys Chem 99(7):2224–2235. https://doi.org/10.1021/j100007a062
30. te Velde G, Bickelhaupt FM, Baerends EJ et al (2001) Chemistry with ADF. J Comput Chem 22(9):931–967. https://doi.org/10.1002/jcc.1056
31. Stenutz R (2023) Rolf Stenutz's chemistry pages. https://www.stenutz.eu/chem/
32. Perdew JP, Burke K, Ernzerhof M (1996) Generalized gradient approximation made simple. Phys Rev Lett 77(18):3865–3868. https://doi.org/10.1103/physrevlett.77.3865
33. Adamo C, Barone V (1999) Toward reliable density functional methods without adjustable parameters: the PBE0 model. J Chem Phys 110(13):6158–6170. https://doi.org/10.1063/1.478522
34. Grimme S, Ehrlich S, Goerigk L (2011) Effect of the damping function in dispersion corrected density functional theory. J Comput Chem 32(7):1456–1465. https://doi.org/10.1002/jcc.21759
35. SCM TC (2023) PLAMS interfaces: conformers. https://www.scm.com/doc/plams/interfaces/conformers.html, software for Chemistry & Materials, Amsterdam, The Netherlands
36. SCM TC (2023) PLAMS components: RDKit. https://www.scm.com/doc/plams/components/mol_rdkit.html, software for Chemistry & Materials, Amsterdam, The Netherlands
37. Zankov D, Madzhidov T, Varnek A et al (2023) Chemical complexity challenge: is multi-instance machine learning a solution? Wiley Interdiscip Rev Comput Mol Sci 14(1). https://doi.org/10.1002/wcms.1698

Chapter

6 Conclusions

This thesis has led to the development of a coherent and versatile methodology for studying chemical reactivity in solution, grounded in the tools of modern theoretical chemistry.

From a methodological standpoint, we have implemented new routines in the AMS software to compute atomic dipole moments and polarisabilities, as well as improved the calculation of local properties and extended their implementation in QTAIM. These features enrich the set of available descriptors, providing a local and physically interpretable perspective on chemical reactivity.

We also proposed a protocol for modelling conformational effects in solution. This protocol relies on conformer sampling followed by statistical averaging based on the Boltzmann distribution. It has been successfully applied to the study of thermodynamic properties such as entropy, contributing to a more realistic and comprehensive description of reaction mechanisms in the condensed phase.

In parallel, a structured database was constructed from results obtained via QTAIM and CDFT, also including solvent aggregates. While the development of machine learning models remains incomplete, the dataset generated is already sufficient to support exploratory ML approaches and lays the groundwork for future generalisation.

Finally, a module dedicated to the study of excited states was developed within the scope of this thesis, with a view towards future spectroscopic applications. This development is functional but still requires finalisation and validation.

In summary, this thesis lays the foundation for a complete and extensible digital toolbox for studying chemical reactivity in solution. The avenues opened —especially those related to excited states and the integration of machine learning techniques— offer promising prospects for future work at the interface between theory, simulation, and data modelling.

Appendix

A Supplementary Scientific Details

This chapter gathers scientific material that, for reasons of scope and readability, could not be fully inserted in the main body of the thesis. While not essential to follow the principal narrative, these details are provided to offer a fuller and more transparent account of the work undertaken. They include explicit proofs, coordinates of systems, output files, structural data of developed code, plain text output files, and benchmarks that were taken during the course of the research.

In some cases, the material here expands upon topics briefly mentioned in earlier chapters; in others, it provides technical depth or alternative perspectives that may be of interest to the reader.

A.1 Creation of a Geodesic Polyhedron (Icosphere)

To initialise the gradient path tracing process, we need to generate a structure that resembles a sphere around the (R/C)CP. Computationally, this involves implementing a set of equidistant points around the CP, which will serve as starting points for tracing the gradient.

A geodesic polyhedron provides an effective means of distributing equidistant points. It offers a balance between computational efficiency and accuracy. This can be achieved in two main ways: the brute-force method and the recursive method.

While the recursive method is more elegant and efficient for constructing higher-order geodesic polyhedra, the brute-force method is better suited to our requirements. The polyhedra we use do not demand a high level of recursion, and as illustrated in Figure A.1, for the levels of icosphere relevant to our study, the brute-force method is computationally less expensive.

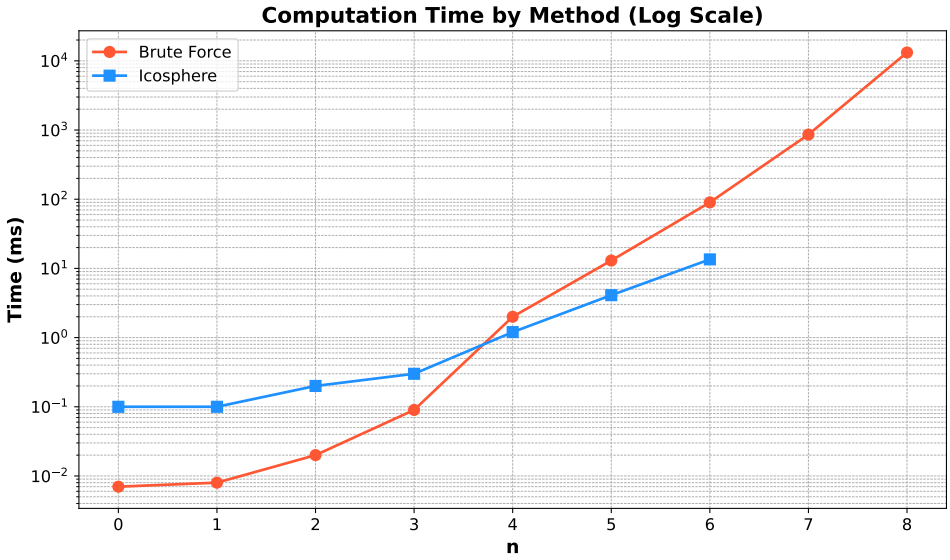


Figure A.1. Time comparison between the brute-force and recursive methods for creating an icosphere, time in milliseconds.

The brute-force method is detailed in Algorithm 8, while the recursive method is described in Algorithm 9.

Algorithm 8: Brute-force method for creating an icosphere.

```
1  φ ← (1+√5)/2;
2  radius ← √(2+φ);
3  norm ← 2φ;
4  nver ← 10 · 4^order + 2 ;                          // number of vertices
5  edge ← 2 ;                                         // analytical edge length
6  vertex ← zeros((nver, 3));
7  vertex[1:12,:] ← [0, ±1, ±φ], [±1, ±φ, 0], [±φ, 0, ±1]
8  for i ← 1 to order do
9      nverb ← 10 × 4^(i−1) + 2 ;                     // number of vertices at previous order
10     l ← nverb;
11     for j ← 1 to nverb-1 do
12         for k ← j + 1 to nverb do
13             if distance(j,k) ≤ edge then
14                 l ← l + 1;
15                 vertex[l,:] ← radius × midPoint(j,k)/norm;
16     edge ← distance(vertex[1,:], vertex[nverb+1,:])
```

Algorithm 9: Recursive method for creating an icosphere.

```
φ ← (1+√5)/2;
nver ← 10 · 4^order + 2 ;                                   // number of vertices
vertex ← zeros((nver, 3));
vertex[1:12,:] ← [0, ±1, ±φ], [±1, ±φ, 0], [±φ, 0, ±1];
triangles ← Array( ;                                        // how the triangles are connected
0, 11, 5, 0, 5, 1, 0, 1, 7, 0, 7, 10, 0, 10, 11, ;
11, 10, 2, 5, 11, 4, 1, 5, 9, 7, 1, 8, 10, 7, 6, ;
3, 9, 4, 3, 4, 2, 3, 2, 6, 3, 6, 8, 3, 8, 9, ;
9, 8, 1, 4, 9, 5, 2, 4, 11, 6, 2, 10, 8, 6, 7)
midCache = {};
Function addMidPoint(a, b):
    key ← math.floor((a + b)*(a + b + 1)/2) + math.min(a, b);
    i ← midCache.get(key);
    if i ≠ undefined then
        midCache.delete(key);
        return i;
    midCache.set(key, v);
    for k ← 0 to 3 do
        vertices[3 * v + k] ← (vertices[3 * a + k] + vertices[3 * b + k]) / 2;
    i ← v++;
    return i;
trianglesPrev ← triangles;
for i ← 0 to order do
    triangles ← Array(trianglesPrev.length * 4);
    for k ← 0 to trianglesPrev.length step 3 do
        for n ← 1 to 3 do
            v_n ← trianglesPrev[k + (n - 1)];
        t ← k * 4;
        triangles[t++] ← v1; triangles[t++] ← a; triangles[t++] ← c;
        triangles[t++] ← v2; triangles[t++] ← b; triangles[t++] ← a;
        triangles[t++] ← v3; triangles[t++] ← c; triangles[t++] ← b;
        triangles[t++] ← a; triangles[t++] ← b; triangles[t++] ← c
    trianglesPrev ← triangles;
vertices ← normalise(vertices);
```

A.2 AMS Directory Structure

The AMS driver is a large project composed of many interdependent components, and its directory structure reflects this complexity. For the benefit of any future developer reading this document, this section provides a concise overview of the organisation of the AMS directory tree, outlining the purpose of its main folders.

