Doktor der Naturwissenschaften (Dr. rer. nat.)

PhD Thesis: Computer Science

Multifidelity Machine Learning Methods for Quantum Chemical Properties

Vivin Vinod

Faculty of Mathematics and Natural Sciences

University of Wuppertal

Gaußstraße 20

Wuppertal - 42119, Germany

Multifidelity Machine Learning Methods for Quantum Chemical Properties

A thesis submitted in fulfillment of the requirements for the degree of

Doktor der Naturwissenschaften (Dr. rer. nat.)

in

Computer Science

by

Dottore Magistrale Vivin Vinod

to

Faculty of Mathematics and Natural Sciences

BERGISCHE
UNIVERSITÄT
WUPPERTAL

University of Wuppertal
Gaußstraße 20
Wuppertal - 42119, Germany

Supervised by: Prof. Dr. Peter Zaspel

AUTHOR'S DECLARATION

I, *Vivin Vinod*, declare that this work, submitted in fulfillment of the requirements for Doktor der Naturwissenschaften (Dr. rer. nat.) in *Computer Science* at the *Faculty of Mathematics and Natural Sciences*, University of Wuppertal, Germany, is in its entirety my own work presented as a compiled thesis. All information sources and related literature are indicated and cited. This document has not been submitted for assessment at any other academic institutions.

Since the zeitgeist demands it, I also declare that no aspect or component of this dissertation has been generated using Generative AI or Large Language Models.

SIGNATURE: _____

Vivin Vinod

DATE: 19-MAY-2025

PLACE: Wuppertal, Germany

DEDICATION

To making rectifiable mistakes

Talent is a pursued interest. Anything that you're willing to practice, you can do.

— Bob Ross

... a tale
Told by an idiot, full of sound and fury,
Signifying nothing.

— William Shakespeare, *Macbeth (Act 5 Scene 5)*

The map is not to be confused with the journey.

— Gabor Maté, *Scattered Minds*

ABSTRACT

Machine learning has made significant progress in being used as an augmented tool to progress research in many scientific fields. One such area is the field of quantum chemistry where machine learning methods have been employed to accelerate calculations by replacing high cost quantum chemistry computations with trained machine learning models. While this has significantly reduced the computational resource usage in making new calculations, there still remains the high cost of generating high accuracy, or high *fidelity*, training data for such models.

This new overhead can be mitigated by the use of multifidelity methods which reduces the cost of training data required to achieve a certain model accuracy. This compiled-format dissertation develops and studies the use of multifidelity methods with machine learning for the prediction of quantum chemical properties. The Multifidelity Machine Learning approach is developed as a time-efficient alternative to the single fidelity machine learning methods. Novel methodological developments such as optimized Multifidelity Machine Learning and the Γ-curve are developed in this dissertation to demonstrate that low-cost high-accuracy machine learning for quantum chemistry is a viable option with multifidelity approaches. The benefit of using such multifidelity models for the prediction of several quantum chemistry properties ranging from atomization energies to excitation energies is benchmarked. Assessments in terms of cost of generating training data versus model accuracy of several multifidelity models are studied for the prediction of diverse quantum chemistry properties. In addition, a multifidelity benchmark dataset with computational time-costs is generated for the benchmarking of multifidelity models and for future development in this field. The effect of the multifidelity data hierarchy on model accuracy and the time-cost efficiency of several multifidelity models is also studied to provide a comprehensive outlook on the use of multifidelity methods for quantum chemical properties.

Two specific applications using the herein developed multifidelity methods are presented. The first, prediction of ground state energies of several small monomers which are atmospherically relevant, is used as an additional assessment of the efficiency of the multifidelity methods in comparison to existing state of art machine learning methods used in

quantum chemistry. Second, the novel Γ-curve is applied to predicting excitation energies of a 16 porphyrin molecules over a 40 pico-second trajectory at a high fidelity quantum chemical method showing significant reduction in the cost incurred to train the machine learning model without sacrificing accuracy.

Finally, conclusions are drawn about the efficiency of multifidelity machine learning methods for use in quantum chemistry. Extensions of the use of optimized MFML and the Γ-curve method are discussed in order to direct future research arising from the works of this dissertation.

CONTRIBUTION STATEMENT

This dissertation is a compiled collection of works of the author that were generated over the course of the PhD research between 2022-2025. Each chapter in the main body of this dissertation is taken from the corresponding publication.

Since this dissertation is a compiled collection of several published/under consideration works, this section will briefly provide statements regarding contributions to the published works. In particular, if the chapter corresponds to a publication involving authors other than the author of this dissertation and his supervisor, Peter Zaspel, the specific contributions that are made in that chapter/publication by other authors are enumerated.

Chapter 5 is a publication that had collaborative authors, namely, Sayan Maity and Ulrich Kleinekathöfer of Constructor University (CU) Bremen gGmbH. The QC calculations of the arenes and the QC time-domain computations were carried out by these authors. The remainder of the work, such as, development of the MFML model, learning curves, and time-cost assessments were carried out by the author of this dissertation. Similarly, Chapter 6 is a publication in collaboration with Ulrich Kleinekathöfer. The dataset of arenes (benzene, naphthalene, and anthracene) for this publication was generated by collaborators at CU. Subsequent chapters from the second part of the thesis contain sole contributions from the author and his supervisor, Prof. Dr. Peter Zaspel. Any pre-existing datasets that were utilized have been appropriately cited in these chapters.

In the third part of this thesis, two applications of the herein developed multifidelity methods are delineated. These are taken from collaborative research performed in ref. [1] and ref. [2] respectively. Only the contributions of the author from these works are presented in this dissertation. Any dataset used and described in these chapters is correspondingly cited. For instance, in ref. [2], the entire system of porphyrins on clay surface was designed and developed by collaborators from CU. The multifidelity method utilized in the chapter was developed and evaluated by the author and is therefore the only portion of ref. [2] that is presented in the chapter. A similar strategy is employed for the second application listed in Chapter 11 for the predictions of ground state energies of monomers. The dataset and corresponding contribution for the chemistry computations are appropriately

cited wherever introduced. Focus is retained only on the multifidelity assessments made by the author.

ACKNOWLEDGMENTS

There is the security of knowing one has a statistically smaller chance of getting shot with an arrow. And then there is the security of knowing that there are people in the world who will care deeply if one is.

— David Graeber & David Wengrow, *The Dawn of Everything*

What work is complete without thanks due to those who made the journey a pleasure? Like all good works of literature, a work of academic research is something that demands the acknowledgment of the many who were present in the background holding it all together. This work is not just mine but also theirs to share with. I am greatly indebted to all who have come in contact with me, even if it were briefly so. I mention some here, but I am assured that no amount of writing would reflect the true measure of debt I owe to each of them and others who might not make an appearance below.

I thank God for His marvelous ways and redemptive grace through faith in Christ. To Him be glory forever, Amen. What is man that you are mindful of him, the son of man that you care for him?

Vinod, Anitha, Vytik, I thank not only for their unwavering support but also for their pretense in understanding my work, no matter how convoluted I may have presented it to them. As little as I mention them here, I will be forever indebted to them for all I have and will have. *Mummy* and *Daddy* I thank for all that they did to raise not just their own children but also us, not simply in the ways of the Lord, but also wise to the ways of this world. Anil is to be thanked for spurring me to think. Simi for thinking with me. Hanoch and Ruth for giving me brief pauses between the thinking with their *non sequiturs*, if I may say so. *Kunjamma* is to be thanked for being the first one to introduce me to Road Rash which inexplicably, I hold, paved way for my interest not just in video games but also the machines they ran on. Jimmy is to be thanked for the letting me twiddle with the sax and countless other gizmos. Aaron and Sharon remain the first cousins I ever held, how you have grown.

I want to thank my friends who stood by me through these past years, in no particular order. Arunima, who being oceans away, remains close and said '*Mere ko dedicate kar*'. Iyer for therapy chat. Kapil for promising to clean up, without questions being asked. Chelsea

for letting me get her biscuits at Stephen's. Sayan for the OG metro walk, and Koushani for being my fellow trickster. Aditya for breaking 'just the one thing' which still remains a mystery. Satvik for considering my Biriyani the second best in the world. CV and Tony for being the first names popping up on the share button. Kavs for being a bulwark of a support, and making poor decisions with me. The entire Jülich gang for the fun times and board-game-bottomless-pit. Bhuvan for the drive through the rain with 0 visibility without killing us all. Vatsala for suffering DB with me for months *after* the visit. To AG for forcing me to download Enthought Canopy and Visual Python, which is where all of this began. Saumya for everything we have been through and somehow still sticking around each other. KTK for being a good DM and player likewise, roll on. Nate for LOTR marathons. Nihal for his mom's chicken masala. Pets for her unwavering friendship. Mo for his friendship (read as *Kaffee Kuchen*). René, Ryo, Jesús, Kshema, Matthias, Louise, for the fun time at work, bowling, and ramen; to many more hidden pathways at the university. Bernardo, Sundeep, Paul, and others for the splendid time at IBC. Marie for bouldering and the cute little bundle of joy. Nandana for suggesting 'To folks on Delhi metro from your looooong journey from greater NOIDA to viswavidyalaya who taught you patience' as a dedication statement. Ronika for everything at Stephens. Jain for forcing me to go to some lake in the morning which I enjoyed. Alex for trying very hard to get me to watch the original Star Wars, and failing. And to the countless others who might have been brief shooting stars but whose essence I regularly find in the nooks and crannies of my identity; to those I have laughed with and thought 'Oh, this is the point of it all' and to those very very few who have grieved with me and said 'This is also the point of it all'.

I would also like to thank my collaborators at Constructor University Bremen, Ulrich Kleinekathöfer, Sayan Maity, and Dongyu Lu for their support in the research reflected in this work. In particular for being patient in explaining the concepts in fields that were well beyond my expertise.

Last, certainly not the least, Peter is to be thanked. From supervision to editing, from congratulating to comforting, Peter has been ubiquitous in this journey. Not only was he present, he was also actively involved. The research in this work would have been impossible without his support and guidance over the past years. This work is an ode to his style of supervision and how he strove to make the entire process of my PhD as memorable and comfortable as possible, often at the expense of his own comfort. I thank him for making this journey one of great joy and helping me train in the world of academia without the loss of individuality (except maybe using words like *impetus* and *cul-de-sac* in papers). To many more years of collaborative research!

TABLE OF CONTENTS

List of Abbreviations

BoB Bag of Bonds representation

CCSD(T) Coupled Cluster Singles Doubles (Triplets) method

CM Coulomb Matrix representation

CQML Combination technique Quantum Machine Learning

DFT Density Functional Theory

DFTB Density Functional Tight Binding method

DLPNO-CCSD(T) domain-based local pair natural orbital CCSD(T)

DNN Deep Neural Network

eV electron Volts (unit of energy)

FCHL Faber-Christensen-Huang-Lilienfeld representation

GP Gaussian Process

GPR Gaussian Process Regression

hE Hartree Energy (unit of energy)

HF High Fidelity

HOMO Highest Occupied Molecular Orbital

kcal kilo calories (unit of energy)

KRR Kernel Ridge Regression

LC-DFTB Long range Corrected Density Functional Tight Binding method

LF Low Fidelity

LUMO Lowest Unoccupied Molecular Orbital

MAE Mean Absolute Error

MD Molecular Dynamics

meV milli electron Volts (unit of energy), i.e. 10^{-3} eV

MF Multifidelity

MFML Multifidelity Machine Learning

mhE Milli Hartree, that is 10^{-3} hE

ML Machine Learning

MP2 Møller-Plesset perturbation theory

MT Multi-task

MTGPR Multi-task GPR

NN Neural Network

o-HBDI 4-(2-hydroxybenzylidene)-1,2-dimethyl-1H-imidazol-5(4H)-one

o-MFML Optimized Multifidelity Machine Learning

OLS Ordinary Least Squares

PDE Partial Differential Equation

PES Potential Energy Surface

PINN Physics Informed Neural Network

QC Quantum Chemistry

QeMFi Quantum Chemistry MultiFidelity (benchmark dataset)

RKHS Reproducing Kernel Hilbert Space

SCF Self Consistent Field Method

SGCT Sparse Grid Combination Technique

SLATM The Spectrum of London and Axilrod-Teller-Muto representation

SMA 2-(methyliminomethyl)phenol

SMILES Simplified Molecular Input Line Entry System of molecular representations

SOAP Smooth Overlap of Atomic Positions representation

TD-DFT Time Dependent Density Functional Theory

TZVP Triple Zeta for Valence electrons plus Polarization function

WFT Wave Function Theory

ZINDO Zerner's intermediate neglect of differential orbital method with spectroscopic parameters together with configuration interaction using single excitations including the 10 HOMO and the 10 LUMO as an active space

List of Symbols and Notations

Symbol	Meaning
$\hbar$	Reduced Planck's constant; 1.054×10^{-34} Joule second
∇_n^2	Laplacian operator in n Cartesian dimensions
$\hat{\boldsymbol{H}}$	Hamiltonian Operator
$\hat{\boldsymbol{H}}_{el}$	Electronic Hamiltonian Operator
$\Psi(\boldsymbol{r}, t)$	Wave function in space($\boldsymbol{r}$) and time(t)
Z_i	Atomic Number of the ith atom
$\boldsymbol{R}_i$	Cartesian coordinate of the ith atom
$\mathbb{R}$	Set of all real numbers
$\mathbb{R}^D$	Set of all points in D-dimensional space
$\mathbb{R}_+$	Set of all non-zero positive real numbers; that is $\{x \in \mathbb{R} : x > 0\}$
$\mathbb{N}$	Set of all natural numbers
$\boldsymbol{A}^T$	Transpose of matrix $\boldsymbol{A}$
$\|\boldsymbol{a}\|_p$	p-norm of vector $\boldsymbol{a}$
$\mathcal{T}$	Training set
$GP(m, k)$	Gaussian Process with mean function m and covariance function k
$\mathbb{E}[G]$	Expectation value of a random variable G
$x \sim \mathcal{N}(\mu, \sigma^2)$	Random variable x is normally distributed with mean μ and variance σ^2
$\mathscr{H}$	Reproducing Kernel Hilbert Space (RKHS)
$A := B$	A is equal to and defined as B
$a \propto b$	a is proportional to b
$\boldsymbol{a} \perp \boldsymbol{b}$	vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ are orthogonal, that is, $\langle a, b \rangle = 0$
$\boldsymbol{A} \bigodot \boldsymbol{B}$	Element by element matrix multiplication for matrices $\boldsymbol{A}$ and $\boldsymbol{B}$
$\boldsymbol{A} \bigotimes \boldsymbol{B}$	Kronecker product between matrices $\boldsymbol{A}$ and $\boldsymbol{B}$
$\boldsymbol{X}$	Molecular descriptor / Representation; that is, input feature to ML model
$k(\cdot, \cdot)$	Kernel function
λ	Lavrentiev regularizer/constant

Symbol	Meaning
$\boldsymbol{I}_n$	Identity matrix of size $n \times n$
exp	The exponential function
σ	Kernel width
f_b	Baseline fidelity
F	Target fidelity
$\mathscr{T}$	Training set
$\mathscr{V}_{val}$	Validation set
$\mathscr{V}_{test}$	Test/Evaluation set
QC_b	Conventionally computed quantum chemistry baseline
$P_{ML}(\boldsymbol{X})$	Prediction of an ML model for a representation $\boldsymbol{X}$
γ	Ratio of training samples used at fidelities f and $f+1$; also called scaling factor
θ	Time informed scaling factor determining training samples at each fidelity based on QC compute costs
$\lfloor n \rceil$	Nearest integer rounding of n
$\lvert v \rangle$	ket in Dirac notation; denotes vector v in an abstract vector space (V). Physically represents the state of some quantum mechanical system
$\langle u \rvert$	bra in Dirac notation; denotes a linear map $u : V \to \mathbb{C}$ with $\langle u \rvert v \rangle \in \mathbb{C}$
$\mathbb{C}$	set of all complex numbers of the form $a + ib$ with $a, b \in \mathbb{R}$

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

"Begin at the beginning," the King said gravely, "and go on till you come to the end: then stop."

