Gerfried Millner, BSc.

Exploring Symbolic Regression for hypothesis testing of London-Dispersion corrections in theoretical molecular physics

MASTER'S THESIS

to achieve the university degree of
Diplom-Ingenieur

Master's degree programme:
Advanced Materials Science

submitted to
Graz University of Technology

Supervisor

Assoc.Prof. Dr.phil. Dr.techn. Andreas Hauser

Institute of Experimental Physics

Graz, November 2020

AFFIDAVIT

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present master's thesis.

Date, Signature

Danksagung

An dieser Stelle möchte ich mich bei allen bedanken, welche diese Arbeit möglich gemacht haben. Zuallererst bei meinem Betreuer Andreas Hauser, welcher mir bei vielen fachlichen Fragen stets weitergeholfen und in etlichen Meetings den Fortschritt, sowie die inhaltliche Qualität meiner Masterarbeit maßgeblich zum Positiven beeinflusst hat.

Weiters möchte ich mich auch bei meinem Arbeitgeber, der Materials Center Leoben Forschung GmbH, bedanken. Dort durfte ich, zuerst als studentischer Mitarbeiter und später als Diplomand, aktiv in die Forschung eintauchen und habe so meine Begeisterung dafür entdeckt.

Dank gilt weiteres meinen Kollegen Christoph, Franz-Martin, Lukas und Sebastian, welche mir hilfreiche Tipps zu git, MLflow und Co gegeben und den Büroalltag angenehmer gestaltet haben. Natürlich und insbesondere auch meinen operativen Betreuer und Teamleiter Manfred Mücke möchte ich meinen Dank aussprechen, welcher mir im Bereich Maschinelles Lernen und Symbolische Regression enorm weitergeholfen hat und ohne den ich möglicherweise nie auf die Idee gekommen wäre ein solch interessantes und vielversprechendes Thema zu behandeln.

Weiters ein großes Dankeschön an alle Entwickler und Entwicklerinnen der zahlreichen (open-source) Software Projekte, welche diese Arbeit möglich gemacht haben. Darunter Python, LaTeX, git, Molpro, der Equation Learner (EQL) und viele andere.

Ohne mein Studium wäre es nie zu dieser Masterarbeit gekommen und deswegen möchte ich auch all jenen danken, die mich dabei unterstützt haben. Schließlich gebührt meinen Eltern ganz besonderen Dank, die mir finanzielle und moralische Unterstützung durch die gesamte Studiumzeit zukommen ließen.

Der Autor bedankt sich für die finanzielle Unterstützung im Rahmen des COMET-Programms im K2 Zentrum „Integrated Computational Material, Process and Product Engineering (IC-MPPE)“ (Projektnummer 859480). Dieses Programm wird von den österreichischen Bundesministerien für Verkehr, Innovation und Technologie (BMVIT) und für Digitalisierung und Wirtschaftsstandort (BMDW), vertreten durch die österreichische Forschungsförderungsgesellschaft (FFG), und den Bundesländern Steiermark, Oberösterreich und Tirol gefördert.

Abstract

In quantum chemistry, material science, physics and other fields, modeling atoms and molecular systems is becoming increasingly popular over the last decades. Approaches, like the Hartree-Fock method (HF), do not include the total electronic energy as compared to more advanced ones (e.g. Coupled Cluster), which are computationally much more demanding and therefore several orders of magnitudes slower to simulate the required task. The difference of HF and post-HF methods is improved by adding the London-dispersion interaction, an attractive van der Waals force. While its principle dependence on interatomic distance is well known, several improvements have been suggested in the past.
In this work interpretable correlations for this correction are searched using a machine learning method called Symbolic Regression and the data input of atomic pairs moving apart from each other.

Kurzfassung

Das Modellieren von molekularen Strukturen ist eine zentrale Prblemstellung in der Quantenchemie, der Physik, den Materialwissenschaften und anderen Forschungsgebieten, welche in den letzten Jahrzehnten signifikante Verbesserungen erfahren hat. Frühe Ansätze, wie die Hartree-Fock Methode (HF), basieren auf einer Näherung der elektronischen Energie, welche die Korrelation der Elektronenbewegung vernachlässigt. Verbesserungen wie zB. die Coupled-Cluster-Methode benötigen jedoch deutlich längere Berechnungszeiten. Der Unterschied zwischen HF und diesen späteren Ansätzen kann in gewissen Fällen über die die London-Dispersion, eine anziehende Van-der-Waals Wechselwirkung, beschrieben werden. Während die prinzipielle Abhängigkeit vom interatomaren Abstand gut bekannt ist, gibt es mehrere verschiedene Hypothesen betreffend ihrer genauere Form.
In dieser Arbeit werden interpretierbare analytische Ausdrücke für diese Korrektur mit Hilfe der Symbolischen Regression, einer Methode des maschinellen Lernens, und einem Datensatz bestehend aus Potentialkurven zweiatomiger Moleküle.

Contents

1 Introduction

The total energy of two atoms in close contact to each other is the sum of multiple contributions describing the interaction between nuclei and electrons of both atoms, which originate from attractive (e.g. van der Waals forces including the London-dispersion interaction) and repulsive (e.g. Pauli repulsion) forces.

Via the conventional Hartree method London-dispersion interaction can not be captured. Experimental observations show that the dispersion energy (E_{disp}) depends to the sixth power on the distance (R), multiplied by a dispersion coefficient (C_6), at least in regions with negligible overlap of the electron clouds between two atoms: $E_{disp} \sim -C_6/R^6$. Damping functions are introduced for higher accuracy. Many attempts have been made to find the correct damping function, leading to a set of hypotheses regarding their shape, more details are given in Section 1.4. In this thesis we use machine learning to build mathematical models by automatic learning with data input (see for example Ref. [3]). Symbolic Regression (SR) allows to mine complex models for interpretable correlations. The core objective of this work is to compile via SR a list of analytical expressions giving a close fit to the dispersion energy curve of the full set of selected diatomic molecules. These expressions shall have low analytical complexity and will be collected in a database for further investigation. In Figure 1.1, the whole process is shown in a block diagram.

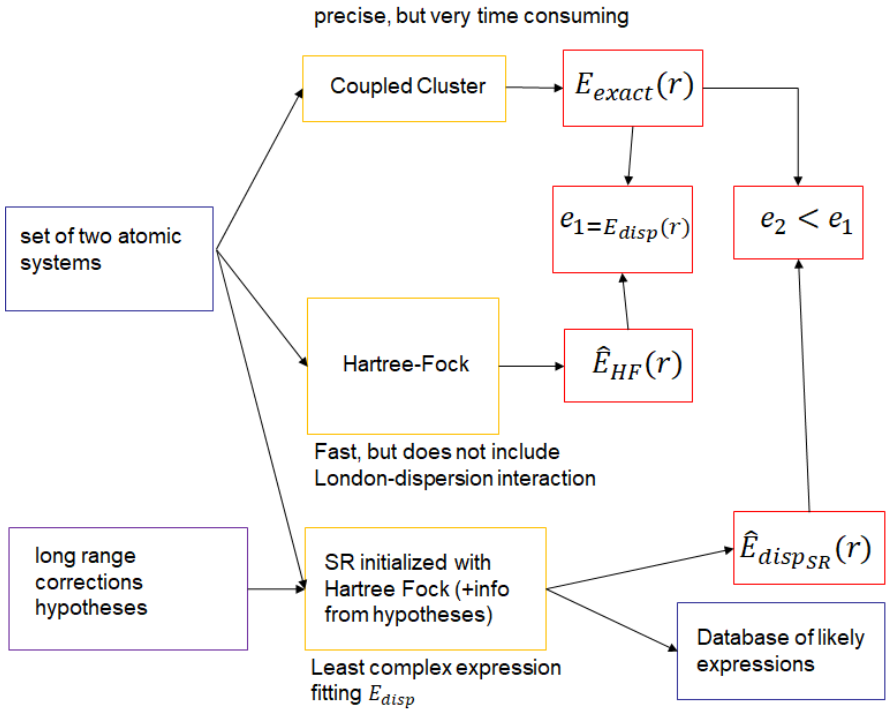

Figure 1.1: Block diagram of the workflow of this Master Thesis as described above.

1.1 Outline

In Section 1.2 a brief overview of the interaction between atoms governed by the famous Schrödinger equation is given. This is followed by Section 1.3, which is dedicated to quantum chemical methods for the calculation of the total energy. An attractive part named the London dispersion interaction is introduced in Section 1.4. In Section 1.5 Symbolic Regression and the used tool throughout this thesis, the software package "'Equation Learner"', is described. In Chapter 2 the calculation of London dispersion interaction between atomic pairs using ab initio methods is reported. In the next chapter 3, Symbolic Regression is used to find interpretable correlations in the simulated data, followed by the discussion of the results in chapter 4.

1.2 Total electron energy and Schrödinger equation

The following chapter is based on **Density Functional Theory in Quantum Chemistry** by Takao Tsuneda [40].

The state of a quantum mechanical system is described by the time-independent Schrödinger equation (equation 1.1), originally proposed in the work Quantisierung als Eigenwert problem (Quantization as an Eigenvalue Problem) [33]. It is an eigenvalue equation inluding the Hamilton operator $\hat{H}$ acting on the wave function Ψ and considering the energy interactions of the system.

$$\hat{H}\Psi = E\Psi, \tag{1.1}$$

In general, the Hamiltonian of an N-body system can be written as

$$\hat{H} = \sum_{i}^{N} \frac{p_i^2}{2m_i} + V, \tag{1.2}$$

with p_i and m_i denoting the momentum and the mass of the $i-th$ particle and V as some interaction potential.

1.2.1 Hartree method

For all systems with more than two bodies involved (e.g. every atom other than the hydrogen atom), solving the Schrödinger equation is not straight-forward due to the so called *three-body problem*. It states that for three bodies interacting with each other the state of the motion can not be solved analytically and a numerical solution needs to be found. Hartree proposed a method for solving the Schrödinger equation for many-electron systems in 1928: the *Hartree method* [14]. The Hamiltonian for such a system is given by equation 1.3

$$\hat{H} = -\frac{\nabla_1^2}{2} - \frac{\nabla_2^2}{2} + V_{ne}(\mathbf{r}_1) + V_{ne}(\mathbf{r}_2) + V_{ee}(\mathbf{r}_1, \mathbf{r}_2), \tag{1.3}$$

The position of the $n-th$ electron is given by $\mathbf{r_n}$ and its gradient vector operator by ∇_n. The first two terms of equation 1.3 are kinetic energy operators, the next two terms are expressing nuclear-electron interaction potentials (called *one-electron operators*) and the last term is considering electron-electron electrostatic interaction potential (*two-electron operator*). For this equation and all other equations in this chapter, atomic units are used as they are commonly present and considered handy in electronic property calculations. Natural constants such as

the electron mass m_e, electron charge e, reduced Planck constant $\hbar = h/2\pi$ and the Coulomb force constant $1/(4\pi\epsilon_0)$ are assumed to be 1. Therefore the energy is also expressed in a new unit called Hartree with $E_h = 4.3597447222071 \cdot 10^{-18}$ J [38]. The potentials are given by

$$V_{ne}(r) = -\sum_{i=1}^{N}\sum_{A=1}^{M}\frac{Z_A}{r_{iA}}, \tag{1.4}$$

and

$$V_{ee}(r_{12}) = \sum_{i=1}^{N}\sum_{j>1}^{N}\frac{1}{r_{ij}}, \tag{1.5}$$

with Z_A as the atomic number of nucleus A, N as the numbers of electrons and M as the number of nuclei [37]. Furthermore Hartree assumes the electrons to be independent from one another, introducing an effective potential (V_{eff}) for the electron-electron interaction potential. The Hamiltonian expressed in equation 1.3 can be divided into two independent terms for each electron.

$$\hat{H} = [-\frac{\nabla_1^2}{2} + V_{ne}(\mathbf{r}_1) + V_{eff}(\mathbf{r}_1)] + [-\frac{\nabla_2^2}{2} + V_{ne}(\mathbf{r}_2) + V_{eff}(\mathbf{r}_2)] = \hat{h}(\mathbf{r}_1) + \hat{h}(\mathbf{r}_2), \tag{1.6}$$

In this equation, $\hat{h}(\mathbf{r_i})$ stands for the Hamiltonian operator for the $i-th$ electron. The total electronic wavefunction can be expressed by a product of functions:

$$\Psi(\mathbf{r}_1, \mathbf{r}_2) = \Phi_1(\mathbf{r}_1)\Phi_2(\mathbf{r}_2), \tag{1.7}$$

With ϵ_i as the eigenenergy for the motion of the $i-th$ electron one obtains the following eigenequation 1.8.

$$\hat{H}\Psi = (\epsilon_1 + \epsilon_2)\Phi_1\Phi_2 = \epsilon\Psi, \tag{1.8}$$

Solving equation 1.8 and obtaining the eigenenergies is described in section 1.3.1.

1.2.2 Slater Determinant

The Slater determinant is a normalized determinant representing an antisymmetrized n-particle wavefunction, which is providing a suitable ansatz for solving the Schrödinger equation and also satisfies the Pauli exclusion principle by its antisymmetric nature. It was proposed by Slater et al. in 1929 [36].

The Hamilton operator of a helium atom can serve as an example for observing that the exchange of electrons is independent of the indexing of the two electron coordinates.

$$\hat{H}(\mathbf{r}_1, \mathbf{r}_2) = \hat{H}(\mathbf{r}_2, \mathbf{r}_1), \tag{1.9}$$

From this follows that the an additionally introduced permutation operator $\hat{P_{12}}$, which replaces the coordinates of two electrons, commutes with H:

$$\hat{H}(\mathbf{r}_1, \mathbf{r}_2)\hat{P_{12}}\Psi(\mathbf{r}_1, \mathbf{r}_2) = \hat{P_{12}}\hat{H}(\mathbf{r}_1, \mathbf{r}_2)\Psi(\mathbf{r}_1, \mathbf{r}_2), \tag{1.10}$$

or

$$[\hat{H}(\mathbf{r}_1, \mathbf{r}_2), \hat{P_{12}}]\Psi(\mathbf{r}_1, \mathbf{r}_2) = 0, \tag{1.11}$$

Therefore, $\hat{P_{12}}^2 = 1$ and the eigenvalues of $\hat{P_{12}}$ are ± 1, corresponding to the symmetric (equation 1.12) and antisymmetric (equation 1.13) wavefunction, respectively.

$$\Psi(\mathbf{r}_1, \mathbf{r}_2)^{(S)} = \frac{1}{\sqrt{2}}(\Psi(\mathbf{r}_1, \mathbf{r}_2) + \Psi(\mathbf{r}_2, \mathbf{r}_1)), \tag{1.12}$$

$$\Psi(\mathbf{r}_1, \mathbf{r}_2)^{(A)} = \frac{1}{\sqrt{2}}(\Psi(\mathbf{r}_1, \mathbf{r}_2) - \Psi(\mathbf{r}_2, \mathbf{r}_1)) = -\Psi(\mathbf{r}_2, \mathbf{r}_1)^{(A)}, \tag{1.13}$$

with $\frac{1}{\sqrt{2}}$ as a normalization constant.
When combining these expressions with equation 1.7, the wavefunctions are given by

$$\Psi(\mathbf{r}_1, \mathbf{r}_2)^{(S)} = \frac{1}{\sqrt{2}}(\Phi_1(\mathbf{r}_1)\Phi_2(\mathbf{r}_2) + \Phi_1(\mathbf{r}_2)\Phi_2(\mathbf{r}_1)), \tag{1.14}$$

and

$$\Psi(\mathbf{r}_1, \mathbf{r}_2)^{(A)} = \frac{1}{\sqrt{2}}(\Phi_1(\mathbf{r}_1)\Phi_2(\mathbf{r}_2) - \Phi_1(\mathbf{r}_2)\Phi_2(\mathbf{r}_1)), \tag{1.15}$$

The Pauli exclusion principle demands that the wavefunctions must be zero when two electrons occupy the same orbital. Only the antisymmetric wavefunction (equation 1.15) is fulfilling this requirement and therefore electronic motions always have antisymmetric wavefunctions. Equation 1.15 can be written as a determinant:

$$\Psi(\mathbf{r}_1, \mathbf{r}_2) = \frac{1}{\sqrt{2}}\begin{vmatrix}\Phi_1(\mathbf{r}_1) & \Phi_1(\mathbf{r}_2)\\ \Phi_2(\mathbf{r}_1) & \Phi_2(\mathbf{r}_2)\end{vmatrix}$$

Generalizing this to cases of three or more electrons leads to the so called Slater determinant formalism ([36]).

$$\Psi(\mathbf{r}_1, \mathbf{r}_2, ..., \mathbf{r_N}) = \frac{1}{\sqrt{N!}}\begin{vmatrix}\Phi_1(\mathbf{r}_1) & \Phi_1(\mathbf{r}_2) & \cdots & \Phi_1(\mathbf{r_N}\\ \Phi_2(\mathbf{r}_1) & \Phi_2(\mathbf{r}_2) & \cdots & \Phi_2(\mathbf{r_N}\\ \vdots & \vdots & \vdots & \ddots\\ \Phi_N(\mathbf{r}_1) & \Phi_N(\mathbf{r}_2) & \cdots & \Phi_N(\mathbf{r_N}\end{vmatrix}$$

1.3 Ab initio methods

In quantum chemistry, material science, physics and other fields, modeling atoms and molecule systems is becoming increasingly popular over the last decades. One of the tools providing such a simulation are ab initio methods. Ab initio is latin for "'from the beginning"' and ab initio methods are describing the computational solution of the electronic Schrödinger equation and obtaining of properties like positions of a collection of atomic nuclei, the total number of electrons in the system, the total electronic energy, the electron density and other. [7]

Two of this methods are described in this chapter. The Hartree Fock method as an earlier approach and Coupled Cluster, a computationally much more demanding but more accurate method that also take the correlated motion of electrons into account. [40]

1.3.1 Hartree Fock

The Hartree-Fock method was proposed by Fock 1930 and combines the Slater determinant ansatz (see Section 1.2.1) with the Hartree method (see Section 1.2.2) [6]. In the same year, Slater independently suggested a similar approach [35].

In this section only the important principles and results of this method will be discussed. A more detailed mathematical derivation is provided by e.g. Tsuneda [40]. When following this derivation, one obtains the Coulomb operator $\hat{J}_j$ and the exchange operator $\hat{K}_j$. They are defined as

$$\hat{J}_j(\mathbf{r}_1)\Phi_i(\mathbf{r}_1) = \int d^3\mathbf{r}_2 \Phi_j^*(\mathbf{r}_2)\Phi_j(\mathbf{r}_2)\frac{1}{r_{12}}\Phi_i(\mathbf{r}_1), \tag{1.16}$$

and

$$\hat{K}_j(\mathbf{r}_1)\Phi_i(\mathbf{r}_1) = \int d^3\mathbf{r}_2 \Phi_j^*(\mathbf{r}_2)\Phi_i(\mathbf{r}_2)\frac{1}{r_{12}}\Phi_i(\mathbf{r}_1), \tag{1.17}$$

The Fock matrix is

$$\hat{F} = \hat{h} + \sum_j^n (2\hat{J}_j - \hat{K}_j), \tag{1.18}$$

with $\hat{h}$ as the corresponding core Hamilton operator. Aiming at the minimization of the electronic ground state energy by variation of the orbitals Φ_i one obtains the effective one-electron equation known as the Hartree-Fock equation:

$$\hat{F}\Phi_i = \epsilon_i \Phi_i \tag{1.19}$$

with ϵ as the so-called orbital energies.

$$\epsilon_i = \int d^3\mathbf{r}_1 \Phi_i^*(\mathbf{r}_1)\hat{F}\Phi_i(\mathbf{r}_1), \tag{1.20}$$

To solve this nonlinear equation 1.19, the self-consistent field (SCF) method is usually applied. This process is described in Figure 1.2.