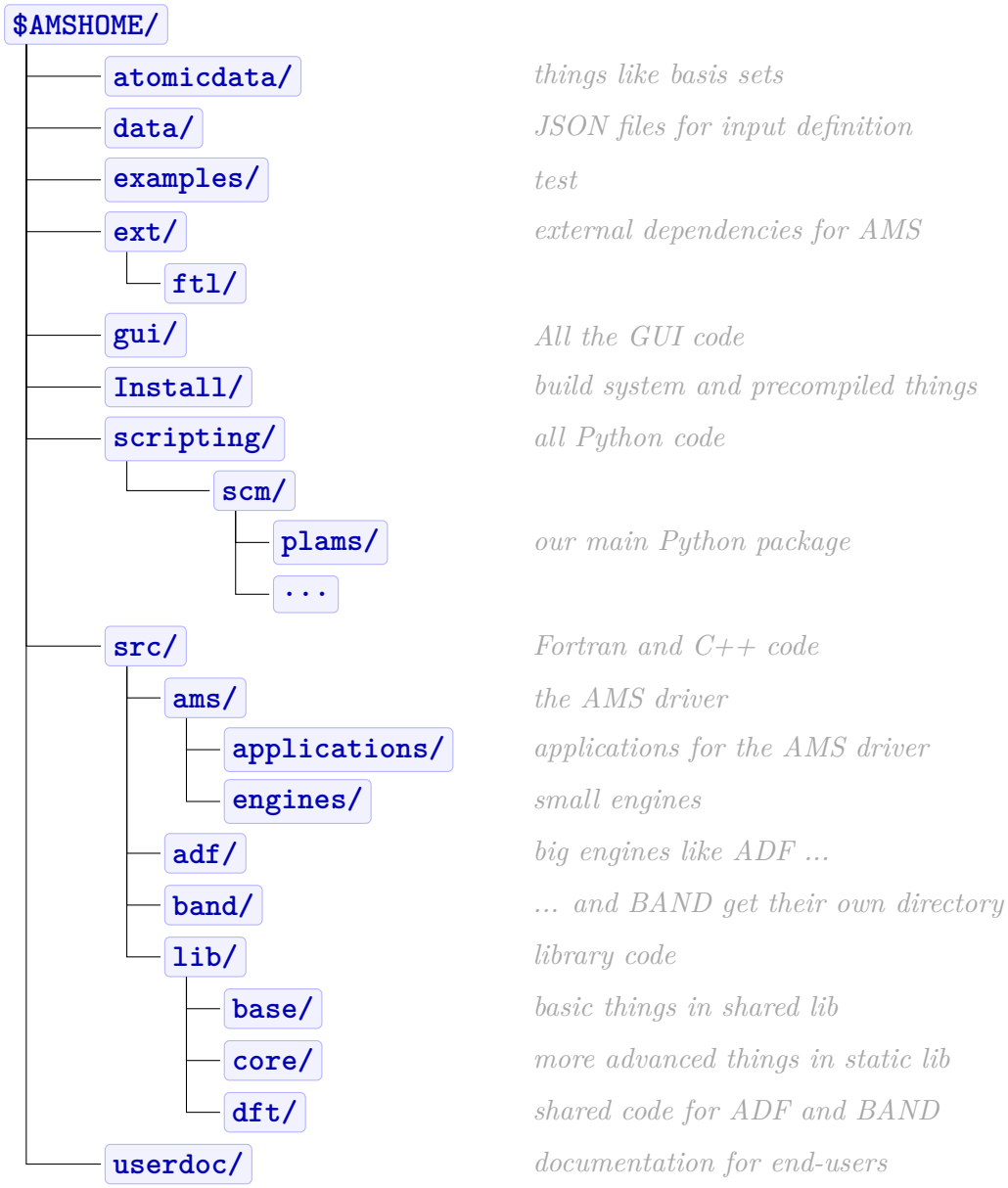


Figure A.2. AMS directory structure with perfectly aligned comments.

A.3 Zero Nuclear Contribution

The following graphs illustrate all possible subsets that can occur within a given system, each of which may be regarded as part of a larger set. In the particular cases shown here, all atoms within the subset carry zero charge. If this condition holds, then the inclusion of any additional atom into the subset will also yield zero charge, as the added atom will necessarily be bonded to an entire subsystem already carrying zero charge. In practice, a nuclear contribution of the dipole moment of exactly zero implies all atoms have zero charge, a situation that is highly improbable in most chemical systems.

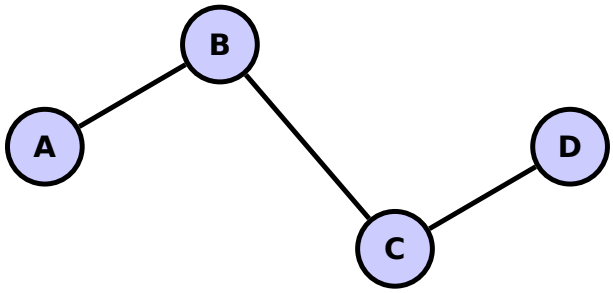


Figure A.3. System of atoms bonded in a chain.

trivial case:

$$if\ q(\mathrm{A}|\mathrm{B}) = 0$$
$$\implies q(\mathrm{A}) = q(\mathrm{B}) = 0 \tag{A.1}$$

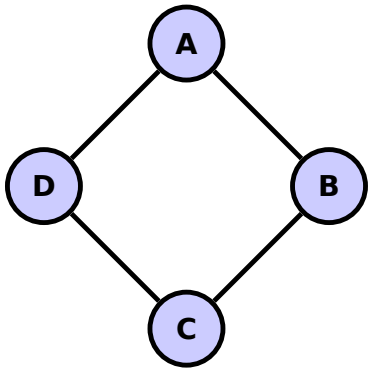


Figure A.4. System of atoms bonded in a ring.

$$
\begin{aligned}
&if\ q(\mathrm{A}|\mathrm{B}) = 0 \\
&\qquad \implies q(\mathrm{A}) = q(\mathrm{A}|\mathrm{D}) \\
&\qquad \implies q(\mathrm{B}) = q(\mathrm{B}|\mathrm{C}) \\
&\qquad \implies q(\mathrm{C}) = q(\mathrm{C}|\mathrm{B}) + q(\mathrm{C}|\mathrm{D}) \\
&\qquad \implies q(\mathrm{A}|\mathrm{B}) + q(\mathrm{B}|\mathrm{C}) + q(\mathrm{C}|\mathrm{D}) + q(\mathrm{D}|\mathrm{A}) = 0 \\
&\qquad \implies q(\mathrm{B}) + q(\mathrm{C}) - q(\mathrm{C}|\mathrm{B}) - q(\mathrm{A}) = 0 \\
&\qquad \implies q(\mathrm{D}) = q(\mathrm{D}|\mathrm{A}) + q(\mathrm{D}|\mathrm{C}) \\
&\qquad \implies q(\mathrm{A}) = -q(\mathrm{D}) + q(\mathrm{D}|\mathrm{C}) \\
&\qquad \implies 2q(\mathrm{B}) + q(\mathrm{C}) = q(\mathrm{A}) \\
&\qquad \implies q(\mathrm{A}) = -q(\mathrm{D}) + q(\mathrm{D}|\mathrm{C}) \\
&\qquad \implies q(\mathrm{A}) = q(\mathrm{B}) + q(\mathrm{C}) - q(\mathrm{C}|\mathrm{B}) \\
&\qquad \therefore q(\mathrm{B}) = -q(\mathrm{C}|\mathrm{B}) = -q(\mathrm{B})
\end{aligned}
\tag{A.2}
$$

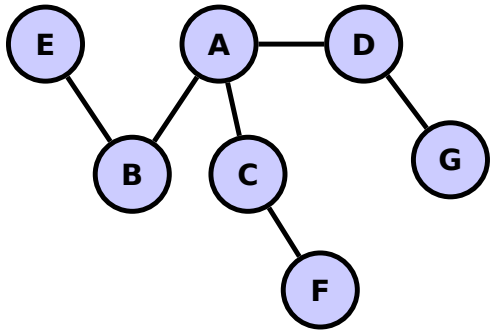


Figure A.5. System of atoms bonded in a branched chain.

$$
\begin{aligned}
&if\ q(\mathrm{A}|\mathrm{B}) = 0 \\
&\qquad \implies q(\mathrm{B}) = q(\mathrm{E}) \\
&\qquad \implies q(\mathrm{A}) = q(\mathrm{A}|\mathrm{C}) + q(\mathrm{A}|\mathrm{D}) \\
&\qquad \implies q(\mathrm{D}) = q(\mathrm{D}|\mathrm{G}) + q(\mathrm{D}|\mathrm{A}) \\
&\qquad \implies q(\mathrm{G}) = q(\mathrm{G}|\mathrm{D}) \\
&\qquad \implies q(\mathrm{D}) = -q(\mathrm{G}) - q(\mathrm{A}) + q(\mathrm{A}|\mathrm{C}) \\
&\qquad \implies q(\mathrm{A}) = q(\mathrm{A}|\mathrm{C}) + q(\mathrm{A}|\mathrm{D}) + q(\mathrm{A}|\mathrm{B}) \\
&\qquad \implies q(\mathrm{A}) = -q(\mathrm{G}) - q(\mathrm{D}) + q(\mathrm{A}|\mathrm{C}) \\
&\qquad \implies q(\mathrm{A}|\mathrm{D}) + q(\mathrm{A}|\mathrm{B}) = -q(\mathrm{G}) - q(\mathrm{D}) \\
&\qquad \implies q(\mathrm{A}|\mathrm{D}) = -q(\mathrm{G}) - q(\mathrm{D}) = -q(\mathrm{A}) - q(\mathrm{A}|\mathrm{C}) \\
&\qquad \implies q(\mathrm{A}|\mathrm{D}) = -q(\mathrm{A}|\mathrm{C}) - q(\mathrm{A}|\mathrm{B}) - q(\mathrm{A}|\mathrm{C}) \\
&\qquad \implies 2q(\mathrm{A}|\mathrm{D}) = -2q(\mathrm{A}|\mathrm{C}) \\
&\qquad \implies q(\mathrm{C}|\mathrm{A} = -q(\mathrm{A}) - q(\mathrm{A}|\mathrm{C}) \\
&\qquad \therefore q(A) = q(\mathrm{A}|\mathrm{C}) - q(\mathrm{A}|\mathrm{C}) = 0
\end{aligned}
\tag{A.3}
$$

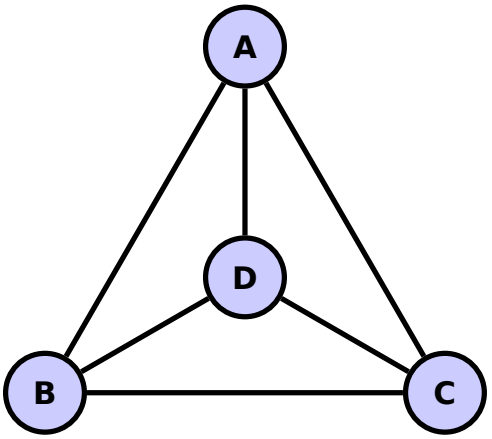


Figure A.6. System of atoms bonded in a cage.

$$
\begin{aligned}
&if\ q(\mathrm{A}|\mathrm{B}) = 0 \\
&\qquad \implies q(\mathrm{A}) = q(\mathrm{A}|\mathrm{C}) + q(\mathrm{A}|\mathrm{D}) \\
&\qquad \implies q(\mathrm{B}) = q(\mathrm{B}|\mathrm{C}) + q(\mathrm{B}|\mathrm{D}) \\
&\qquad \implies q(\mathrm{B}|\mathrm{C}) = -q(\mathrm{C}|\mathrm{D}) \\
&\qquad \implies q(\mathrm{B}|\mathrm{D}) = -q(\mathrm{D}|\mathrm{A}) \\
&\qquad \implies q(\mathrm{B}|\mathrm{C}) + q(\mathrm{C}|\mathrm{D}) + q(\mathrm{D}|\mathrm{B}) = 0 \\
&\qquad \implies q(\mathrm{A}|\mathrm{C}) + q(\mathrm{C}|\mathrm{D}) + q(\mathrm{D}|\mathrm{A}) = 0 \\
&\qquad \implies q(\mathrm{B}) - q(\mathrm{B}|\mathrm{D}) + q(\mathrm{C}|\mathrm{D}) = 0 \\
&\qquad \implies q(\mathrm{B}) - q(\mathrm{B}|\mathrm{C}) + q(\mathrm{D}|\mathrm{A}) = 0 \\
&\qquad \implies q(\mathrm{B}|\mathrm{C}) - q(\mathrm{D}|\mathrm{A}) - q(\mathrm{B}|\mathrm{C}) + q(\mathrm{C}|\mathrm{D}) = 0 \\
&\qquad \implies q(\mathrm{D}|\mathrm{B}) + q(\mathrm{D}|\mathrm{A}) - q(\mathrm{B}|\mathrm{C}) = 0 \\
&\qquad \implies q(\mathrm{B}|\mathrm{C}) + q(\mathrm{B}) - 2q(\mathrm{B}|\mathrm{C}) = 0 \\
&\qquad \implies q(\mathrm{B}) = q(\mathrm{B}|\mathrm{C}) \implies q(\mathrm{B}|\mathrm{D}) = 0 \\
&\qquad \implies q(\mathrm{B}|\mathrm{C}) = q(\mathrm{C}|\mathrm{D}) = q(\mathrm{B}) \\
&\qquad \therefore q(\mathrm{B}) = 0
\end{aligned}
\tag{A.4}
$$

A.4 System's Coordinates

Throughout this thesis, various molecular systems were computed both to analyse their intrinsic properties and to validate our implementation. For brevity, the Cartesian coordinates of these systems have been omitted from the main chapters. They are instead collected here in `xyz` format for reference and reproducibility. All coordinates are reported in Å.