— Lewis Carroll, *Alice in Wonderland*

In recent years, machine learning (ML) has seen widespread use across industries and areas of application. From self-driving cars [3, 4] to drug discovery [5], the use of ML methods has significantly reduced human effort required in progress and discovery. ML has become integral to most day-to-day activities with the advent of OpenAI's ChatGPT models with applications ranging from generating images and text to using the GPT models in teaching frameworks. We are truly in an augmented era of human intelligence and artificial intelligence working side-by-side to further the frontiers of science, research, and discovery.

Quantum Chemistry (QC) is a broad term that studies the behavior of molecules and atoms at a quantum level to better understand the world around us. Although QC covers a large range of themes and topics, in the recent years and in this manuscript, the term is used synonymous to computational QC, the simulation of atoms and molecules using computational hardware. With the rapid development of computational methods in tandem with the progress on the frontiers of hardware, QC calculations have become the norm in research. Computational software for QC such as ORCA [6, 7], Gaussian [8], PSI4 [9], NewtonX [10], and RDkit [11] have become the staple of modern QC research. With a varying range of numerical accuracies possible with such software, these have been used to significantly broaden the extent of our understanding of how the quantum world works. At its core, QC computations involve the numerical solution of the Schrödinger equation

which in it's time dependent form [12, 13] is given as

$$i\hbar\frac{\partial}{\partial t}\Psi(\boldsymbol{r},t) = \left[-\frac{\hbar^2}{2m}\nabla_3^2 + V(\boldsymbol{r},t)\right]\Psi(\boldsymbol{r},t)\,, \tag{1.1}$$

where Ψ is the wave-function of a particle of mass, m, in space and time, ∇_3^2 is the Laplacian operator in 3-D Cartesian coordinates, $\hbar$ is the reduced Planck constant, $i = \sqrt{-1}$, and V is the potential representing the environment that the particle is in. Often, only the time independent, also called the stationary, version of the Schrödinger equation is of interest. In such a case, Eq. (1.1) reduces to

$$\left[-\frac{\hbar^2}{2m}\nabla_3^2 + V(\boldsymbol{r})\right]\Psi(\boldsymbol{r}) = E_n\Psi(\boldsymbol{r})\,, \tag{1.2}$$

which is often written in operator form [13] as

$$\hat{\boldsymbol{H}}\Psi = E_n\Psi\,. \tag{1.3}$$

Here, E_n is the n-th eigenvalue of the electronic Hamiltonian operator $\hat{\boldsymbol{H}}$ for the system. In all applications described in this dissertation, the system of interest is some molecule. The molecule is fully categorized by the number of nuclei N_p in addition to the number of electrons N_e. The e^{th} electron, $\mathscr{E}_e$ can be identified with its position $\boldsymbol{r}_e \in \mathbb{R}^3$, while the i^{th} nucleus, $\mathscr{N}_i$, is characterized by the tuple $(\boldsymbol{r}_i, m_i, Z_i) \in \mathbb{R}^3 \times \mathbb{R} \times \mathbb{R}$ where m_i is its mass and Z_i is its atomic number. For a molecule that is considered without the surrounding environment (that is $V(\boldsymbol{r}) = 0$), the stationary electronic Hamiltonian operator is defined as [14]:

$$\begin{aligned}\hat{\boldsymbol{H}}_{el} := -\frac{1}{2}\sum_{j=1}^{N_e}\nabla_{\boldsymbol{r}_j}^2 - \sum_{j=1}^{N_e}\sum_{i=1}^{N_p}Z_i\frac{1}{\|\boldsymbol{r}_j - \boldsymbol{r}_i\|_2} \\ + \sum_{j=1}^{N_e}\sum_{j'>j}^{N_e}\frac{1}{\|\boldsymbol{r}_j - \boldsymbol{r}_{j'}\|_2} + \sum_{i=1}^{N_p}\sum_{i'>i}^{N_p}\frac{Z_iZ_{i'}}{\|\boldsymbol{r}_i - \boldsymbol{r}_{i'}\|_2} - \frac{1}{2}\sum_{i}^{N_p}\frac{1}{2m_i}\nabla_{\boldsymbol{r}_i}^2\,.\end{aligned} \tag{1.4}$$

In Eq. (1.4), the first term describes the kinetic energy of the electrons. Subsequent terms describe the electron-nuclei, electron-electron, and nuclei-nuclei interactions respectively. The last term encodes the kinetic energy of the nuclei. The eigenvalues of this operator are denoted as E_n in Eq. (1.3). The smallest eigenvalue, E_0 is the ground state energy of the molecule. With increasing values of n, one achieves the values of the excited state energies.

Depending on the amount of computational resources one is willing to expend, these numerical QC calculations of the energies or other QC properties can get close to experimental values [15]. QC methods that are faster and less expensive in terms of resource

requirements do so at the cost of accuracy. For instance, the gold standard in QC computations is Coupled Cluster theory with Single, Double, and a perturbative treatment of Triple excitations (CCSD(T)) which scales approximately as $\mathcal{O}(O^2 \cdot N^8)$ where N is the number of basis functions and O is the occupied orbitals [15]. Density functional theory (DFT) approaches and semi-empirical approaches such as LC-DFTB and ZINDO are less expensive, and less accurate. Thus, if one is interested in the QC calculations of large molecules or systems, it is often the case that such cheaper methods are used. For example, the use of semi-empirical LC-DFTB has been used to study the conversion of light energy in bacteriochlorophyll molecules in a 100 million atoms scale model of a chromatophore [16]. Even the use of DFT methods for such large systems has been impossible due to the large computational times required for a single point calculations. Within DFT, there are different levels of accuracy with respect to the ground truth that can be achieved based on the computational effort applied. This is often called the Jacob's Ladder of DFT [17]. The key takeaway of this discussion is this: higher accuracy requires higher amount of computational resources. The higher the accuracy, the better the QC computation is to make conclusions about the chemical system being studied. The term *fidelity* refers to the level of accuracy of the QC calculation with respect to the ground truth value of that property. Thus, CCSD(T) would be a high fidelity method while DFT would be a lower fidelity method.

1.1 Motivation

Solving the Schrödinger equation in Eq. (1.1) falls under the larger category of numerically solving partial differential equations (PDEs) and is a key aspect of understanding almost all physical problems ranging from deflection of a cantilever beam in a modern building to modeling aerodynamics in F1 cars. While conventional solvers for such methods rely on numerically solving the PDEs, a rapidly advancing field of research and application is the use of data-driven models, or surrogate models (SM) to approximate the solutions to the PDEs. That is, in order to solve a PDE such as the one in Eq. (1.1), instead of numerically solving the non-parametric problem, one replaces this workflow with SMs such as ML models. The high cost of running conventional QC computations poses a major challenge in broadening our understanding. For the past few years, ML has been making significant progress in the prediction of QC properties and expediting discovery and research [18, 19, 20]. This has greatly reduced the cost-budget of making new QC calculations.

The process of performing QC computations has now shifted to first training an ML model on high fidelity training data for a QC property of interest for a given system, say,

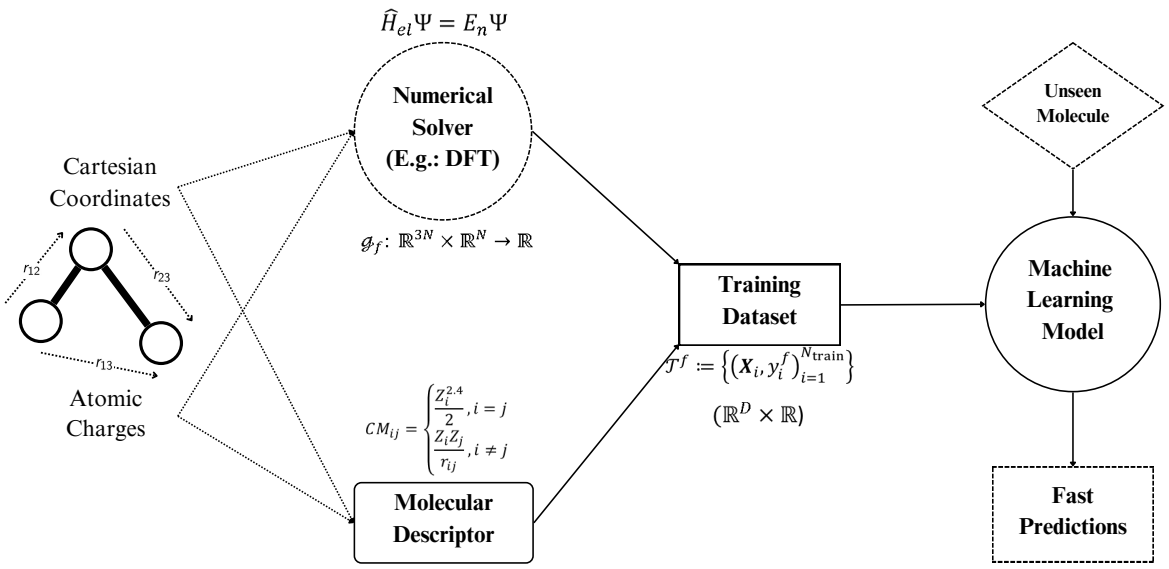


Figure 1.1: A general ML-QC workflow for the prediction of QC properties. Once trained, an ML model can provide fast and accurate predictions of the property of interest on molecules which were not part of the training data.

excitation energies of a molecule. This trained ML model is then used to make predictions on new molecules. ML models essentially learn the mapping between the Cartesian coordinates and atomic charges of the atoms constituting the molecules and the corresponding QC property. To this end, the pipeline involves generating what are called *molecular descriptors* or *representations* which convert Cartesian coordinates to machine learnable input features [21, 19]. The next step is using a training data set consisting of the pair of representations and QC property to train an ML method of choice such as neural networks (NN) or kernel ridge regression (KRR). The representations for the unseen molecules from the test set are generated and this forms the input to the trained ML model. The above described workflow is graphically represented in Figure 1.1. The numerical PDE solver, say for instance at the DFT fidelity, is used to generate a training dataset $\mathcal{T}^f$ consisting of the computed QC properties and machine learnable input descriptors, in this case, the Coulomb Matrices (see Chapter 2). This can be used to then train an ML model of interest. This model can then be used on unseen molecules, that is molecules that was not used to train the model. This results in a fast prediction of the QC property that the ML model was trained on at the fidelity of the training dataset. Notice that the numerical solver for some fidelity f is the function $g_f : \mathbb{R}^{3N} \times \mathbb{R}^N \to \mathbb{R}$ for scalar properties such as excitation energies, where N is the number of atoms in the molecule. The training dataset is of dimensions $\mathbb{R}^D \times \mathbb{R}$ where D is the dimension of the molecular descriptor.

Although the use of ML in QC has significantly reduced the cost of making new calculations, or rather predictions, this has opened up a new dimension of costs. ML models can only predict at the fidelity they are trained on. If a ML model is trained on a high fidelity dataset, it will predict at that fidelity. If trained on a low fidelity dataset, it can only predict at the low fidelity for the QC property of interest. Further, a common observation in ML models is that the larger the amount of training data, the more accurate the ML model is [22, 23]. This creates a new bottle-neck in the process of making QC calculations: the cost of generating high accuracy training data. It is a common observation that an ML model trained on a specific fidelity can predict at that fidelity. Furthermore, more training data often corresponds to a more accurate ML model. Thus in order to achieve a high accuracy ML model that predicts at high fidelity, one needs to generate a large amount of training data placing strain on the compute resources that are available. In order to overcome such an obstacle, over the past years, several methods have been proposed. For instance, active learning strategies have been proposed to effectively pick training samples and thereby reduce the redundant training data costs [24, 25, 26, 27]. A different approach is the use of training data from more than one fidelity, that is, multifidelity methods [28]. In such methods, training data consists of molecular descriptors and the QC property calculated at the highest fidelity, often called the *target fidelity*, and the QC property calculated also at cheaper and less accurate fidelities. Such methods have been shown to effectively reduce the number of high fidelity training data needed to achieve a specific model accuracy.

The earliest usage of multifidelity methods in ML for QC is the Δ-ML method where a two fidelity dataset is used and the ML model is used to learn the difference, or Δ, between the two [29]. The final model involved the prediction of this difference added to the QC computation of the cheaper fidelity. Due to its simplicity, this approach has since become a common tool in the ML-QC pipeline with applications ranging from energy band gaps to excited state dynamics of molecules [22, 30, 31]. This was followed by a methodological development over the Δ-ML method, termed the Combination technique Quantum Machine Learning (CQML) [32]. In this approach, several fidelities were used as opposed to a two-fidelity approach. Further, several *sub-models*, identified by fidelity and number of training set size, were trained. These sub-models were then combined in a systematic manner to result in a generalized Δ-ML like model with the lowest fidelity itself being predicted by a sub-model rather than being conventionally computed. This method was shown to be efficient in predicting CCSD(T) atomization energies for several organic molecules. The hierarchy structure in CQML was chosen across two dimensions, the QC method, such as CCSD(T) or MP2, and the basis set size, such as STO-3G or 6-31G. The combination of

the sub-models for these fidelities was carried out using approaches analogous to those in sparse grid combination techniques (SGCT) [33, 34, 35, 36, 37, 38] which resulted in the CQML being in some sense, a combination of Δ-ML over these different fidelities.

1.2 Objectives

Motivated by the above status of the use of ML in QC, the main aim of this dissertation is to develop multifidelity methods that result in construction of cheaper ML models. The developed multifidelity methods will reduce the cost of the overhead of generating training data that restricts scalable research with high fidelity QC predictions. In order to address this overarching theme, the following objectives are presented as checkpoints.

O1: Develop the Multifidelity Machine Learning (MFML) Method

Multifidelity methods for data driven models such as ML have been prominently used in several fields of research. As will be discussed in Chapter 3, literature on general multi-fidelity methods primarily deals with a bi-fidelity structure, that is one high fidelity and one low fidelity method. This objective will further work in this area by utilizing the MFML method to combine several fidelities with a one-dimensional hierarchy structure resulting in a low-cost high-accuracy ML model. The multifidelity machine learning (MFML) can be seen as a case of the CQML method where the fidelity hierarchy structure is assumed to be along a single dimension. This development would simplify assumed hierarchy structures within the multifidelity training data for the use of these methods in ML for QC.

> Design, devise, and develop the Multifidelity Machine Learning (MFML) method for combining several fidelities of training data in order to reduce the overall cost of an ML model.

O2: Study Overall Cost of the ML Model Instead of Number of Training Samples Used

The cost of generating training data is the newfound overhead of training high accuracy ML models in QC. As discussed previously, high accuracy, high fidelity ML models need a large amount of training data. This encumbers the ML-QC pipeline with further hurdles of generating a lot of high fidelity training data. However, the use of MFML has been seen to be effective in reducing the number of training samples needed at the higher fidelities

[32]. An important question that is to be asked is if the newly developed MFML method also reduces the overall time-cost of the training data. Through the work presented in this dissertation the focus of model efficiency will shift from the number of high fidelity training data needed to the overall cost of generating training data across all included fidelities. This will ensure a meaningful metric of assessment for the model accuracy of single fidelity and multifidelity models.