Solving the Hartree-Fock equation for molecules with such a iterative method cannot be done by hand in general and the help of computers is required. [40]

1.3.2 Coupled Cluster

The Coupled cluster method is an advanced ab initio method which takes all electron correlation into consideration in a highly effective way. It was developed by Čìžek in 1966 [5]. The wavefunction is expanded in this method to

$$\Psi^{CC} = exp(\hat{T})\Phi = (1 + \hat{T} + \frac{1}{2}\hat{T}^2 + \cdots)\Phi, \tag{1.21}$$

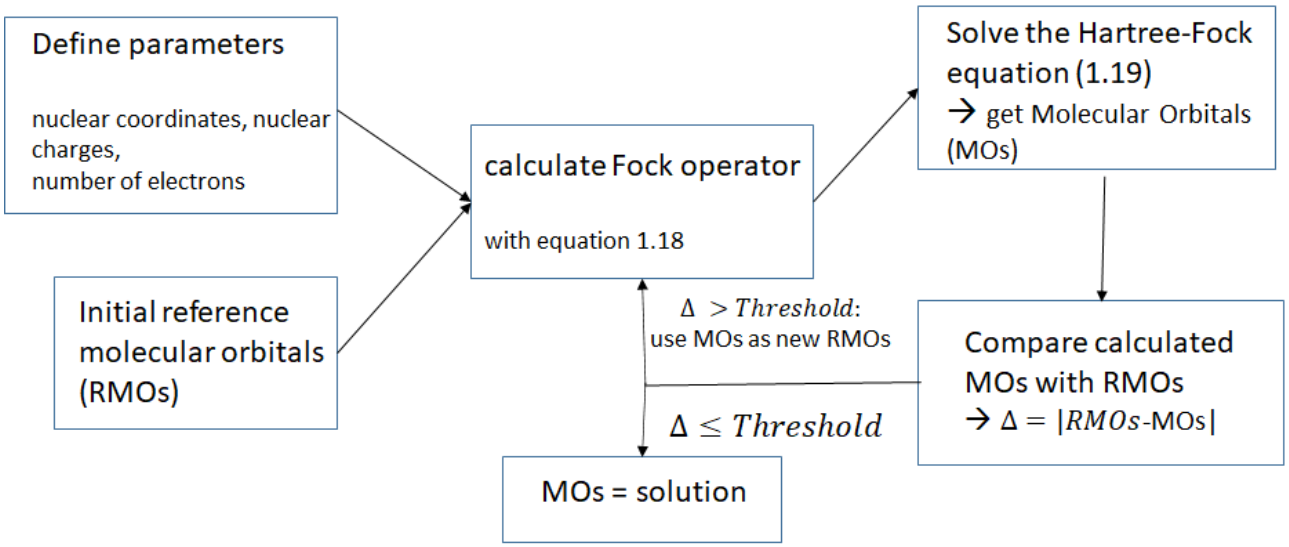

Figure 1.2: Block diagram of the Hartree-Fock method. The solution of equation 1.19 is accepted, if the difference between the calculated molecular orbitals and the reference molecular orbitals (RMOs) is below a threshold value. If not, the calculation is repeated, using the calculated molecular orbitals as new RMOs. The initial RMOs are guessed.

with Φ as the Slater determinant of the ground electron configuration and $\hat{T}$ as an operator summing all excitations (see equation 1.22)

$$\hat{T}\Phi = \hat{T}_1\Phi + \hat{T}_2\Phi + \cdots , \tag{1.22}$$

with $\hat{T}_1$, $\hat{T}_2$ specifying the operators of single, double and higher excitations, respectively. Using equation 1.22, equation 1.21 can be written as

$$\Psi^{CC} = exp(\hat{T})\Phi = (1 + \hat{T}_1 + \hat{T}_2 + \frac{1}{2}\hat{T}_1{}^2 + \hat{T}_1\hat{T}_2 + \frac{1}{2}\hat{T}_1{}^2 + \cdots)\Phi, \tag{1.23}$$

The coupled cluster singles and doubles (CCSD) method is calculating $\hat{T}$ up to the double excitation($\hat{T}_2$), while CCSD(T) additionally account for perturbative triples coming from triple excitation. [40]

More details about the Coupled cluster method can be found in Ref. [22].

1.4 London dispersion interaction

Besides Coloumb interaction, several other types of interaction contribute to the total energy. Some of them repulsive (e.g. Pauli repulsion) while others are attractive. Such an attractive part is the London dispersion interaction, which is a van der Waals interaction acting between all bodies, independent of charge or multipole moment. First described by Fritz London in Ref. [23] via a principle relation and proportionality as stated in equation 1.24 - with element specific parameters as the atomic ionization potential IP and the dipole polarizability α of atom A and B, [8]

$$E_{disp} \propto -3/2 \, \frac{IP^A \, IP^B}{IP^A + IP^B} \, \alpha^A \, \alpha^B \, R^{-6} = -\frac{C_6}{R^6}. \tag{1.24}$$

R denotes the distance between two bodies (from center to center) and C_6 is the sixth-order dispersion coefficient. The exact nature of this coefficient and additional damping functions, multiplied to it for compensation at short range, are still under discussion and several hypotheses exist, regarding their shape. The most prominent suggestions are the DFT-D3 variants by Grimme et al. [13, 12], which are refined versions based on their earlier approaches DFT-D1 [9] and DFT-D2 [11], along other methods like e.g. the Tkatchenko–Scheffler model (DFT+TS) [39].

1.4.1 The DFT-D2 Correction

Although the DFT-D2 correction has been surpassed and replaced by later approaches (see section 1.4.2) it is still worth a detailed description before proceeding to DFT-D3, as it gives a simpler model for the London-dispersion correction.

The dispersion correction of the DFT-D2 variant is stated in equation 1.25.

$$E_{disp}^{DFT-D2} = -\frac{1}{2} s_6 \sum_{A \neq B} \frac{C_6^{AB}}{R_6^{AB}} f_{damp,n}^{DFT-D2}(R_{AB}), \tag{1.25}$$

with s_6 as a scaling parameter, C_6^{AB} as the dispersion coefficient (see equation 1.27 and 1.28), R_6^{AB} as the distance between two bodys and a damping function f_{damp}^{DFT-D2} (see equation 1.26), which is introduced to take the contribution of overlapping electron clouds into account and damp the sixth-order decay in this region.

$$f_{damp,n}^{DFT-D2}(R_{AB}) = \frac{1}{1 + exp(-20(R_{AB}/R_r - 1))}, \tag{1.26}$$

where R_r stands for the sum of the van der Waals radii of atom A and B. The dispersion coefficient (C_6^{AB}) is given as the geometric mean of two element-specific coefficients (C_6^A).

$$C_6^{AB} = \sqrt{C_6^A C_6^B}, \tag{1.27}$$

$$C_6^A = 0.05 \; N \; IP^A \; \alpha^A, \tag{1.28}$$

with N as a number depending on the row in the periodic table of the respected element: 2,8,15,36, or 54. $N = 2$ for all elements in the first row (H and He) and analogous for row two to five. IP^A stands for the atomic ionization potential and α^A for the dipole polarizability of the respected element.

1.4.2 The DFT-D3 Correction

The DFT-D3 correction has various versions, with different damping functions. It was introduced first in 2010 [13] - here called DFT-D3(0). 2011 Grimme published a refinement called the Becke–Johnson damping version (DFT-D3(BJ), [12]) using a different damping function. Another modification of this approach was done by Schwabe et al. 2015 [32]. All of those DFT-D3 versions are applicable to the first 94 elements of the periodic table and use the same dispersion coefficients. Finally, Grimme added another modification in 2019, called DFT-D4 [2], which covers all elements up to radon (Z = 86) and include also three-body effects that are not in the scope of this master thesis.

DFT-D3(0)

The proposed DFT-D3(0) dispersion correction is stated in equation 1.29. With s as a scaling parameter, R_{AB} as the distance between two bodys, C^{AB} as dispersion coefficient and damping functions $f_{damp}^{DFT-D3(0)}$ for sixth and eight order terms.

$$E_{disp}^{DFT-D3(0)} = -\frac{1}{2}\sum_{A\neq B}\sum_{n=6,8} s_n \frac{C_n^{AB}}{R_{AB}^n} f_{damp,n}^{DFT-D3(0)}(R_{AB}), \tag{1.29}$$

In the following, the damping function in sixth (equation 1.30) and eight order (equation 1.30) are given. R_0^{AB} is the cut-off radius for the atom pair AB, which is the combined van der Waals radius of A and B. More details about this radius and why it should not be confused with the often used term "'vdW radii"' is given in [13]. α_6 determines the steepness of the damping function and was adjusted manually to a value of 14, in order to limit the dispersion correction to 1 % of the maximal dispersion energy.

$$f_{damp,6}^{DFT-D3(0)}(R_{AB}) = \frac{1}{1+6(R_{AB}/(R_0^{AB})^{-\alpha_6}}, \tag{1.30}$$

$$f_{damp,8}^{DFT-D3(0)}(R_{AB}) = \frac{1}{1+6(R_{AB}/(s_{r,6}R_0^{AB})^{-(\alpha_6+2)}}, \tag{1.31}$$

The C_6^{AB} dispersion coefficients are calculated from atomic C_6 values and also considering the chemical environment of each atom. More details about the derivation can be found in Ref. [8]. C_8^{AB} dispersion coefficients are calculated with equation 1.32.

$$C_8^{AB} = 3C_6^{AB}\sqrt{Q_A Q_B}, \tag{1.32}$$

with

$$Q_A = \frac{1}{2}\sqrt{Z_A}\frac{\langle r^4\rangle_A}{\langle r^2\rangle_A}, \tag{1.33}$$

With Z_A as the nuclear charge of the respected atom, $\langle r^4\rangle_A$ as the quadrupole- and $\langle r^2\rangle_A$ as the dipole-moment-type expectation values derived from atomic densities. See Ref. [13] for more information about the derivation.

DFT-D3(BJ)

The DFT-D3(BJ) variant was proposed 2011 by Grimme et al. [12] and uses a damping function proposed by Becke and Johnson [1].

$$E_{disp}^{DFT-D3(BJ)} = -\frac{1}{2}\sum_{A\neq B}\sum_{n=6,8} s_n \frac{C_n^{AB}}{R_{AB}^n + [f_{damp,n}^{DFT-D3(BJ)}(R_{BJ}^{AB})]^n}, \tag{1.34}$$

While using s_n and C_n^{AB} similar to DFT-D3(0), the damping function is different. With α_1 and α_2 as adjustable parameters.

$$f_{damp,n}^{DFT-D3(BJ)}(R_{BJ}^{AB}) = a_1 R_{BJ}^{AB} + a_2 \tag{1.35}$$

The cut-off radii (R_0^{AB}) in DFT-D3(0) is replaced by R_{BJ}^{AB} and depends on the dispersion coefficients as stated in equation 1.36.

$$R_{BJ}^{AB} = \sqrt{\frac{C_8^{AB}}{C_6^{AB}}} = \sqrt{\frac{3}{2}}\sqrt{\sqrt{Z_A}\frac{\langle r^4\rangle_A}{\langle r^2\rangle_A}\sqrt{Z_B}\frac{\langle r^4\rangle_B}{\langle r^2\rangle_B}}, \tag{1.36}$$

DFT-D3(CSO)

This variant was proposed by Schwabe et al. 2015 and is called the "'C-six-only"' version of DFT-D3. Eliminating the R^{-8} term completely with the argumentation that the interpolation of this term can be expressed with sigmoidal damping functions as well [32]. This variant is reported to have similar results as DFT-D3(BJ) while only including a R^{-6} term [32].

$$E_{disp}^{DFT-D3(CSO)} = -\frac{1}{2}\sum_{A\neq B}[s_6 + \frac{c_1}{1+exp(R_{AB}-2.5R_{AB}^{BJ})}]\frac{C_6^{AB}}{R_{AB}^6+(2.5^2)^6}, \tag{1.37}$$

With c_1 as a scaling parameter (in the original publication named a_1).

1.4.3 The DFT+TS Correction

In 2009 Tkatchenko and Scheffler proposed a parameter-free version of the London-dispersion interaction. Using a summation of C_6 dispersion coefficients which are derived from the electron density of a molecule or solid and accurate reference data for the free atoms [39]. To avoid confusion the dispersion energy is called E_{disp}^{TS}, similar like in the DFT-D variants, opposed to the term E_{vdW} in the original publication. R_A^0 and R_B^0 denote the van der Waals radii of the respected atoms and R_{AB} the distance between the atoms.

$$E_{disp}^{TS} = -\frac{1}{2}\sum_{A\neq B} f_{damp}(R_{AB}, R_A^0, R_B^0)C_{6AB}R_{AB}^{-6}, \tag{1.38}$$

with

$$C_{6AB} = \frac{2C_{6AA}C_{6BB}}{[\frac{\alpha_B^0}{\alpha_A^0}C_{6AA} + \alpha_A^0\alpha_B^0 C_{6AA}]}, \tag{1.39}$$

with α_A^0 as the static polarizability of A and C_{6AA} as homonuclear parameters, calculated from the respected parameters of free atoms, which are obtained from Ref. [4].

$$C_{6AA} = v_A^2 C_{6,free}^{AA} \tag{1.40}$$

and

$$\alpha_A^0 = v_A\alpha_{A,free}^0, \tag{1.41}$$

with v_A as the ratio between the effective volume of atom A in a bound state and the volume of the free atom calculated by a Hirshfeld partioning. [17]

$$v_A = \frac{\int d\mathbf{r} w_A(\mathbf{r})\rho(\mathbf{r})r^3}{\int d\mathbf{r}\rho_A^{free}(\mathbf{r})r^3} \tag{1.42}$$

with $\rho(\mathbf{r})$ as the total density of the system, $\rho_A^{free}(\mathbf{r})$ as reference density of the free atom A and w_A as the atomic Hirshfeld partitioning weight for atom A, given by

$$w_A = \frac{\rho_A^{free}(\mathbf{r})}{\sum_B \rho_B^{free}(\mathbf{r})}. \tag{1.43}$$

The damping function is stated as

$$f_{damp}(R_{AB}, R_{AB}^0) = \frac{1}{1 + exp(-d(\frac{R_{AB}}{s_r R_{AB}^0} - 1))}, \tag{1.44}$$

with d and s_r as free parameters. The d parameter is controlling the steepness of the damping function. Tkatchenko and Scheffler suggest a value of $d = 20$, empirically found by an investigation done by Grimme et al. [11]. The van der Waals radii $R_{AB}^0 = R_A^0 + R_B^0$ in the damping function are modified similar to equation 1.40 and 1.41.

$$R_A^{free} = v_A^{1/3} R_A^{free}, \tag{1.45}$$

1.5 Symbolic Regression

Machine Learning methods like deep learning with neural networks are showing promising results (e.g. Schmidhuber et al. [31]), but lack in interpretability of the found solutions. Symbolic Regression on the other hand is delivering explainable models by learning analytical connections from data input.

In this master thesis an Symbolic Regression method called Equation Learner (EQL) from Martius et al. [26] is used to investigate the ability to test hypotheses on a real problem in molecular chemistry.

1.5.1 Equation Learner (EQL)

The Equation Learner, introduced by Martius et al. in 2018 [26] is identifying underlying equations from data for systems governed by an analytical formula, meaning that the data must originate from an (unknown) analytical expression. It is doing so by using a multi-layer feed-forward network with units (operations such as multiplication or functions like sinus) which represent the building blocks of algebraic expressions, learning the connection by a given data set providing input (x) and output (y) values. The possible units provided in the EQL are divided into unitary units (e.g sine, cosine, exponential and identity), needing only a single input and binary units (e.g. multiplication) which require two of them. In 2018 Martius et al.[30] improved the EQL by adding regularized division units, creating a pole for a/d at $d \to 0$, which is bypassed by equation 1.46.

$$h^{\Theta}(a, d) = \begin{cases} \frac{a}{d}, & if\ d > \Theta_D \\ 0, & otherwise \end{cases}, \tag{1.46}$$

with a threshold value $\Theta_D > 0$. With the choice of units to be included in the network, the number of identical units and the number of hidden layers, the network architecture is influenced, which is displayed in Figure 1.3. In this description the terms weight and bias are used to denote floating point numbers (floats) multiplied to the various unitary and binary units ($\to$ weights) and stand-alone floats ($\to$bias).

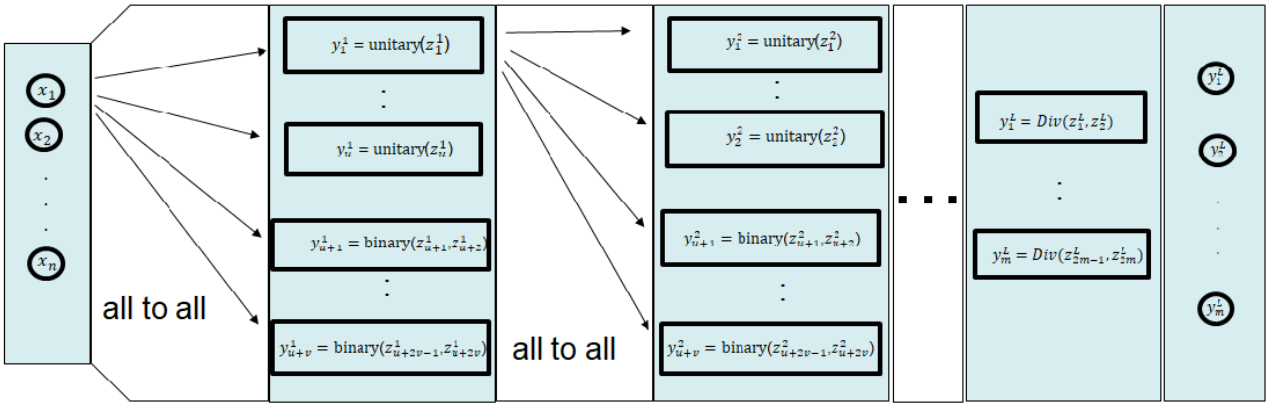

Figure 1.3: The network structure of the improved EQL by Martius et al. [30]. With intermediate outputs $z_k^{\vec{l}} = \sum_{i=1}^{n}(W_{ki}^{l}\ y^{\vec{l-1}}) + b_k,\ k = 1, 2, \cdots, (u + 2v), l = 1, 2, \cdots, L$ ($W \cdots$ weight, $b \cdots$ bias, $n \cdots$ number of inputs, $m \cdots$ number of outputs, $L \cdots$ number of layers, $u \cdots$ number of unitary units, $v \cdots$ number of binary units). The division units are added in the final layer.

The output from unitary and binary units are joined to form a formula expression in a linear read-out,

$$\mathbf{y}^L = \hat{W}^L \mathbf{y}^{L-1} + \mathbf{b}^L, \tag{1.47}$$

with L denoting the last layer, $\hat{W}$ as weight matrix and $\mathbf{b}$ as bias vector. Details about the network architecture are provided in The work on extrapolation and learning equations by Martius et al. [26].

In the following two sections the training and regularization mechanism of the EQL are explained on the modifications done by myself regarding regularization strength and duration as well as the total training duration.

Training

A loss function is introduced to obtain a quantitative measure of how well a model is predicting,

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N}(\|\Psi(x_i) - y_i\|^2) + \lambda \sum_{l=1}^{L}\left|\hat{W}^l\right|_1 + P^{\Theta}, \tag{1.48}$$

with N as the number of values in y, L for the number of layers in the network and $\left|\hat{W}\right|_1$ as the L1-norm of the weight matrix, summing the absolute values of all entries. The loss is consisting of three parts. First the regular loss, giving the difference between calculated values from the model (Ψ) and the results provided (y_i). A regularization term is encouraging networks with sparse connections and prevents overfitting by penalizing complex models with λ as regularization strength, controlling the contribution to the total loss. Finally, a penalty term is added for small and negative denominators, calculated via

$$P^{\Theta} = max(\Theta_D - d) \tag{1.49}$$

with Θ_D as division threshold and d as denominator (see equation 1.46). The goal of the training is to minimize the loss using back-propagation. Updates are calculated with the Adam

algorithm ([20]). The training consists of a minimum of 20 training episodes per hidden layer (layer in between first and last). Each with 50 normal training epochs, where in each of them all data is fed to the network in mini-batches (denoting the number of training examples) and one penalty epoch, where randomly sampled input data points in the expected test range is added to the whole data to prevent overfitting [30]. More details about the concept of overfitting can be found in Ref. [15].

The input data is split into training data (usually around 80 to 90 percent of all data) used to train the network and validation data to validate its process, which are both taken from the same data input domain. Optionally, test data can be provided to extrapolate to unseen domains as well. To monitor the training process, the loss is calculated after every training episode, resulting in train loss, validation loss and test loss. Additionally, the number of active nodes in the network is calculated, meaning that the number of weights bigger than a given threshold value, which represent terms in the formula that are contributing to the evaluated result in a specified amount. This value is called (network) complexity.

The network is trained for 20 training episodes per hidden layer. After that the training is continued, till train loss ($\mathcal{L}^{train}$), validation loss ($\mathcal{L}^{validation}$) and complexity (C) are considered to be converged. Precisely until the sum of train loss and validation loss over the last 5 training episodes are not changing for more than 10 percent of the current loss value and the complexity is not changing at all over the last 5 training episodes:

$$\sum_{i=1}^{5} \left| \mathcal{L}_{i+1}^{train} - \mathcal{L}_{i}^{train} \right| < 0.1 \mathcal{L}_{1}^{train}$$

$$\sum_{i=1}^{5} \left| \mathcal{L}_{i+1}^{validation} - \mathcal{L}_{i}^{validation} \right| < 0.1 \mathcal{L}_{1}^{validation}$$

$$\sum_{i=1}^{5} |C_{i+1} - C_i| = 0,$$

with i counting from the current calculated loss ($i = 1$) backward till $i = 5$, representing the fifth last entry of the monitored value.