For the case of the Be_3^{-2} system, the geometries can be generated using the code example provided in Section 2.10.1. In that example, a series of equilateral triangles is constructed with side lengths of 2.700, 2.200, 2.150, 2.120, 2.090, 2.070, 1.960, and 1.800 Å.

For the set of ∼45k systems used to compare the dipole moment implementation, the full dataset (∼180 MB) is available upon request from the author.

$(\eta^5\text{-}C_5H_5)_2Fe$ Ferrocene

```
21
Fe         0.000000    0.000000    0.000000
C          1.215650    0.000000    1.600813
C          0.375656   -1.156152    1.600813
C         -0.983481   -0.714541    1.600813
C         -0.983481    0.714541    1.600813
C          0.375656    1.156152    1.600813
C          1.215650    0.000000   -1.600813
C          0.375656    1.156152   -1.600813
C         -0.983481    0.714541   -1.600813
C         -0.983481   -0.714541   -1.600813
C          0.375656   -1.156152   -1.600813
H          2.310827    0.000000    1.629796
H          0.714085   -2.197727    1.629796
H         -1.869498   -1.358270    1.629796
H         -1.869498    1.358270    1.629796
H          0.714085    2.197727    1.629796
H          2.310827    0.000000   -1.629796
H          0.714085    2.197727   -1.629796
H         -1.869498    1.358270   -1.629796
H         -1.869498   -1.358270   -1.629796
H          0.714085   -2.197727   -1.629796
```

$C_{10}H_4$ Deformated C-ring

```
14
C -0.098296  0.033726  0.000358
C  0.960203  0.950023  0.015468
C  2.278483  0.545408  0.014216
C  3.427410  0.063394  0.011682
C  4.639871 -0.593498  0.008144
C  4.727889 -1.990708 -0.009113
H  5.702426 -2.460995 -0.011669
C  3.549127 -2.704959 -0.022249
C  2.296096 -2.732170 -0.026925
C  1.076724 -2.220698 -0.024487
C  0.218107 -1.307710 -0.015738
H  5.559802 -0.019573  0.018432
H  0.725030  2.008469  0.028042
H -1.116708  0.399408  0.001653
```

gdb_str_10035 Extra atom in a ring

35

```
C  5.2243243922   3.7528607775  -0.5434970940
C  4.8531893626   3.7954419414  -1.7869426732
C  5.6006347196   3.2764308271  -2.9310574593
C  5.5366699842   1.7920523889  -3.1405919799
O  5.8495941156   1.3987111954  -4.4483969720
C  6.3458302802   0.8895691709  -2.1961630992
C  7.7829243276   0.9824346857  -1.8028417201
C  8.2393854952   0.4888878488  -0.6660737631
C  7.4181788045  -0.2763628819   0.3664226785
C  8.2958514624  -0.9145162272   1.4754740210
C  8.6882674777   1.8243330057  -2.7217745509
C  8.8731031795   3.1214967664  -1.9010847295
C  10.0077787239  1.1887588552  -3.0187918765
H  4.5965096141   4.1515561122   0.2008188585
H  6.1378799344   3.2626787157  -0.2882259666
H  3.9586761330   4.3673379728  -2.0793200106
H  6.6473675102   3.4283877406  -2.7675753548
H  5.2799636740   3.8624288736  -3.9172475040
H  4.4613704798   1.5064780507  -3.0278889658
H  5.2100470529   1.8218485509  -5.0361251036
H  6.3097708215  -0.1484404307  -2.7114127393
H  5.7634847758   0.6982725490  -1.2965429164
H  9.2620459698   0.6292930773  -0.3802456750
H  6.8225024908  -1.0218233117  -0.1367130443
H  6.7326385172   0.4105657441   0.8427750721
H  9.2966166771  -0.7902729519   1.0166550814
H  8.0102729037  -1.9370614815   1.6870318472
H  8.4421411568  -0.4237294470   2.4143101061
H  8.1368562989   2.0550509149  -3.5716959073
H  9.7323074571   2.9522297587  -1.2576228368
H  8.0676484938   3.2719238021  -1.2056055693
H  9.0696459552   4.0599414957  -2.4224585628
H  10.6529545440  1.2408423862  -2.1091429336
H  10.6173445404  1.7371421590  -3.7491005670
H  9.9024472541   0.1063793097  -3.1114731341
```

gdb_str_10410 A CCP between two rings

31

```
C 3.0583589618 -1.5268605492  0.4895642720
C 2.7936846691 -0.2918901828 -0.2739277581
C 1.2706580188  0.0134899561 -0.2178445828
C 3.4190257622  0.9945762486  0.3941816882
C 4.9923235415  1.0017847849  0.6379138300
N 5.6280632217  0.6345534052  1.6347645046
N 4.9740909377  0.3079717667  2.8030256966
O 5.6946207353  1.3511141692 -0.4746866841
C 5.3153663061  2.4562305427 -1.1866245882
C 3.8745459277  2.4641613720 -1.5886436258
C 3.4770226059  3.6419742594 -2.5019506728
C 3.0641645669  2.3380437576 -0.2713000937
O 3.2204176493  3.5626447602  0.4321001942
H 2.7150134476 -2.3725859339 -0.1177187911
H 2.5011876633 -1.5508866523  1.4174295525
H 4.0982417399 -1.6861561547  0.6968867325
H 3.1094867442 -0.5045379305 -1.3415928649
H 0.8246432052  0.6398615017  0.5639394284
H 0.6776250077 -0.9439233998 -0.0648877033
H 0.8291446600  0.3166765180 -1.1661130628
H 3.1465090673  0.9856823856  1.4711652536
H 4.1374815577 -0.2870507138  2.6720040788
H 5.6204241508  0.1525239343  3.5592060150
H 6.0294692630  2.5010391625 -2.0436087620
H 5.6349487436  3.3619912624 -0.6641787497
H 3.6265096329  1.5773865427 -2.1514702223
H 3.5512912996  4.5785252168 -1.9628383734
H 2.4434331078  3.5990937234 -2.8260684956
H 4.1346165742  3.6426252875 -3.3885323320
H 2.0022163598  2.2390914723 -0.5025507847
H 3.9631111228  3.4698131808  0.9921469288
```

Dicyclopentadiene Transition state

22

```
H        -1.10532017523107       1.96140412357569      -1.27240532794643
C        -0.12233795920601       1.93675797138191      -0.80168624921797
C        -0.12534453646893       1.99381594931195       0.69777268888986
C         0.91287327114044       1.23947038990251       1.16757197718591
C         1.46911293841416       0.47150655480099       0.11812227736679
C         0.67985420632880       0.66525181077679      -1.04982077604919
H         0.45415643793236       2.78886674222036      -1.19068373103846
C        -1.46911293841416      -0.47150655480099       0.11812227736679
H        -0.74739788804598       2.65617908104104       1.28195899397973
H         1.22524975545037       1.18213001451158       2.20142177744971
H         2.34395930311060      -0.15471310341308       0.19580072015395
C        -0.67985420632880      -0.66525181077679      -1.04982077604919
H         1.11179701329136       0.49647581407003      -2.03129035077390
H        -1.11179701329136      -0.49647581407003      -2.03129035077390
H        -2.34395930311060       0.15471310341308       0.19580072015395
C        -0.91287327114044      -1.23947038990251       1.16757197718591
C         0.12534453646893      -1.99381594931195       0.69777268888986
H        -1.22524975545037      -1.18213001451158       2.20142177744971
H         0.74739788804598      -2.65617908104104       1.28195899397973
C         0.12233795920601      -1.93675797138191      -0.80168624921797
H         1.10532017523107      -1.96140412357569      -1.27240532794643
H        -0.45415643793236      -2.78886674222036      -1.19068373103846
```

A.5 KF files

During the execution of the AMS code, most data is transferred between FORTRAN SUBROUTINES via explicit arguments. In some cases, however, certain values must remain available across multiple calls without being permanently stored in memory. Retaining such data in large variables over the entire runtime can lead to unnecessary memory consumption. To address this, the suite employs temporary storage in a standardised binary file format known as KF files.

KF files serve two main purposes. First, they act as a temporary storage mechanism, allowing intermediate results to be written to disk and later retrieved by other SUBROUTINES. Second, they are used as a persistent interface: the same files can be read by the GUI and by PLAMS, enabling external tools to access calculation data without requiring a direct in-memory transfer.

The KF format was developed by SCM and follows a fixed set of conventions. Section and key names are limited to 32 characters; any name exceeding this limit is silently truncated, potentially leading to mismatches. Internally, the FORTRAN code interfaces with a `C` code, this detail is important because `C` stores two-dimensional arrays in *column-major order*, whereas FORTRAN uses *row-major order*.

The FORTRAN interface provides dedicated SUBROUTINES for reading and writing integers and reals of rank 1 through 4. The variable type is detected automatically by the SUBROUTINE. If the array size is not given explicitly, it assumes the full array length. A typical write call follows the syntax:

```
kfwrite(iu, key, array, len, istride)
```

where:

- `iu` is the unit name of the KF file [`TAPE21/TAPE10/...`],
- `key` is the string identifier under which the data will be stored,
- `array` is the scalar or array to be written,
- `len` (optional) specifies how many elements to write (if omitted, the full array is used),
- `istride` (optional) is the stride, *i. e.* the number of elements skipped between consecutive writes (1 by default, no skipping).