> Study the cost-accuracy trade-off and overall efficiency of the MFML model in comparison to the single fidelity model. Replace the number of training samples used in ML models with cost of generating training data to assessing model error as a function of time-cost.

O3: Examine and Evaluate the Degrees of Freedom in MFML

The MFML method is built by combining sub-models trained at several fidelities. This combination is weighted by unitary weights (± 1; see section 4.1) based on research from SGCT. Furthermore, like CQML and Δ-ML, MFML too is built with training data is nested in nature, that is the high-fidelity data also has low-fidelity counterparts within the training structure based on previous research in multifidelity methods for Gaussian Processes (GP)[39, 40]. Although not a strict condition for the working of multifidelity methods, it has been recommended in applications especially in QC. Additionally, the number of training samples used in the two fidelities of Δ-ML are identical while in CQML and MFML, each subsequent cheaper fidelity uses twice as much data as the costlier one, that is, scaled by a factor of 2. These are three vital points of investigation for the application of MFML in QC. Therefore it is imperative that these three areas be further investigated to understand their contribution to the efficiency and accuracy of the MFML method. First, in understanding how optimally combining the sub-models can affect prediction error and increase the methods robustness to non-ideal training data; this is referred to as optimized multifidelity machine learning (o-MFML). Second, the study of both MFML and o-MFML in the case of a strictly non-nested training data structure in QC would investigate the extremities of the multifidelity methods. Thirdly, and finally, the study of the number of training samples to be used at each fidelity while training sub-models and understanding the resulting cost/accuracy trade-off would push the MFML method to its limits. The investigation into the different degrees of freedom for the MFML model will result in a better understanding of not just the efficiency of the multifidelity methods but also about the contribution of each fidelity to the overall accuracy of the MFML model. This is crucial to being able to

reliably use the MFML method for large-scale systems in QC.

> Investigating the degrees of freedom in MFML by considering optimal combination of sub-models, non-nested training data structure, and scaling of number of training samples per fidelity.

O4: Benchmark Time-Cost of Multifidelity Methods

After developing various multifidelity methods for ML-QC, it becomes imperative to assess them against the primarily used multifidelity approach in QC, that is against Δ-ML. This form of assessment needs to be carried out not just in terms of the accuracy of the models but the overall reduction in the time-cost of generating training data, closely associated with **O2**. Since existing multifidelity datasets lack the time-cost of the QC calculations for each fidelity, this objective would require the generation of a new benchmark dataset which also contains this information for the fidelities generated. This dataset can then be used to carry out the test of overall efficiency of the diverse multifidelity methods in the prediction of QC properties.

> Cost-benefit benchmarking of herein developed multifidelity methods for QC using time to generate training data and model accuracy as metrics. To this end develop a benchmark multifidelity dataset.

1.3 Roadmap

With the above mentioned objectives, the primary focus is ensured to be the reduction of the overall cost of the training data for an ML model in the prediction of QC properties. In order to achieve this overarching aim, the individual objectives need to be fulfilled in a systematic manner. This section will delineate how the objectives are accomplished with the help of the methodological developments that are presented in this dissertation.

The Multifidelity Machine Learning (MFML) method is developed in this dissertation as a direct fulfillment of **O1**. This method involves the combination of training data from several fidelities as opposed to the use of a more common bi-fidelity approach in most applications of multifidelity methods. Once developed this method is first assessed in the prediction of excitation energies of small to medium size arenes as a proof of concept. Not only is the method evaluated for the reduction in the prediction error but also for the total time-cost of generating the training data required to achieve that error addressing **O2**. The

MFML method is shown to indeed reduce the cost incurred in generating training data for ML-QC for a desired error in prediction of the ML model. In these preliminary measure of the MFML method, it is seen to be sensitive to the data quality of the cheaper fidelities. In cases where the multifidelity training data is poor in quality and lacks a clear hierarchy as a result, the MFML model struggles to provide significant benefit.

In order to develop a multifidelity method that is robust to the quality of training data, this dissertation studies the tuning of the degrees of freedom that occur in MFML by virtue of construction. One of these is the use of a data-adapted approach to the optimal combination of fidelities in the MFML model resulting in the development of the optimized MFML (o-MFML) method. The herein developed o-MFML method is shown to be robust in cases of poor data quality in addition to actively lowering ML model error. Another degree of freedom that can be manipulated is the use of heterogeneous training data across fidelities. In several state of art techniques for multifidelity methods in ML-QC such as CQML and Δ-ML, the training data at the different fidelities involves the use of homogeneous or nested training data. That is, if a training molecular geometry is chosen at a higher fidelity then it is *de facto* included also in the lower fidelities. If this assumption is relaxed with the use of heterogeneous training data, one could potentially combine training data at several fidelities arising from different existing datasets in order to further reduce the cost associated with an ML model in QC. An assessment of this degree of freedom in this work reveals that the o-MFML model is robust even in cases of heterogeneous training data while the MFML model breaks down. Regardless, a key takeaway from the study on this degree of freedom in the multifidelity training data is to retain the nested structure of the data. Yet another degree of freedom that can be tuned is the amount of training data that is used across each fidelity used in the MFML model. This relates to changing the *scaling factor*, γ, which has conventionally been set to 2. That is, if N^f_{train} training samples are used at fidelity f, then at fidelity $f-1$, the number of training samples used would be $N^{f-1}_{\text{train}} := \gamma \cdot N^f_{\text{train}}$. The value of γ implicitly encodes the notion of sparsity of data as the fidelity of the data increases. In this dissertation, the tuning of γ results in the development of a novel error metric for multifidelity methods and subsequently in the introduction of the Γ-curve MFML method which is shown to be a high-accuracy low-cost multifidelity model. These three developments assist in addressing **O3** and provide to the ML-QC community a robust, low-cost, and high-accuracy multifidelity method, that is, the Γ-curve MFML. In all these developments, **O2** is imperatively fulfilled with the focus of model efficiency shifting from a purely training set size assessment to studying model error as a function of the time-cost incurred in generating training data at the different fidelities. It is shown that the Γ-curve

MFML method significantly reduces the cost in comparison to both single fidelity ML and conventional MFML method.

After sufficient methodological development and accomplishing the respective objectives, it becomes important to assess the standing of these newly developed methods vis-á-vis the existing standard of single fidelity ML and Δ-ML methods. The key indicator for this benchmarking is the time-cost for generating training data versus the model error. This metric for efficiency requires that the dataset used contains the information of time-cost for each fidelity. While there exist several multifidelity datasets, very few of them, if any, provide this information to the ML-QC community. In order to bridge this gap, a multifidelity benchmarking dataset, QeMFi (**Q**uantum Ch**e**mistry **M**ulti**Fi**delity) is produced as reported in Chapter 7. The QeMFi dataset contains several organic QC molecules with a diverse collection of QC properties. Moreover, the wall-time cost of each fidelity computed in the dataset is provided. This contributes greatly towards **O4**. With the benchmark dataset, since **O2** has been sufficiently accomplished, multifidelity models developed in this dissertation are systematically assessed and benchmarked for the time-cost incurred in generating the training data in order to predict several QC properties such as excitation energies and magnitude of molecular dipole moments. Within this framework, yet another multifidelity method, the MFΔML method is introduced and shown to be superior to Δ-ML [29] in cases where only few evaluations are to be made with ML models. In cases where several predictions are to be made using the trained ML model, MFML is shown to be far superior in terms of high-accuracy low-cost ML techniques.

1.4 Structure

This section presents to the reader the structure of the remainder of the dissertation. Chapter 2 provides a review of existing methods in both ML and QC relevant to the dissertation including the regression problem and molecular descriptors. This is followed by Chapter 3, which describes the state of art in multifidelity methods. Details on how fidelities can be combined are discussed in addition to their relevance to this dissertation. Key state of art multifidelity methods developed for ML-QC are also presented to complete the foundation for this thesis.

Part II of this thesis contains the compiled publications of the author. Since the methods section of multiple publications have overlaps, the methodological contributions have been collected into Chapter 4. This chapter is in its completeness the author's own contribution to the field of multifidelity ML methods in QC. The various methods are taken

almost as is from the original publications with the notations harmonized for this dissertation. Chapter 4 develops the MFML method and provides a framework for the methodological study of the degrees of freedom in this method resulting in o-MFML and the Γ-curve methods. In addition, a cross-validation scheme with the use of homogeneous or nested training data across multiple fidelities is also introduced.

Chapter 5 offers the reader the first application of MFML to QC, in particular to the prediction of excitation energies of arenes. The time-cost of generating training data versus the accuracy of the resulting ML model is introduced as a measure of model efficiency for assessment. Chapter 6 reports the use of o-MFML and MFML in the prediction of excitation energies and atomization energies of several molecules. The results indicate that o-MFML is a successor to the MFML method that is robust to poor quality of multifidelity data and results in a lower model error for prediction of QC properties.

Chapter 7 presents a benchmarking multifidelity dataset, QeMFi, consisting of five fidelities calculated with the TD-DFT formalism. The fidelities differ in their basis set choice: STO-3G, 3-21G, 6-31G, def2-SVP, and def2-TZVP. QeMFi offers to the community a variety of QC properties such as vertical excitation properties and molecular dipole moments, further including QC computation times allowing for a time benefit benchmark of multifidelity models for ML-QC. The chapter also discussed associated code scripts which can be readily used to benchmark MFML and o-MFML models on the QeMFi dataset. This work has been published as [41].

Chapter 8 assesses the use of non-nested (that is heterogeneous) training data configuration for MFML and o-MFML for the prediction of ground state energies and first vertical excitation energies of a diverse collection of molecules of the QeMFi dataset. Results indicate that the MFML method still requires a nested structure of training data across the fidelities. However, the o-MFML method shows promising results for non-nested multifidelity training data with model errors comparable to the nested configurations. This chapter has been published as ref. [42].

Chapter 9 compares the data costs associated with Δ-ML, MFML, and o-MFML in contrast with a newly introduced MultifidelityΔ-Machine Learning (MFΔML) method for the prediction of several QC properties from the QeMFi dataset. This assessment is made on the basis of training data generation cost associated with each model and is compared with the single fidelity KRR case. The results indicate that the use of multifidelity methods surpasses the standard Δ-ML approaches in cases of a large number of predictions. For cases, where Δ-ML method might be favored, such as small test set regimes, the MFΔML method is shown to be more efficient than conventional Δ-ML.

Chapter 10 investigates the data hierarchies for MFML and its effect on model efficiency and accuracy in the prediction of vertical excitation energies using the QeMFi dataset. A novel error metric, error contours of MFML, is proposed to provide a comprehensive view of model error contributions from each fidelity. The results indicate that high model accuracy can be achieved with just 2 training samples at the target fidelity when a larger number of samples from lower fidelities are used. This is further illustrated through a novel concept, the Γ-curve, which compares model error against the time-cost of generating training samples, demonstrating that multifidelity models can achieve high accuracy while minimizing training data costs. This chapter is published as ref. [43].

In Chapter 11, two practical applications of the herein developed MF methods is presented. First, the MF methods developed in this thesis are used to predict high accuracy energies of several small molecules. Secondly, the use of Γ-curve for the prediction of excitation energies for a system of 16 porphyrins on clay surface is discussed and shown to be highly efficient. Only the contributions of the author are detailed. Any datasets used or experiment designs that lie outside the ambit of the author's own contribution are appropriately cited. Finally, Chapter 12 provides conclusive remarks and future outlooks for the works presented in this dissertation.

Appendices are provided towards the end of the dissertation. Appendix A provides supplementary results which are taken from the Supplementary Information files of the publications from List of Publications. Appendix B provides selected QC methods for the sake of completeness of this thesis.

Part I

State of Art

CHAPTER 2

A PRIMER OF FUNDAMENTALS

That which can be asserted without evidence, can be dismissed without evidence.

— Christopher Hitchens

The ambit of this part of the thesis is to present the reader with brief details about the methodological developments that are the foundation of the work developed in the dissertation. The part is divided into two chapters. This chapter introduces fundamentals such as regression and Gaussian Process Regression while the forthcoming chapter in this part describes in detail the multifidelity approach, in particular with surrogate models (SM) such as ML. The rest of this chapter is structured as follows: section 2.1 describes key concepts such as molecular descriptors which are a key component of the ML-QC pipeline, converting the Cartesian coordinates and elemental species of the molecules into machine learnable input features. The concept of regression is introduced in section 2.2 with special focus on kernel based methods such as Kernel Ridge Regression (KRR) and Gaussian Process Regression (GPR). KRR and GPR are the ML models of choice for this dissertation. Details on the metrics used to evaluate ML models are provided in section 2.7 for completeness.

2.1 Molecular Descriptors

As presented in Chapter 1, ML in QC learns some property of interest by mapping molecular geometry and other physical details such as atomic number to the property. In the world of ML-QC frameworks, it is pertinent to first convert the Cartesian coordinates of the molecules into a machine learnable format, which are called *molecular descriptors* or *repre-*

sentations [44, 19, 45]. Generally speaking, molecular descriptors are required to be invariant under translation, rotation, and permutation of the molecule and constituent atoms [21]. In certain cases such as learning dipole moments, molecular descriptors are expected to be equivariant under rotation since the dipole moment is a QC property that depends on the orientation of the molecule itself [46, 47]. In addition, the uniqueness of the descriptor is considered to be a required trait [48]. Other desirable features include computational efficiency and universal applicability of the descriptor [22].

Several molecular descriptors are used in the ML-QC framework and the specific choice for application depends on the QC property and scope of application of the ML model. Since the central aim is to present multifidelity methods for ML and not to establish the best possible molecular descriptor, only two descriptors are utilized across the different chapters which are described below. For a comprehensive analysis of several molecular descriptors in predicting QC properties, the interested reader is directed to ref. [48] as a port of first entry.

Unsorted Coulomb Matrices (CM) [44, 49] are the molecular descriptors that are commonly used across this work. For a molecule, the entries of CM, C, are computed as

$$C_{i,j} := \begin{cases} \frac{Z_i^{2.4}}{2}, & i = j \\ \frac{Z_i \cdot Z_j}{\|\boldsymbol{R}_i - \boldsymbol{R}_j\|}, & i \neq j, \end{cases} \tag{2.1}$$

where the Cartesian coordinate of the i-th atom is $\boldsymbol{R}_i$ with Z_i being the atomic charge. The indices i, j run over the atoms of the molecule. Note that the CM are symmetric matrix representations, that is $C^T = C$. That is, only the upper triangular entries of the matrix C would be unique. Thus, for a molecule consisting of m atoms, the number of unique entries of the corresponding CM are $m(m+1)/2$. Notice that the size CM representation of a molecule depends on the number of atoms of the molecule. In cases where the CM representation is used to learn QC properties of molecules with different number of atoms such as the case in Chapters 7 and 8 the CM are padded by zeros to maintain a uniform size of the descriptor. For example, in Chapter 7 the largest molecule in the QeMFi dataset is o-HBDI with 22 atoms resulting in the padded CM size of 253 entries. Some literature uses a row-norm sorted CM as molecular representation where the rows of the CM descriptor are reordered such that:

$$\sum_j C_{1j} \geq \sum_j C_{2j} \geq \ldots \geq \sum_j C_{Nj},$$

where j indexes the atoms in the molecule. The row-norm sorted CM is avoided in most of this dissertation since this form of sorting is known to produce discontinuities in the repre-

sentation [44, 50]. However, in Chapter 10, row-sorted CM are used to assess transferability of the therein developed Γ-curve MFML model.