Regularization

Regularization is used to encourage sparsity in the network and therefore get analytically more simple models. The regularization scheme used in the work of this master thesis is different from the one Martius et al. proposed [26]. First the regularization strength λ is set to zero, eliminating the second term from the loss calculation (see equation 1.48). After $2L$ training episodes, where L is representing the number of hidden layers, λ is set to a value calculated as

$$\lambda = \lambda_{\%} \frac{\mathcal{L}[-1]}{W_{sum}[-1]} \tag{1.50}$$

with $W_{sum} = \sum_{l=1}^{L} \left| W^l \right|_1$ as the sum of the L1-norm of the weight matrices and $\mathcal{L}[-1]$ as the loss calculated at the end of the previous training episode. The regularization stays active as long as it is not changing over three training episodes and is then switched off, meaning λ is set to zero again to prevent a large impact of the regularization on the whole training procedure. During this phase the weights are pressed down to lower values and after the regularization process, all weights smaller than a given threshold value (complexity threshold) are set to zero.

2 Numerical calculations of London-dispersion interaction between atomic pairs

The goal of this master thesis is to fit the London-dispersion interaction between atomic pairs with models found by Symbolic Regression, including prior knowledge about the correction from hypotheses (see section 1.4). To do so, a database has to be created that can be provided to the machine learning algorithm as training data. Therefore, 51 combinations of atoms are simulated at the Coupled Cluster (see section 1.3.2) and Hartree-Fock (see section 1.3.1) level of theory. The first three noble gas atoms in the periodic table (helium, neon and argon) are combined with each element having an atomic number smaller than argon. Involving noble gas atoms in the considered atomic pairs enforces that van der Waals interactions, like the London-dispersion, are practically the only attractive contributions to the interaction potential (see Introduction to Computational Chemistry by Jensen et al. page 35 [18]). An example is shown in Figure 2.1. As the London-dispersion interaction are not covered in the Hartree-Fock method (E_{HF}), but in Coupled Cluster (E_{CC}), the dispersion energy (E_{disp}) is considered to be the difference $E_{disp} \approx E_{CC} - E_{HF}$. This approximation is very precise in the long-range regime, but does not necessarily hold true in the short-range regime, where exchange effects are dominating the total electron energy, as stated in Ref. [10].

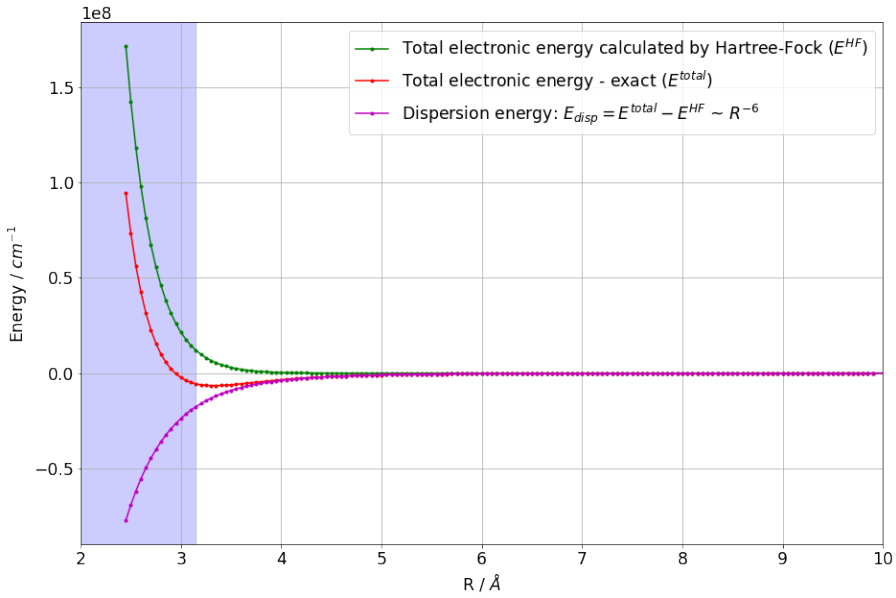

Figure 2.1: Total electronic energy as calculated by Hartree-Fock, Coupled Cluster (exact) and the dispersion energy (E_{disp}), as wavenumber in reciprocal centimetres, simulated for the atomic pair of helium and chlorine with increasing distance in Ångström.

2.1 Details about ab initio calculations

In this section the settings for the simulation of atomic pairs Coupled Cluster and Hartree-Fock, using the Molpro package [42], are explained. The atomic pairs are separated from each other by the distance R (as can be seen in Figure 2.1), which is increased from 2.45 to 20 Ångström (Å) in very small steps to get enough training data for the neural network of the Equation Learner (see section 1.5.1). A discretization from 2.45 to 10 Å in 0.05 Å steps and from 10 to 20 Å in 0.1 Å steps is chosen. These very small step sizes are causing effects as discussed in section 2.4. As an one-electron basis set the augmented correlation-consistent basis set quintuple zeta (aug-cc-pV5Z) of Dunning et al. [43] is used.

All computations were performed on the Quantum Chemistry Cluster of the Institute of Experimental Physics at Technical University Graz using Molpro version 2012.1.

2.2 Results

The simulated dispersion energies (E_{disp}) of 51 atomic pairs, specified at the start of chapter 2, are presented in the following. At first an overview over all resulting E_{disp} is given in Figure 2.2. Showing that the dispersion energies are starting at very different energy values with different order of magnitude.

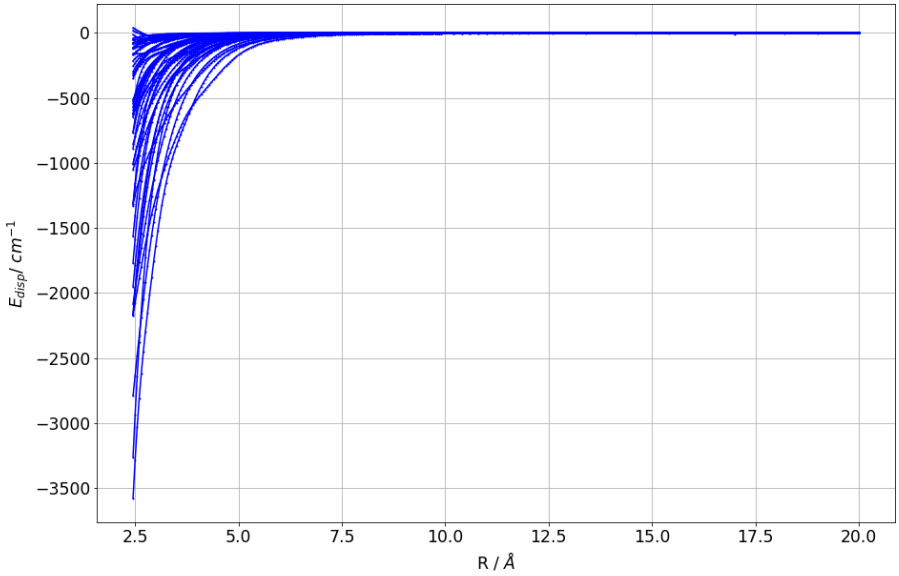

Figure 2.2: The simulated dispersion energies E_{disp} of all 51 atomic pairs, specified at the start of chapter 2, over the whole distance range.

For a better comparability, the energy values are normalized by their respective minimum value. As a result all dispersion energies should start with 1 as the minimum value is supposed to be the one with the closest distance and decreasing until they approach 0 asymptotically. As we can see in Figure 2.3 not all E_{disp} trajectories have this curve course. Some of them (namely HeN, HeP, HeS and NeN) are not starting at their respected minimum value at all (see Figure 2.4), which is considered to be an unphysical behavior of London-dispersion forces

and can be explained by a flawed calculation due to multi-reference configuration interaction (see [18], page 158f).

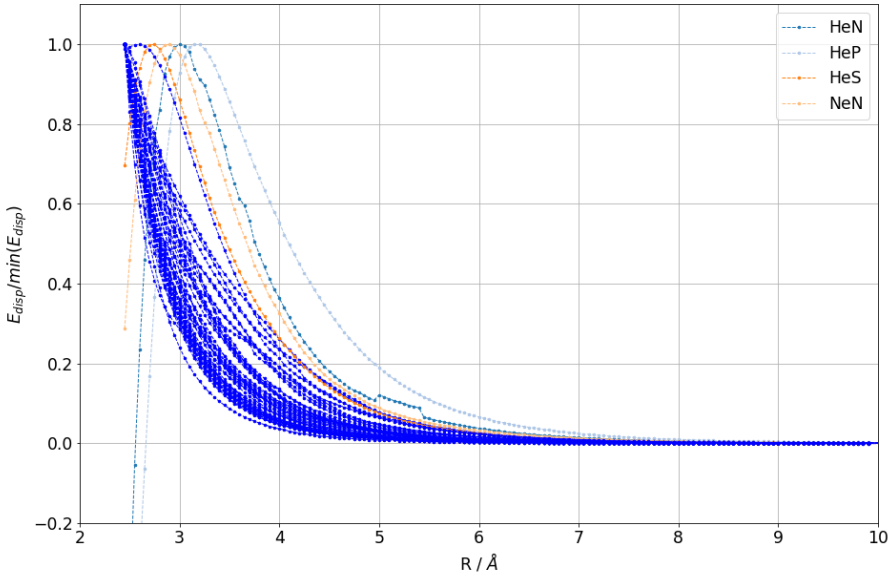

Figure 2.3: The simulated dispersion energies E_{disp} of all 51 atomic pairs, normalized to their respected minimum value. Those starting from a value other than their respected minimum are colored differently and listed in the legend. The x-axis is cropped at 10 Å for better visibility.

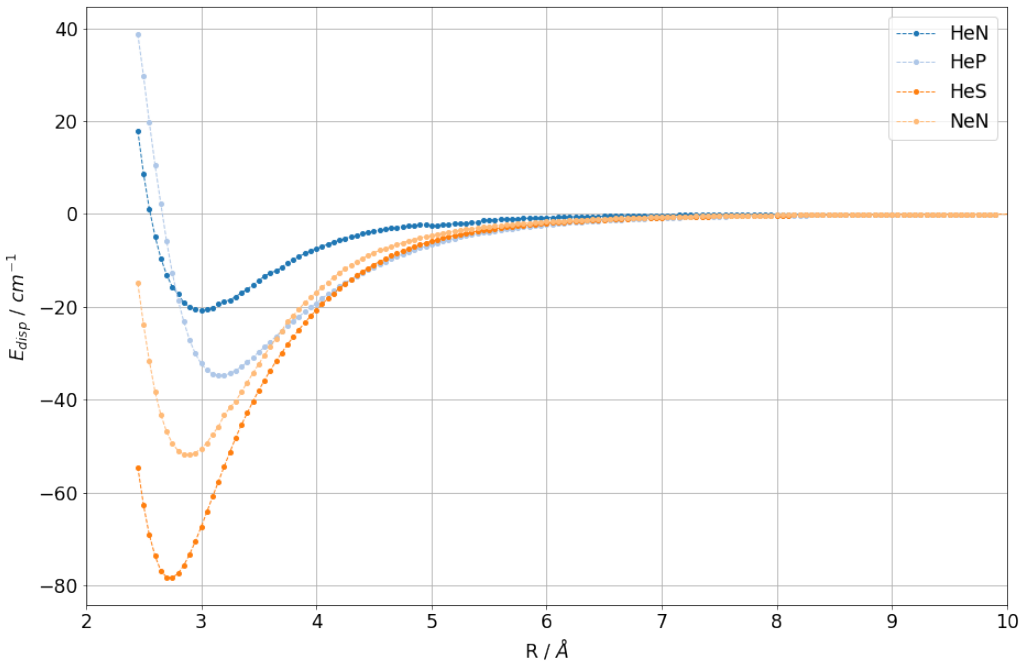

Figure 2.4: Simulated dispersion energies E_{disp} of atomic pairs showing a unphysical behavior.

2.2.1 Curves with outliers

In order for machine learning algorithms to find a clear fit of the dispersion energy with the simulated data described in section 2.1, it has to be free from physically inconsistent curvatures. Curves with outliers caused by details of the used simulation techniques are dropped since a further investigation of these issues lies beyond the scope of this research. These outliers may cause the neural network trained in the EQL to learn wrong connections (see e.g. the work of Khamis et al. [19]) and therefore result in analytical expressions not depicting the physical nature of the desired London-dispersion interaction. To get a clearer view of this problem, the dispersion energy is displayed logarithmic in Figure 2.5. Note that the natural logarithm is not defined for negative values and zero. The London-dispersion force is an attractive force and has negative values approaching zero asymptotically. Therefore, as soon as E_{disp} is reaching zero, the values are cropped, although values coming from simulated points with larger distance are nonzero again, which is considered to be numerical noise. The remaining values of E_{disp} are multiplied by -1 and logarithmized: $log(-E_{disp})$. The curves with outliers are shown in Figure 2.5.

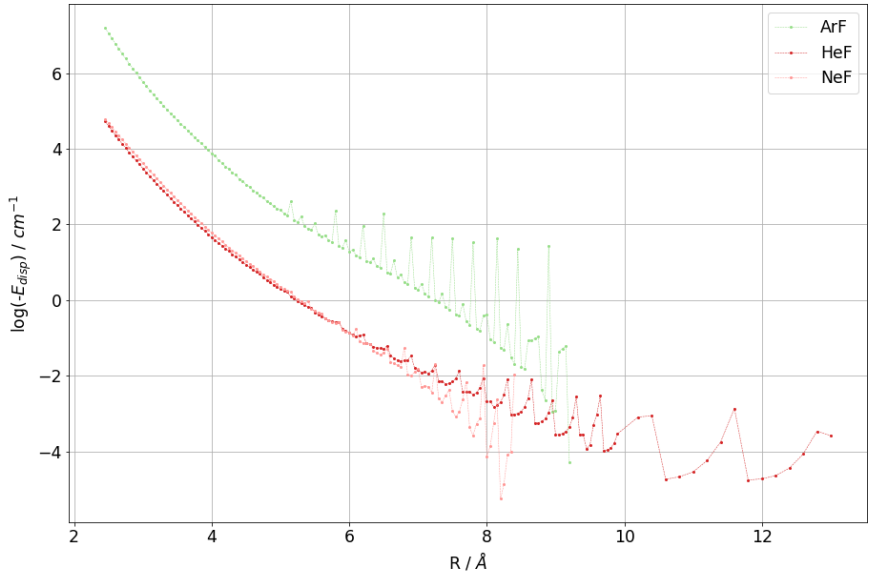

Figure 2.5: The natural logarithm of the dispersion energy obtained from the performed simulation of atomic pairs with curve courses including some outliers.

Additionally to the E_{disp} of atomic pairs shown in Figure 2.4 and 2.5, another atomic pair is excluded from further investigation. The E_{disp} of NeS, consisting of the atoms neon and sulfur, does not decrease monotonic with decreasing distance, but has a minimum at $R \approx 2.6$ Å. Therefore its curvature, shown in Figure 2.6, is not considered to be unphysical, but still not suitable for the training of machine learning algorithms.

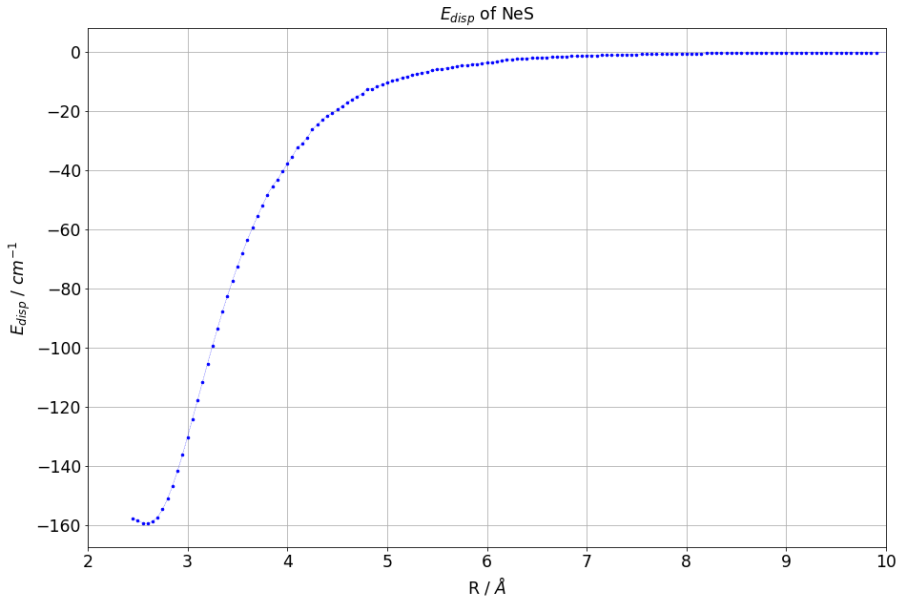

Figure 2.6: E_{disp} of the atomic pair NeS with increasing distance R. The respective minimum value is not at the lowest distance. The x-axis is cropped at 10 Å to see this more clearly.

After the exclusion of problematic dispersion energy curves, due to numerical details of the simulation and to obtain a clean data set, 43 (of 51 in total) simulated dispersion energy calculations of atomic pairs remain (see Figure 2.7).

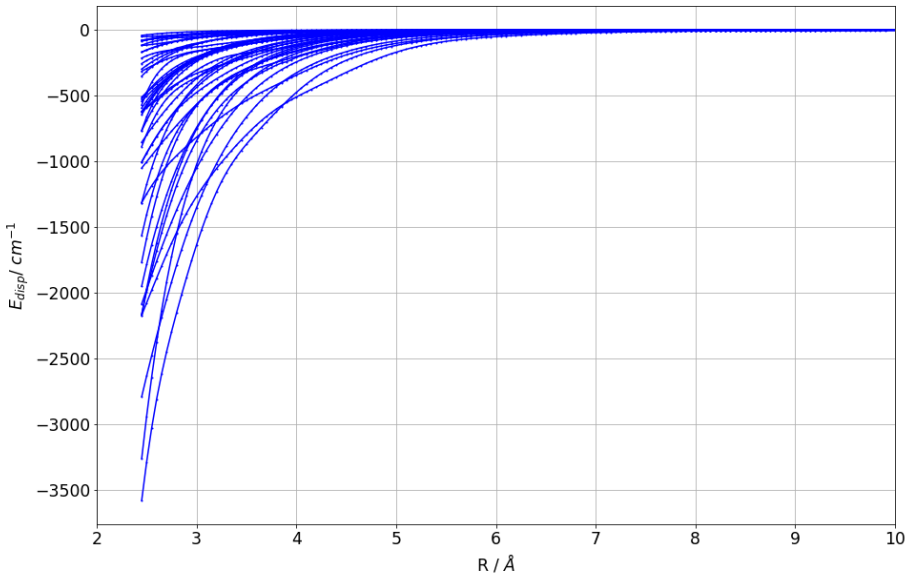

Figure 2.7: The simulated dispersion energies E_{disp} of all 43 atomic pairs chosen for further investigation, specified in section 2.2, over the distance. The x-axis is cropped at 10 Å for better visibility.

2.3 Classification

The 43 remaining dispersion energy curves (shown in Figure 2.7) are categorized according to the respective noble gas involved. As two noble gases are combined as well and double counting is avoided, there are more combination involving Ar (17, shown in Figure 2.10) than with Ne (13, shown in Figure 2.9) and He (13, shown in Figure 2.8). Note that some of the atomic pairs are excluded from this investigation, as described in section 2.2.

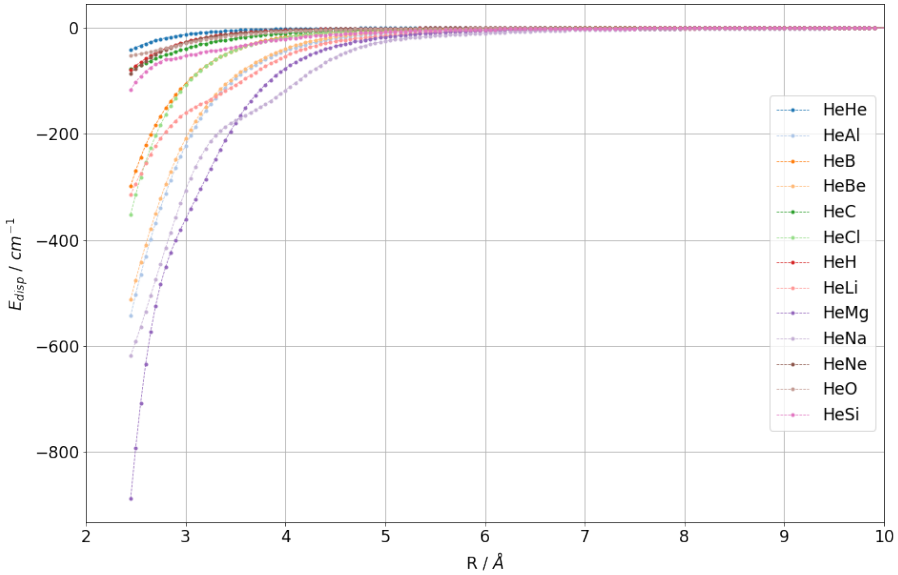

Figure 2.8: The simulated dispersion energies E_{disp} of all atomic pairs involving He over the distance. The x-axis is cropped at 10 Å for better visibility.