The next code illustrates how to write to the KF file:

```

! Write to KF file
use KF
! ...
! Open KF file for writing; iu21 is the TAPE21
call kfopfl (iu21, filename)
! If the section does not exist, it will be created
if (.not.kfexsc (iu21, 'Properties')) call kfcrsc (iu21, 'Properties')
! Open the section
call kfopsc (iu21, 'Properties')

! Write some properties
call kfwrite (iu21, 'CP number of', nna)
call kfwrite (iu21, 'CP coordinates', CPpoints(1:nna,1:3))
call kfwrite (iu21, 'CP density at', CPpoints(1:nna,4))
call kfwrite (iu21, 'CP density gradient at', CPpoints(1:nna,5:7))
call kfwrite (iu21, 'CP density Hessian at', CPpoints(1:nna,8:13))
call kfwrite (iu21, 'CP code (Rank,Signature)', CPpoints(1:nna,14))
call kfwrite (iu21, 'CP eigenvalues of Hessian', CPpoints(1:nna,15:17))
call kfwrite (iu21, 'CP eigenvectors of Hessian', CPpoints(1:nna,18:26))

! Close the file
call kfclfl (iu21)
```

The next code illustrates how to read from the KF file:

```

! Read from KF file
use KF
! ...
call kfopfl(iu21, gADFFiles%main) ! Open KF file
call kfopsc(iu21, 'Properties')   ! Open the section
call kfread(iu21, 'CP number of', AIMLocalProperties%ncp); ncp = AIMLocalProperties%ncp
allocate(AIMLocalProperties%CPpoints(1:ncp,26)) ! Allocate the array
! Read the properties
call kfread(iu21, 'CP coordinates', AIMLocalProperties%CPpoints(1:ncp,1:3))
! ...
```

A.6 AMS Output (plain text)

This section illustrates the structure of the plain text output generated by the AMS driver. While the overall format of the output varies between engines, the QTAIM partition is presented in a consistent style for both ADF and DFTB, with some differences for BAND, since not all properties are available in BAND calculations.

In ADF and DFTB, the local properties are introduced with a header that includes references for the computed quantities, their corresponding authors, followed by the atomic coordinates (where we include the NNA in case they are present). If the grid used for the QTAIM partition has been modified, this is explicitly noted in the output. A summary of the CPs is then provided, as illustrated in the example below (azulene molecule):

```
            +--------------------------------+
            | Summary of the Critical Points |
            +--------------------------------+

        Poincare-Hopf relationship satisfied
      ----------------------------------------+-------
       NUMBER OF     NUCLEAR CRITICAL POINTS |    18
       NUMBER OF NON-NUCLEAR CRITICAL POINTS |     0
       NUMBER OF        BOND CRITICAL POINTS |    19
       NUMBER OF        RING CRITICAL POINTS |     2
       NUMBER OF        CAGE CRITICAL POINTS |     0
      ----------------------------------------+-------
                    TOTAL OF CRITICAL POINTS |    39

 ------------------------------------------------------------------------
                         Bond Path(s) Summary
 ------------------------------------------------------------------------
     #BP   #CP     Atom   Atom   BP Length     Distance    Atom Type
       1    37       1      2      1.392         1.392        C-C
       2    35       1     11      1.084         1.084        C-H
# ... all bonds
 ------------------------------------------------------------------------
                         Ring Path(s) Summary
  #RP    #CP     Size     Atom,  Atom,  Atom, ...
 ------------------------------------------------------------------------
    1     19       5         3     4     8     9    10
# ... all rings (and cages)
```

For the properties of each CP, nuclear (and non-nuclear) attractors are reported with only the electron density and coordinates, whereas the remaining CPs include the full set of computed properties. Compared with previous implementations, the current version also lists the atoms involved in each RCP and CCP, both in the summary and in the detailed property tables.

```
 +-----------------------------------------------------------------------+
 |  CP #   1                                                             |
 |  Type               : Nuclear Attractor (3,-3)
 |  Associated atom    :           1 C

       CP Coordinates :       0.0000000      -1.2683395       1.8873492
                  Rho :   0.1266696E+03
# ... all attractors
 +-----------------------------------------------------------------------+
 |  CP #  19                                                             |
 |  Type               :  Ring Critical Point (3,+1)
 |  Ring size          :     5
 |  With the atoms     :   3 C   4 C   8 C   9 C  10 C

       CP Coordinates :      -0.0000000      -0.0000000      -1.4679307
                  Rho :   0.5323605E-01
         Gradient Rho :  -0.5283257E-19  -0.2201300E-16  -0.1094178E-15
        |Gradient Rho|:   0.1116102E-15
              Hessian :
         -0.4974860E-01  -0.8786186E-11   0.2206157E-10
                          0.1903828E+00  -0.3282384E-10
                                         0.1878770E+00

                Eigenvalues of the Hessian:
         -0.4974860E-01   0.1878770E+00   0.1903828E+00

  (orthonormal) Eigenvectors of the Hessian [by columns]:
          0.1000000E+01  -0.9284171E-10  -0.3658922E-10
          0.3658918E-10  -0.1309889E-07   0.1000000E+01
         -0.9284171E-10  -0.1000000E+01  -0.1309889E-07

            Laplacian :   0.3285112E+00
# ... all properties
          Inhomogeneity parameters at CPs
              Delta_u :   0.1527487E+00
# ... all properties
```

For the dipole moment, the atomic properties section has been extended to include additional tables showing the electron and nuclear contributions to the dipole moment, as well as their sum, together with a comparison against the value obtained from the Debye analysis.

```
ATOMIC DIPOLE MOMENTS
=====================

          (electronic contribution)   (nuclear contribution)
=======================================================
  Atom        X      ...      Z     ...      Z
-------------------------------------------------------
  1 C  :    0.0000   ...   -0.0512  ...    0.0103
  2 C  :    0.0000   ...   -0.0128  ...    0.0442
# ...
-------------------------------------------------------
Sum (a. u.) 0.0000   ...   -0.0370  ...    0.3913
    (Debye) 0.0000   ...   -0.0942  ...    0.9946

         (electronic + nuclear) contribution
=======================================================
  Atom            X              Y              Z
-------------------------------------------------------
  1 C  :        0.0000         0.0661        -0.0409
  2 C  :        0.0000        -0.0101         0.0314
## ...
-------------------------------------------------------
Sum (a. u.)     0.0000        -0.0000         0.3543
    (Debye)     0.0000        -0.0000         0.9005

Magnitude (a. u.)     0.3543
Magnitude (Debye)     0.9005
Difference between QTAIM and Analytic Dipole Moment (Debye):       0.0005
```

The inclusion of NNAs is also reflected in the output. These appear under the label `NNA`, as shown in the following example for the lithium dimer:

```
                    Nuclei coordinates

   No  |  Atom  |      X      |      Y      |      Z
-------+--------+-------------+-------------+-------------
    1      Li       0.00000       0.00000       0.00000
    2      Li       0.00000       0.00000       2.67300
    3      NN       0.00000      -0.00000       1.33650

           +--------------------------------+
           | Summary of the Critical Points |
           +--------------------------------+

       Poincare-Hopf relationship satisfied
     ---------------------------------------+-------
      NUMBER OF     NUCLEAR CRITICAL POINTS |     2
      NUMBER OF NON-NUCLEAR CRITICAL POINTS |     1
      NUMBER OF        BOND CRITICAL POINTS |     2
      NUMBER OF        RING CRITICAL POINTS |     0
      NUMBER OF        CAGE CRITICAL POINTS |     0
     ---------------------------------------+-------
                  TOTAL OF CRITICAL POINTS |     5

------------------------------------------------------------------------
                       Bond Path(s) Summary
------------------------------------------------------------------------
    #BP   #CP     Atom   Atom   BP Length     Distance    Atom Type
      1     3       1      3      1.337         1.336       Li-NNA
      2     5       2      3      1.337         1.337       Li-NNA
# ...
 +----------------------------------------------------------------------+
 |  CP #   4                                                            |
 |  Type               :  Non-Nuclear Attractor (3,-3)
 |  With no associated nucleus by its nature

       CP Coordinates :       0.0000000      -0.0000000       1.3364995
                  Rho :   0.1382686E-01
```

Appendix

B Author's Notes

This chapter in contrast to the previous one has no physical chemistry or mathematical content, but provides additional information about the workflow of the thesis that in fact, was also a part of the main project of the PhD.

B.1 How this thesis was written

This thesis was written using LaTeX, bibliography management handled by BibTeX. The compilation was done via LuaLaTeX, using a `Makefile` that automates the compilation process as well as some of the imagen generation.

Figures were generated using Python (Matplotlib and Seaborn), and Ti*k*Z. Image editing was carried out using GIMP and Inkscape. Various diagrams were created with Graphviz and Mermaid, this last one for the README file. For the vector graphics, the svgrepo was used to obtain some starting points, and then modified with Inkscape.

For the Figure 2.5, a modified version of qtaim.wl was used, which is a Wolfram language script, to analyse the QTAIM partition from `wfn` files. The original code is available at /ecbrown/QTAIM.wl. The original code is quite experimantal, and my modificated version is even more unestable, no guarantees are made about the generalised version of the code.

Any issue or bug will be tracked in the repository. However, since the code functions as intended for this thesis, no further development is currently planned. All warnings i have during *local compilation* were analysed and just ignored, as they do not affect the final result.

The project is intended to be used from the command line. Nonetheless, due to the widespread use of Overleaf, a dedicated strategy is provided to make it compatible. Note that Overleaf does not support all features of the original setup, such as `Makefile` options (compaling just one Chapter or don't compilation for bibliography) or multi-branch workflows, typographies are limited and the compilation time would be longer since many images/plots are complex TikZ figures.

The Overleaf branch should be used mainly for minor text edits. It is recommended to treat it as a mirror of the main `writing` branch. To synchronize changes, we suggest using selective cherry-picks of `*.tex` files to avoid merge conflicts.

Important: Overleaf does not allow privileged git operations such as `git push -force`, which prevents proper handling of merges with unrelated histories. To initialise the Overleaf branch, i recommend, the following steps:

```
vcastor@ada - bash
$ # From Overleaf track everything
$ git pull
$ # Delete all Overleaf files via Overleaf UI or locally
$ # Then merge with unrelated histories allowed:
$ git merge writing --allow-unrelated-histories
$ git commit -am "My first commit on Overleaf"
$ git push
```

The full source code of this manuscript is available in a public GitHub repository: /vcastor/PhDManuscript.

My recommendation is to use the >_ terminal to compile the manuscript, as this project was designed, to compile in local you can follow the next steps:

```
vcastor@ada - bash
$ # Download the repository
$ git clone https://github.com/vcastor/PhDManuscript.git
$ cd PhDManuscript
$ # Compile the entire manuscript
$ make all  # imagenes, plots, bibliography and text
$ # Or just what you need
$ make tex  # no imagenes or plots but bibliography
$ make fast # no imagenes or plots neither bibliography
$ make chapter # just the chapter [READ THE README.md for that]
$ make style   # style template
```

B.2 חלום יעקב (Jacob's Dream)

the author writes here not as a scientist,

As a theoretical chemist and an *enthusiast* of art, i've always found it a little unsatisfying that most of the visual representations of Jacob's Ladder in DFT are, let's say, not exactly the most beautiful images ever created (no offence to their authors!). We might be missing a chance to reimagine it, to create a representation that captures not only the technical content but also the metaphorical and symbolic richness behind it.