The second molecular representation that is used in this work is the Spectral London and Axilrod-Teller-Muto (SLATM) representation [51]. The theoretical foundation of this representation is the atoms-in-molecules formalism of QC [52]. Subsequently, the electronic density of atoms is considered in an ensemble fashion with the projection of the system charge density on internal degrees of freedom of the system. The derivation of this representation lies outside the scope of this work. However, the interested reader is directed to refs. [51, 21] for a comprehensive derivation of the SLATM representation.

2.2 The Regression Problem

This section is adapted from ref. [53] on regression. Consider the set of input features $\mathscr{X} \subset \mathbb{R}^d$ which is non empty, and a function $g : \mathscr{X} \to \mathbb{R}$, one has the training set $\mathscr{T} = (\boldsymbol{X}_i, y_i)_{i=1}^N \subset \mathscr{X} \times \mathbb{R},\ \forall N \in \mathbb{N}$ where $\boldsymbol{X}_i$ is the input feature with corresponding *target* y_i. One can collect the targets into a vector $\boldsymbol{y}$, and the input features into $\hat{\boldsymbol{X}}$ in order to rewrite the training dataset as $\mathscr{T} := (\hat{\boldsymbol{X}}, \boldsymbol{y})$.

The standard linear regression model is interested in approximating some function g such that $g(\hat{\boldsymbol{X}}) = \hat{\boldsymbol{X}}^T \boldsymbol{w}$, where $\boldsymbol{w}$ is the vector of weight (or parameters). Then the standard linear regression model is

$$\boldsymbol{y} = g(\hat{\boldsymbol{X}}) + \boldsymbol{\epsilon}, \tag{2.2}$$

where the noise is Gaussian with zero mean and σ_N^2 variance:

$$\boldsymbol{\epsilon} \sim \mathscr{N}(0, \sigma_N^2)\ . \tag{2.3}$$

Assuming this form of noise in the observations results in the following likelihood:

$$p(\boldsymbol{y}|\hat{\boldsymbol{X}}, \boldsymbol{w}) = \prod_{i=1}^{N} p(y_i|\boldsymbol{X}_i, \boldsymbol{w}) \tag{2.4}$$

$$= \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma_N}} \exp\left(-\frac{(y_i - \boldsymbol{X}_i^T \boldsymbol{w})^2}{2\sigma_N^2}\right) \tag{2.5}$$

$$= \frac{1}{(2\pi\sigma_N^2)^{N/2}} \exp\left(-\frac{1}{2\sigma_N^2 |\boldsymbol{y} - \hat{\boldsymbol{X}}^T \boldsymbol{w}|_1^2}\right) \tag{2.6}$$

$$= \mathscr{N}(\hat{\boldsymbol{X}}^T \boldsymbol{w}, \sigma_N^2 \boldsymbol{I}_N) \tag{2.7}$$

where the first step arises due to the independence assumption of training data. If one employs Bayesian formalism, that is, one enforces a *prior* over the weights $\boldsymbol{w}$, then

$$\boldsymbol{w} \sim \mathcal{N}(0, \Sigma_p)$$

that is, a zero mean Gaussian with Σ_p covariance matrix. The inference is made using Baye's rule as

$$p(\boldsymbol{w}|\boldsymbol{y}, \hat{\boldsymbol{X}}) = \frac{p(\boldsymbol{y}|\hat{\boldsymbol{X}}, \boldsymbol{w}) \cdot p(\boldsymbol{w})}{p(\boldsymbol{y}|\hat{\boldsymbol{X}})}, \tag{2.8}$$

where

$$p(\boldsymbol{y}|\hat{\boldsymbol{X}}) = \int p(\boldsymbol{y}|\hat{\boldsymbol{X}}, \boldsymbol{w}) \cdot p(\boldsymbol{w}) d\boldsymbol{w}$$

is called the marginal likelihood and is independent of $\boldsymbol{w}$.

Thus, Eq. (2.8) of the posterior uses the prior and the likelihood in order to use all information available about the weights. If one writes the posterior in terms of likelihood and prior which depend on the weights[1], we have

$$p(\boldsymbol{w}|\hat{\boldsymbol{X}}, \boldsymbol{y}) \propto \exp\left(-\frac{1}{2}(\boldsymbol{w} - \bar{\boldsymbol{w}})^T \cdot (\frac{1}{\sigma_N^2}\hat{\boldsymbol{X}}\hat{\boldsymbol{X}}^T + \Sigma_p^{-1}) \cdot (\boldsymbol{w} - \bar{\boldsymbol{w}})\right), \tag{2.9}$$

with $\bar{\boldsymbol{w}} := \sigma_N^{-2} \cdot (\sigma_N^{-2}\hat{\boldsymbol{X}}\hat{\boldsymbol{X}}^T + \Sigma_p^{-1})\hat{\boldsymbol{X}}\boldsymbol{y})$.

With this setup, the predictive distribution $g_q \equiv g(\boldsymbol{X}_q)$ for an unseen query input feature $\boldsymbol{X}_q$ is made as

$$p(g_q|\boldsymbol{X}_q, \mathcal{T}) = \int p(g_q|\boldsymbol{X}_q) \cdot p(\boldsymbol{w}|\mathcal{T}) d\boldsymbol{w} \tag{2.10}$$

$$= \mathcal{N}\left(\frac{1}{\sigma_N^2}\boldsymbol{X}_q^T\left(\frac{1}{\sigma_N^2}\hat{\boldsymbol{X}}\hat{\boldsymbol{X}}^T + \Sigma_p\right)^{-1}\boldsymbol{y}, \boldsymbol{X}_q\left(\frac{1}{\sigma_N^2}\hat{\boldsymbol{X}}\hat{\boldsymbol{X}}^T + \Sigma_p\right)^{-1}\boldsymbol{X}_q\right). \tag{2.11}$$

The prediction itself is a Gaussian distribution with the mean being related to the posterior mean of $\boldsymbol{w}$ and the query input. The variance of the predictive distribution is a quadratic terms which is dependent on the query input and posterior covariance.

2.3 Reproducing Kernel Hilbert Spaces

While there are several ways to approximate the regression function g from Eq. (2.2), one method is the use of kernel based methods such as Gaussian Process Regression or Kernel Ridge Regression. In order to understand how these methods function, this section

[1] Since the marginal likelihood is independent of $\boldsymbol{w}$

is a primer in the concept of Reproducing Kernel Hilbert Spaces (RKHS) and is primarily adapted from ref. [54]. A reproducing kernel k for an arbitrary Hilbert space $\mathscr{H}$, for some domain $\mathscr{X}$, is a function $k : \mathscr{X} \times \mathscr{X} \to \mathbb{R}$ such that

1. $k(\cdot, \boldsymbol{X}) \in \mathscr{H}\ \forall \boldsymbol{X} \in \mathscr{X}$, and
2. $f(\boldsymbol{X}) = (u, k(\cdot, \boldsymbol{X}))_{\mathscr{H}}\ \forall u \in \mathscr{H}, \boldsymbol{X} \in \mathscr{X}$.

A given $\mathscr{H}$ is said to be reproducing (that is, a reproducing kernel Hilbert space, RKHS) if there exists $k : \mathscr{X} \times \mathscr{X} \to \mathbb{R}$, that is, there is a reproducing kernel for the Hilbert Space. A given continuous k is said to be positive definite on $\mathscr{X} \subseteq \mathbb{R}^d$ if for all $n \in \mathbb{N}, \alpha \in \mathbb{R}^n \setminus \{0\}, \hat{\boldsymbol{X}} = \{\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n\}$ it is that $\sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j k(\boldsymbol{X}_i, \boldsymbol{X}_j) > 0$. Different kernel functions such as the Matérn kernel satisfy this requirement. These kernels are part of radial basis functions depicted as

$$k(\boldsymbol{X}, \boldsymbol{X}') := \varphi(\|\boldsymbol{X} - \boldsymbol{X}'\|)\,, \tag{2.12}$$

where $\varphi : [0, \infty) \to \mathbb{R}$. Consider the specific example of a Matérn kernel which is used in most applications in this dissertation. In the form of Eq. (2.12), this corresponds to setting

$$\varphi(x) \equiv \frac{K_{\chi - \frac{d}{2}}(x) \cdot x^{\chi - \frac{d}{2}}}{2^{\chi - 1} \cdot \Gamma(\chi)}, \chi > \frac{d}{2}\,,$$

where K_o is the Bessel function of second kind with order o, Γ is the gamma function, and χ is a positive parameter of covariance of the kernel.

Given a symmetric positive definite kernel k, one can construct for it a *native space* $\mathscr{N}_k(D)$ by completing the pre-Hilbert space $\mathscr{H}_k(\mathscr{X}) := \text{span}\{k(\cdot, \boldsymbol{X} | \boldsymbol{X} \in \mathscr{X}\}$. This native space can be shown to be a RKHS for k [54].

2.4 Gaussian Process Regression

The following section on Gaussian processes (GP) is adapted from refs. [53, 55]. Consider the positive definite kernel $k : \mathscr{X} \times \mathscr{X} \to \mathbb{R}$ and a real-valued function $m : \mathscr{X} \to \mathbb{R}$. Then, the function $G : \mathscr{X} \to \mathbb{R}$ is called a GP if for $\hat{\boldsymbol{X}} = (\boldsymbol{X}_1, \ldots, \boldsymbol{X}_N) \subset \mathscr{X}$ for some $N \in \mathbb{N}$,

$$G_{k,\hat{\boldsymbol{X}}} = (G(\boldsymbol{X}_1), \ldots, G(\boldsymbol{X}_N))^T \in \mathbb{R}^N$$

follows the multivariate Gaussian distribution with mean function m and covariance kernel k. that is, $G_{k,\hat{\boldsymbol{X}}} \sim \mathscr{N}(m_{\hat{\boldsymbol{X}}}, k_{\hat{\boldsymbol{X}},\hat{\boldsymbol{X}}})$ with

$$m_{\hat{\boldsymbol{X}}} := (m(\boldsymbol{X}_1), \ldots, m(\boldsymbol{X}_N))^T$$

and

$$k_{\hat{\boldsymbol{X}},\hat{\boldsymbol{X}}} := (k(\boldsymbol{X}_i, \boldsymbol{X}_j))_{i,j,=1}^{N} \in \mathbb{R}^{N\times N} .$$

The GP is often denoted as $GP(m,k)$. Ref. [55] shows that there is a one-one correspondence between $G \equiv GP(m,k)$ and the pair (m,k). It follows that

$$k(\boldsymbol{X},\boldsymbol{X}') = \mathbb{E}\left[(G(\boldsymbol{X}) - m(\boldsymbol{X}))\cdot(G(\boldsymbol{X}') - m(\boldsymbol{X}'))\right],\ \boldsymbol{X},\boldsymbol{X}' \in \mathscr{X} . \tag{2.13}$$

Gaussian process regression (GPR) or *kriging*, is a commonly used Bayesian method in solving regression problems such as that defined in section 2.2. The key task in GPR is the estimation of a posterior distribution for the unknown function g given three components:

(a) training dataset $\mathscr{T}$,

(b) a prior distribution Θ_0 defined as a GP

$$g \sim GP(m,k) , \tag{2.14}$$

where, m and k are chosen specific to the problem at hand and are influenced by the knowledge of the application domain, and

(c) a likelihood function,

$$l_{\hat{\boldsymbol{X}},\hat{\boldsymbol{X}}'}(g) = \prod_{i=1}^{N} \mathscr{N}\left(y_i, \sigma^2\right) , \tag{2.15}$$

where y_i is as presented in Eq. (2.2). The above is defined for the independent and identically distributed (i.i.d.)[2] noise variables $\epsilon_i \sim \mathscr{N}(0,\sigma^2)$ for $\sigma^2 > 0$ where the symbols are as expressed previously for Eq. (2.2).

The posterior distribution, $\Theta_N(g|\boldsymbol{X},\boldsymbol{X}')$ is given as

$$d\Theta_N(g|\boldsymbol{X},\boldsymbol{X}') \propto l_{\boldsymbol{X},\boldsymbol{X}'}(g)\, d\Theta_0(g) = \prod_{i=1}^{N} \mathscr{N}\left(y_i|g(\boldsymbol{X}_i),\sigma^2\right) d\Theta_0(g) . \tag{2.16}$$

it can be shown [53, 55, 57] that the posterior distribution is a GP denoted as $GP(\hat{m},\hat{k})$ with the mean function (also called posterior mean) for a query input feature $\boldsymbol{X}_q \in \mathscr{X}_q \subset \mathbb{R}^d$

$$\hat{m}(\boldsymbol{X}_q) = m(\boldsymbol{X}) + \boldsymbol{K}_{\boldsymbol{X}_q,\hat{\boldsymbol{X}}}\left(\boldsymbol{K}_{\hat{\boldsymbol{X}},\hat{\boldsymbol{X}}} + \sigma^2 \boldsymbol{I}_N\right)^{-1}(\boldsymbol{y} - m_{\hat{\boldsymbol{X}}}) , \tag{2.17}$$

[2] It should be noted that GPR does however, allow for non zero correlation between ϵ_i, ϵ_j $i,j, \in \{1,2,\ldots,N\}$ [56, 57].

and covariance function (also called posterior covariance function) given by

$$\hat{k}(\boldsymbol{X}, \boldsymbol{X}_q) = \boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X}_q) - \boldsymbol{K}_{\boldsymbol{X},\hat{\boldsymbol{X}}} \left(\boldsymbol{K}_{\hat{\boldsymbol{X}},\hat{\boldsymbol{X}}} + \sigma^2 \boldsymbol{I}_N \right)^{-1} \boldsymbol{K}_{\hat{\boldsymbol{X}},\boldsymbol{X}_q} , \tag{2.18}$$

where the symbols are as explained above. The posterior mean function is used to make predictions for an unseen query input feature $\boldsymbol{X}_q$, while the posterior covariance function is used in quantifying the uncertainty of the GPR model. The hyper-parameters such as the prior mean, prior covariance, and σ^2 are determined by standard hyper-parameter optimization frameworks such as grid searches or by maximizing the marginal likelihood [53]. Making the predictions with the GP over $\hat{\boldsymbol{X}}$ is often referred to as the inference. If one designates $\hat{\boldsymbol{g}}$ as the inferences, the joint distribution of the predictions and observations can be written as

$$\begin{bmatrix} \boldsymbol{y} \\ \hat{\boldsymbol{g}} \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} m \\ \hat{m} \end{bmatrix}, \begin{bmatrix} \boldsymbol{K}_{\boldsymbol{X},\boldsymbol{X}} + \sigma^2 \boldsymbol{I}_N & \boldsymbol{K}_{\boldsymbol{X},\hat{\boldsymbol{X}}} \\ \boldsymbol{K}^T_{\boldsymbol{X},\hat{\boldsymbol{X}}} & \boldsymbol{K}_{\boldsymbol{X},\hat{\boldsymbol{X}}} \end{bmatrix} \right) . \tag{2.19}$$

Further, the probability distribution of the predictions conditioned to the training data observations $\boldsymbol{y}$ is given as

$$\hat{\boldsymbol{g}} | \boldsymbol{y} \sim \mathcal{N} \left(\hat{m}, \hat{k} \right) \tag{2.20}$$

2.5 Kernel Ridge Regression

Kernel ridge regression (KRR) is a SM approach that is also called regularized least-squares or spline smoothing [58, 59]. KRR is the resulting SM when one performs regularized risk minimization over an RKHS (section 2.3) as a hypothesis space [60]. This can be formally stated as solving the optimization problem given a training dataset $\mathcal{T}$:

$$\hat{g} = \arg\min_{g \in \mathcal{H}} \frac{1}{N} \sum_{i=1}^{N} L(\boldsymbol{X}_i, y_i, g(\boldsymbol{X})) + \lambda \| g \|^2_{\mathcal{H}} , \tag{2.21}$$

where $\| \cdot \|_{\mathcal{H}}$ denotes the norm in the RKHS, $L : \mathcal{X} \times \mathbb{R} \times \mathbb{R}$ is a loss function which penalizes the difference between the real value y_i and prediction $g(\boldsymbol{X}_i)$ for input features $\boldsymbol{X}_i$, and $\lambda > 0$ is a constant. The term $\lambda \| g \|^2_{\mathcal{H}}$ is a regularization term that prevents overfitting.