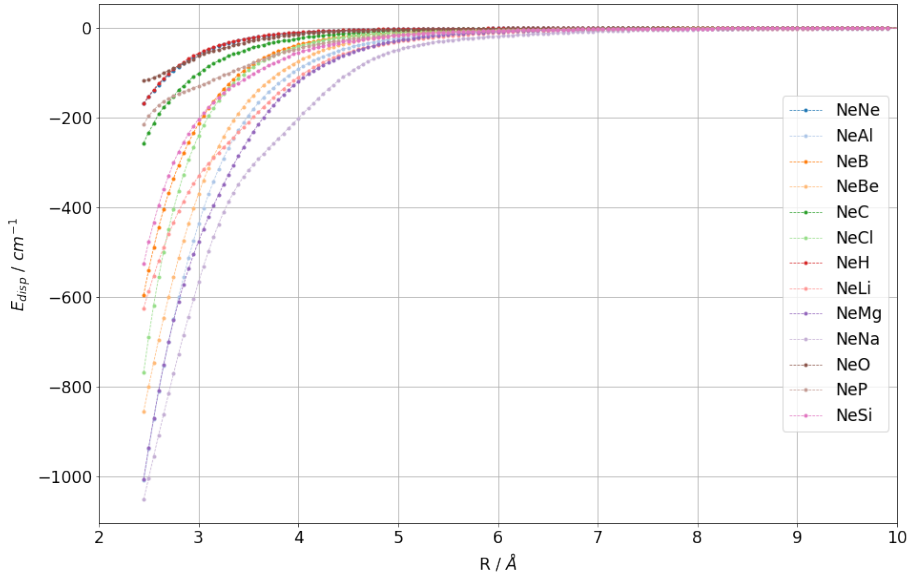

Figure 2.9: The simulated dispersion energies E_{disp} of all atomic pairs involving Ne over the distance. The x-axis is cropped at 10 Å for better visibility.

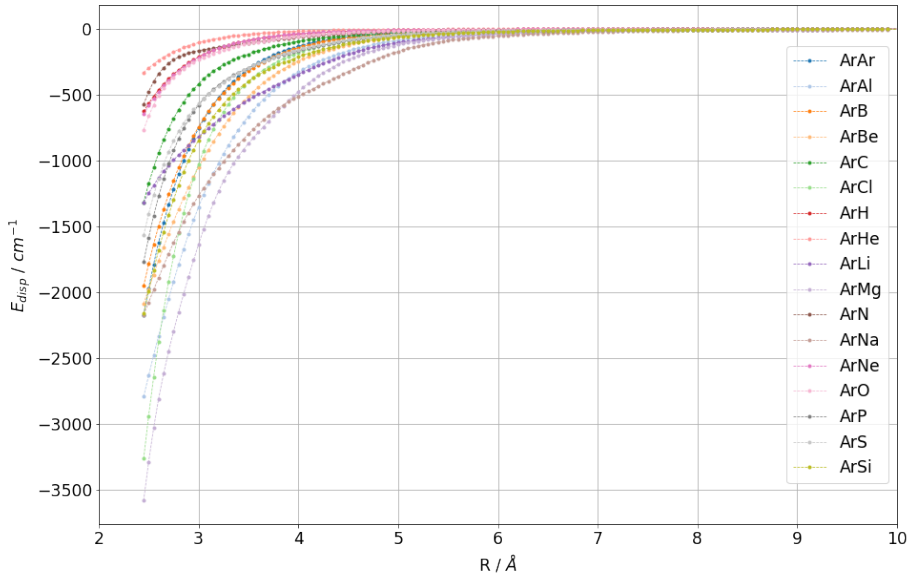

Figure 2.10: The simulated dispersion energies E_{disp} of all atomic pairs involving Ar over the distance. The x-axis is cropped at 10 Å for better visibility.

2.4 Quantization

When investigating the results of ab initio calculations described in section 2.1, a quantization can be observed, which is clearly visible, when looking at the logarithmized E_{disp} of some of the considered atomic pairs, shown in Figure 2.11.

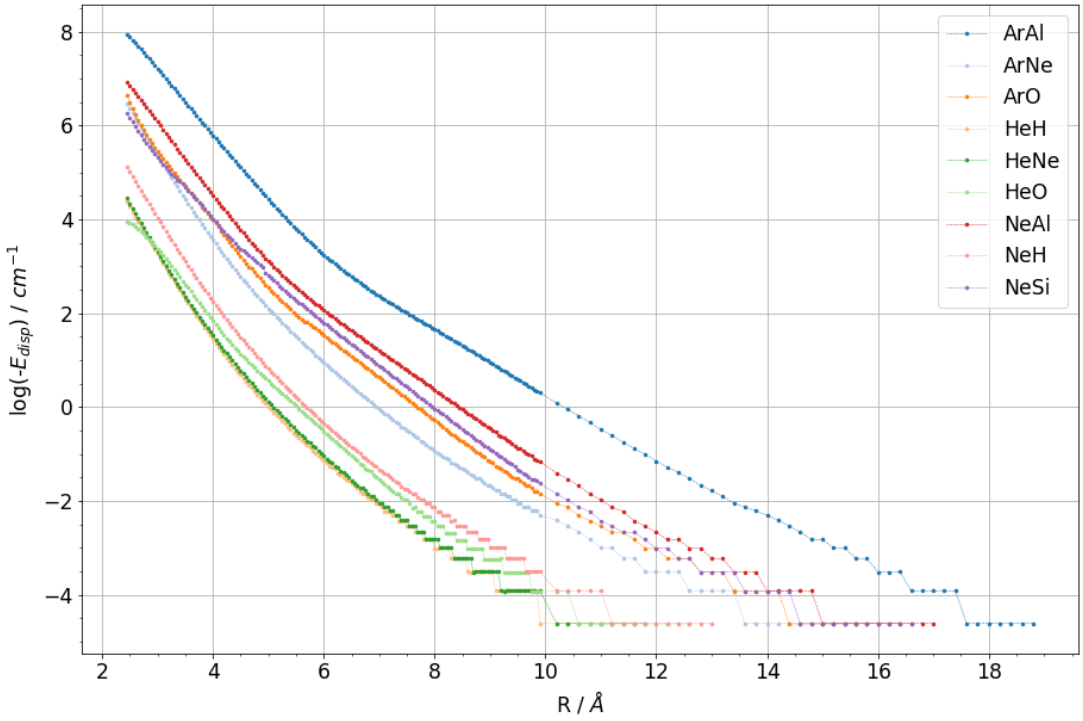

Figure 2.11: The simulated dispersion energies E_{disp} of some of the considered atomic pairs, showing a quantization starting at different distances, but with the same (increasing) stepsize.

The dispersion energies displayed in Figure 2.11 consist of the difference of Hartree-Fock and Coupled Cluster calculations. However, due to the energy threshold in Molpro chosen for efficiency reasons, E_{disp} can only be resolved up to the second decimal place in units of inverse centimeters. Steps of 0.01 cm^{-1} are visible, which can be seen more clearly in the logarithmized E_{disp} curves.

3 Application of Symbolic Regression

In this chapter the attempt of finding correlations proposed by hypotheses (see section 1.4) with Symbolic Regression, providing all necessary building blocks (e.g. variables). First the London formula, as stated in equation 1.24, is tested and additionally damping functions similar to the ones proposed in the London-dispersion corrections described in section 1.4.1 to 1.4.3 are identified.

As Symbolic Regression Tool the Tensorflow implementation of the Equation Learner (EQL) is used, which is an open source software prototype with fully available code at github (Equation Learner). The modified code of the EQL used in this master thesis, including training procedure explained in section 1.5.1, is available at: github (Equation Learner - ETC). Python version 3.6.10, tensorflow version 1.14.0, numpy version 1.18.5 and sympy version 1.6.2 are used throughout this thesis.

3.1 Training data

Before beginning to search for connections, proposed by hypothesis in section 1.4, in the dispersion energy curves, a further step in data cleaning is done by setting $E_{disp} = 0$ as soon as $E_{disp} \geq -0.1\ cm^{-1}$. When E_{disp} is reaching this fluctuations and quantization effects occur (see Figure 3.1), supposedly caused by the accuracy limits of the used ab initio calculation for calculating the total energy (CCSDT). Since the London-dispersion is increasing monotonously until reaching 0 with increasing distance, all values obtained at larger distance are considered to be numerical noise and do not represent a connection that our machine learning algorithms are supposed to learn. After this last data cleaning step, a training data set of 6347 values remains.

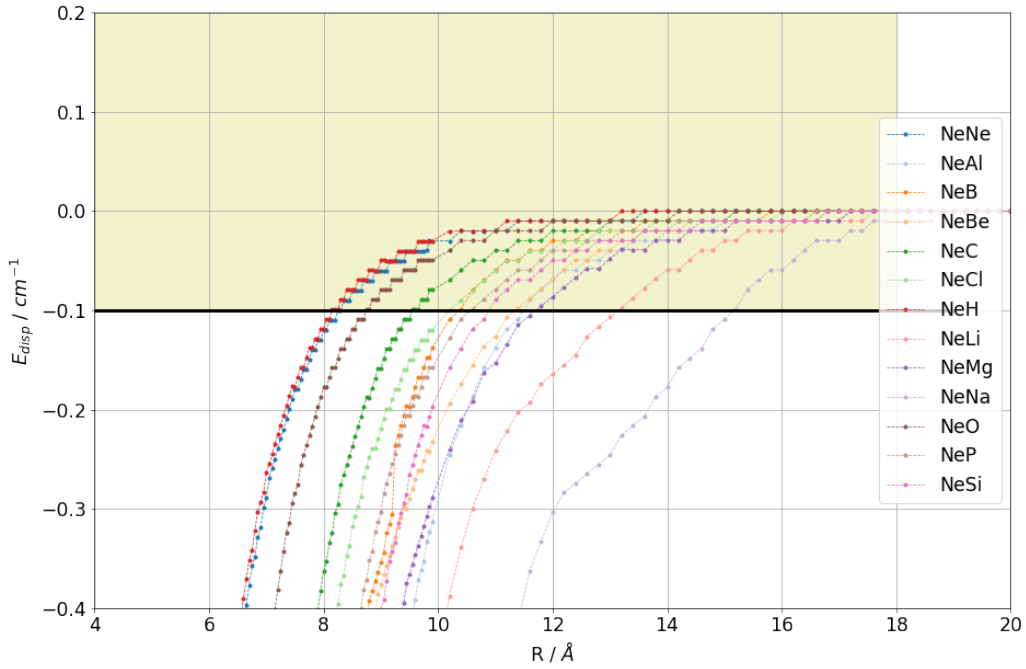

Figure 3.1: The simulated dispersion energies E_{disp} of a selection of all considered atomic pairs involving neon over the distance. The displayed E_{disp} shows fluctuations in the curves and quantization as soon as $E_{disp} \geq -0.1\ cm^{-1}$. This value is marked with a black line in the figure and all values at larger distance are no longer considered as training data.

3.2 Network architecture

In the following the decisions influencing the architecture of the neural network in the EQL in the experiments done in this chapter are described.

3.2.1 Input variables

In order to seek for the proposed hypotheses of the London formula (equation 1.24) and damping functions as proposed in e.g. equation 1.26, or equation 1.44, the same input parameters as used in the respected hypotheses are needed. For the acquisition of the atomic ionization potential IP, the static dipole polarizability α and the sum of the van der Waals radii R_{vdw} of the respected atomic pairs the mendeleev package is used [27], which is extracting this information from literature (Ref. [21], Ref. [34] and Ref. [16]). For predicting analytical expressions similar to the London formula, R^6, the distance between the atomic pair to the power of 6, the IP and α is needed. The investigation of the damping function demands the introduction of R and R_{vdw}.

3.2.2 Network parameters

The EQL is used with network parameters as proposed from Martius et al. in Ref. [30] and included in the default settings of the EQL tensorflow implementation, including the following:

- batch size = 20: Number of samples that are fed to the network at one iteration.
- layer width = 10: Number of identical nodes of each unit per hidden layer.

- train val split = 0.9: Percentage of data used for training of the neural network (here 90 %), with the rest used as validation data (here 10 %) to verify the training progress.
- l0 threshold = 0.05: Threshold value for the regularization process. All weights smaller than this value are set to zero at the end of the regularization process.

3.2.3 Unitary and binary units

Besides the respected input parameters described in section 3.2.1 the functions used to combine them and resulting in an analytical expression predicting the respected output are needed to be defined as well. These functions are handed over to the EQL with unitary and binary units as explained in section 1.5.1. For the London formula investigation only multiplication, subtraction, division and identity are needed, but the damping function investigation needs exponential functions as well.

3.3 Model selection and training procedure

In the original proposed model selection procedure (Ref. [26]), the absolut regularization strength λ and the number of hidden layers L are varied to find the parameter pair creating the best result, which is considered to be the one with the lowest score, according to equation 3.1

$$\text{score} = \sqrt{(\mathcal{L}_{norm}^{validation})^2 + (C_{norm})^2} \tag{3.1}$$

with $\mathcal{L}_{norm}^{validation}$ as the validation loss normed by the respective maximum and C_{norm} as the network complexity normed by the respective maximum.
In the modifications described in section 1.5.1, λ is replaced by $\lambda_{\%}$, as can be sees in Equation 1.50. Therefore in the new proposed model selection $\lambda_{\%}$ and L are variied.
As the searched formulas are considered to be analytical simple and EQL runs with $L > 1$ are resulting in a diverging of the neural network during training, $L = 1$ is set for all runs in this investigation. Furthermore multiple runs are done with each parameter, to account for the random initialization of the weights and possible diverging of the neural network during training resulting in incomplete data (formula expression, validation error $L^{validation}$ and network complexity C), which reason is not further investigated in this master thesis.
The regularization, controlled by $\lambda_{\%}$, is influencing C by lowering the weights and set them to zero, when they are lower than a given threshold value (e.g. 0.05), which is desired, and $\mathcal{L}^{validation}$, which is not desired. If $\lambda_{\%}$ is chosen too low, C is not lowered and it has no effect on the resulting formula expression. If it is too high, all weights are set to zero. Additionally, $\mathcal{L}^{validation}$ is influenced by the regularization at all times depending on the respected strength and this influence should be as low as possible. In order to find the best fitting $\lambda_{\%}$ for the given problem, resulting in runs with the lowest score, a two-step model selection is proposed. First $\lambda_{\%}$ is varied over six orders of magnitude and the two best are identified. Then, a second finer variation around the two found $\lambda_{\%}$ is done. The three best values are identified and ten independent runs with these suggested $\lambda_{\%}$ are performed, resulting in 30 individual runs, which are further investigated. In each run a split of training and validation data is done (here 90 % is used as training data) and the weights of the network are initialized randomly.

3.4 Results

In the following, the results of two investigations are presented. First an attempt to find correlations of the distance between atomic pairs, the atomic ionization potential and the static dipole polarizability of the involved atoms, similar to the famous London formula stated in equation 1.24, is done. Then a damping function depending on the distance of the atomic pair and the sum of the van der Waals radii of the involved atoms, similar to e.g. equation 1.26, is searched. The model selection is done as described in section 3.3 and to quantify the performance of a run, the score is used as defined in equation 3.1.

3.4.1 London formula

The results of the first model selection are summarized in Table 3.1.

id	$\lambda_{\%}$	$\mathcal{L}^{validation}$	$\mathcal{L}^{validation}_{norm}$	**C**	$\mathbf{C}_{norm}$	**score**
0	0.00001	0.000234	0.413832	10	1.0	1.082246
1	0.00001	0.000316	0.558336	9	0.9	1.059122
2	0.00001	0.000228	0.403014	10	1.0	1.078156
mean						1.073175
3	0.00010	0.000202	0.356302	8	0.8	0.875758
4	0.00010	0.000185	0.327615	9	0.9	0.957774
5	0.00010	0.000314	0.555215	8	0.8	0.973789
mean						0.935773
6	0.00100	0.000211	0.373511	5	0.5	0.624108
7	0.00100	0.000195	0.343669	6	0.6	0.691454
8	0.00100	0.000213	0.377051	6	0.6	0.708638
mean						0.674733
9	0.01000	0.000194	0.342561	3	0.3	0.455355
10	0.01000	0.000185	0.327339	7	0.7	0.772756
11	0.01000	0.000235	0.415844	2	0.2	0.461440
mean						0.563183
12	0.10000	0.000555	0.980512	1	0.1	0.985598
13	0.10000	0.000566	1.000000	1	0.1	1.004988
14	0.10000	0.000527	0.930640	4	0.4	1.012961
mean						1.001182
15	1.00000	0.000535	0.944418	4	0.4	1.025634
16	1.00000	0.000546	0.964181	4	0.4	1.043861
17	1.00000	0.000563	0.994423	1	0.1	0.999438
mean						1.022978

Table 3.1: Results of the first model selection of the London formula investigation with Symbolic Regression. Three individual runs are done with each regularization percentage and the mean of the score of those three is calculated. $\lambda_{\%}$ ... regularization percentage, $\mathcal{L}^{validation}$ ... validation error, C ... network complexity

The lowest resulting mean score over three individual runs in the first model selection are at $\lambda_{\%} = 0.001$ and 0.01. Therefore $\lambda_{\%}$ is varied around those values in the second model selection, which is summarized in Table 3.2.

id	$\lambda_{\%}$	$\mathcal{L}^{validation}$	$L_{norm}^{validation}$	**C**	$\mathbf{C}_{norm}$	**score**
0	0.00050	0.000270	0.473336	6.0	0.666667	0.817613
1	0.00050	0.000199	0.349070	8.0	0.888889	0.954973
2	0.00050	0.000196	0.343867	4.0	0.444444	0.561939
mean						0.778175
3	0.00075	0.000206	0.361412	5.0	0.555556	0.662767
4	0.00075	0.000214	0.375449	4.0	0.444444	0.581802
5	0.00075	0.000237	0.414720	9.0	1.000000	1.082586
mean						0.775718
6	0.00100	0.000252	0.441803	6.0	0.666667	0.799772
7	0.00100	0.000214	0.374439	6.0	0.666667	0.764623
8	0.00100	-	-	-	-	-
mean						0.782197
9	0.00250	0.000208	0.364490	2.0	0.222222	0.426891
10	0.00250	0.000260	0.454454	9.0	1.000000	1.098421
11	0.00250	0.000200	0.350585	7.0	0.777778	0.853140
mean						0.792817
12	0.00500	0.000238	0.415843	5.0	0.555556	0.693950
13	0.00500	0.000267	0.466828	5.0	0.555556	0.725651
14	0.00500	0.000208	0.364735	4.0	0.444444	0.574945
mean						0.664849
15	0.00750	0.000271	0.474311	2.0	0.222222	0.523788
16	0.00750	0.000174	0.304692	9.0	1.000000	1.045388
17	0.00750	0.000217	0.379866	2.0	0.222222	0.440092
mean						0.669756
18	0.01000	0.000369	0.646772	5.0	0.555556	0.852617
19	0.01000	0.000251	0.439121	3.0	0.333333	0.551306
20	0.01000	0.000301	0.527696	5.0	0.555556	0.766228
mean						0.723384
21	0.02500	0.000550	0.963639	0.0	0.000000	0.963639
22	0.02500	0.000571	1.000000	2.0	0.222222	1.024394
23	0.02500	0.000209	0.366231	2.0	0.222222	0.428378
mean						0.80547
24	0.05000	0.000547	0.958388	0.0	0.000000	0.958388
25	0.05000	0.000204	0.356608	2.0	0.222222	0.420181
26	0.05000	0.000347	0.607681	5.0	0.555556	0.823358
mean						0.733975

Table 3.2: Results of the second model selection of the London formula investigation with Symbolic Regression. Three individual runs are done with each regularization percentage and the mean of the score of those three is calculated. $\lambda_{\%}$ ... regularization percentage,$\lambda_{\%}$ ... regularization percentage, $\mathcal{L}^{validation}$ ... validation error, C ... network complexity

The three lowest mean scores in the second model selection, according to Table 3.2, are with $\lambda_{\%} = 0.005$, 0.0075 and 0.01. In consequence those three values are used in the final runs of this investigation summarized in Table 3.3 and shown in Figure 3.2.

id	$\lambda_{\%}$	$\mathcal{L}^{validation}$	$\mathcal{L}^{validation}_{norm}$	C	C_{norm}	*score*
0	0.0050	0.000303	0.549877	3.0	0.428571	0.697165
1	0.0050	-	-	-	-	-
2	0.0050	0.000532	0.964551	3.0	0.428571	1.055477
3	0.0050	0.000307	0.556068	3.0	0.428571	0.702058
4	0.0050	0.000220	0.398467	4.0	0.571429	0.696640
5	0.0050	0.000207	0.374536	3.0	0.428571	0.569167
6	0.0050	0.000215	0.389677	5.0	0.714286	0.813666
7	0.0050	0.000225	0.408268	4.0	0.571429	0.702291
8	0.0050	-	-	-	-	-
9	0.0050	0.000205	0.370952	2.0	0.285714	0.468229
10	0.0075	0.000214	0.387252	2.0	0.285714	0.481245
11	0.0075	0.000205	0.371484	7.0	1.000000	1.066771
12	0.0075	-	-	-	-	-
13	0.0075	-	-	-	-	-
14	0.0075	0.000247	0.447006	6.0	0.857143	0.966699
15	0.0075	0.000191	0.346153	5.0	0.714286	0.793742
16	0.0075	0.000201	0.365037	2.0	0.285714	0.463557
17	0.0075	0.000551	1.000000	2.0	0.285714	1.040016
18	0.0075	0.000198	0.359224	4.0	0.571429	0.674961
19	0.0075	0.000224	0.405984	2.0	0.285714	0.496443
20	0.0100	0.000112	0.202610	6.0	0.857143	0.880764
21	0.0100	0.000217	0.392658	3.0	0.428571	0.581252
22	0.0100	-	-	-	-	-
23	0.0100	0.000216	0.390942	2.0	0.285714	0.484219
24	0.0100	0.000185	0.335657	5.0	0.714286	0.789221
25	0.0100	0.000239	0.433833	5.0	0.714286	0.835713
26	0.0100	0.000231	0.418433	2.0	0.285714	0.506674
27	0.0100	0.000230	0.417325	2.0	0.285714	0.505760
28	0.0100	0.000202	0.366598	7.0	1.000000	1.065079
29	0.0100	0.000229	0.414962	2.0	0.285714	0.503812

Table 3.3: Results of the London formula investigation with Symbolic Regression, performing ten individual runs with each choice of regularization percentage. $\lambda_{\%}$ ... regularization percentage, $\mathcal{L}^{validation}$ ... validation error, C ... network complexity

The runs with the lowest five scores are resulting in five formulas that are subject of further investigation (id = 9, 10, 16, 19 and 23). The runs with id = 10 and 9 as well as with id = 19 and 23 are providing analytical expressions with the same structure, only differing in their weights. Therefore the formulas resulting from the runs with the lower score are chosen (id = 9 and 23). All those formulas can be seen in equation 3.2 (id = 9), 3.3 (id = 23) and 3.4 (id = 19). Additionally the formula resulting from the run with id = 20 is inspected more closely, as it results in the lowest validation error, and simplified by using the sympy package for better representation [28], leading to equation 3.5. Note that these expressions are not actual energy terms, but rather show a proportionality of atom-specific features to the dispersion energy.