On one hand, i get it. Not everyone wants their science mixed with metaphors, religion, or artistic flourishes. Some just want equations, data, numbers and that's completely fine. It's also true that many of us are cautious about including anything that might be interpreted as pushing a particular faith or worldview into scientific spaces.

But even so... we're missing something! We're missing the opportunity to make something —something visual, symbolic, maybe even a bit poetic and just for fun, as a side project during free time— that reflects the deeper meaning of this metaphor.

The "Jacob's Ladder" in DFT is, of course, a reference to the biblical one. According to the Book of Genesis, the patriarch Jacob had a dream while resting for the night, using a stone as his pillow. In that dream, he saw a ladder reaching from the earth up to the heavens, with angels ascending and descending upon it.

In the context of DFT, the ladder becomes a metaphor: we begin with HF, and we aim for the heaven, that is, for chemical accuracy. We climb rung by rung, from LDA to GGA, and onward to meta-GGAs, hybrids, double hybrids... each rung represents a systematic improvement, a deliberate step upward in complexity and (ideally) accuracy.

Just like in the biblical interpretations, we don't take a single miraculous leap from earth to sky. Instead, we follow a structured path, becuase... if angels can fly, why would they need a ladder?, maybe because the ladder wasn't built for them. It was built for us. We don't need wings. We don't need to mimic the angels. We just need a clear and methodical way to ascend. Not by miracle, but by method.

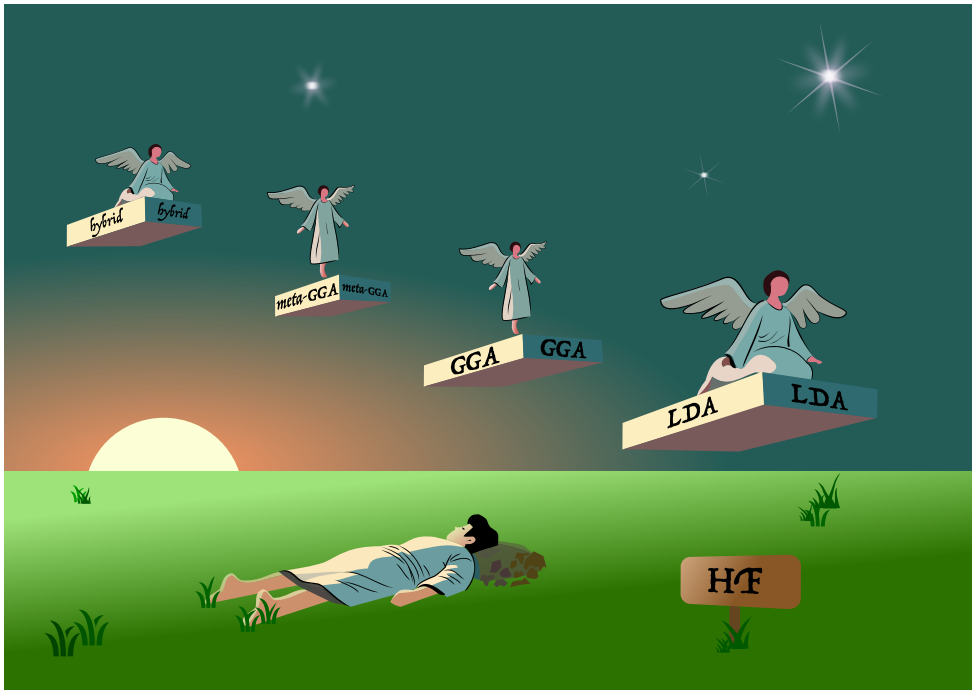


Figure B.1. Jacob's Ladder in DFT. Only angels going down, constructing the ladder, the rungs are being arrived by the angels.

Perdew himself offered a beautiful interpretation of his metaphor in [155]. In his words, the angels on the ladder are the users: researchers who choose the rung that best suits their needs, depending on the balance they seek between accuracy and computational cost.

Personally, I like to think of the angels in two ways: some go up, carrying the prayers —input files, basis sets, SCF settings— of the users; others come down bearing blessings converged energies and properties. Just like a biblical interpretation, where the angels ascending represent the prayers and the ones descending are the messengers of God (*intentional redundancy*) with the divine blessing for those who pray.

Note that we can't know which angel is in front of other, just by the knowledge of the rung, we need to know the direction of the angel; as well as in the rungs of DFT Jacob's Ladder, some times our goal is to ascend in the ladder, seeking more accuracy, and sometimes we need to descend, looking for a faster calculation.

Our feet stay grounded in practical considerations, but our eyes are fixed on the sky. It's not blind faith, it's a slow, structured journey up the ladder.

Bibliography

[1] Software for Chemistry & Materials (SCM). Gibbs free energy change for a gas phase reaction, 2025. https://www.scm.com/doc/AMS/Vibrational_Spectroscopy.html. Section "Gibbs free energy change for a gas phase reaction" in "Vibrational Spectroscopy - AMS 2025.1 Documentation".

[2] R. G. Parr and W. Yang. Density-functional theory of the electronic structure of molecules. *Annual Review of Physical Chemistry*, 46(1):701 – 728, October 1995. ISSN 1545-1593. http://dx.doi.org/10.1146/annurev.pc.46.100195.003413.

[3] R. F. W. Bader. *Atoms in Molecules: A Quantum Theory (International Series of Monographs on Chemistry)*. Oxford University Press, USA, 1994. ISBN 0198558651.

[4] G. te Velde, F. M. Bickelhaupt, E. J. Baerends, C. Fonseca Guerra, S. J. A. van Gisbergen, J. G. Snijders, and T. Ziegler. Chemistry with adf. *Journal of Computational Chemistry*, 22(9):931–967, 2001. ISSN 1096-987X. http://dx.doi.org/10.1002/jcc.1056.

[5] M. Elstner and G. Seifert. Density functional tight binding. *Philosophical Transactions of the Royal Society A*, 372(2011):20120483, 2014.

[6] P. Hohenberg and W. Kohn. Inhomogeneous electron gas. *Physical Review*, 136(3B): B864–B871, November 1964. https://doi.org/10.1103/physrev.136.b864.

[7] J. S. Rowlinson. The maxwell – boltzmann distribution. *Molecular Physics*, 103(21 – 23):2821 – 2828, November 2005. ISSN 1362-3028. http://dx.doi.org/10.1080/00206897050004474 9.

[8] M. T. P. Beerepoot, A. H. Steindal, N. H. List, J. Kongsted, and J. M. H. Olsen. Averaged

solvent embedding potential parameters for multiscale modeling of molecular properties. *Journal of Chemical Theory and Computation*, 12(4):1684–1695, March 2016. https://doi.org/10.1021/acs.jctc.5b01000.

[9] J. M. H. Olsen and P. Reinholdt. Pyframe: Python framework for fragment-based multiscale embedding, 2021. https://zenodo.org/record/775113.

[10] E. Schrödinger. An undulatory theory of the mechanics of atoms and molecules. *Physical Review*, 28(6):1049–1070, December 1926. https://doi.org/10.1103/physrev.28.1049.

[11] P. A. M. Dirac. The quantum theory of the electron. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, 117 (778):610–624, February 1928. https://doi.org/10.1098/rspa.1928.0023.

[12] R. Čurík, D. Hvizdoš, and C. H. Greene. Validity of the born-oppenheimer approximation in the indirect-dissociative-recombination process. *Physical Review A*, 98(6), December 2018. ISSN 2469-9934. http://dx.doi.org/10.1103/PhysRevA.98.062706.

[13] B. T. Sutcliffe and R. G. Woolley. Comment on 'on the quantum theory of molecules' [j. chem.phys. **137**, 22a544 (2012)]. 2014. https://arxiv.org/abs/1401.0873.

[14] D. R. Hartree. The wave mechanics of an atom with a non-coulomb central field. part i. theory and methods. *Mathematical Proceedings of the Cambridge Philosophical Society*, 24(1):89 – 110, January 1928. ISSN 1469-8064. http://dx.doi.org/10.1017/S0305004100011919.

[15] V. Fock. Konfigurationsraum und zweite quantelung. *Zeitschrift für Physik*, 75(9 – 10): 622 – 647, September 1932. ISSN 1434-601X. http://dx.doi.org/10.1007/BF01344458.

[16] C. C. J. Roothaan. A study of two-center integrals useful in calculations on molecular structure. i. *The Journal of Chemical Physics*, 19(12):1445–1458, December 1951. https://doi.org/10.1063/1.1748100.

[17] I. Mayer. *The Linear Variational Method and Löwdin's Orthogonalization Schemes*, page 45 – 68. Springer US, 2003. ISBN 9781475765199. http://dx.doi.org/10.1007/978-1-4757-6519-9_3.

[18] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen. *LAPACK Users' Guide*. Society for Industrial and Applied Mathematics, Philadelphia, PA, third edition, 1999. ISBN 0-89871-447-8 (paperback).

[19] P. Echenique and J. L. Alonso. A mathematical and computational review of hartree – fock

scf methods in quantum chemistry. *Molecular Physics*, 105(23 – 24):3057 – 3098, December 2007. ISSN 1362-3028. http://dx.doi.org/10.1080/00268970701757875.

[20] C. Yang, W. Gao, and J. C. Meza. On the convergence of the self-consistent field iteration for a class of nonlinear eigenvalue problems. *SIAM Journal on Matrix Analysis and Applications*, 30(4):1773 – 1788, January 2009. ISSN 1095-7162. http://dx.doi.org/10.1137/080716293.

[21] A. Szabo and N. S. Ostlund. *Modern Quantum Chemistry: Introduction to Advanced Electronic Structure Theory (Dover Books on Chemistry)*. Dover Publications, 1996. ISBN 0486691861.

[22] F. Jensen. *Introduction to Computational Chemistry*. Wiley, 2013.

[23] T. H. Dunning. Gaussian basis sets for use in correlated molecular calculations. i. the atoms boron through neon and hydrogen. *The Journal of Chemical Physics*, 90(2):1007–1023, January 1989. https://doi.org/10.1063/1.456153.

[24] F. Weigend and R. Ahlrichs. Balanced basis sets of split valence, triple zeta valence and quadruple zeta valence quality for h to rn: Design and assessment of accuracy. *Physical Chemistry Chemical Physics*, 7(18):3297, 2005. ISSN 1463-9084. http://dx.doi.org/10.1039/B508541A.

[25] F. Jensen. Polarization consistent basis sets: Principles. *The Journal of Chemical Physics*, 115(20):9113 – 9125, November 2001. ISSN 1089-7690. http://dx.doi.org/10.1063/1.1413524.

[26] S. F. Boys. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 200(1063):542 – 554, February 1950. ISSN 2053-9169. http://dx.doi.org/10.1098/rspa.1950.0036.

[27] E. Besalú and R. Carbó-Dorca. The general gaussian product theorem. *Journal of Mathematical Chemistry*, 49(8):1769 – 1784, June 2011. ISSN 1572-8897. http://dx.doi.org/10.1007/s10910-011-9857-9.