On setting the loss function to be equivalent to the square loss, $L(\boldsymbol{X}, \hat{y}, y) = (\hat{y} - y)^2$, one arrives at a unique solution by employing the representer theorem [61, 62] to Eq. (2.21) given as

$$\hat{g}(\boldsymbol{X}_q) = \boldsymbol{K}_{\boldsymbol{X}_q,\hat{\boldsymbol{X}}} (\boldsymbol{K}_{\hat{\boldsymbol{X}},\hat{\boldsymbol{X}}} + N\lambda \boldsymbol{I}_N)^{-1} \boldsymbol{y} = \sum_{i=1}^{N} \alpha_i k(\boldsymbol{X}_q, \boldsymbol{X}_i), \boldsymbol{X}_i \in \hat{\boldsymbol{X}} , \tag{2.22}$$

where, α_i are often referred to as the coefficients of KRR.

Equivalence of GPR and KRR

On comparing Eq. (2.17) and Eq. (2.22), it can be readily deduced that GPR and KRR are equivalent, that is, $\hat{g} \equiv \hat{m}$ if $\sigma^2 = N\lambda$. In other words, under the condition of a specific variance σ^2, the posterior mean function of GPR is the solution to the KRR optimization problem. Loosely speaking, one can state that KRR is the expected value of a GPR.

Consider a training data set $\mathcal{T} := \{(\boldsymbol{X}_i, y_i)\}_{i=1}^{N_{\text{train}}}$ of size N_{train} with molecular descriptors or representations $\boldsymbol{X}_i$ and their corresponding QC properties, y_i, such as the first vertical excitation energy. The KRR model for the prediction of the excitation energies y_i for an unseen query descriptor $\boldsymbol{X}_q$ is denoted by

$$P_{\text{KRR}}\left(\boldsymbol{X}_q\right) := \sum_{i=1}^{N_{\text{train}}} \alpha_i k\left(\boldsymbol{X}_q, \boldsymbol{X}_i\right) , \tag{2.23}$$

where k is the kernel function. The unknown coefficient vector $\boldsymbol{\alpha}$ is trained by solving the linear system of equations $(\boldsymbol{K} + \lambda \boldsymbol{I}_{N_{train}})\boldsymbol{\alpha} = \boldsymbol{y}$, with $\boldsymbol{K} = \left(k(\boldsymbol{X}_i, \boldsymbol{X}_j)\right)_{i,j=1}^{N_{\text{train}}}$ the kernel matrix, $\boldsymbol{I}$ the identity matrix, $\boldsymbol{y} = \left(y_1, y_2, \ldots, y_{N_{\text{train}}}\right)^T$ the vector of QC properties, and λ a regularization parameter.

This work uses two kernels throughout. First, the Matérn Kernel of order 1 with the discrete L-2 norm

$$k\left(\boldsymbol{X}_i, \boldsymbol{X}_j\right) = \exp\left(-\frac{\sqrt{3}}{\sigma}\left\|\boldsymbol{X}_i - \boldsymbol{X}_j\right\|_2^2\right)\left(1 + \frac{\sqrt{3}}{\sigma}\left\|\boldsymbol{X}_i - \boldsymbol{X}_j\right\|_2^2\right) , \tag{2.24}$$

where σ denotes a length scale hyperparameter that determines the width of the kernel. This property is, in some sense, a measure of the degree of correlation associated with the training samples [63, 64, 65]. The second kernel used in this work is the Laplacian kernel given by:

$$k\left(\boldsymbol{X}_i, \boldsymbol{X}_j\right) = \exp\left(-\frac{1}{\sigma}\left\|\boldsymbol{X}_i - \boldsymbol{X}_j\right\|_1\right) \tag{2.25}$$

2.6 Cholesky Decomposition

By the construction, the kernel matrix $\boldsymbol{K}$, is symmetric, that is $\boldsymbol{K}^T = \boldsymbol{K}$. Further, it is positive definite. That is, $\forall \boldsymbol{X} \neq 0$,

$$\boldsymbol{X}^T \boldsymbol{K} \boldsymbol{X} > 0 .$$

This implies that the determinant of the kernel matrix is positive and all the principle proper sub-matrices of the kernel matrix also have positive determinants. Since we are interested in solving the system of linear equations presented in section 2.5, one is interested

in the factorization of the kernel matrix[3] given as:

$$\boldsymbol{K} = \boldsymbol{L}\boldsymbol{L}^T \,,$$

where $\boldsymbol{L}$ is a lower triangular matrix that is called the Cholesky factor of $\boldsymbol{K}$ and the right hand side of the equation is the corresponding Cholesky decomposition. if the diagonal elements of the factor are strictly positive, then the factorization is unique. once the Cholesky decomposition is performed, the solution to the system of linear equations can be written as

$$\boldsymbol{K}\boldsymbol{\alpha} = \boldsymbol{L}(\boldsymbol{L}^T\boldsymbol{\alpha}) = \boldsymbol{L}\boldsymbol{z} = \boldsymbol{y} \,, \tag{2.26}$$

where $\boldsymbol{z} = \boldsymbol{L}^T\boldsymbol{\alpha}$. Therefore, one can solve first for $\boldsymbol{z}$ using Eq. (2.26) and subsequently solve for $\boldsymbol{\alpha}$ with $\boldsymbol{L}^T\boldsymbol{\alpha} = \boldsymbol{z}$.

The work presented in this dissertation uses the qmlcode package [66] to perform the Cholesky decomposition and solve the linear system of equations to arrive at the values of α_i from Eq. (2.23). For a detailed description on the algorithm and derivation of the Cholesky method, the interested reader is directed to ref. [67].

2.7 Machine Learning Model Evaluation Metrics

In order to assess the ML models studied in this thesis, certain error metrics discussed below are employed. Learning curves are a well known metric in the field of KRR-based ML methods. These depict the change in prediction error of the ML model for increasing training set size. Throughout this dissertation, all prediction errors have been reported on a test set, $\mathcal{V}^F_{\text{test}} := \{(\boldsymbol{X}^{\text{ref}}_q, y^{\text{ref}}_q)\}^{N_{\text{test}}}_{q=1}$, which consist of evaluation representations and their corresponding reference values for property of interest (for example, excitation energy) calculated at the target fidelity F (for example, TZVP). These errors are reported mostly as Mean Absolute Errors (MAE) which are defined by a discrete L_1 norm

$$\text{MAE} = \frac{1}{N_{\text{test}}} \sum_{q=1}^{N_{\text{test}}} \left| P_{\text{ML}}\left(\boldsymbol{X}^{\text{ref}}_q\right) - y^{\text{ref}}_q \right| \,. \tag{2.27}$$

The model P_{ML} can either be identified by the standard single fidelity KRR model or by the various multifidelity models discussed in this dissertation. In some cases, such as the prediction of excitation energies of diverse molecules in Chapter 10, relative MAE (RMAE) is used since excitation energies are system specific. Relative error measures eliminate this

[3] Since we also use Lavrentiev regularization, w.l.o.g. we use the term kernel matrix to imply $\boldsymbol{K} + \boldsymbol{I}\lambda$.

system effect on the ML model assessment in such cases. RMAE is computed over the hold-out test set as

$$\text{RMAE} = \frac{1}{N_{\text{test}}} \sum_{q=1}^{N_{\text{test}}} \left| \frac{P_{\text{ML}}\left(\boldsymbol{X}_q^{\text{ref}}\right) - y_q^{\text{ref}}}{y_q^{\text{ref}}} \right| . \tag{2.28}$$

CHAPTER 3

MULTIFIDELITY MODELS

The poet only asks to get his head into the heavens. It is the logician who seeks to get the heavens into his head. And it is his head that splits.

—G. K. Chesterton, *Orthodoxy*

In the previous chapter, several fundamental to ML in QC were introduced such as molecular descriptors and kernel based regression such as KRR. This chapter intends to introduce to the reader the second segment of literature, namely, multifidelity methods. Multifidelity methods in numerical simulations deal with combining simulation data across different fidelities. In this chapter, firstly, the concept of high and low fidelity models is discussed. A primer on categorizing these models is presented along with details on how such models can be combined is delineated. Multifidelity methods are then discussed in detail with focus on a recursive GP approach based on refs. [39, 40]. Several multifidelity methods such as Δ-ML [29], CQML method [32], and multi-task GPR are also presented. This is followed by an examination of application of MF methods in literature, in particular to quantum chemistry (QC). Throughout the chapter, boxes are provided to indicate where in the dissertation the specific method is further developed for the MF methods of this dissertation.

In several areas of science where computational methods are used, different *numerical models* can be used to study a given system of interest [39]. A numerical model, as the name suggests, simulates the system of interest *in silico*, numerically. A resource extensive numerical model, also called a HF model, usually results in a high accuracy description of the system. In contrast, a LF model results in a less accurate description while also being less computationally expensive. MF methods is an umbrella term to cover the approaches

which combine outputs of HF and LF models to approximate the output of the former [68]. Mathematically, one can denote a numerical model as a mapping between some inputs and an output. Then the HF and LF fidelity models can be defined as

$$g_h : \mathscr{X} \to \mathscr{Y} \ ,$$

and

$$g_l : \mathscr{X} \to \mathscr{Y} \ ,$$

respectively, which map some input $\boldsymbol{X} \in \mathscr{X} \subseteq \mathbb{R}^M$ to the output $\boldsymbol{y} \in \mathscr{Y} \subseteq \mathbb{R}^N$. The model g_h has an associated cost $c_h \in \mathbb{R}_+$ for a single *realization*[1], while the LF model g_l has a cost $c_l \in \mathbb{R}_+$. By assumption, $c_h > c_l$, thereby giving rise to the concept of HF and LF. Since there can be several LF models, one can consider the general LF models $g_l^{(f)}$ for $f \in \{1,2,\ldots,F-1\}$ with corresponding cost $c_l^{(f)}$ such that $f = F$ denotes the HF model. For ease of writing the notation, hereon, the HF model is denoted as $g_{(F)}$ and the LF models as $g_{(f)}$ $\forall f \in \{1,2,\ldots,F-1\}$.

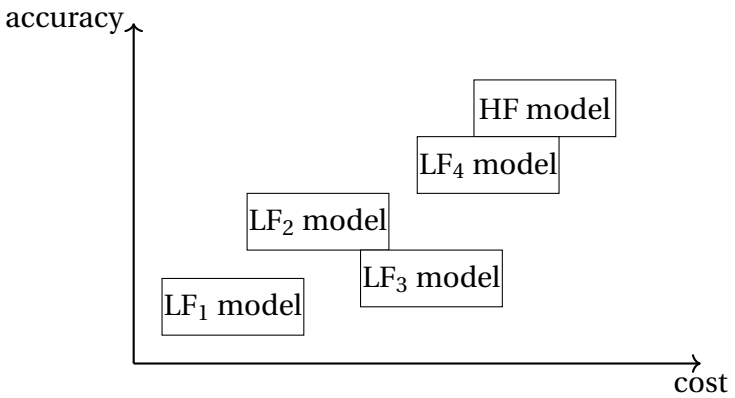


Figure 3.1: A hypothetical HF and LF model graphic depicting the cost and accuracy of the models. Often, the different LF models themselves can be diverse for cost and accuracy. MF methods employ the use of LF models to achieve HF accuracy at a lower cost.

Figure 3.1 provides a graphical layout of a hypothetical case of cost versus accuracy of HF and LF models in order to elucidate the concept of fidelity in numerical models. Although not to scale, one can see more intuitively what the concept of fidelity means. The model LF_1 is the cheapest numerical model which is also the least accurate. On the other hand, the model HF denotes the highest fidelity model with a high cost and a high accuracy. In between the least accurate and most accurate numerical models, lie several other fidelity models indicates by LF_f. It is often the case that some numerical model is more accurate than another even though it is cheaper, such as is the case for LF_2 and LF_3

[1] That is, making a single point calculation with the model.

in this hypothetical setup. In such a setup, usually the less accurate model is considered to be the lower fidelity [28].

In multifidelity methods, one is interested in being able to meaningfully combine the cheaper and less accurate LF models to achieve the accuracy of the HF model at a cumulative cost of making one realization of the model being lower than c_h. Within the collection of LF models itself, as shown in the example from Figure 3.1, there can be some which are costlier than the others, thereby forming a hierarchy of fidelities. Thus the first step in using MF methods, is establishing the hierarchy of fidelities. That is, ascertaining which is more accurate (and costly) than the others. In some cases, there is a direct relation between some parameter and the fidelity of the resulting model. More often, a clear hierarchy between the models is not visible and depends mostly on the context, aim, and physical property that is being modeled [68].

3.1 Categorizing Low Fidelity Models

At this juncture, it is interesting to note that the low fidelity models can be broadly categorized into the following briefly discussed types.

Model simplification

This corresponds to change in the mathematical model that is used to generate the data. That is, *simplifying the model* that is used to study the phenomena of interest [68, 28]. This is usually carried out by reducing the complexity of the PDE that governs the process. An example of such LF models is when non-linear aspects of the modeling are ignored [69].

Several MF methods with fidelity being defined by SM complexity have been applied to reduce the cost of sensitivity analysis with a demonstrated speed up of about 10× over single fidelity methods in application to the Ishigami function [70, 71] which is a fundamental benchmark test case for sensitivity analysis. Rijn *et al.* set up a comparison between the co-kriging method, a co-radial basis function (RBF) approach, and a new co-Random Forest method. They establish that the latter is faster with close to no difference in the accuracy [72].

Optimal mixture of substrates in biogas has been modeled with co-kriging in ref. [73]. Therein, the the HF data is generated with a full-scale simulation of the biogas plant while the HF data is generated with a faster and less accurate bio-methane potential model. Since kriging models are best suited for continuous problems [74] and the optimization search space for the modeling of biogas plants is discontinuous, the work implements a two-layer

modeling approach wherein two kriging models are simultaneously created which the optimization procedure then switches between on the detection of a discontinuity. This approach is shown to be highly effective for such an application.

Reduced dimensionality of the problem

This approach deals with tuning the accuracy with which one solves the PDE governing the physical phenomena, usually by manipulating grid size or time steps [28]. These are referred to as *projection based LF models* and are results of employing mathematical techniques to reduce the dimension of the problem structure as opposed to using the knowledge of the problem domain [75]. Another example is ref. [76], where Molléro *et al.* propose a MF approach in designing a cardiac electromechanical model. In this case, the LF model is a 0-D (dimension) model of the cardiac system while the HF is the 3-D model of the cardiac system. Parameters of the cardiac model are estimated using the 0-D model and shown to transfer to the 3-D model resulting in a fast and accurate overall model for the cardiac system benchmarked on over 100 different cardiac geometries.