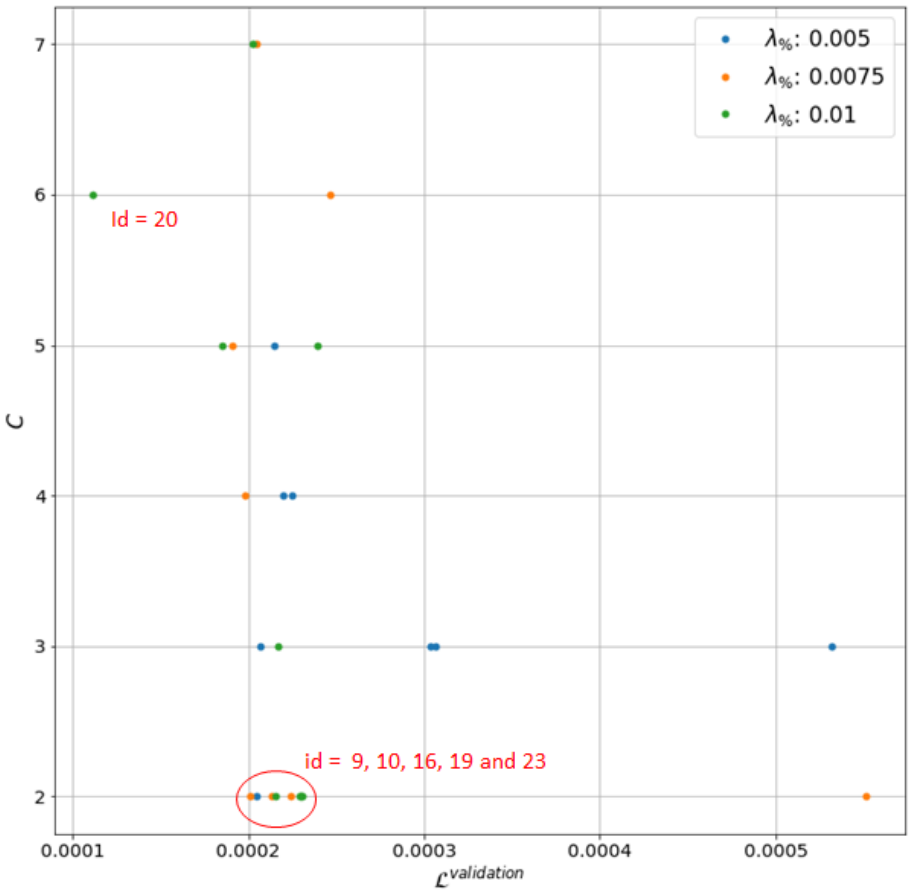

Figure 3.2: Network complexity C over validation error $\mathcal{L}^{validation}$ for the second model selection of the London formula investigation with Symbolic Regression. For each regularization percentage $\lambda_{\%}$, ten individual runs are done and the runs which are subject of further investigation are marked.

$$E_{disp}^{M1} = \frac{-1.337(6.16\alpha^A + 0.192\alpha^B)}{R^6} \propto E_{disp} \tag{3.2}$$

$$E_{disp}^{M2} = \frac{-0.54\alpha^B}{R^6} \propto E_{disp} \tag{3.3}$$

$$E_{disp}^{M3} = \frac{-0.168\alpha^B}{R^6 + 39.126\alpha^B} \propto E_{disp} \tag{3.4}$$

$$E_{disp}^{M4} = \frac{-3.961\alpha^A - 0.372\alpha^B + 7.519}{0.473R^6 + 26.276908\alpha^A + 39.162832\alpha^B + 24.584} \propto E_{disp} \tag{3.5}$$

3.4.2 Damping Functions

The results of the first model selection in the damping function investigation are summarized in Table 3.4.

id	$\lambda_{\%}$	$\mathcal{L}^{validation}$	$\mathcal{L}^{validation}_{norm}$	**C**	$\mathbf{C}_{norm}$	**score**
0	0.00001	0.327528	0.777272	14	0.8750	1.170374
1	0.00001	0.308614	0.732385	14	0.8750	1.141058
2	0.00001	0.308998	0.733298	11	0.6875	1.005178
mean						1.105537
3	0.00010	0.321801	0.763680	14	0.8750	1.161392
4	0.00010	0.306223	0.726711	16	1.0000	1.236167
5	0.00010	0.338434	0.803152	16	1.0000	1.282596
mean						1.226718
6	0.00100	0.332008	0.787902	13	0.8125	1.131789
7	0.00100	0.318094	0.754883	15	0.9375	1.203642
8	0.00100	0.306150	0.726539	16	1.0000	1.236066
mean						1.190499
9	0.01000	0.302698	0.718345	10	0.6250	0.952179
10	0.01000	0.317810	0.754209	10	0.6250	0.979518
11	0.01000	0.353423	0.838724	4	0.2500	0.875190
mean						0.935629
12	0.10000	0.312689	0.742057	4	0.2500	0.783038
13	0.10000	0.343272	0.814634	7	0.4375	0.924681
14	0.10000	0.314179	0.745593	7	0.4375	0.864474
mean						0.857397
15	1.00000	0.421382	1.000000	3	0.1875	1.017426
16	1.00000	0.419827	0.996310	2	0.1250	1.004120
17	1.00000	0.416983	0.989560	4	0.2500	1.020651
mean						1.014066

Table 3.4: Results of the first model selection of the damping function investigation with Symbolic Regression. Three individual runs are done with each regularization percentage and the mean of the score of those three is calculated. $\lambda_{\%}$ . . . regularization percentage, $\mathcal{L}^{validation}$ . . . validation error, C . . . network complexity

The lowest resulting mean score over three individual in the first model selection runs are at $\lambda_{\%} = 0.01$ and 0.1 and therefore $\lambda_{\%}$ is varied around those values in the second model selection, which results are summarized in Table 3.5.

id	$\lambda_{\%}$	$\mathcal{L}^{validation}$	$\mathcal{L}^{validation}_{norm}$	**C**	$\mathbf{C}_{norm}$	**score**
0	0.0050	0.325266	0.761678	10.0	0.588235	0.962380
1	0.0050	0.304184	0.712312	12.0	0.705882	1.002825
2	0.0050	0.297897	0.697589	9.0	0.529412	0.875732
mean						0.946979
3	0.0075	0.318777	0.746485	9.0	0.529412	0.915159
4	0.0075	0.305783	0.716055	9.0	0.529412	0.890512
5	0.0075	0.343462	0.804290	7.0	0.411765	0.903567
mean						0.903079
6	0.0100	0.313037	0.733042	17.0	1.000000	1.239900
7	0.0100	0.310185	0.726364	7.0	0.411765	0.834958
8	0.0100	0.374089	0.876009	2.0	0.117647	0.883873
						0.986244
9	0.0250	0.314174	0.735705	6.0	0.352941	0.815984
10	0.0250	-	-	-	-	-
11	0.0250	0.308789	0.723094	13.0	0.764706	1.052445
mean						0.934214
12	0.0500	0.309035	0.723671	7.0	0.411765	0.832617
13	0.0500	0.414124	0.969760	1.0	0.058824	0.971543
14	0.0500	0.313306	0.733672	10.0	0.588235	0.940370
mean						0.914843
15	0.0750	0.405414	0.949364	2.0	0.117647	0.956626
16	0.0750	0.323913	0.758510	7.0	0.411765	0.863069
17	0.0750	0.317339	0.743117	3.0	0.176471	0.763783
mean						0.861159
18	0.1000	-	-	-	-	-
19	0.1000	0.312834	0.732567	4.0	0.235294	0.769427
20	0.1000	0.321941	0.753894	6.0	0.352941	0.832421
mean						0.800924
21	0.2500	0.311862	0.730290	5.0	0.294118	0.787292
22	0.2500	0.391258	0.916214	6.0	0.352941	0.981842
23	0.2500	0.305094	0.714441	4.0	0.235294	0.752190
mean						0.840442
24	0.5000	0.341541	0.799791	5.0	0.294118	0.852156
25	0.5000	0.326700	0.765036	4.0	0.235294	0.800402
26	0.5000	0.427038	1.000000	2.0	0.117647	1.006897
mean						0.886485

Table 3.5: Results of the second model selection of the damping function investigation with Symbolic Regression. Three individual runs are done with each regularization percentage and the mean of the score of those three is calculated. $\lambda_{\%}$ ... regularization percentage, $\mathcal{L}^{validation}$ ... validation error, C ... network complexity

The three lowest mean scores in the second model selection, according to Table 3.2, are with $\lambda_{\%} = 0.075$, 0.1 and 0.25. In consequence, those three values are used in the final runs of this investigation summarized in Table 3.6 and shown in Figure 3.3.

id	$\lambda_{\%}$	$\mathcal{L}^{validation}$	$\mathcal{L}^{validation}_{norm}$	**C**	$\mathbf{C}_{norm}$	**score**
0	0.075	0.318716	0.758083	5.0	0.625	0.982505
1	0.075	0.312987	0.744459	6.0	0.750	1.056749
2	0.075	0.332993	0.792043	4.0	0.500	0.936660
3	0.075	0.318643	0.757912	3.0	0.375	0.845609
4	0.075	0.307957	0.732494	6.0	0.750	1.048354
5	0.075	0.297532	0.707697	5.0	0.625	0.944171
6	0.075	0.318590	0.757784	7.0	0.875	1.157524
7	0.075	-	-	-	-	-
8	0.075	0.312540	0.743395	8.0	1.000	1.246048
9	0.075	0.335172	0.797226	5.0	0.625	1.013012
10	0.100	0.337424	0.802583	5.0	0.625	1.017234
11	0.100	0.392480	0.933536	3.0	0.375	1.006039
12	0.100	0.399320	0.949806	3.0	0.375	1.021154
13	0.100	0.321778	0.765367	5.0	0.625	0.988136
14	0.100	0.369400	0.878640	3.0	0.375	0.955318
15	0.100	0.300912	0.715738	7.0	0.875	1.130445
16	0.100	0.314205	0.747355	7.0	0.875	1.150724
17	0.100	0.369199	0.878162	4.0	0.500	1.010529
18	0.100	-	-	-	-	-
19	0.100	0.319373	0.759648	5.0	0.625	0.983712
20	0.250	0.389672	0.926858	5.0	0.625	1.117896
21	0.250	0.365646	0.869710	4.0	0.500	1.003193
22	0.250	0.385373	0.916631	6.0	0.750	1.184362
23	0.250	0.332922	0.791873	7.0	0.875	1.180122
24	0.250	0.340025	0.808769	6.0	0.750	1.102999
25	0.250	0.315599	0.750670	8.0	1.000	1.250402
26	0.250	0.360408	0.857252	4.0	0.500	0.992412
27	0.250	0.343835	0.817831	6.0	0.750	1.109661
28	0.250	0.420423	1.000000	1.0	0.125	1.007782
29	0.250	-	-	-	-	-

Table 3.6: Results of the damping function investigation with Symbolic Regression, where ten individual runs with each regularization percentage are done. $\lambda_{\%}$ . . . regularization percentage, $\mathcal{L}^{validation}$ . . . validation error, C . . . network complexity

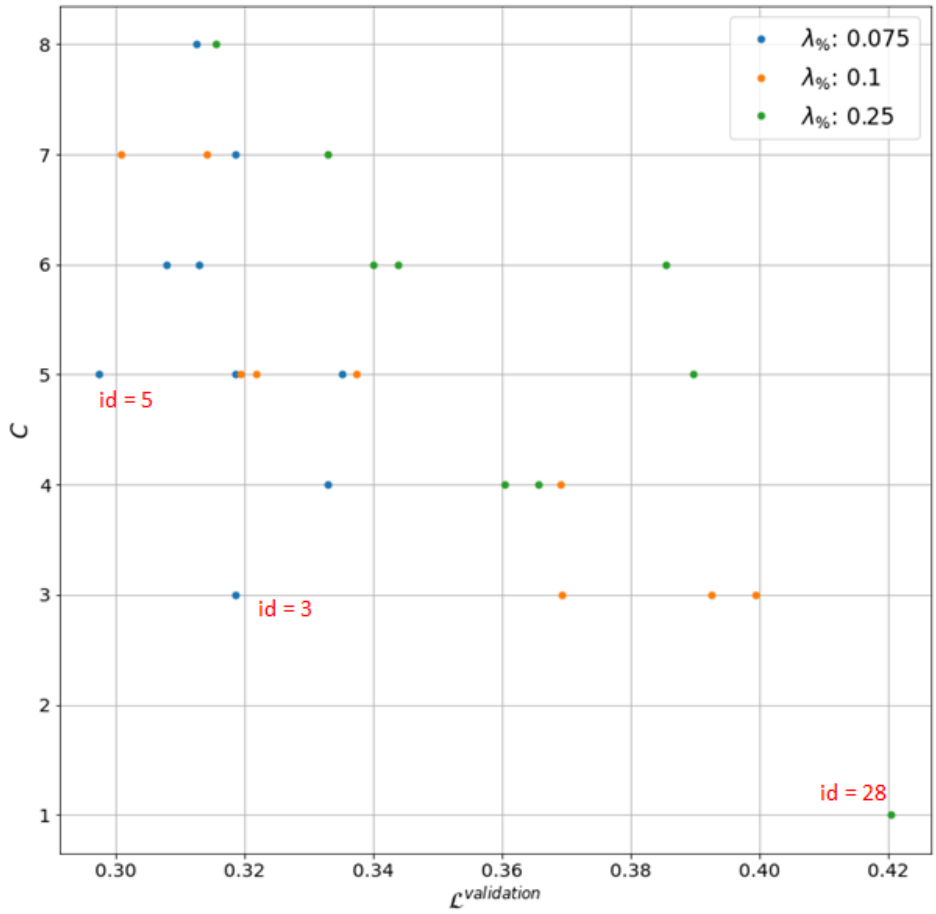

Figure 3.3: Network complexity C over validation error $\mathcal{L}^{\text{validation}}$ for the second model selection of the London formula investigation with Symbolic Regression. For each regularization percentage $\lambda_{\%}$, ten individual runs are done and the three runs which are subject of further investigation are marked.

The run with the lowest score (id=3) is resulting in the formula shown in equation 3.6. The runs with the lowest complexity (id=28, equation 3.8) and lowest validation error (id=5, equation 3.7) are also selected for further investigation. Equation 3.7 has been simplified again by using the sympy package [28].

$$f_{damp}^{M1} = \frac{0.269\ R^2 + 0.332 \exp(0.22\ R + 0.13\ R_{vdw})}{0.225\ R^2 + 0.796(0.568\ R + 0.741\ R_{vdw})\ (-0.542\ R + 0.705\ R_{vdw})} \tag{3.6}$$

$$f_{damp}^{M2} = (0.519\ R\ R_{vdw} - 0.48 \exp(-0.268\ R + 0.776\ R_{vdw}) + 5.57 \exp(0.136\ R + 0.155\ R_{vdw}) + 0.143 \exp(0.338\ R + 0.22\ R_{vdw})) \cdot \frac{1}{2.21 \exp(0.258\ R) + 3.586 \exp(0.136\ R + 0.155\ R_{vdw})} \tag{3.7}$$

$$f_{damp}^{M3} = 1.451 \tag{3.8}$$

4 Discussion

In this chapter the results shown in chapter 3 are discussed, starting with the investigation of the London formula and followed by a detailed analysis of damping functions for the London-dispersion corrections found in section 3.4.2.

4.1 London formula

In this investigation it was attempted to find correlation similar to the proposed London formula describing the London-dispersion interaction (E_{disp}), see equation 1.24, by providing meaningful variables (distance between two atoms, ionization energy and dipole polarization of the respected atoms) as an input and corresponding simulated dispersion energy as an output to a neural network that creates analytical expressions called Equation Learner (EQL).

In the resulting expressions in equation 3.2 to 3.5 the R^{-6} dependence of E_{disp} is found in all results and the ionization energy (IP) is not considered in the correlations in contrast to the dipole polarization (α), which suggests to a stronger α dependence of E_{disp} and a rather weak connection to the IP.

4.1.1 Performance

In order to quantitatively measure the quality of a model, the average difference between the simulated dispersion energy (E_{disp}) calculated in chapter 2 and E_{disp}^{model} calculated by models is given as

$$\Delta(E_{disp}, E_{disp}^{model}) = \frac{1}{N} \sum_{i=0}^{N} \left| E_{disp}(i) - E_{disp}^{model}(x_i) \right| \tag{4.1}$$

with N as the number of calculated values and x as the input parameters of the respected model.

$\Delta(E_{disp}, E_{disp}^{model})$ is shown in Figure 4.1 to 4.4 for all considered atomic pairs, grouped by the involved noble gas atom and ranked by the dipole polarization (α) of the second involved atom. A higher α seems to imply a worse approximation of E_{disp} in all models, which can be partly explained by larger relative amount of correlation energy involved. In the short range regime, where the distance is smaller than the sum of the van der Waals radii of the respected atoms, a higher $\Delta(E_{disp}, E_{disp}^{model})$ can be observed. This can be explained by the fact that none of the models is considering contributions from the R^{-8} term, as described in section 1.4.2 and other electronic correlation effects not considered in the Hartree-Fock theory, as mentioned in Ref. [24].

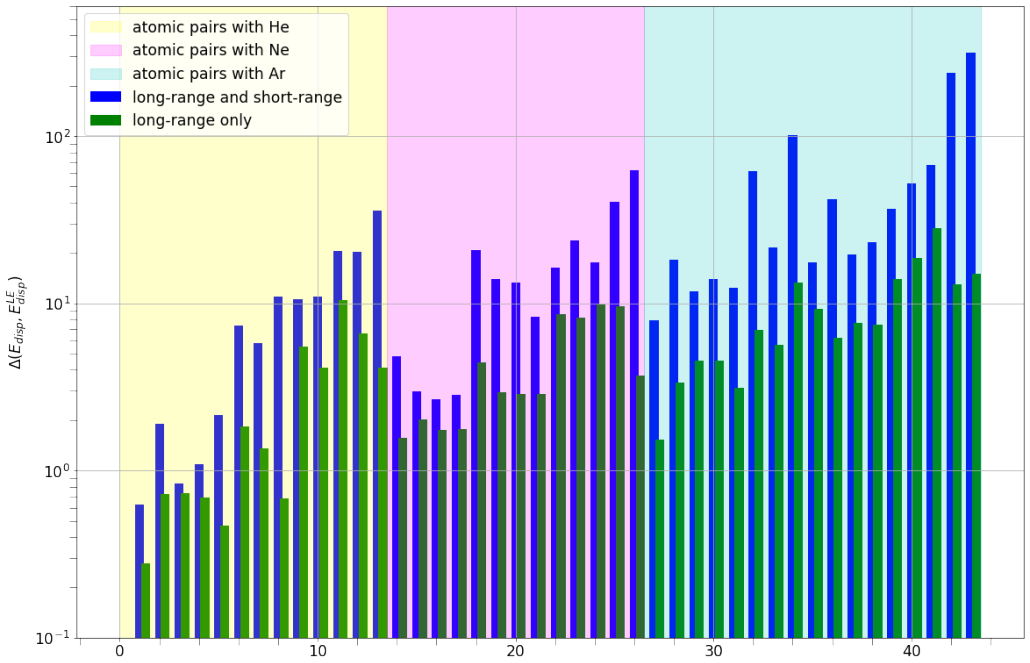

Figure 4.1: Barplot of the average difference between the ab initio dispersion energy (E_{disp}) and E_{disp} calculated by the London formula, see equation 1.24, according to equation 4.1, for all considered atomic pairs over the whole distance and only in the long range regime, where R is bigger than the combined van der Waals radii of the respected atoms.