[28] E. J. Baerends, N. F. Aguirre, N. D. Austin, J. Autschbach, F. M. Bickelhaupt, R. Bulo, C. Cappelli, A. C. T. van Duin, F. Egidi, C. Fonseca Guerra, A. Förster, M. Franchini, T. P. M. Goumans, T. Heine, M. Hellström, C. R. Jacob, L. Jensen, M. Krykunov, E. van Lenthe, A. Michalak, M. M. Mitoraj, J. Neugebauer, V. P. Nicu, P. Philipsen, H. Ramanantoanina, R. Rüger, G. Schreckenbach, M. Stener, M. Swart, J. M. Thijssen, T. Trnka, L. Visscher, A. Yakovlev, and S. van Gisbergen. The amsterdam modeling

suite. *The Journal of Chemical Physics*, 162(16):162501, 04 2025. ISSN 0021-9606. https://doi.org/10.1063/5.0258496.

[29] M. J. Frisch, G. W. Trucks, H. B. Schlegel, G. E. Scuseria, M. A. Robb, J. R. Cheeseman, G. Scalmani, V. Barone, G. A. Petersson, H. Nakatsuji, X. Li, M. Caricato, A. V. Marenich, J. Bloino, B. G. Janesko, R. Gomperts, B. Mennucci, H. P. Hratchian, J. V. Ortiz, A. F. Izmaylov, J. L. Sonnenberg, D. Williams-Young, F. Ding, F. Lipparini, F. Egidi, J. Goings, B. Peng, A. Petrone, T. Henderson, D. Ranasinghe, V. G. Zakrzewski, J. Gao, N. Rega, G. Zheng, W. Liang, M. Hada, M. Ehara, K. Toyota, R. Fukuda, J. Hasegawa, M. Ishida, T. Nakajima, Y. Honda, O. Kitao, H. Nakai, T. Vreven, K. Throssell, J. A. Montgomery, Jr., J. E. Peralta, F. Ogliaro, M. J. Bearpark, J. J. Heyd, E. N. Brothers, K. N. Kudin, V. N. Staroverov, T. A. Keith, R. Kobayashi, J. Normand, K. Raghavachari, A. P. Rendell, J. C. Burant, S. S. Iyengar, J. Tomasi, M. Cossi, J. M. Millam, M. Klene, C. Adamo, R. Cammi, J. W. Ochterski, R. L. Martin, K. Morokuma, O. Farkas, J. B. Foresman, and D. J. Fox. Gaussian16 Revision C.01, 2016. Gaussian Inc. Wallingford CT.

[30] F. Neese. The orca program system. *WIRES Comput. Molec. Sci.*, 2(1):73–78, 2012.

[31] P. Giannozzi, S. Baroni, N. Bonini, M. Calandra, R. Car, C. Cavazzoni, D. Ceresoli, G. L. Chiarotti, m. Cococcioni, I. Dabo, A. D. Corso, S. Fabris, G. Fratesi, S. de Gironcoli, R. Gebauer, U. Gerstmann, C. Gougoussis, A. Kokalj, M. Lazzeri, L. Martin-Samos, N. Marzari, F. Mauri, R. Mazzarello, S. Paolini, A. Pasquarello, L. Paulatto, C. Sbraccia, S. Scandolo, G. Sclauzero, A. P. Seitsonen, A. Smogunov, P. Umari, and R. M. Wentzcovitch. Quantum espresso: a modular and open-source software project for quantum simulations of materials. 2009. https://arxiv.org/abs/0906.2569.

[32] G. Kresse and J. Furthmüller. Efficiency of ab-initio total energy calculations for metals and semiconductors using a plane-wave basis set. *Comput. Mater. Sci.*, 6:15–50, 1996.

[33] B. P. Pritchard, D. Altarawy, B. Didier, T. D. Gibson, and T. L. Windus. New basis set exchange: An open, up-to-date resource for the molecular sciences community. *Journal of Chemical Information and Modeling*, 59(11):4814 – 4820, October 2019. ISSN 1549-960X. http://dx.doi.org/10.1021/acs.jcim.9b00725.

[34] Y. W. Robert G. Parr. *Density-Functional Theory of Atoms and Molecules.* Oxford University Press Inc, 1994. ISBN 978-0-19-509276-9.

[35] H. Kasai, K. Tolborg, M. Sist, J. Zhang, V. R. Hathwar, M. Ø. Filsø, S. Cenedese, K. Sugimoto, J. Overgaard, E. Nishibori, and B. B. Iversen. X-ray electron density investigation

of chemical bonding in van der waals materials. *Nature Materials*, 17(3):249–252, February 2018. https://doi.org/10.1038/s41563-017-0012-2.

[36] P. Coppens and A. Vos. Electron density distribution in cyanuric acid. II. Neutron diffraction study at liquid nitrogen temperature and comparison of X-ray neutron diffraction results. *Acta Crystallographica Section B Structural Crystallography and Crystal Chemistry*, 27(1):146–158, January 1971. https://doi.org/10.1107/s0567740871001808.

[37] P.-O. Löwdin. Quantum theory of many-particle systems. i. physical interpretations by means of density matrices, natural spin-orbitals, and convergence problems in the method of configurational interaction. *Physical Review*, 97(6):1474 – 1489, March 1955. ISSN 0031-899X. http://dx.doi.org/10.1103/PhysRev.97.1474.

[38] R. McWeeny and B. Sutcliffe. *Methods of Molecular Quantum Mechanics*. Pure and Applied Mathematics. Academic Press, 1969. ISBN 9780124865501. https://books.google.fr/books?id=D_Ph20OZu0YC.

[39] R. McWeeny. Some recent advances in density matrix theory. *Reviews of Modern Physics*, 32(2):335 – 369, April 1960. ISSN 0034-6861. http://dx.doi.org/10.1103/RevModPhys.32.335.

[40] T. Helgaker, P. Jørgensen, and J. Olsen. *Molecular Electronic-Structure Theory*. Wiley, 2000. ISBN 0471967556.

[41] L. H. Thomas. The calculation of atomic fields. *Mathematical Proceedings of the Cambridge Philosophical Society*, 23(5):542–548, January 1927. https://doi.org/10.1017/s0305004100011683.

[42] E. Fermi. Statistical method to determine some properties of atoms. *Rend. Accad. Naz. Lincei*, 6(602-607):5, 1927.

[43] W. Kohn and L. J. Sham. Self-consistent equations including exchange and correlation effects. *Physical Review*, 140(4A):A1133–A1138, November 1965. https://doi.org/10.1103/physrev.140.a1133.

[44] J. P. Perdew. Jacob's ladder of density functional approximations for the exchange-correlation energy. In *AIP Conference Proceedings*, volume 577, page 1 – 20. AIP, 2001. http://dx.doi.org/10.1063/1.1390175.

[45] L. Goerigk and S. Grimme. Double - hybrid density functionals. *WIREs Computational Molecular Science*, 4(6):576 – 600, July 2014. ISSN 1759-0884. http://dx.doi.org/10.1002/wcms.1193.

[46] W. Koch and M. C. Holthausen. *A Chemist's Guide to Density Functional Theory*. Wiley, July 2001. https://doi.org/10.1002/3527600043.

[47] J. C. Slater and J. C. Phillips. Quantum Theory of Molecules and Solids vol. 4: The Self-consistent Field for Molecules and Solids. *Physics Today*, 27(12):49–50, December 1974. https://doi.org/10.1063/1.3129035.

[48] S. H. Vosko, L. Wilk, and M. Nusair. Accurate spin-dependent electron liquid correlation energies for local spin density calculations: a critical analysis. *Canadian Journal of Physics*, 58(8):1200–1211, August 1980. https://doi.org/10.1139/p80-159.

[49] D. M. Ceperley and B. J. Alder. Ground state of the electron gas by a stochastic method. *Physical Review Letters*, 45(7):566–569, August 1980. https://doi.org/10.1103/physrevlett.45.566.

[50] J. P. Perdew and A. Zunger. Self-interaction correction to density-functional approximations for many-electron systems. *Physical Review B*, 23(10):5048–5079, May 1981. https://doi.org/10.1103/physrevb.23.5048.

[51] A. D. Becke. Density-functional exchange-energy approximation with correct asymptotic behavior. *Physical Review A*, 38(6):3098–3100, September 1988. https://doi.org/10.1103/physreva.38.3098.

[52] C. Adamo and V. Barone. Exchange functionals with improved long-range behavior and adiabatic connection methods without adjustable parameters: The mPW and mPW1pw models. *The Journal of Chemical Physics*, 108(2):664–675, January 1998. https://doi.org/10.1063/1.475428.

[53] C. Lee, W. Yang, and R. G. Parr. Development of the colle-salvetti correlation-energy formula into a functional of the electron density. *Physical Review B*, 37(2):785–789, January 1988. https://doi.org/10.1103/physrevb.37.785.

[54] R. Colle and O. Salvetti. Approximate calculation of the correlation energy for the closed shells. *Theoretica Chimica Acta*, 37(4):329–334, 1975. https://doi.org/10.1007/bf01028401.

[55] C. v. Weizsäcker. Zur theorie der kernmassen. *Zeitschrift für Physik*, 96(7):431–458, 1935.

[56] J. P. Perdew. Density-functional approximation for the correlation energy of the inhomogeneous electron gas. *Physical Review B*, 33(12):8822–8824, June 1986. https://doi.org/10.1103/physrevb.33.8822.

[57] J. Sun, R. C. Remsing, Y. Zhang, Z. Sun, A. Ruzsinszky, H. Peng, Z. Yang, A. Paul, U. Waghmare, X. Wu, M. L. Klein, and J. P. Perdew. Scan: An efficient density functional

yielding accurate structures and energies of diversely-bonded materials, 2015. https://arxiv.org/abs/1511.01089.

[58] T. V. Voorhis and G. E. Scuseria. A novel form for the exchange-correlation energy functional. *The Journal of Chemical Physics*, 109(2):400–410, July 1998. https://doi.org/10.1063/1.476577.

[59] A. Gonis, N. Kioussis, and M. Ciftan, editors. *Electron Correlations and Materials Properties*. Springer US, 1999. https://doi.org/10.1007/978-1-4615-4715-0.

[60] A. D. Becke. Density-functional thermochemistry. III. the role of exact exchange. *The Journal of Chemical Physics*, 98(7):5648–5652, April 1993. https://doi.org/10.1063/1.464913.

[61] C. Adamo and V. Barone. Toward reliable density functional methods without adjustable parameters: The pbe0 model. *The Journal of Chemical Physics*, 110(13):6158 – 6170, April 1999. ISSN 1089-7690. http://dx.doi.org/10.1063/1.478522.

[62] M. Ernzerhof and J. P. Perdew. Generalized gradient approximation to the angle- and system-averaged exchange hole. *The Journal of Chemical Physics*, 109(9):3313–3320, September 1998. https://doi.org/10.1063/1.476928.