Experimental and simulated

Experimental data versus simulated data is another categorization of HF and LF data, often shortly referred to as the *source categorization*. Perdikaris *et al.* utilize MF information fusing with GPs to reduce high dimensional problems with high number of training data to linear algorithmic complexity [77]. The method is benchmarked on several use-cases including water transport in boreholes and solving the stochastic Helmholtz equation in 100 dimensions.

Data-fit models

The *data fit LF models* are the another kind of categorization which offer the flexibility of having a *black-box* HF model. This is due to the premise that one is only interested in the input and output of the HF model and not how it arrives at that relation. Data fit models are usually derived as linear combinations of basis functions with the coefficients being fit based on interpolation or regression of inputs and outputs from the HF models [68]. Different kinds of such interpolation and regression are possible and include radial basis functions interpolation with kriging [53, 78].

For instance, in ref. [79], a reinforcement learning framework with MF models is proposed and implemented on remote controlled cars to test self driving capabilities of such

models. The proposed workflow for such approaches involves the transfer learning of parameters from a LF model to a HF model without needing to fully train the latter, thereby reducing computation cost.

In ref. [80], a statistical approach to build a multifidelity SM is proposed with an alternative definition for fidelity being offered for cases where clear hierarchy cannot be achieved. In particular, the definition of fidelity is presented in the form of a *variance metric* which is defined based on the area of application of the surrogate model. The proposed approach is finally used to minimize the drag coefficient for an airfoil with the use of GPs.

In another example, ref. [81] on the other hand, offers a Markov random fields framework for MF co-kriging resulting in a computationally efficient ML approach and employ it for uncertainty quantification of the Burger's equation in fluid dynamics problems. MF importance sampling methods combining outputs of HF and LF models have been used to simulate non-linear time-dependent problems such as deflection of a clamped plate, Burgers' equation, and simulation of acoustic horn with the Helmholtz equation [82]. The work showed that the MF method provides a significant speed up over the single HF Monte Carlo approach.

Furthermore, a NN based MF approach has previously been used in functional approximation and identifying unknown parameters in PDEs [83]. Deep neural network (DNN) and Physics Informed Neural Network (PINN) methods are implemented to approximate functions of both continuous and discontinuous kinds. The same NN architectures are then employed in solving the inverse PDE problem with non-linearity. One such application is to learn the hydraulic conductivity of for non-linear unsaturated flows in a 1-D water column. The MF PINN is shown to be more accurate in the prediction of hydraulic conductivity [83].

Similarly, MF Monte Carlo (MC) strategies have been implemented in the accurate prediction of wildfire spread [84] and shown to be effective to this end. More recently, deep convolutional neural networks have been used along with a MF training data strategy to achieve efficient estimation of the distribution of a quantity of interest for parabolic PDEs of multi-phase flow [85]. On the other hand, Xu *et al.* [86] leverage MF learning and bandit learning [87] to perform fine element approximation of parametric PDEs. Such MF models have also been used to solve the PDE for heat transport [88]. Physics guided ML (PGML) along with MF information fusion have been used in order to predict boundary layer flow using NNs [89].

3.2 Combining Fidelities

A *surrogate model* (SM) is a data driven model that approximates the outcome of a numerical model. ML methods such as GPR and KRR are examples of SMs. While the term SM is a broader categorization of statistical and data driven models, in this chapter, they are used synonymously with ML models.

In multifidelity methods for SMs, the central aim of combining the fidelities is to utilize the characteristics of each of them to arrive at a low-cost high-accuracy SM. For the most part, the study of MF methods in literature has been with its demonstration using two fidelities, ergo a *bi-fidelity* approach, although certain cases do show the extension of such methods to more than two fidelities [28]. Bi-fidelity SMs use data from HF models to make LF models achieve the accuracy of the HF. There are three common archetypes of combining bi-fidelity data using SMs which are discussed below with ref. [28] as an anchor point. While there is a fourth, space mapping, this is omitted due to relevance and the interested reader is referred to ref. [28] for details. Godino classifies the integration of bi-fidelity data in surrogate modeling into two main categories: multifidelity SMs and MF hierarchical models (MFHM) [28]. Bi-fidelity SMs such as co-kriging use data driven surrogates to 'raise' the LF data to the accuracy of the HF. MFHMs, on the other hand, combine fidelities without building a surrogate architecture for MF but rather using optimization specific to the problem domain. Importance sampling is one example of MFHMs. Although MFHMs are actively considered as methods to combine multifidelity data, these methods are not discussed in this dissertation due to relevance. The interested reader is directed to refs. [90, 68, 28] for more details. Below, the approaches related to bi-fidelity SMs is discussed. In particular, the combination of data for a bi-fidelity structure is presented in this section. That is, approaches involved in combining one LF model and one HF model[2]. The section that follows will discuss details on how several fidelities can be recursively combined.

3.2.1 Additive correction

As the name suggests, these methods use additive corrections to improve the accuracy of a LF model. Additive correction is expressed as

$$\hat{y}^{(F)}(\boldsymbol{X}) = y^{(f)}(\boldsymbol{X}) + \Delta(\boldsymbol{X}) \,, \tag{3.1}$$

[2]It remains a matter of semantics to refer to bi-fidelity models as multifidelity models. This decision is left to the reader to make.

where $\Delta(\boldsymbol{X})$ is the additive correction model that bridges the LF model to the HF model accuracy. Such forms of corrections have been used in several cases, for instance, in aerodynamic optimization and other applications of fluid dynamics [91, 92].

In refs. [93, 94] a bi-fidelity additive correction is carried out in order to solve a diffusive optical tomography problem. The fidelities are differentiated on the basis of coarseness of the grid employed. In contrast, ref. [95] uses the simplification of the model as a measure of fidelity. In all these cases, the additive terms is modeled using a Bayesian approximation.

> The additive correction technique from Eq. (3.1) is used in the Δ-ML [29] model which forms a major component of the MF ML models that are discussed in this work. More details are presented in section 3.4.1.

3.2.2 Multiplicative correction

The multiplicative correction for multifidelity SMs is expressed as

$$\hat{y}^{(F)}(\boldsymbol{X}) = \rho(\boldsymbol{X}) \cdot y^{(f)}(\boldsymbol{X}) \,, \tag{3.2}$$

with the term $\rho(\boldsymbol{X})$ denoting the multiplicative correction factor which in some sense is the ratio of the output of the HF and LF models. This type of correction has previously been used in aerodynamic optimization [91, 96].

Ref. [96] utilizes a multiplicative correction between fidelities in order to perform aerodynamic optimization. Similar tasks have been demonstrated in ref. [97] for wing-bending simulations in aerospace engineering. Ref. [98] utilizes the multiplicative correction method in order to solve optimization problems. Several other works in literature perform and assess multiplicative correction for a bi-fidelity model with applications ranging from surface optimization to defect detection [99, 100, 101]. The keen reader is referred to the surveys of Godino and Peherstorfer for more details [68, 28].

3.2.3 Comprehensive correction

When both additive and multiplicative corrections are used simultaneously to combine MF data, it is considered to be a comprehensive correction method. Formally,

$$\hat{y}^{(F)}(\boldsymbol{X}) = \rho(\boldsymbol{X}) \cdot y^{(f)}(\boldsymbol{X}) + \Delta(\boldsymbol{X}) \,. \tag{3.3}$$

In such an approach, a common simplification that is made is to consider $\rho(\boldsymbol{X}) \equiv \rho \in \mathbb{R}$ being a constant. Such a framework has been employed in design optimization, accurate

simulations of a fluidized-bed process using linear regression, and uncertainty quantification in fluid dynamics by studying the Burgers equation [102, 90, 103].

Another flavor of comprehensive correction given as

$$\hat{y}^{(F)}(\boldsymbol{X}) = w(\boldsymbol{X}) \cdot \rho(\boldsymbol{X}) \cdot y^{(f)}(\boldsymbol{X}) + (1 - w(\boldsymbol{X})) \cdot [y^{(f)}(\boldsymbol{X}) + \Delta(\boldsymbol{X})] \,, \tag{3.4}$$

was introduced in ref. [104] for the design of supercritical high lift airfoil. Here, $w(\boldsymbol{X})$ is a weight function which informs which data point is more important while training the SM. This method has since been employed in several applications ranging from solid mechanics to aerodynamic calculations in flight simulations [105, 106, 107].

3.3 Multifidelity Gaussian Processes

As has been discussed in the earlier stages of this chapter, there can exist several LF models of differing accuracy and cost leading to a hierarchy of fidelities. One can consider an ordered hierarchy of fidelities, $f \in \{1, 2, \ldots, F\}$. Given this, the main objective of this section is to derive the essentials that indicate how multiple fidelities can be combined into a multifidelity SM model. In particular, one can focus on the GPR as the SM of choice.

3.3.1 Simple Auto-regressive Model

Here, the simple auto-regressive model presented in ref. [39] is delineated. This multifidelity model builds a GP at fidelity f by using a comprehensive corrected GP model from fidelity $f-1$. Consider again the ordered hierarchy of multiple LF models denoted as $\left(g_f(\boldsymbol{X})\right)_{f=1}^{F}$ with the HF model being identified with $f = F$. Each of these is now being modeled by a GP, $\left(G_f(\boldsymbol{X})\right)_{f=1}^{F}$, for increasing fidelity with increasing f and $\boldsymbol{X} \in \mathscr{X} \subset \mathbb{R}^d$ representing the input features and $\mathscr{X}$ being the input parameter space. With this, the auto-regressive model for $1 < f \leq F$ can be constructed considering a comprehensive correction of the fidelities as from Eq. (3.3)

$$\begin{cases} G_f(\boldsymbol{X}) = \rho_{f-1}(\boldsymbol{X}) G_{f-1}(\boldsymbol{X}) + \Delta_f \boldsymbol{X} \,, \\ G_{f-1}(\boldsymbol{X}) \perp \Delta_f(\boldsymbol{X}) \,, \\ \rho_{f-1}(\boldsymbol{X}) = \boldsymbol{l}_{f-1}^T(\boldsymbol{X}) \boldsymbol{\alpha}_{\rho_{f-1}} \,. \end{cases} \tag{3.5}$$

In the above equation $\boldsymbol{l}_f$ is a vector of n_{f-1} regression functions,

$$\Delta_f(\boldsymbol{X}) \sim GP\left(\boldsymbol{p}_f^T(\boldsymbol{X}) \boldsymbol{\alpha}_f, \sigma_f^2 k_f(\boldsymbol{X}, \boldsymbol{X}')\right) \,, \tag{3.6}$$

and

$$G_1(\boldsymbol{X}) \sim GP\left(\boldsymbol{p}_1^T(\boldsymbol{X})\boldsymbol{\alpha}_1, \sigma_1^2 k_1(\boldsymbol{X}, \boldsymbol{X}')\right) . \tag{3.7}$$

Further $\boldsymbol{p}_f$ is s vector of m_f regression functions, $k_f(\cdot,\cdot)$ is a correlation function, $\boldsymbol{\alpha}_f$ is a vector of m_f dimensions whereas $\boldsymbol{\alpha}_{\rho_{f-1}}$ is a vector of dimension n_{f-1}, and $\sigma_f^2 \in \mathbb{R}_+$. Furthermore, since there is an underlying assumption of a Markov property, it is that given $\boldsymbol{X} \in \mathscr{X}$, if $G_{f-1}(\boldsymbol{X})$ is known, then there is no further information to be gained about $G_f(\boldsymbol{X})$ from $G_{f-1}(\boldsymbol{X}')$ provided $\boldsymbol{X}' \neq \boldsymbol{X}$. This implies [40]

$$\rho_{f-1}(\boldsymbol{X}) = \frac{\operatorname{cov}\left(G_f(\boldsymbol{X}), G_{f-1}(\boldsymbol{X})\right)}{\operatorname{var}\left(G_{f-1}(\boldsymbol{X})\right)} . \tag{3.8}$$

However, in the applications reported in ref. [39], the parameters $\rho_{f-1}\ \forall f \in \{2,\ldots,F\}$ were treated as constants. On the other hand, ref. [40] makes a case for the consideration of non-constant values of ρ_{f-1}.

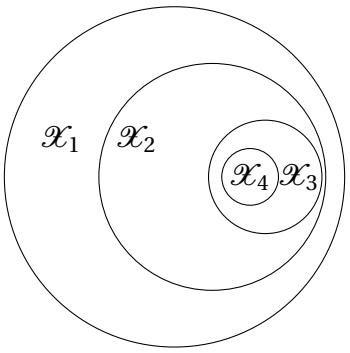


Figure 3.2: A hypothetical parameter input space $\mathscr{X}$ for 4 fidelities showing nested property of the experiment design for the models of different fidelities.

One can now consider the Gaussian vector, $\mathscr{G}^{(F)} = \left(\mathscr{G}_1^T, \ldots, \mathscr{G}_F^T\right)^T$, which contains the evaluation of the random processes defined by the models of different fidelities, $(G_f(\boldsymbol{X}))_{f=1}^F$. These are now evaluated at the points within the finite subsets of $\mathscr{X}$ given as $\mathscr{X}_F \subseteq \mathscr{X}_{F-1} \subseteq ,\ldots, \subseteq \mathscr{X}_1$. This concept is graphically depicted in Figure 3.2. The different circles indicate the input space of the corresponding fidelities. The size of the circles denote the cardinality of the space. As one increases the fidelity of data, the cost increases, thus it is anticipated that the size of the circles also reduces. Notice that the nested fashion of the input space ensures that if some $\boldsymbol{X} \in \mathscr{X}_4$ is chosen, it is also that this $\boldsymbol{X} \in \mathscr{X}_3$, $\boldsymbol{X} \in \mathscr{X}_2$, and $\boldsymbol{X} \in \mathscr{X}_1$. This property of designing the model calculations at different fidelities is called the nested property and is not a strictly necessary pre-requisite for the MF method. However this restriction is often implemented since it allows for easier estimation of model parameters. In Chapter 8, the nested property is assessed for the prediction of several QC properties.

While for the most part, MF methods require homogeneous data on both fidelities, several works in this field have dealt with heterogeneous data structures. Sarkar *et al.* present a MF approach to solving computational fluid dynamics with heterogeneous domains using heterogeneous transfer learning [108]. Further reading on this matter can be found in the detailed survey of this method in ref. [109].

The effect of using nested and non-nested data for MF methods of ML in QC is studied in Chapter 8. While the results indicate that nested training data is preferred, the o-MFML method developed in Chapter 4 and discussed in Chapter 6 results in favorable outcomes even for non-nested training data.