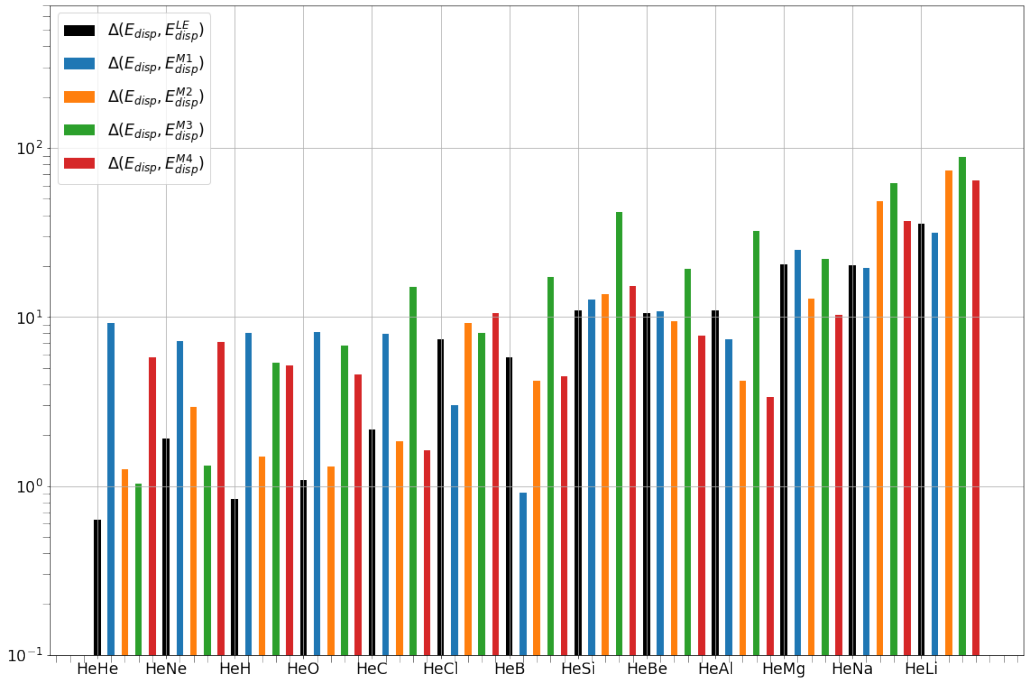

Figure 4.2: Barplot of the average difference between the ab initio dispersion energy (E_{disp}) and E_{disp} calculated by the London formula (LE) and other models found with Symbolic Regression (M1-4), according to equation 4.1, for all considered atomic pairs involving He.

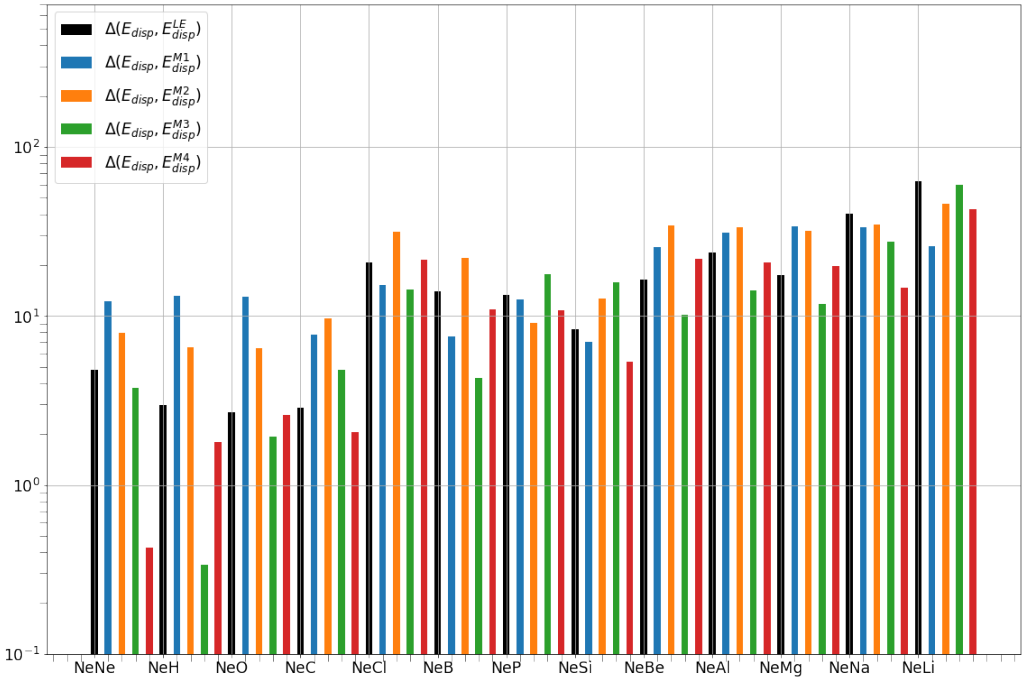

Figure 4.3: Barplot of the average difference between the ab initio dispersion energy (E_{disp}) and E_{disp} calculated by the London formula (LE) and other models found with Symbolic Regression (M1-4), according to equation 4.1, for all considered atomic pairs involving Ne.

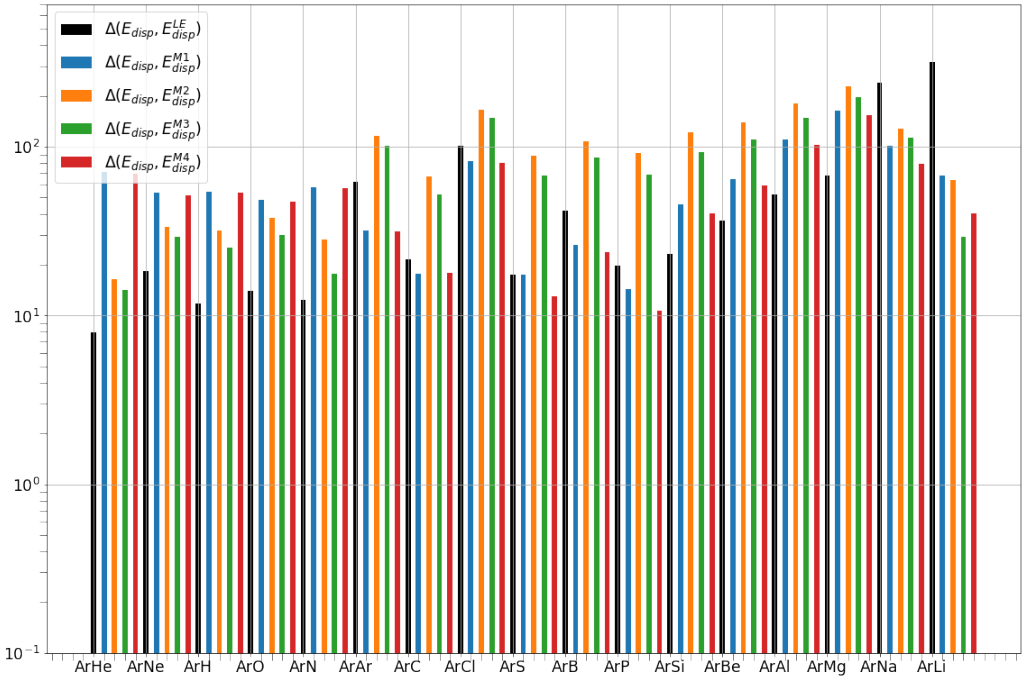

Figure 4.4: Barplot of the average difference between the ab initio dispersion energy (E_{disp}) and E_{disp} calculated by the London formula (LE) and other models found with Symbolic Regression (M1-4), according to Equation 4.1, for all considered atomic pairs involving Ar.

In the following Figures the dispersion energy of two atomic pairs calculated by various models are shown, which are approximating E_{disp} partly very good (Figure 4.5) and much worse (Figure 4.6), due to an overestimation of the contribution of the high dipole polarization of the involved atoms.

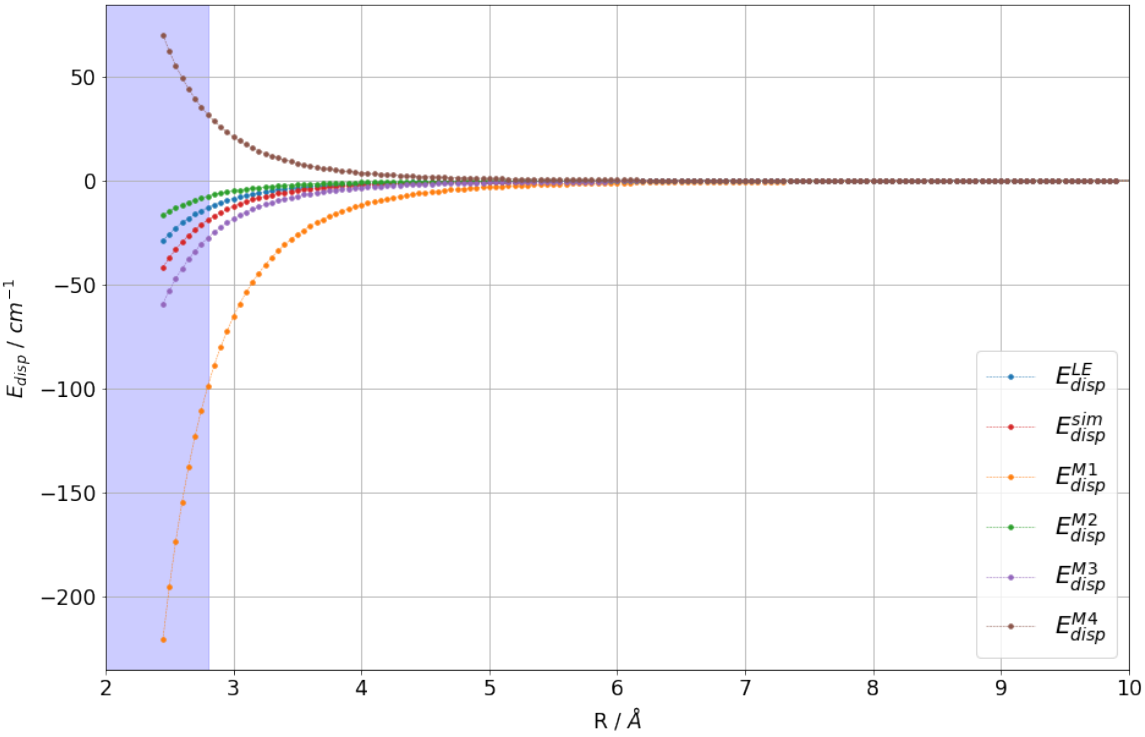

Figure 4.5: Comparison of the ab initio dispersion energy (E^{sim}_{disp}) with the calculated E_{disp} by the London formula (LE) and models found by Symbolic Regression (M1-4) for the helium dimer. M4 shows a non physical behavior at short distances due to the low dipole polarization of He and the factor of +7.519 in equation 3.5. The short range regime, where R is smaller than the combined van der Waals radii of the respected atoms, is indicated by the blue area and the x-axis is cropped at 10 Å for better visibility.

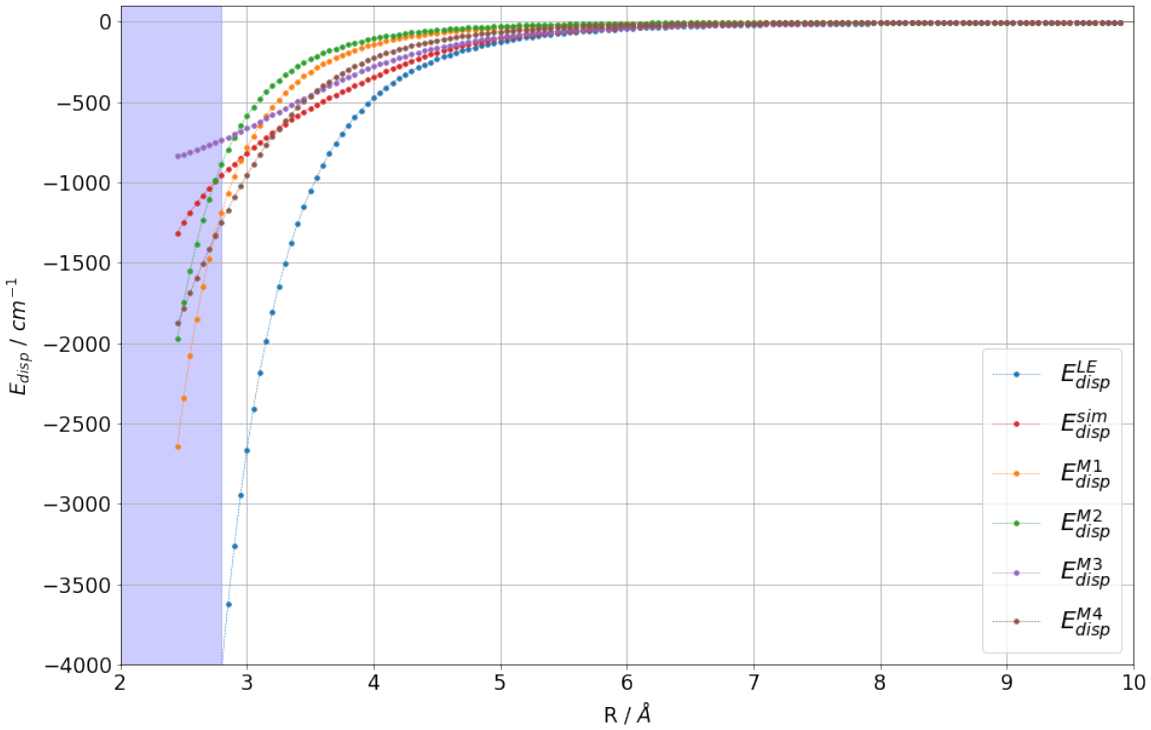

Figure 4.6: Comparison of the ab initio dispersion energy (E^{sim}_{disp}) with the calculated E_{disp} by the London formula (LE) and models found by Symbolic Regression (M1-4) for the atomic pair of argon and lithium. The contribution of the dipole polarization is especially overestimated in the London formula, where its calculated to a value of about $-8976\ cm^{-1}$ at a distance of $R = 2.45$ Å compared to $E^{sim}_{disp} \approx -1315\ cm^{-1}$ at the same distance. Note that the y axis is cropped at -4000 cm^{-1}, as well as the x-axis is cropped at 10 Å for better visibility. The short range regime, where R is smaller than the combined van der Waals radii of the respected atoms, is indicated by the blue area.

4.2 Damping functions

The idea of a damping function in the given context is to damp the contribution of E_{disp} in the short range regime. Therefore a function is multiplied to an approximated dispersion energy depending only on distance and sum of the van der Waals radii of two atoms, e.g. equation 1.26. In this investigation Symbolic Regression is used to find damping functions (f^{M}_{damp}) that correct the approximation of the dispersion energy by the London formula according to equation 1.24 (E^{LE}_{disp}), resulting in a better approximation (E^{DF}_{disp}).

$$E^{DF}_{disp} = E^{LE}_{disp} \cdot f^{M}_{damp} \tag{4.2}$$

The functions identified by the Equation Learner, see equation 3.6 to 3.8, are not only damping the approximation, but also scale it to compensate terms not considered by the London approximation. This can be seen clearly in f^{M3}_{damp}, which is only a scaling factor. The course of the functions are displayed in Figure 4.7 and show a steep rise at greater distances, where the dispersion energy is very low, therefore the impact is not significant and less data was used for training the neural network of the EQL.

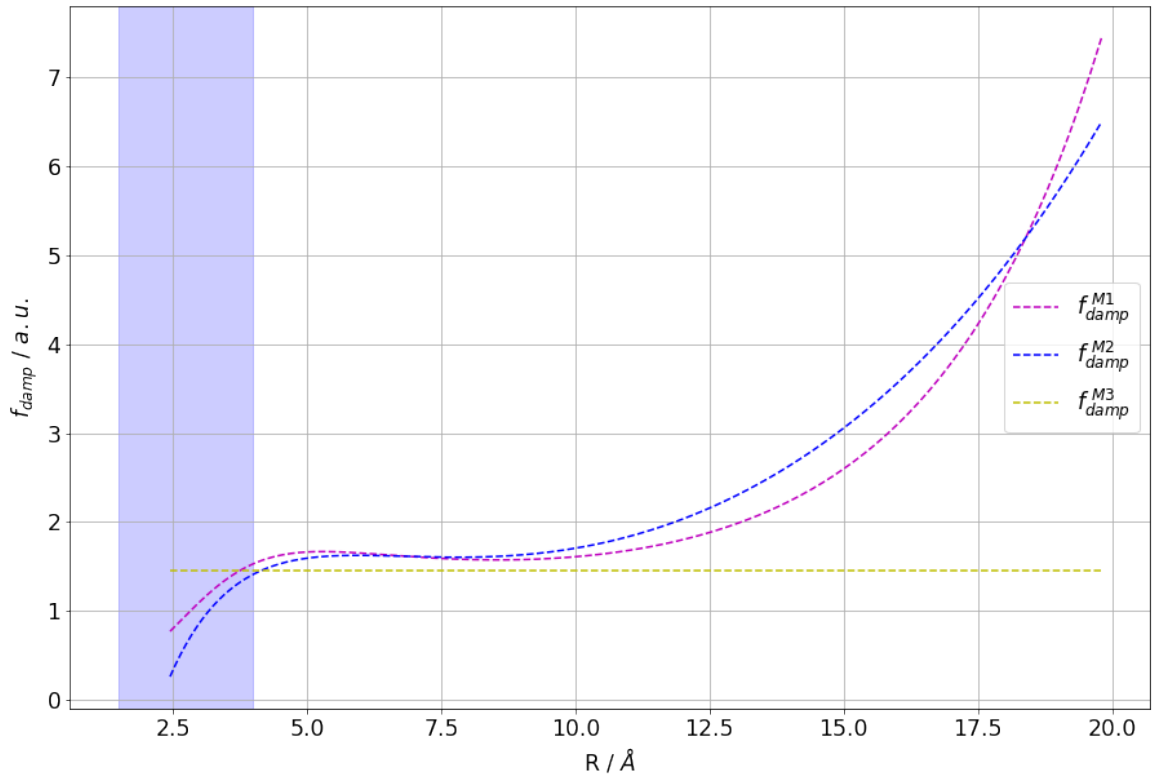

Figure 4.7: Functions found in the investigation in section 3.4.2 for the atomic pair of Argon and Silicon in atomic units over the distance. The short range regime, where R is smaller than the combined van der Waals radii of the respected atoms, is shaded blue.

4.2.1 Performance

The quantitative performance of the found damping functions multiplied with the dispersion energy calculated by the London formula (E_{disp}^{LE}) are compared to the deviation of the approximation by the London formula alone over all considered atomic pairs in Figure 4.8 to 4.10. The results shown are grouped by the involved noble gas atom and ranked by the dipole polarization (α) of the second involved atom.

In Figure 4.11 and 4.12 two examples (HeHe and ArLi) for the approximation of E_{disp} of an atomic pair by the found damping functions multiplied with the London formula are shown. While the correction of the London formula by the the found f_{damp}^{M} works very good in the example of the Helium dimer, E_{disp} is approximating the simulated energy worse than without those corrections in the atomic pair of Argon and Lithium.