[63] Y. Zhao and D. G. Truhlar. The M06 suite of density functionals for main group thermochemistry, thermochemical kinetics, noncovalent interactions, excited states, and transition elements: two new functionals and systematic testing of four M06-class functionals and 12 other functionals. *Theoretical Chemistry Accounts*, 120(1-3):215–241, July 2007. https://doi.org/10.1007/s00214-007-0310-x.

[64] J. P. Perdew, K. Burke, and M. Ernzerhof. Generalized gradient approximation made simple. *Physical Review Letters*, 77(18):3865 – 3868, October 1996. ISSN 1079-7114. http://dx.doi.org/10.1103/PhysRevLett.77.3865.

[65] K. Burke, J. P. Perdew, and Y. Wang. Derivation of a generalized gradient approximation: The pw91 density functional. In *Electronic Density Functional Theory: recent progress and new directions*, pages 81–111. Springer, 1998.

[66] E. H. Lieb. A lower bound for coulomb energies. *Physics Letters A*, 70(5 – 6):444 – 446, April 1979. ISSN 0375-9601. http://dx.doi.org/10.1016/0375-9601(79)90358-X.

[67] E. H. Lieb and S. Oxford. Improved lower bound on the indirect coulomb energy. *International Journal of Quantum Chemistry*, 19(3):427 – 439, March 1981. ISSN 1097-461X. http://dx.doi.org/10.1002/qua.560190306.

[68] E. Engel and R. M. Dreizler. *Density Functional Theory: An Advanced Course*. Springer Berlin Heidelberg, 2011. ISBN 9783642140907. http://dx.doi.org/10.1007/978-3-642-14090-7.

[69] G. Monard and J.-L. Rivail. *Solvent Effects in Quantum Chemistry*, page 727 – 739. Springer International Publishing, 2017. ISBN 9783319272825. http://dx.doi.org/10.1007/978-3-319-27282-5_15.

[70] C. Reichardt. *Solvent Effects in Organic Chemistry*. Verlag Chemie, Weinheim/New York, 1979.

[71] J. Tomasi, B. Mennucci, and R. Cammi. Quantum mechanical continuum solvation models. *Chemical Reviews*, 105(8):2999 – 3094, July 2005. ISSN 1520-6890. http://dx.doi.org/10.1021/cr9904009.

[72] J. G. Kirkwood. Theory of solutions of molecules containing widely separated charges with special application to zwitterions. *The Journal of Chemical Physics*, 2(7):351 – 361, July 1934. ISSN 1089-7690. http://dx.doi.org/10.1063/1.1749489.

[73] J. Tomasi. Cavity and reaction field: "robust" concepts. perspective on "electric moments of molecules in liquids" . *Theoretical Chemistry Accounts: Theory, Computation, and Modeling (Theoretica Chimica Acta)*, 103(3 – 4):196 – 199, February 2000. ISSN 1432-2234. http://dx.doi.org/10.1007/s002149900044.

[74] M. L. Connolly. Analytical molecular surface calculation. *Journal of Applied Crystallography*, 16(5):548 – 558, October 1983. ISSN 0021-8898. http://dx.doi.org/10.1107/S0021889883010985.

[75] M. F. Sanner, A. J. Olson, and J.-C. Spehner. Reduced surface: An efficient way to compute molecular surfaces. *Biopolymers*, 38(3):305 – 320, March 1996. ISSN 1097-0282. http://dx.doi.org/10.1002/(SICI)1097-0282(199603)38:3<305::AID-BIP4>3.0.CO;2-Y.

[76] A. Klamt and G. Schüürmann. Cosmo: a new approach to dielectric screening in solvents with explicit expressions for the screening energy and its gradient. *J. Chem. Soc., Perkin Trans. 2*, (5):799 – 805, 1993. ISSN 1364-5471. http://dx.doi.org/10.1039/P29930000799.

[77] N. L. Allinger, X. Zhou, and J. Bergsma. Molecular mechanics parameters. *Journal of Molecular Structure: THEOCHEM*, 312(1):69 – 83, January 1994. ISSN 0166-1280. http://dx.doi.org/10.1016/s0166-1280(09)80008-0.

[78] Software for Chemistry & Materials (SCM). Cosmo: Conductor like screening model. https://www.scm.com/doc/ADF/Input/COSMO.html, 2025. Accessed: 2025-07-31.

[79] G. Monard, X. Prat-Resina, A. González - Lafont, and J. M. Lluch. Determination of enzymatic reaction pathways using qm/mm methods. *International Journal of Quantum Chemistry*, 93(3):229 – 244, January 2003. ISSN 1097-461X. http://dx.doi.org/10.1002/qua.10555.

[80] F. Lipparini, L. Lagardère, B. Stamm, E. Cancès, M. Schnieders, P. Ren, Y. Maday, and J.-P. Piquemal. Scalable evaluation of polarization energy and associated forces in polarizable molecular dynamics: I. toward massively parallel direct space computations. *Journal of Chemical Theory and Computation*, 10(4):1638 – 1651, March 2014. ISSN 1549-9626. http://dx.doi.org/10.1021/ct401096t.

[81] M. Bondanza, M. Nottoli, L. Cupellini, F. Lipparini, and B. Mennucci. Polarizable embedding qm/mm: the future gold standard for complex (bio)systems? *Physical Chemistry Chemical Physics*, 22(26):14433 – 14448, 2020. ISSN 1463-9084. http://dx.doi.org/10.1039/D0CP02119A.

[82] J. W. Ponder and D. A. Case. *Force Fields for Protein Simulations*, page 27 – 85. Elsevier, 2003. http://dx.doi.org/10.1016/s0065-3233(03)66002-x.

[83] T. S. Koritsanszky and P. Coppens. Chemical applications of X-ray charge-density analysis. *Chemical Reviews*, 101(6):1583–1628, June 2001. https://doi.org/10.1021/cr990112c.

[84] C. F. Matta. *Applications of the Quantum theory of atoms in molecules to chemical and biochemical problems*. PhD thesis, McMaster University, 2002.

[85] J. W. Milnor. *Topology from the Differentiable Viewpoint*. University of Virginia Press, 1965. Reprinted with corrections.

[86] F. W. Biegler-könig, R. F. W. Bader, and T.-H. Tang. Calculation of the average properties of atoms in molecules. II. *Journal of Computational Chemistry*, 3(3):317–328, September 1982. https://doi.org/10.1002/jcc.540030306.

[87] T. A. Keith. Atomic response properties. In *The Quantum Theory of Atoms in Molecules*, pages 61–94. Wiley-VCH Verlag GmbH & Co. KGaA. https://doi.org/10.1002/9783527610709.ch3.

[88] L. D. Landau, J. S. Bell, M. Kearsley, L. Pitaevskii, E. Lifshitz, and J. Sykes. *Electrodynamics of continuous media*, volume 8. elsevier, 2013.

[89] L. H. R. D. Santos, A. Krawczuk, and P. Macchi. Distributed atomic polarizabilities of amino acids and their hydrogen-bonded aggregates. *The Journal of Physical Chemistry A*, 119(13):3285–3298, March 2015. https://doi.org/10.1021/acs.jpca.5b00069.

[90] A. Krawczuk, D. Perez, K. Stadnicka, and P. Macchi. Distributed atomic polarizabilities from electron density. 1. motivations and theory. *American Crystallographic Association*, 42:1–25, 01 2011.

[91] J. F. Nye. *Physical Properties of Crystals: Their Representation by Tensors and Matrices*. Oxford Science Publications. Clarendon Press, Oxford, illustrated, reprint edition, 1985.

[92] J. Franck and E. G. Dymond. Elementary processes of photochemical reactions. *Transactions of the Faraday Society*, 21(February):536, 1926. ISSN 0014-7672. http://dx.doi.org/10.1039/TF9262100536.

[93] E. Condon. A theory of intensity distribution in band systems. *Physical Review*, 28(6): 1182 – 1201, December 1926. ISSN 0031-899X. http://dx.doi.org/10.1103/PhysRev.28.1182.

[94] C. Gatti, Y. Danten, and C. Frayret. Atomic group decomposition of charge transfer excitation global indexes. *The Journal of Physical Chemistry A*, 126(36):6314 – 6328, September 2022. ISSN 1520-5215. http://dx.doi.org/10.1021/acs.jpca.2c04607.

[95] J. Herbert. Visualizing and characterizing excited states from time-dependent density functional theory. December 2023. http://dx.doi.org/10.26434/chemrxiv-2023-gnh1v-v2.

[96] M. J. G. Peach, P. Benfield, T. Helgaker, and D. J. Tozer. Excitation energies in density functional theory: An evaluation and a diagnostic test. *The Journal of Chemical Physics*, 128(4), January 2008. ISSN 1089-7690. http://dx.doi.org/10.1063/1.2831900.

[97] C. A. Guido, P. Cortona, B. Mennucci, and C. Adamo. On the metric of charge transfer molecular excitations: A simple chemical descriptor. *Journal of Chemical Theory and Computation*, 9(7):3118 – 3126, June 2013. ISSN 1549-9626. http://dx.doi.org/10.1021/ct400337e.

[98] R. G. Parr, R. A. Donnelly, M. Levy, and W. E. Palke. Electronegativity: The density functional viewpoint. *The Journal of Chemical Physics*, 68(8):3801 – 3807, April 1978. ISSN 1089-7690. http://dx.doi.org/10.1063/1.436185.

[99] R. G. Parr and R. G. Pearson. Absolute hardness: companion parameter to absolute electronegativity. *Journal of the American Chemical Society*, 105(26):7512 – 7516, December 1983. ISSN 1520-5126. http://dx.doi.org/10.1021/ja00364a005.

[100] R. G. Parr and W. Yang. Density functional approach to the frontier-electron theory of chemical reactivity. *Journal of the American Chemical Society*, 106(14):4049 – 4050, July 1984. ISSN 1520-5126. http://dx.doi.org/10.1021/ja00326a036.

[101] R. G. Pearson. Absolute electronegativity and hardness correlated with molecular orbital theory. *Proceedings of the National Academy of Sciences*, 83(22):8440 – 8441, November 1986. ISSN 1091-6490. http://dx.doi.org/10.1073/pnas.83.22.8440.

[102] R. G. Pearson. *Chemical Hardness*. Wiley, October 1997. ISBN 9783527606177. http://dx.doi.org/10.1002/3527606173.

[103] W. Yang and R. G. Parr. Hardness, softness, and the fukui function in the electronic theory of metals and catalysis. *Proceedings of the National Academy of Sciences*, 82(20):6723 – 6726, October 1985. ISSN 1091-6490. http://dx.doi.org/10.1073/pnas.82.20.6723.

[104] R. G. Parr, L. v. Szentpály, and S. Liu. Electrophilicity index. *Journal of the American Chemical Society*, 121(9):1922 – 1924, February 1999. ISSN 1520-5126. http://dx.doi.org/10.1021/ja983494x.

[105] T. A. Albright, J. K. Burdett, and M. Whangbo. *Orbital Interactions in Chemistry*. Wiley, March 2013. ISBN 9781118558409. http://dx.doi.org/10.1002/9781118558409.