3.3.2 Recursive Multifidelity Model

While the auto-regressive model described above performs a comprehensive correction for the GP model at fidelity $f-1$ in order to 'raise' it to fidelity f, the recursive multifidelity model described here based on ref. [40] does so recursively. The key difference therefore is that the multifidelity model at fidelity F is built up recursively up from fidelity $f=1$ with appropriate comprehensive correction being applied. That is, the LF model as $f-1$ is not computed using a numerical solver but rather is in of itself a multifidelity model built on comprehensively corrected multifidelity model of $f-2$ and so on. In other words, the recursive MF model described in ref. [40] expresses the GP at some fidelity f as a function of the GP $G_{f-1}(\boldsymbol{X})$ which is conditioned on $\boldsymbol{g}_{(f-1)} = (g_{(1)},\dots,g_{(f-1)})$ at input points taken from the input spaces $(\mathscr{X}_i)_{i=1}^{f-1}$ while still assuming the nested property. Formally,

$$\begin{cases} G_f(\boldsymbol{X}) = \rho_{f-1}(\boldsymbol{X})\hat{G}_{f-1}(\boldsymbol{X}) + \Delta_f(\boldsymbol{X}) \,, \\ \hat{G}_{f-1}(\boldsymbol{X}) \perp \Delta_f(\boldsymbol{X}) \,, \\ \rho_{f-1}(\boldsymbol{X}) = \boldsymbol{l}_{f-1}^T(\boldsymbol{X})\boldsymbol{\alpha}_{\rho_{f-1}} \,. \end{cases} \tag{3.9}$$

Here any $\hat{G}_f$ for $1 < f \leq F$ is a GP with a distribution given as

$$\left[G_f(\boldsymbol{X})|\mathscr{G}^{(f)} = g^{(f)}, \boldsymbol{\alpha}_f, \boldsymbol{\alpha}_{f-1}, \sigma_f^2\right] \sim \mathcal{N}\left(\mu_{G_f}(\boldsymbol{X}), s_{G_f}^2(\boldsymbol{X})\right) . \tag{3.10}$$

The mean of the Gaussian distribution in the above equation can be explicitly stated as

$$\mu_{G_f}(\boldsymbol{X}) = \rho_{f-1}(\boldsymbol{X})\mu_{G_{f-1}}(\boldsymbol{X}) + \boldsymbol{p}_f(\boldsymbol{X})^T\boldsymbol{\alpha}_f + \boldsymbol{k}_f(\boldsymbol{X})^T\boldsymbol{K}_f^{-1}\left(g_f - \rho_{f-1}(\mathscr{X}_f)\bigodot g_{f-1}(\mathscr{X}_f) - \boldsymbol{P}_f\boldsymbol{\alpha}_f\right) , \tag{3.11}$$

with $\boldsymbol{K}_f$ being the matrix whose elements are $\left(k_f(\boldsymbol{X}_i,\boldsymbol{X}_j)\right)_{\boldsymbol{x_i},\boldsymbol{x_j}\in\mathscr{X}_f}$ and $\boldsymbol{P}_f$ is the matrix containing $\boldsymbol{p}_f(\boldsymbol{X})$ evaluated on $\mathscr{X}_f$. Further, one has that the ith element of the vector $\boldsymbol{k}_f(\boldsymbol{X})$

is given as $(k_f(\boldsymbol{X}, \boldsymbol{X}_i))_{\boldsymbol{X}_i \in \mathscr{X}_f}$. This mean is the SM at fidelity f for $1 \leq f \leq F$. The variance of the normal distribution from Eq. (3.10) is expressed as

$$\sigma^2_{G_f}(\boldsymbol{X}) = \rho^2_{f-1}(\boldsymbol{X})\sigma^2_{G_{f-1}}(\boldsymbol{X}) + \sigma^2_f \left(1 - \boldsymbol{k}_f(\boldsymbol{X})^T \boldsymbol{K}_f(\boldsymbol{X})^{-1}\right) . \tag{3.12}$$

This variance can be understood as the mean squared error of the SM defined in Eq. (3.11). Since the mean and variance of the GPR are represented recursively, that is, that at fidelity f is expressed in terms of those from fidelity $f-1$, Eq. (3.9) is the recursive MF-GPR model. If the assumption of nested property of $\mathscr{X}_f$ holds, then it can be shown [40] that

$$\mu_{G_f}(\boldsymbol{X}) \equiv m_{G_f}(\boldsymbol{X})$$

and

$$\sigma^2_{G_f}(\boldsymbol{X}) \equiv s^2_{G_f}(\boldsymbol{X}) .$$

As ref. [40] argues, a benefit of the recursive model over the standard auto-regressive model is that once the MF model is built, it can be used to evaluate the SM, $g_f(\boldsymbol{X})$, for all $f \in \{1, \ldots, F\}$. Furthermore, the recursive MF model has a lower computational complexity than the auto-regressive model.

The MFML and o-MFML model developed in this thesis are a formulation of the recursive MF model. In particular, MFML corresponds to setting $\rho_{f-1} \equiv 1$ in Eq. (3.9) and $\Delta_f(\boldsymbol{X}) \equiv G_f(\boldsymbol{X}) - G_{f-1}(\boldsymbol{X})$ with restrictions on the number of training samples to be used to generate each of these SMs. By formulation of the nested training set, in MFML, the orthogonality requirement of Eq. (3.9) is satisfied. This will be detailed in Chapter 4. In o-MFML, the values of ρ_{f-1} are optimized over a validation set which results in a more robust way to combine the different fidelities.

3.3.3 Multi-Task Gaussian Process Regression

Multi-task (MT) methods are another flavor of MF-SM where several regression tasks, say M tasks, are solved simultaneously with an assumption that these tasks are related. This can be used to learn QC properties at several fidelities in a related manner, as is done in ref. [110] for instance. The assumption of relatedness between the tasks is made with care such that learning several tasks simultaneously does not affect the accuracy of the model [111, 112]. In this section, the details of MTGPR are derived along the lines of ref. [113] where inter-task dependencies are learned without the use of task-descriptor features [114,

115]. This form of MTGPR is the most common approach where no explicit relation between the tasks is provided but the algorithm learns the relation purely based on the observation data for each task.

Consider again, N inputs collected in $\hat{\boldsymbol{X}} = \{\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_N\}$ with corresponding outputs given as $\boldsymbol{y} = (y_{11}, \ldots, y_{N1}, \ldots, y_{12}, \ldots, y_{1M}, \ldots, y_{NM})^T$ where each y_{nm} is the output for the mth task of the nth input. One can collect these as a $N \times M$ matrix given as $\boldsymbol{Y}$ with $\boldsymbol{y} \equiv \mathrm{vec}\,\boldsymbol{Y}$. Assuming a zero mean GP prior over the true value functions of the tasks, $\{g_m\}$, for the mth and m'th task, it is that

$$\langle g_m(\hat{\boldsymbol{X}}) g_{m'}(\hat{\boldsymbol{X}}')\rangle = K^g_{m,m'} k_{\hat{\boldsymbol{X}}}(\hat{\boldsymbol{X}}, \hat{\boldsymbol{X}}') \,, \tag{3.13}$$

and

$$y_{i,m} \sim \mathcal{N}(g_m(\hat{\boldsymbol{X}}_i), \sigma^2_m) \,.$$

Here, $\boldsymbol{K}^g$, is the positive semi-definite matrix specifying similarities between tasks, k_X is the covariance function of the inputs, and σ^2_m is the variance for noise in the mth task. Since there is no specification of the inter-task correlation, the matrix $\boldsymbol{K}^g$ is also learned during the training process. The mean prediction of the mth task, $\hat{g}_m$, using MTGPR for a query input $\boldsymbol{X}_q$ is given as

$$\hat{g}_m(\boldsymbol{X}_q) = \left(\boldsymbol{k}^g_m \bigotimes \boldsymbol{k}_{\boldsymbol{X},\boldsymbol{X}_q}\right)^T \boldsymbol{\Lambda}^{-1} \boldsymbol{y} \,, \tag{3.14}$$

with

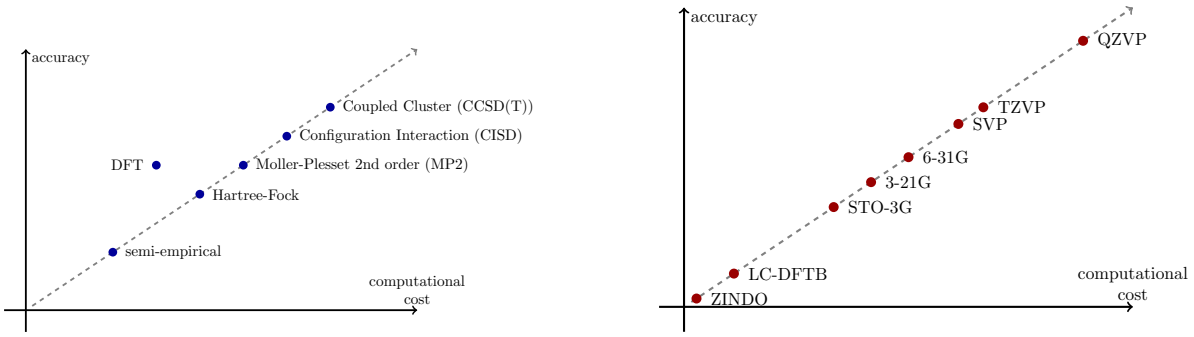
$$\boldsymbol{\Lambda} = \boldsymbol{K}^g \bigotimes \boldsymbol{K}_X + \boldsymbol{\Sigma} \bigotimes \boldsymbol{I}_N \,. \tag{3.15}$$

Here, $\boldsymbol{k}^g_m$ is the mth column of $\boldsymbol{K}^g$, $\boldsymbol{k}_{\boldsymbol{X},\boldsymbol{X}_q}$ collects the covariance between the query input and the training inputs. The matrix $\boldsymbol{\Sigma}$ is a diagonal of size $M \times M$ with $\boldsymbol{\Sigma}_{m,m} = \sigma^2_m$, that is, it models the noise in the data[3]. Thus, it can be deduced that $\boldsymbol{\Lambda}$ is of size $MN \times MN$.

3.4 Multifidelity Methods in Quantum Chemistry

Quantum chemistry allows for the control of accuracy of the properties to be calculated with a numerical simulation due to well established hierarchies [116, 117, 118]. The hierarchy can be established with several metrics. Figure 3.3 depicts a hierarchy structure for QC methods. The axes are not to scale but depict the accuracy of the methods as a

[3] Ref. [113] shows that if there is no noise in the data, that is $\boldsymbol{\Sigma} \equiv 0$, it is impossible to transfer between the tasks.



((a)) ((b))

Figure 3.3: A depiction of hierarchy in QC based on the (a) QC theory used and (b) choice of basis set used for example here in DFT. ZINDO and LC-DFTB are semi-empirical methods which are shown for easy reference.

function of the computational cost associated with them. In Figure 3.3(a), the hierarchy of QC methods is made on the basis of the QC theory used. Semi-empirical methods are the cheapest and often the least accurate. Density functional theory (DFT) [119, 120] is cheaper than most Wave function theory (WFT) methods [121, 15, 122, 123] such as Hartree-Fock and CCSD(T), the latter is often considered the 'gold standard' of computational chemistry but comes at immense cost particularly with larger molecule sizes with complex atomic species [122]. Another approach to determining the hierarchy of fidelities in QC is the use of basis set choices. Consider for example using DFT as the QC theory. There are several basis sets that can be used depending on the system of choice and the desired accuracy of considering QC details [6, 7]. This is graphically depicted in Figure 3.3(b) along with two semi-empirical methods, ZINDO and LC-DFTB. The larger the size of the basis set that is considered, such as TZVP, the more expensive the computation is.

Having developed a general overview of the multifidelity approaches that exist in literature in the preceding sections, one can now proceed to look at some multifidelity methods developed specifically in the field of ML-QC. In this section four such approaches are discussed. The Δ-ML method can be seen as a version of additive correction applied to the fidelities. The CQML method is a recursive sparse approximation method which is akin to the recursive multifidelity model described in section 3.3.2. The h-ML model is a generalization of the Δ-ML method with several fidelities and is another version of recursive multifidelity methods. Finally, the use of a MTGPR method with Δ-ML is presented.

3.4.1 Δ-Machine Learning Approach

The Δ-ML approach uses training data computed at two different fidelities [29] to train a model. Consider a training set computed at some fidelity F given as $\mathscr{T}^F := \{(\boldsymbol{X}_i, y_i^F)\}_{i=1}^{N_{train}^F}$ where $\boldsymbol{X}_i$ are the molecular descriptors and the corresponding QC property are y_i^F. For the same molecular descriptors, consider a set of QC computational calculations made at some lower fidelity $QC_b < F$ given as $\mathscr{T}^{QC_b} := \{(\boldsymbol{X}_i, y_i^{QC_b})\}_{i=1}^{N_{train}^F}$. Then the Δ-ML approach trains a ML model to learn the difference of the property between the two fidelities, that is $\Delta_{QC_b}^F = \boldsymbol{y}^F - \boldsymbol{y}^{QC_b}$. Hereon, F is referred to as the target fidelity. The final prediction of a KRR model with the Δ-ML approach is first predicting the difference between the target and baseline fidelities to which the QC calculation of the baseline fidelity is added for the evaluation samples. In other words, for a given query representation, $\boldsymbol{X}_q$, the Δ-ML prediction is given as:

$$P_{\Delta}^{F;QC_b} := P_{KRR}^{\Delta_{QC_b}^F}(\boldsymbol{X}_q) + y_q^{QC_b}\,, \tag{3.16}$$

where, $P_{KRR}^{\Delta_{QC_b}^F}$ is the KRR prediction of the difference, and $y_q^{QC_b}$ is the QC computation of the baseline fidelity for the query molecule.

Harnessing the effectiveness of ML methods [124, 44, 125], Δ-ML has shown to have high accuracy in the prediction of several QC properties. In the original work from ref. [29], authors show the effectiveness of the approach for the prediction of atomization energies of 6,000 isomers of $C_7H_{10}O_2$ from the GDB dataset [126, 127]. The HF, or the target fidelity was chosen to be G4MP2 method which is considered to be at par with experimental estimations [128, 129]. The Δ-ML model is trained and evaluated for several cheap fidelities resulting a a collection of two-fidelity MF models. The assessment of these models was performed using model error and number of training samples used in training the ML model to predict the difference[4]. The study of the model errors led to the conclusion that the more correlated the high and low fidelities are, the lower the model error is. The key aspect of the Δ-ML method which has made it popular in the QC community is that it shows good transferability. This is due to the fact that the cheaper QC baseline fidelity is computed by conventional QC methods and already incorporates a good deal of the QC theory in it. While the work in ref. [29] introduces the Δ-ML method and performs comprehensive studies of the effect of the QC computed baseline fidelities, it does not consider the time-cost of making such calculations in the assessment of the time-cost of the model in comparison with the time-cost of the single fidelity ML model built with training samples from only the

[4]The mathematical *Ansatz* of the Δ-ML approach is given in section 3.4.1.

target fidelity. The comparison is purely made in terms of model error versus number of training samples, in which case Δ-ML is shown to be superior.

The popularity of Δ-ML in reducing the overall ML model error by training on two fidelities has been utilized in several areas of predicting QC properties. For example, the method has seen usage in prediction electronic properties such as spectra and energies with the TD-DFT formalism for over 20k small organic molecules with up to 8 heavy atoms (C,O,N,F) by learning the difference between CC2 and DFT fidelities of QC theory [130]. Pilania *et al.* have used the Δ-ML approach using NNs to predict several relevant QC properties such as band gaps of solids, atomization energies, and dielectric constants at the DFT level [131]. Elsewhere, Palizhati use the Δ-ML model to predict experimental accuracy level band gaps using HF DFT data and HF experimental data [132]. Egorova *et al.* use a two fidelity GPR model to predict the energy and crystal structure ranking of three small organic molecules. The method is shown to cost as less as ~6% of the compute cost of running a full QC calculation at the HF method of DFT(PBE0). The LF employed was the GGA DFT(PBE) method [133]. Other applications of the Δ-ML method include the prediction of potential energy surfaces (PES) with CCSD(T) accuracy [134], crystal bandgap prediction with PBE accuracy [135], material screening [136], and chemical reaction PES [137].

In this dissertation, the developed multifidelity methods are compared against Δ-ML models for prediction of several QC properties in Chapter 9 and Chapter 11.