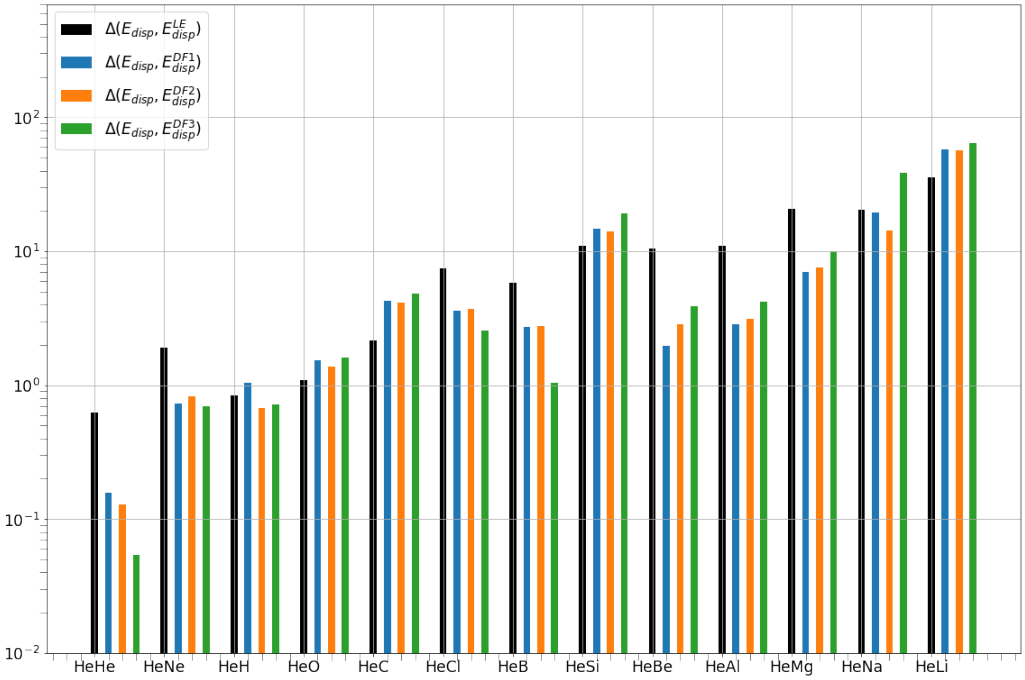

Figure 4.8: Barplot of the average difference between the simulated dispersion energy (E_{disp}) and E_{disp} calculated by the London formula (LE) and other models (DF1-4), according to equation 4.1, for all considered atomic pairs involving He.

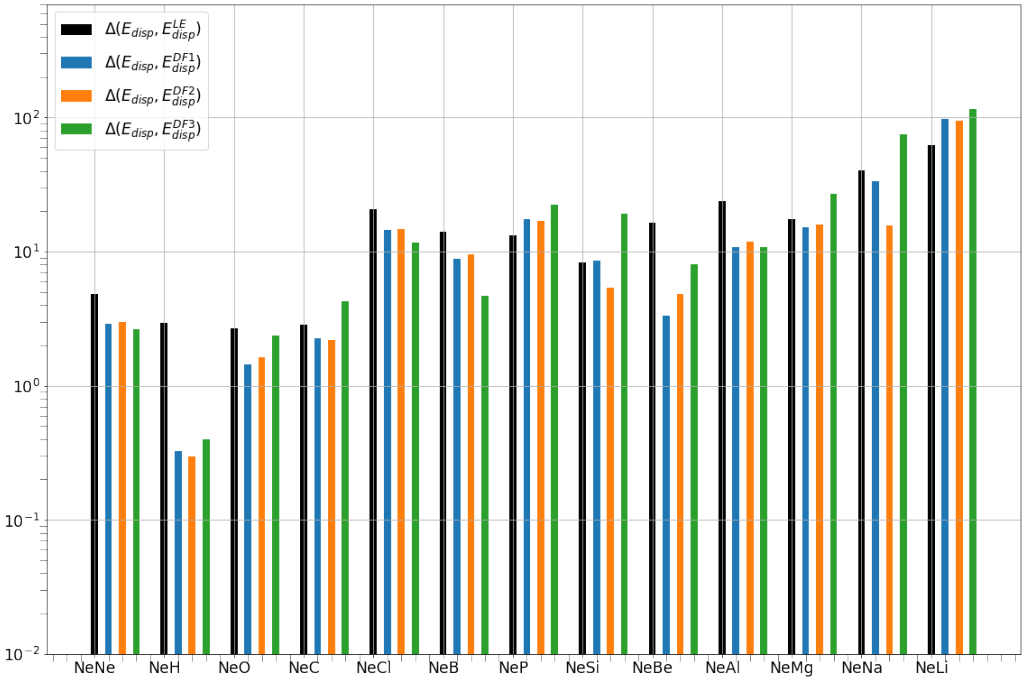

Figure 4.9: Barplot of the average difference between the simulated dispersion energy (E_{disp}) and E_{disp} calculated by the London formula (LE) and other models (DF1-4), according to equation 4.1, for all considered atomic pairs involving Ne.

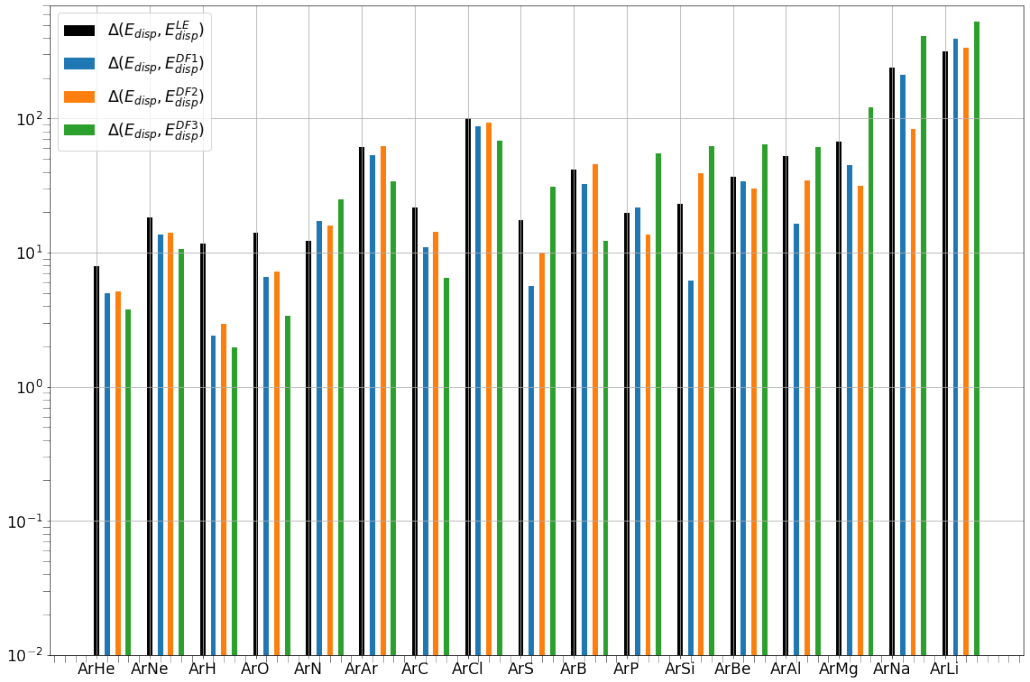

Figure 4.10: Barplot of the average difference between the simulated dispersion energy (E_{disp}) and E_{disp} calculated by the London formula (LE) and other models (DF1-4), according to equation 4.1, for all considered atomic pairs involving Ar.

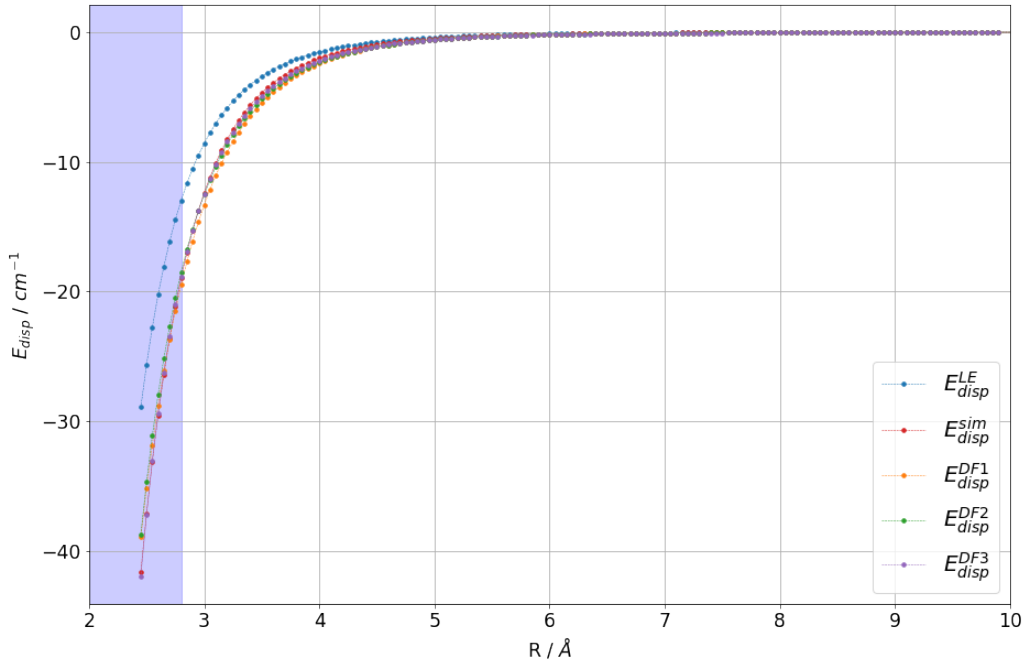

Figure 4.11: Comparison of the simulated dispersion energy (E_{disp}^{sim}) with the calculated E_{disp} by the London formula (LE) and multiplied with damping functions found by Symbolic Regression (DF1-3) for the Helium dimer, which shows a very good approximation. The short range regime, where R is smaller than the combined van der Waals radii of the respected atoms, is shaded blue and the x-axis is cropped at 10 Å for better visibility.

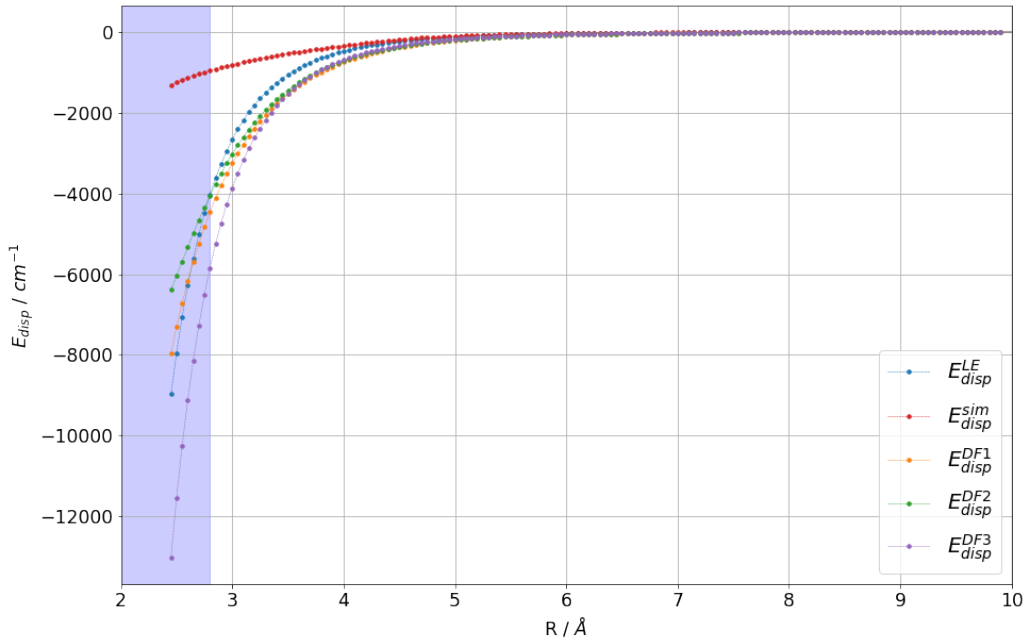

Figure 4.12: Comparison of the ab initio dispersion energy (E^{sim}_{disp}) with the calculated E_{disp} by the London formula (LE) and multiplied with damping functions found by Symbolic Regression (DF1-3) for the atomic pair of Argon and Lithium. The short range regime, where R is smaller than the combined van der Waals radii of the respected atoms, is shaded blue and the x-axis is cropped at 10 Å for better visibility.

4.2.2 Optimization

In this section Equation 3.6 is investigated in more detail by changing the prefactors to fitting parameters $(a_0, a_1, \ldots, a_9)$, as can be seen in Equation 4.3. Now an optimization is done using the scipy package [41] with the Powell algorithm [29]. On the one hand, the parameters are optimized for the whole considered data set, which consists of 43 atomic pairs, on the other hand they are optimized for every atomic pair individually. The difference of the calculated E_{disp} from the simulated one is shown in Figure 4.13.

$$f_{damp}^{M1} = \frac{a_0\ R^2 + a_1 \exp(a_2\ R + a_3\ R_{vdw})}{a_4\ R^2 + a_5(a_6\ R + a_7\ R_{vdw})\ (a_8\ R + a_9\ R_{vdw})} \tag{4.3}$$

With R as the as the distance between two atomes (from center to center) and R_{vdw} as the sum of the van der Waals radii of the respected atoms.

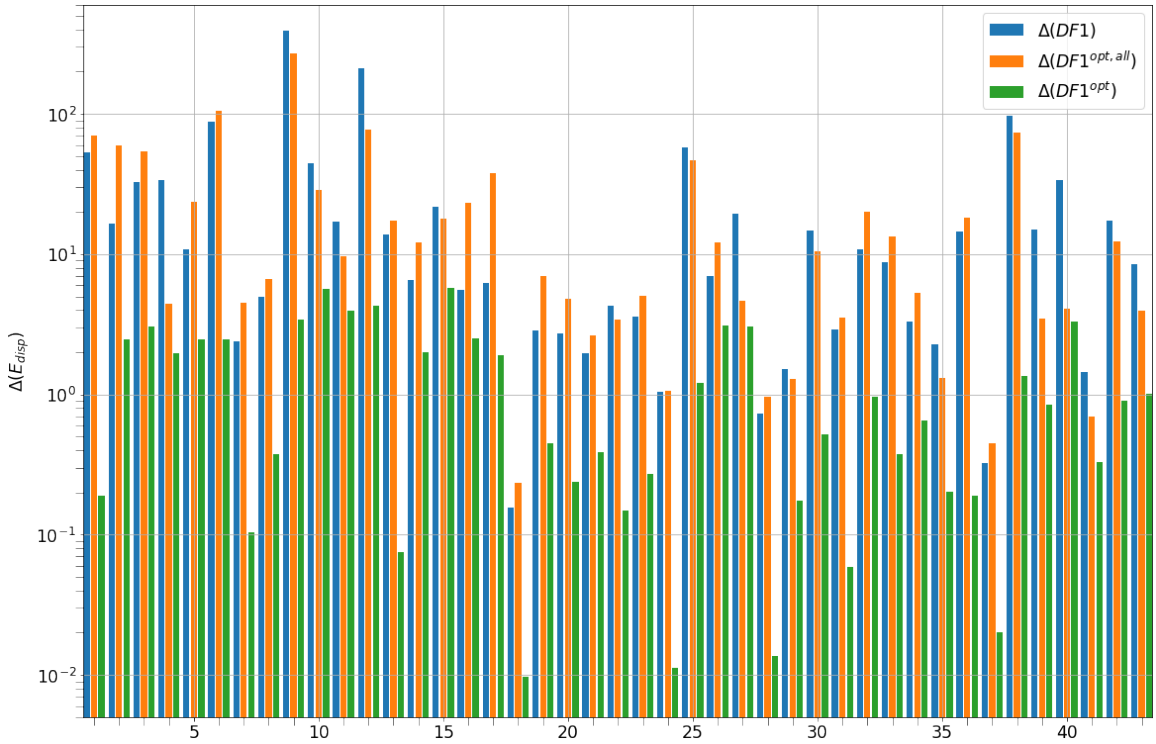

Figure 4.13: Barplot comparing the average difference between the ab initio dispersion energy (E_{disp}) and E_{disp} calculated by multiplying the London formula (Equation 1.24) with Equation 4.3 using parameters found by the EQL ($\Delta(DF1)$), optimized for the whole data set ($\Delta(DF1^{opt,all})$) and optimized individually ($\Delta(DF1^{opt})$). The Difference is calculated according to Equation 4.1.

When inspecting Figure 4.13 two things become evident. First, the prefactors found by the EQL perform in a comparable way to the optimized parameters found by the Powell algorithm, which is expected as the EQL is simultaneously optimizing formula structure and prefactors. Additionally the individually optimized parameters are resulting in a significantly better fit in some cases, e.g. for the atomic pairs ArAr (1 in Figure 4.13) and HeLi (24 in Figure 4.13) among others. Those two examples are inspected more detailed, the parameters are shown in Figure 4.14 and 4.16 and the calculated E_{disp} in Figure 4.15 and 4.17.

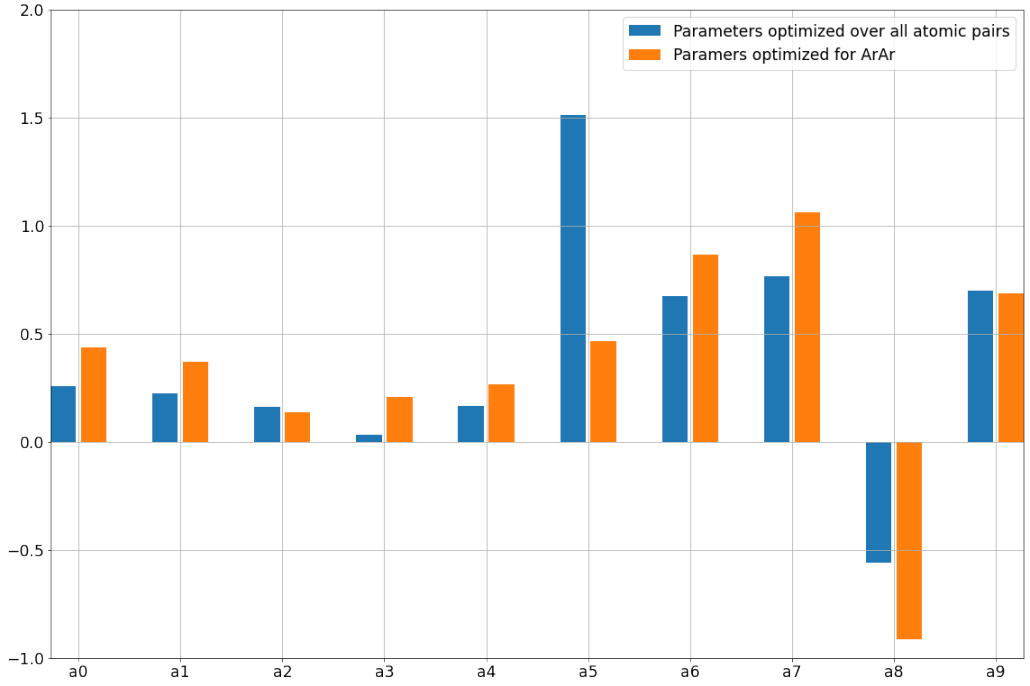

Figure 4.14: Barplot of the parameters in f_{damp}^{M1} optimized for the whole considered data set consisting of 43 atomic pairs and optimized for ArAr.

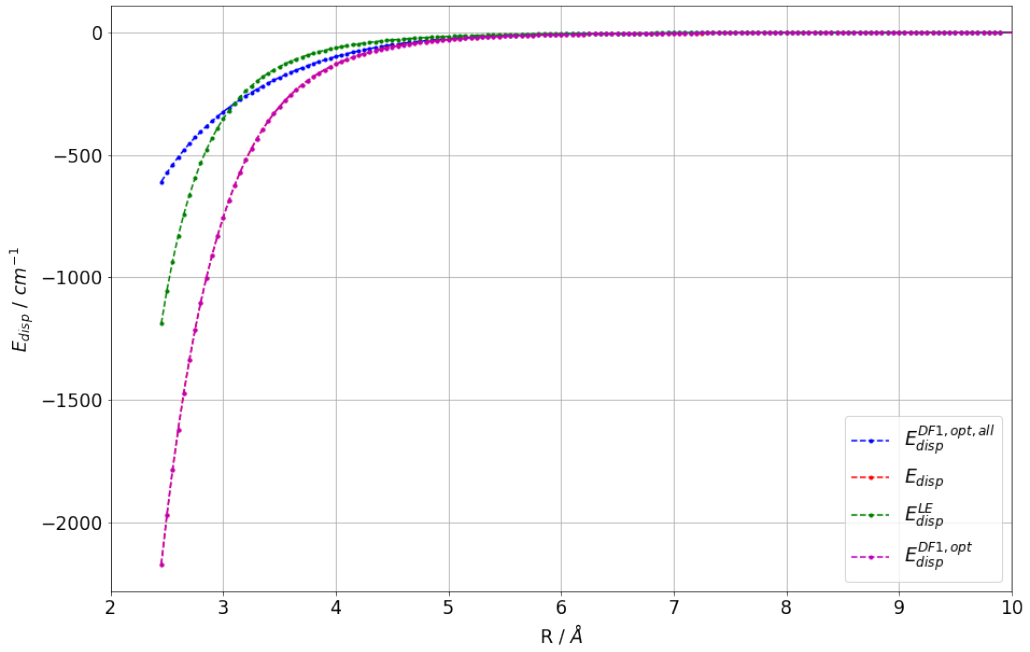

Figure 4.15: Comparison of ab initio dispersion energy (E_{disp}) and those energies obtained with the London formula E_{disp}^{LE} and E_{disp}^{LE} multiplied with f_{damp}^{M1} with optimized parameters for all 43 atomic pairs ($E_{disp}^{DF1,opt,all}$) and for ArAr ($E_{disp}^{DF1,opt}$) over the distance. The x-axis is cropped at 10 Å for better visibility.