[106] P. W. Ayers and M. Levy. *Perspective on "Density functional approach to the frontier-electron theory of chemical reactivity"*, page 353 – 360. Springer Berlin Heidelberg, 2000. ISBN 9783662104217. http://dx.doi.org/10.1007/978-3-662-10421-7_59.

[107] W. Yang, R. G. Parr, and R. Pucci. Electron density, kohn-sham frontier orbitals, and fukui functions. *The Journal of Chemical Physics*, 81(6):2862 – 2863, September 1984. ISSN 1089-7690. http://dx.doi.org/10.1063/1.447964.

[108] J. Melin, P. W. Ayers, and J. V. Ortiz. The electron-propagator approach to conceptual density-functional theory. *Journal of Chemical Sciences*, 117(5):387 – 400, September 2005. ISSN 0973-7103. http://dx.doi.org/10.1007/BF02708342.

[109] L. J. Bartolotti and P. W. Ayers. An example where orbital relaxation is an important contribution to the fukui function. *The Journal of Physical Chemistry A*, 109(6):1146 – 1151, January 2005. ISSN 1520-5215. http://dx.doi.org/10.1021/jp0462207.

[110] J. S. M. Anderson, J. Melin, and P. W. Ayers. Conceptual density-functional theory for general chemical reactions, including those that are neither charge- nor frontier-orbital-controlled. 1. theory and derivation of a general-purpose reactivity indicator. *Journal of Chemical Theory and Computation*, 3(2):358 – 374, February 2007. ISSN 1549-9626. http://dx.doi.org/10.1021/ct600164j.

[111] J. P. Perdew, R. G. Parr, M. Levy, and J. L. Balduz. Density-functional theory for fractional particle number: Derivative discontinuities of the energy. *Physical Review Let-*

ters, 49(23):1691 – 1694, December 1982. ISSN 0031-9007. http://dx.doi.org/10.1103/PhysRevLett.49.1691.

[112] Y. Zhang and W. Yang. Perspective on "density-functional theory for fractional particle number: derivative discontinuities of the energy". *Theoretical Chemistry Accounts: Theory, Computation, and Modeling (Theoretica Chimica Acta)*, 103(3 – 4):346 – 348, February 2000. ISSN 1432-2234. http://dx.doi.org/10.1007/s002149900021.

[113] C. Morell, A. Grand, and A. Toro-Labbé. New dual descriptor for chemical reactivity. *The Journal of Physical Chemistry A*, 109(1):205 – 212, December 2004. ISSN 1520-5215. http://dx.doi.org/10.1021/jp046577a.

[114] W. Langenaeker and S. Liu. The response of atomic electron densities to point perturbations in the external potential. *Journal of Molecular Structure: THEOCHEM*, 535(1 – 3): 279 – 286, January 2001. ISSN 0166-1280. http://dx.doi.org/10.1016/S0166-1280(00)00579-0.

[115] J. L. Gázquez, M. Franco - Pérez, P. W. Ayers, and A. Vela. Conceptual density functional theory in the grand canonical ensemble, August 2021. http://dx.doi.org/10.1002/9781119683353.ch11.

[116] P. Geerlings, F. De Proft, and W. Langenaeker. Conceptual density functional theory. *Chemical Reviews*, 103(5):1793 – 1874, April 2003. ISSN 1520-6890. http://dx.doi.org/10.1021/cr990029p.

[117] L. Meneses, W. Tiznado, R. Contreras, and P. Fuentealba. A proposal for a new local hardness as selectivity index. *Chemical Physics Letters*, 383(1 – 2):181 – 187, January 2004. ISSN 0009-2614. http://dx.doi.org/10.1016/j.cplett.2003.11.019.

[118] T. Gál, P. Geerlings, F. De Proft, and M. Torrent-Sucarrat. A new approach to local hardness. *Physical Chemistry Chemical Physics*, 13(33):15003, 2011. ISSN 1463-9084. http://dx.doi.org/10.1039/C1CP21213C.

[119] H. Chermette. Chemical reactivity indexes in density functional theory. *Journal of Computational Chemistry*, 20(1):129 – 154, January 1999. ISSN 1096-987X. http://dx.doi.org/10.1002/(SICI)1096-987X(19990115)20:1<129::AID-JCC13>3.0.CO;2-A.

[120] S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, 3rd edition, 2010. ISBN 978-0136042594.

[121] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[122] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. V. D. Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham,

N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.

[123] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need, 2017. https://arxiv.org/abs/1706.03762.

[124] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016. ISBN 978-0262035613. https://www.deeplearningbook.org.

[125] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2nd edition, 2009. ISBN 978-0387848570. https://web.stanford.edu/~hastie/ElemStatLearn/.

[126] K. P. Murphy. *Machine Learning: A Probabilistic Perspective*. MIT Press, 2012. ISBN 978-0262018029.

[127] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006. ISBN 978-0387310732.

[128] S. Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28(2):129 – 137, March 1982. ISSN 0018-9448. http://dx.doi.org/10.1109/TIT.1982.1056489.

[129] S. C. Johnson. Hierarchical clustering schemes. *Psychometrika*, 32(3):241 – 254, September 1967. ISSN 1860-0980. http://dx.doi.org/10.1007/BF02289588.

[130] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, 2003. http://jmlr.org/papers/v3/guyon03a.html.

[131] W. S. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4):115 – 133, December 1943. ISSN 1522-9602. http://dx.doi.org/10.1007/BF02478259.

[132] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84 – 90, May 2017. ISSN 1557-7317. http://dx.doi.org/10.1145/3065386.

[133] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[134] J. R. Quinlan. Induction of decision trees. *Machine Learning*, 1(1):81 – 106, March 1986. ISSN 1573-0565. http://dx.doi.org/10.1007/BF00116251.

[135] C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20(3):273 – 297, September 1995. ISSN 1573-0565. http://dx.doi.org/10.1007/BF00994018.

[136] B. Schölkopf and A. J. Smola. Learning with kernels. In *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, 2002. ISBN 978-0262194754.

[137] A. Ben-Hur, D. Horn, H. T. Siegelmann, and V. Vapnik. Support vector clustering. *Journal of Machine Learning Research*, 2:125–137, 2001. http://www.jmlr.org/papers/v2/ben-hur01a.html.

[138] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and É. Duchesnay. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830, 2011. https://jmlr.org/papers/v12/pedregosa11a.html.

[139] scikit-learn developers. Plot classification boundaries with different svm kernels (scikit-learn example). https://scikit-learn.org/stable/auto_examples/svm/plot_svm_kernels.html, 2025. Accessed: 2025-08-13.

[140] L. Breiman. Random forests. *Machine Learning*, 45(1):5 – 32, October 2001. ISSN 1573-0565. http://dx.doi.org/10.1023/A:1010933404324.

[141] M. E. Tipping. Sparse bayesian learning and the relevance vector machine. *Journal of Machine Learning Research*, 1:211–244, 2001. http://www.jmlr.org/papers/v1/tipping01a.html.

[142] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006. ISBN 9780262182539. http://www.gaussianprocess.org/gpml/.

[143] I. S. Dhillon. Current inverse iteration software can fail. *BIT Numerical Mathematics*, 38(4):685 – 704, December 1998. ISSN 1572-9125. http://dx.doi.org/10.1007/BF02510409.

[144] J. W. Demmel, O. A. Marques, B. N. Parlett, and C. Vömel. Performance and accuracy of lapack' s symmetric tridiagonal eigensolvers. *SIAM Journal on Scientific Computing*, 30(3):1508 – 1526, January 2008. ISSN 1095-7197. http://dx.doi.org/10.1137/070688778.

[145] J. I. Rodríguez, A. M. Köster, P. W. Ayers, A. Santos - Valle, A. Vela, and G. Merino. An efficient grid - based scheme to compute qtaim atomic properties without explicit

calculation of zero - flux surfaces. *Journal of Computational Chemistry*, 30(7):1082 – 1092, October 2008. ISSN 1096-987X. http://dx.doi.org/10.1002/jcc.21134.

[146] P. W. Ayers and S. Jenkins. Bond metallicity measures. *Computational and Theoretical Chemistry*, 1053:112 – 122, February 2015. ISSN 2210-271X. http://dx.doi.org/10.1016/j.comptc.2014.10.040.

[147] V. Tognetti, L. Joubert, P. Cortona, and C. Adamo. Toward a combined dft/qtaim description of agostic bonds: The critical case of a nb(iii) complex. *The Journal of Physical Chemistry A*, 113(44):12322 – 12327, October 2009. ISSN 1520-5215. http://dx.doi.org/10.1021/jp9045534.

[148] V. Tognetti and L. Joubert. Density functional theory and bader's atoms-in-molecules theory: towards a vivid dialogue. *Physical Chemistry Chemical Physics*, 16(28):14539, 2014. ISSN 1463-9084. http://dx.doi.org/10.1039/C3CP55526G.

[149] V. Tognetti and L. Joubert. On critical points and exchange-related properties of intramolecular bonds between two electronegative atoms. *Chemical Physics Letters*, 579: 122 – 126, July 2013. ISSN 0009-2614. http://dx.doi.org/10.1016/j.cplett.2013.06.006.

[150] Y. A. Abramov. On the possibility of kinetic energy density evaluation from the experimental electron-density distribution. *Acta Crystallographica Section A Foundations of Crystallography*, 53(3):264 – 272, May 1997. ISSN 0108-7673. http://dx.doi.org/10.1107/S010876739601495X.

[151] E. Espinosa, E. Molins, and C. Lecomte. Hydrogen bond strengths revealed by topological analyses of experimentally observed electron densities. *Chemical Physics Letters*, 285(3 – 4):170 – 173, March 1998. ISSN 0009-2614. http://dx.doi.org/10.1016/S0009-2614(98)00036-0.

[152] K. Pineda-Urbina, R. D. Guerrero, A. Reyes, Z. Gómez-Sandoval, and R. Flores-Moreno. Shape entropy's response to molecular ionization. *Journal of Molecular Modeling*, 19(4):1677 – 1683, January 2013. ISSN 0948-5023. http://dx.doi.org/10.1007/s00894-012-1725-4.

[153] T. Goswami, S. Paul, S. Mandal, A. Misra, A. Anoop, and P. K. Chattaraj. Unique bonding pattern and resulting bond stretch isomerism in Be_3^{2-}. *International Journal of Quantum Chemistry*, 115(7):426 – 433, January 2015. ISSN 0020-7608. http://dx.doi.org/10.1002/qua.24866.

[154] M. Gallegos, J. M. Guevara-Vela, and Á. M. Pendás. NNAIMQ: A neural network model

for predicting QTAIM charges. *The Journal of Chemical Physics*, 156(1):014112, January 2022. https://doi.org/10.1063/5.0076896.

[155] J. P. Perdew. Climbing the ladder of density functional approximations. *MRS Bulletin*, 38(9):743 – 750, September 2013. ISSN 1938-1425. http://dx.doi.org/10.1557/mrs.2013.178.