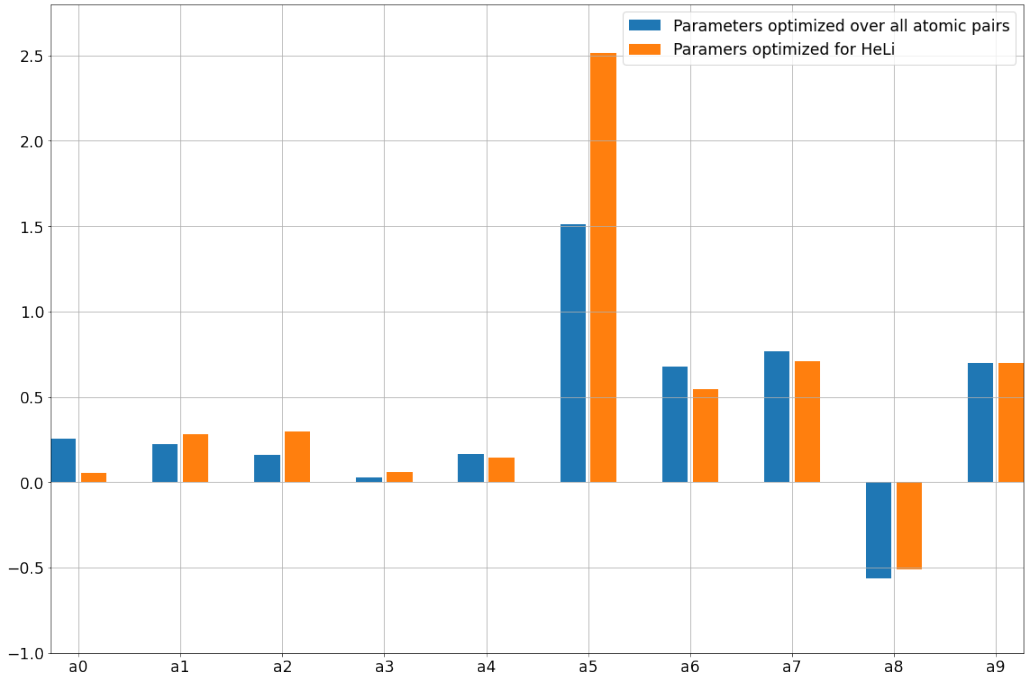

Figure 4.16: Barplot of the parameters in f_{damp}^{M1} optimized for the whole considered data set consisting of 43 atomic pairs and optimized for HeLi.

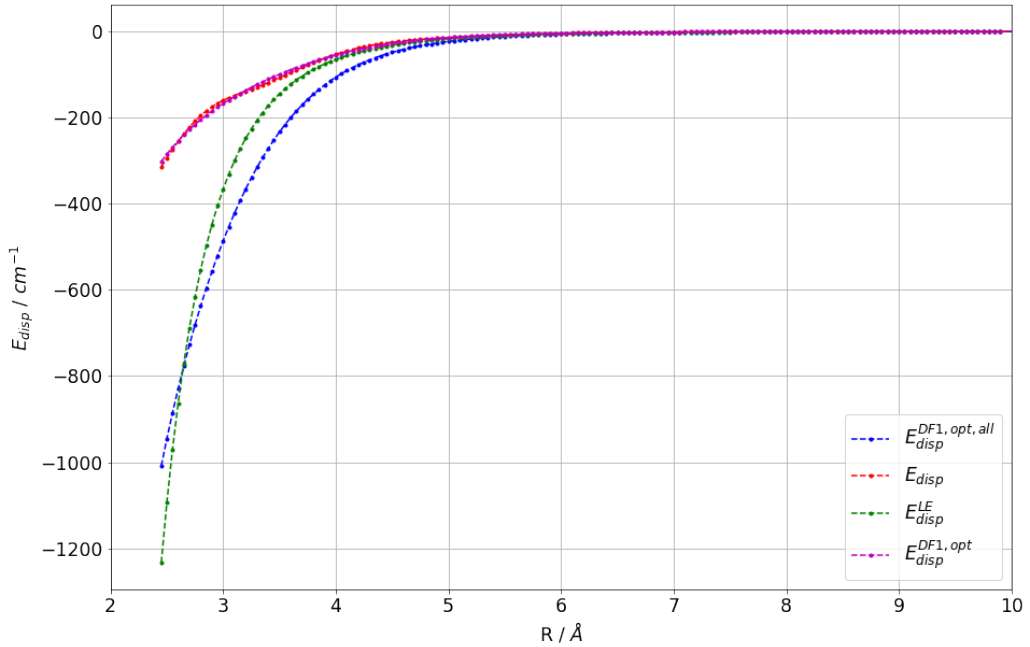

Figure 4.17: Comparison of ab initio dispersion energy (E_{disp}), calculated using the London formula E_{disp}^{LE} and E_{disp}^{LE} multiplied with f_{damp}^{M1} with optimized parameters for all 43 atomic pairs ($E_{disp}^{DF1,opt,all}$) and for HeLi ($E_{disp}^{DF1,opt}$) over the distance. The x-axis is cropped at 10 Å for better visibility.

While the individually optimized parameters to caclulate E_{disp} are increasing the drop of the trajectory for smaller distances in ArAr (see Figure 4.15), they are damping it in case of HeLi (see Figure 4.17). This different behavior can also be seen when inspecting the optimized parameters for those atomic pairs in Figure 4.16 and 4.14, e.g. a_5 is much higher in the optimization for HeLi than for ArAr and the value of a_0 is the other way round. This indicates a lack of of one or more element specific variables in f_{damp}^{M1}.
The Pearson correlation coefficient of a_0 and a_5 with some material specific coefficients are calculated and a correlation matrix is shown in Figure 4.18. The highest correlation is with a_0 and *alpha*, the sum of the dipole polarizability of the involved atoms and its regression line is shown in Figure 4.19.

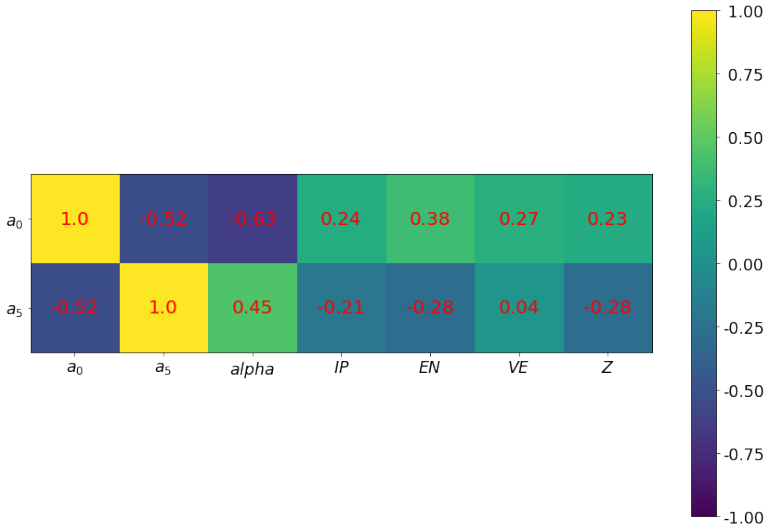

Figure 4.18: Correlation matrix of the fit parameters a_0 and a_5, optimized individually for every atomic pair in the considered data set and element specific variables of the involved atoms: *alpha* . . . sum of the dipole polarizability (Ref. [34]), IP . . . sum of the ionization potential (Ref. [16]), EN . . . sum of the Allen electronegativity (Ref. [25]), VE . . . sum of the valence electrons, Z . . . sum of the atomic number.

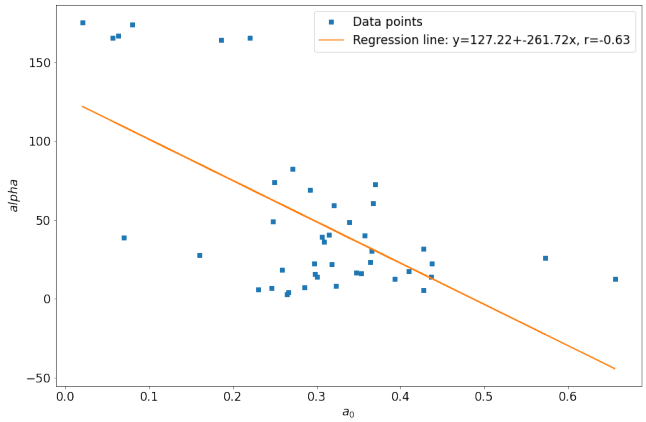

Figure 4.19: The sum of the dipole polarizability of the involved atoms (*alpha*) over the fitting parameter a_0, optimized individually for every atomic pair in the considered data set, with the respected regression line and the Pearson correlation coefficient given in the legend.

5 Conclusions

In this master thesis it was attempted to find simple analytical expressions linking known atomic features and other variables derived by well established hypotheses to the London-dispersion interaction (E_{disp}) between two atoms.

A data set had to be created containing E_{disp}, the distance between the atoms, the ionization energy, dipole polarization and van der Waals radii for a set of atomic pairs. Hartree-Fock (HF) and Coupled Cluster (CC) simulations have been performed for 51 atomic pairs combining the first 18 atoms in the periodic table with the noble gas atoms helium, neon and argon and E_{disp} was calculated as the difference between the total energy calculated by HF and CC. The other variables were gathered from literature. After a detailed investigation of E_{disp} curves, some of them were excluded from further investigation and a set of 43 atomic pairs was obtained.

After a last data cleaning step, cropping the E_{disp} values for large distances where they are approaching zero, the resulting data set was given as training data to a Symbolic Regression application called Equation Learner (EQL), a neural network that finds analytical expressions from a data input, to find on the one hand expression similar to the proposed London formula (Equation 1.24) and damping functions that multiplied with the London formula are approximating E_{disp} more closely.

The EQL finds formula expressions similar to the London formula with similar performance but without including the ionization potential (IP) of the involved atoms. This indicates that the IP of both atoms in atomic pairs do not have a large impact on E_{disp} and can be ignored for the correction. Additionally, the corrections seem to be much less effective in the short range regime, where the distance between the two atoms is smaller than the sum of the van der Waals radii of the respected atoms and the electron orbitals are overlapping. In this region forces not captured by HF and differing from the R^{-6} term of E_{disp} are contributing to the total electronic energy in a larger extent and our central approximation that calculates E_{disp} as the difference of the electronic energy calculated by HF and CC is compromised. This could be avoided by including R^{-8}, R^{-10} and other terms in the search and simultaneously limiting it to an analytical expressions with a formula structure that separates R^{-N} ($N = 6, 8 \dots$) terms, which remains as an interesting topic for further research.
In the damping function investigation, several corrections were found that improve the fit of E_{disp}, as calculated by the London formula, over the whole considered data set in the short range and also in the long range regime, but failed to do this for E_{disp} curves of some atomic pairs. A more detailed investigation revealed that this might be due to the lack of atom-specific features in the damping function. A (weak) correlation of a fitting parameter with the dipole polarizability of the involved atoms can be noticed.

Bibliography

[1] Axel D. Becke and Erin R. Johnson. "A density-functional model of the dispersion interaction". In: *The Journal of Chemical Physics* 123.15 (2005), p. 154101. DOI: 10.1063/1.2065267. eprint: https://doi.org/10.1063/1.2065267. URL: https://doi.org/10.1063/1.2065267.

[2] E. Caldeweyher et al. "A generally applicable atomic-charge dependent London dispersion correction". English. In: *Journal of Chemical Physics* 150.15 (2019). Cited By :49.

[3] C. Chen et al. "A Critical Review of Machine Learning of Energy Materials". English. In: *Advanced Energy Materials* 10.8 (2020). Cited By :4.

[4] X. Chu and A. Dalgarno. "Linear response time-dependent density functional theory for van der Waals coefficients". In: *The Journal of Chemical Physics* 121.9 (2004), pp. 4083–4088. DOI: 10.1063/1.1779576. eprint: https://doi.org/10.1063/1.1779576. URL: https://doi.org/10.1063/1.1779576.

[5] J Cizek. "On the Correlation Problem in Atomic and Molecular Systems. Calculation of Wavefunction Components in Ursell-Type Expansion Using Quantum-Field Theoretical Methods". English. In: *The Journal of chemical physics* 45.11 (1966). Cited By :906, pp. 4256–4266.

[6] V. Fock. "Näherungsmethode zur Lösung des quantenmechanischen Mehrkörperproblems". German. In: *Zeitschrift für Physik* 61.1-2 (1930). Cited By :1007, pp. 126–148.

[7] R. A. Friesner. "Ab initio quantum chemistry: Methodology and applications". English. In: *Proceedings of the National Academy of Sciences of the United States of America* 102.19 (2005). Cited By :172, pp. 6648–6653.

[8] L. Goerigk. "A Comprehensive Overview of the DFT-D3 London-Dispersion Correction". English. In: Non-Covalent Interactions in Quantum Chemistry and Physics: Theory and Applications. Cited By :13. 2017, pp. 195–219.

[9] S. Grimme. "Accurate description of van der Waals complexes by density functional theory including empirical corrections". English. In: *Journal of Computational Chemistry* 25.12 (2004). Cited By :3156, pp. 1463–1473.

[10] S. Grimme. "Dispersion Interaction and Chemical Bonding". English. In: vol. 9783527333158. The Chemical Bond: Chemical Bonding Across the Periodic Table. Cited By :17. 2014, pp. 477–500.

[11] S. Grimme. "Semiempirical GGA-type density functional constructed with a long-range dispersion correction". English. In: *Journal of Computational Chemistry* 27.15 (2006). Cited By :14671, pp. 1787–1799.

[12] S. Grimme, S. Ehrlich, and L. Goerigk. "Effect of the damping function in dispersion corrected density functional theory". English. In: *Journal of Computational Chemistry* 32.7 (2011). Cited By :6125, pp. 1456–1465.

[13] S. Grimme et al. “A consistent and accurate ab initio parametrization of density functional dispersion correction (DFT-D) for the 94 elements H-Pu”. English. In: *Journal of Chemical Physics* 132.15 (2010). Cited By :14422.

[14] D. R. Hartree. “The Wave Mechanics of an Atom with a Non-Coulomb Central Field. Part I. Theory and Methods”. In: *Mathematical Proceedings of the Cambridge Philosophical Society* 24.1 (1928), pp. 89–110. DOI: 10.1017/S0305004100011919.

[15] D. M. Hawkins. “The Problem of Overfitting”. English. In: *Journal of chemical information and computer sciences* 44.1 (2004). Cited By :971, pp. 1–12.

[16] W.M. Haynes. *CRC Handbook of Chemistry and Physics*. CRC Handbook of Chemistry and Physics. CRC Press, 2014. ISBN: 9781482208689. URL: https://books.google.no/books?id=bNDMBQAAQBAJ.

[17] F. L. Hirshfeld. “Bonded-atom fragments for describing molecular charge densities”. English. In: *Theoretica Chimica Acta* 44.2 (1977). Cited By :3607, pp. 129–138.

[18] Frank Jensen. *Introduction to Computational Chemistry*. Hoboken, NJ, USA: John Wiley and Sons, Inc., 2006. ISBN: 0470011874.

[19] Azme Khamis et al. “The Effects of Outliers Data on Neural Network Performance”. In: *Journal of Applied Sciences* 5 (Jan. 2005).

[20] D. P. Kingma and J. L. Ba. “Adam: A method for stochastic optimization”. English. In: *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*. Cited By :6557. 2015.

[21] Reader Kramida Ralchenko and NIST ASD Team. *NIST Atomic Spectra Database (version 5.7.1)*. 2020. URL: https://physics.nist.gov/asd (visited on 09/02/2020).

[22] H. G. Kümmel. “A biography of the coupled cluster method”. English. In: *International Journal of Modern Physics B* 17.28 II (2003). Cited By :16, pp. 5311–5325.

[23] F. London. “Zur Theorie und Systematik der Molekularkräfte”. German. In: *Zeitschrift für Physik* 63.3-4 (1930). Cited By :804, pp. 245–279.

[24] Per-Olov Löwdin. “Quantum Theory of Many-Particle Systems. III. Extension of the Hartree-Fock Scheme to Include Degenerate Systems and Correlation Effects”. In: *Phys. Rev.* 97 (6 Mar. 1955), pp. 1509–1520. DOI: 10.1103/PhysRev.97.1509. URL: https://link.aps.org/doi/10.1103/PhysRev.97.1509.

[25] Joseph B. Mann, Terry L. Meek, and Leland C. Allen. “Configuration Energies of the Main Group Elements”. In: *Journal of the American Chemical Society* 122.12 (2000), pp. 2780–2783. DOI: 10.1021/ja992866e. eprint: https://doi.org/10.1021/ja992866e. URL: https://doi.org/10.1021/ja992866e.

[26] G. Martius and C. H. Lampert. “Extrapolation and learning equations”. English. In: *5th International Conference on Learning Representations, ICLR 2017 - Workshop Track Proceedings*. 2017.

[27] Łukasz Mentel. *mendeleev – A Python resource for properties of chemical elements, ions and isotopes*. Version 0.6.0. 2014. URL: https://github.com/lmmentel/mendeleev.

[28] Aaron Meurer et al. “SymPy: symbolic computing in Python”. In: *PeerJ Computer Science* 3 (Jan. 2017), e103. ISSN: 2376-5992. DOI: 10.7717/peerj-cs.103. URL: https://doi.org/10.7717/peerj-cs.103.

[29] M. J. D. Powell. "An efficient method for finding the minimum of a function of several variables without calculating derivatives". In: *The Computer Journal* 7.2 (Jan. 1964), pp. 155–162. ISSN: 0010-4620. DOI: 10.1093/comjnl/7.2.155. eprint: https://academic.oup.com/comjnl/article-pdf/7/2/155/959784/070155.pdf. URL: https://doi.org/10.1093/comjnl/7.2.155.

[30] S. S. Sahoo, C. H. Lantpert, and G. Martius. "Learning equations for extrapolation and control". English. In: *35th International Conference on Machine Learning, ICML 2018*. Vol. 10. Cited By :3. 2018, pp. 7053–7061.

[31] J. Schmidhuber. "Deep Learning in neural networks: An overview". English. In: *Neural Networks* 61 (2015). Cited By :5406, pp. 85–117.

[32] H. Schröder, A. Creon, and T. Schwabe. "Reformulation of the D3(Becke-Johnson) Dispersion Correction without Resorting to Higher than C6 Dispersion Coefficients". English. In: *Journal of Chemical Theory and Computation* 11.7 (2015). Cited By :31, pp. 3163–3170.

[33] E. Schrödinger. "Quantisierung als Eigenwertproblem". German. In: *Annalen der Physik* 384.4 (1926), pp. 361–376.

[34] Peter Schwerdtfeger and Jeffrey K. Nagle. "2018 Table of static dipole polarizabilities of the neutral elements in the periodic table". In: *Molecular Physics* 117.9-12 (2019), pp. 1200–1225. DOI: 10.1080/00268976.2018.1535143.

[35] J. C. Slater. "Note on hartree's method [5]". English. In: *Physical Review* 35.2 (1930). Cited By :226, pp. 210–211.

[36] J. C. Slater. "The theory of complex spectra". English. In: *Physical Review* 34.10 (1929). Cited By :496, pp. 1293–1322.

[37] Attila Szabo and Neil S. Ostlund. *Modern Quantum Chemistry: Introduction to Advanced Electronic Structure Theory*. First. Mineola: Dover Publications, Inc., 1996.

[38] *The NIST Reference on Constants, Units and Uncertainty. Hartree energy*. https://physics.nist.gov/cgi-bin/cuu/Value?hr. Accessed: 2020-08-07.

[39] A. Tkatchenko and M. Scheffler. "Accurate molecular van der Waals interactions from ground-state electron density and free-atom reference data". English. In: *Physical Review Letters* 102.7 (2009). Cited By :2895.

[40] T. Tsuneda. "Density functional theory in quantum chemistry". English. In: vol. 9784431548256. Density Functional Theory in Quantum Chemistry. 2013, pp. 1–200.

[41] Pauli Virtanen et al. "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python". In: *Nature Methods* 17 (2020), pp. 261–272. DOI: 10.1038/s41592-019-0686-2.

[42] H.-J. Werner et al. *MOLPRO, version 2019.2, a package of ab initio programs*. Cardiff, UK, 2019. URL: https://www.molpro.net.

[43] D. E. Woon and T. H. Dunning Jr. "Gaussian basis sets for use in correlated molecular calculations. V. Core-valence basis sets for boron through neon". English. In: *The Journal of chemical physics* 103.11 (1995). Cited By :2177, pp. 4572–4585.