3.4.2 Combination Technique Quantum Machine Learning

Zaspel *et al.* introduced a systematic generalization of the Δ-ML method by not only extending to more than two fidelities, but also by replacing the QC computed baseline with a ML predicted baseline fidelity. This approach, titled the Combination technique Quantum Machine Learning (CQML) method was used for the prediction of atomization energies of diverse molecules with CCSD(T) as the HF QC method[32]. For this approach, one assumes that the function to be approximated, g, lies in a function space $V := V^{(1)} \bigotimes \dots V^{(d)}$, a tensor product space of d dimensions. Next, a series of lower dimensional subspaces are introduced as $V_0^m \subset \dots \subset V_{L_f}^{(m)}$ for each of the L_m dimensional $V^{(m)}$. Instead of the full tensor product approximation for fidelity f, $V_f := V_f^{(1)} \bigotimes V_f^{(d)}$, one uses sparse grid combination techniques (SGCT)[37, 36, 35] to approximate g with a cheaper approach by recursive sparse approximation of $\hat{V}_f$ given as

$$\hat{V}_f^{(d)} := \sum_{i=0}^{f} \left(V_{f-i}^{(d)} - V_{f-1-i}^{(d)} \right) \bigotimes \hat{V}_i^{(d-1)} \,. \tag{3.17}$$

Once a general ML model is identified with $P_{ML}^{(\boldsymbol{s})}$ with $\boldsymbol{s} = (t, b, n)$ being a composite index for a subspace $V_t^1 \bigotimes V_b^2 \bigotimes V_n^3$, the SGCT can be transferred to ML for QC. This is called the CQML approach. For a specific example, if one takes the case of 2-D CQML, say by fixing $b = B$, then one has

$$P_{\text{CQML}}^{(t,B,n)}(\boldsymbol{X}_q) := \sum_{f_t=0}^{t} P_{\text{ML}}^{(t-f_t,B,f_t)}(\boldsymbol{X}_q) - P_{\text{ML}}^{t-1-f_t,B,f_t}(\boldsymbol{X}_q) \,, \tag{3.18}$$

where $\boldsymbol{X}_q$ is some query input features and $P_{ML}^{(t,b,n)} \equiv 0 \iff t = 0$ or, $b = 0$ or, $n = 0$. The hierarchy of fidelities was projected into a 2-D concept, the QC level of theory and the basis set used at the level of theory. This allowed for a more systematic study of the QC hierarchies for each of these dimensions. The number of training samples at subsequent fidelities in the case of CQML are set to account for sparsity of data available with each increasing fidelity. Then the CQML model is identified as

$$P_{\text{CQML}}(\boldsymbol{X}_q) := \sum_{\boldsymbol{s}} \beta_{\boldsymbol{s}} P_{\text{ML}}^{(\boldsymbol{s})} \,, \tag{3.19}$$

where the sum runs over the composite index $\boldsymbol{s}$. As in the case of Δ-ML, the assessment of the CQML method is performed with the number of training samples used. Since the CQML method varies the number of training samples across fidelities, the model error is reported as a function of the number of training samples used at the target fidelity. The effectiveness of the multi-dimensional (QC method and basis set size) MF method is established for the prediction of atomization energies over the QM7b dataset [138, 49]. The 1-D CQML, wherein only the basis set is modified while fixing the QC theory is called the MF machine learning (MFML) model and is introduced in Chapter 5 of this dissertation.

> The MFML method developed in Chapter 4 is a lower dimension version of the CQML approach where the composite index corresponds to the fidelity of training data and the number of training samples used at that fidelity.

3.4.3 Hierarchical Machine Learning

Yet another generalization of the Δ-ML that has been introduced is the hierarchical ML (hML) approach wherein several Δ-ML models are trained for several fidelities [139]. The method involves the use of an *ad hoc* optimization procedure to select the number of training samples to be used at each fidelity in the construction of the Δ-ML models. The hML model for a target potential energy surface (PES) is defined as

$$P_{hML}(\boldsymbol{X}_q) = \sum_{m} P_{\Delta_m}^{(N_m)}(\boldsymbol{X}_q) \,, \tag{3.20}$$

where each Δ-ML model P_{Δ_m} is trained a number of training samples N_m. The work firstly presents and *ad hoc* optimization procedure to determine the optimal number of training samples for each Δ-ML model to achieve a target accuracy ϵ_u with some time budget t_{hML}. If the error of the hML model is denoted as ϵ_{hML} and the maximum time cost of conventionally calculating the PES as t_{full}, and s_u be the user defined time benefit over t_{full}, then the following objective function is minimized for the number of training samples N_m

$$\frac{\epsilon_{\text{hML}}}{\epsilon_u} + \frac{\frac{t_{\text{hML}}}{t_{\text{full}}}}{1 - s_u} + \frac{p}{\epsilon_u}, \tag{3.21}$$

where the last term penalizes extremely small values of N_m. The error ϵ_{hML} is estimated with a small validation set with 100 samples. This eliminates the need to train all the Δ-ML models. With this set-up the hML model is used to predict the PES of CH_3Cl at the CCSD(T)-F12b fidelity. The Δ-ML models are built with several cheaper fidelities. The final hML model is then built with the appropriate number of training samples selected for each Δ-ML model. This hML model is shown to produce the PES with an error of 0.03 kcal/mol at a fraction of the cost of t_{full}. Note that the reported benefit of the hML is with respect to the use of conventional QC computation at the CCSD(T)-F12b fidelity and does not compare the cost of a single fidelity model trained at CCSD(T)-F12b with respect to the hML model. That is, the cost benefit s_u, is a measure of the cost of the hML model to the QC computation. As with the case of CQML, hML also requires the nestedness of training data, that is $\mathscr{X}_F \subseteq \ldots \subseteq \mathscr{X}_1$ for an ordered hierarchy of fidelities $\{1,\ldots,F\}$.

> In Chapter 9, an elementary h-ML model is assessed against the multifidelity methods developed in the dissertation. This assessment is made on the basis of training data cost versus ML model error.

3.4.4 Multi-Task Δ-Machine Learning

As opposed to using the different fidelities to build a recursive model, ref. [110] treats the different fidelities as related tasks to be estimated simultaneously. This is achieved by using MTGPR and trains a single model for the prediction of QC properties with heterogeneous data, that is relaxing the nestedness assumption of the MF data. Within this approach, each task is assumed to be its own regression problem given as

$$\boldsymbol{y}_p = g(\boldsymbol{X}) + \epsilon_p \tag{3.22}$$

for the primary task and

$$\boldsymbol{y}_{s_n} = g(\boldsymbol{X}) + \epsilon_{s_n}, \forall\, n = 1,2,\ldots,N-1 \tag{3.23}$$

for the $N-1$ secondary tasks. Each $\epsilon_.$ is i.i.d. from a centered Normal distribution $\mathcal{N}(0,\sigma^2)$. With such a set-up, one then defines the MT based on the primary task [140] as

$$g_{s_n}(\boldsymbol{X}) = \rho_{s_n}(\boldsymbol{X}) g_p(\boldsymbol{X}) + \Delta_{s_n}(\boldsymbol{X}) \,. \tag{3.24}$$

Notice how this resembles the comprehensive correction for MF data from Eq. (3.3). The prior for this formulation is taken as

$$g_p \sim GP(\boldsymbol{\mu}_p(\boldsymbol{X}), k_p(\boldsymbol{X})) \,,$$

and

$$\Delta_{s_n} \sim GP(\boldsymbol{\mu}_{s_n}(\boldsymbol{X}), k_{s_n}(\boldsymbol{X})) \,,$$

that is, these follow GP distributions with a given mean and kernel function. Furthermore, $\Delta_{s_n} \perp \Delta_{s_m} \forall m \neq n$. If $\boldsymbol{R}$ is considered to be a diagonal matrix of size $N-1$ with its diagonal entries to be the correlation of the corresponding secondary tasks, then the joint mean and joint covariance of the MT can be stated as

$$\mu_{\text{MT}} := \begin{pmatrix} \boldsymbol{\mu}_p \\ \boldsymbol{R}\boldsymbol{\mu}_p + \boldsymbol{\mu}_s \\ \boldsymbol{\mu}_s \end{pmatrix} \tag{3.25}$$

and

$$\Sigma = \begin{pmatrix} K^p_{pp} + \sigma^2 \boldsymbol{I} & K^p_{ps}\boldsymbol{R} & K^p_{pq} \\ \boldsymbol{R}K^p_{sp} & \boldsymbol{R}K^p_{ss}\boldsymbol{R} + K^\Delta + \sigma^2 \boldsymbol{I} & \boldsymbol{R}K^p_{sq} \\ K^p_{qp} & K^p_{qs}\boldsymbol{R} & K^p_{qq} \end{pmatrix} \tag{3.26}$$

respectively. In the joint covariance matrix, the super-scripts are used to indicate the GP priors used. The q subscripts indicate the input queries for which the prediction is to be made. Combined with Δ-ML, the MTGPR method is shown to be effective in the prediction of QC properties such as the three-body interaction energy for water trimers and the ionization potential of small molecules consisting of {C,O,N,H} atoms [110].

The MFΔML method developed in Chapter 4 and further assessed in Chapter 9 and Chapter 11 builds an MFML-like model with several Δ-ML models. This approach melds the two methods similar to what is done in the Multi-task Δ-ML approach where the two methods that are combined are the MT-GPR and Δ-ML.

3.5 Revisiting the Objectives

Through the last two chapters, several methodological pre-requisites to the dissertation have been presented. Furthermore, several applications of multifidelity methods have been delineated. An extensive survey of the applications of such methods can be found in refs. [141, 109, 68, 28]. With a better view of the landscape, it becomes easier to notice the niches which can and will be explored in this dissertation. The objectives described in Chapter 1 can be better understood.

In most of the applications described above, the standard procedure is to establish the method for a bi-fidelity setup of the model or training data. The CQML method developed in ref. [32] was shown to be effective for the prediction of atomization energies but with a very specific 3-D setup of the fidelities, along QC theory and basis set size. Often, computing such intensive variations in both QC theory and basis set sizes is not feasible especially when dealing with large systems. This is specially where **O1** - developing the MFML method - would come in. Not only does it relax the assumptions made on the fidelity hierarchy structure but it also allows, like the CQML method, for the use of several fidelities.

While several examples above discuss the reduction in number of training samples required at the highest fidelity incorporated, not many discuss the time cost incurred in generating the training data for a specific ML model accuracy. **O2** - focusing on the cost of the entire multifidelity training data instead of number of most expensive training samples - fills this precise gap in literature. The focus now shifts from merely considering ML model accuracy and number of training samples but now moves to studying the cost incurred in building that ML model (which is explicitly related to the number of training samples used in all fidelities). This approach also enables one to fulfill **O3** - time-cost benchmarks of multifidelity methods. In this dissertation several multifidelity methods described above are benchmarked for the prediction of QC properties.

The recursive multifidelity model presented in section 3.3.2 allows for several degrees of freedom that can be studied, for example, the choice of the multiplicative and additive correction terms. The use if o-MFML studies how these can be optimally computed. Further, the nested assumption within the training data is analyzed in Chapter 8. These work in the direction of completing **O3** - investigating degrees of freedom in MFML.

Overall, the last two chapters have provided a working framework and foundation for this dissertation. In the forthcoming part, contributions made by this dissertation in the identified niches will be discussed. The methods that are developed are presented compiled in Chapter 4 with specific elaborations relegated to individual chapters that follow.

Part II

Development of Technical Methods and Applications to Quantum Chemical Properties

CHAPTER 4

METHODOLOGICAL CONTRIBUTIONS

One recognizes one's course by discovering the paths that stray from it.

—Albert Camus

In the previous part of this thesis, the state of art was discussed in addition to certain key existing concepts that are used throughout the work in this dissertation. This current chapter aims to highlight the technical and methodological developments from this dissertation. The methods mentioned in this chapter are a harmonized version of the methods referred to in the different publications that are compiled in this larger body of work with some minor changes in interest of uniformity of the document itself. To fully place them in the context of the chapters that follow, the reader is referred to the original works in refs. [142, 143, 41, 42, 144, 43]. However, contrary to the case of the individual publications, the notations are unified in this chapter and used across this present work. Furthermore, the methodical details from the QC side of this work are retained within the specific chapters for easier reference.

4.1 Multifidelity Machine Learning

Consider an ordered hierarchy of fidelities indexed by $f = 1, 2, \ldots, F$ where the cost of calculation (and usually, therefore, accuracy) increases with an increase in the index. The training set for data at some fidelity f can be then defined as $\mathscr{T}^{(f)} := \left\{\left(\boldsymbol{X}_i^{(f)}, y_i^{(f)}\right)\right\}_{i=1}^{N^{(f)}}$. One can define the set of molecular descriptors as $\mathscr{X}^f := \left\{\boldsymbol{X}_i^f | \left(\boldsymbol{X}_i^f, y_i^f\right) \in \mathscr{T}^{(f)}\right\}$. Based on previous work in this field as detailed in refs. [32, 142] the current state of the multifidelity method

recommends the nestedness $\mathscr{X}^F \subseteq \ldots \subseteq \mathscr{X}^2 \subseteq \mathscr{X}^1$ of the training data. In other words, if a molecular conformation has a quantum chemistry property (such as the excitation energy) calculated at the highest fidelity, then the property is also calculated at all other lower fidelities. Although this nested property of training data is retained for the most part of this dissertation, Chapter 8 studies the effect of using non-nested training data across different fidelities. As shown in ref. [142], a MFML model together with KRR as the ML model of choice can iteratively be built for an ordered hierarchy of fidelities as

$$P^{(F;f_b)}_{\text{MFML}}\left(\boldsymbol{X}_q\right) := P^{(f_b)}_{\text{KRR}}\left(\boldsymbol{X}_q\right) + \sum_{f_b \leq f < F} P^{(f,f+1)}_{\text{KRR}}\left(\boldsymbol{X}_q\right) , \tag{4.1}$$

where F is the target fidelity and $f_b = 1,2,\ldots,F-1$ is some baseline fidelity, and $\boldsymbol{X}_q$ is the molecular representation of a query molecule. The term inside the summation is calculated as

$$P^{(f,f+1)}_{\text{KRR}}\left(\boldsymbol{X}_q\right) := \sum_{i=1}^{N^{(f+1)}_{\text{train}}} \alpha^{(f,f+1)}_i k\left(\boldsymbol{X}_i, \boldsymbol{X}_q\right) . \tag{4.2}$$

The coefficients of KRR, $\alpha^{(f,f+1)}_i$, are calculated by solving a linear system of equations given by

$$\left(\boldsymbol{K} + \lambda \boldsymbol{I}_{N^{(f+1)}}\right) \boldsymbol{\alpha}^{(f,f+1)} = \boldsymbol{\Delta y}^{(f,f+1)} . \tag{4.3}$$

It is to be noted that $\boldsymbol{\Delta y}^{(f,f+1)} = \boldsymbol{y}^{f+1} - \boldsymbol{y}^{(f,f+1)}$, where $\boldsymbol{y}^{f+1}$ is the vector of energies in the training set $\mathscr{T}^{(f+1)}$ and $\boldsymbol{y}^{(f,f+1)}$ is the vector of energies in the training set $\mathscr{T}^{(f)}$ restricted to those conformations only found on fidelity level $f+1$. Thus, this definition of MFML can be seen as one that works on the difference between the data, or simply put, data difference MFML.

Example 4.1 (Data Difference MFML)**.** *A MFML model for a target fidelity $F = 5$ with a baseline of $f_b = 3$ can be iteratively built as*

$$P^{(5;3)}_{\text{MFML}}\left(\boldsymbol{X}_q\right) := P^{(3)}_{\text{KRR}}\left(\boldsymbol{X}_q\right) + P^{(3,4)}_{\text{KRR}}\left(\boldsymbol{X}_q\right) + P^{(4,5)}_{\text{KRR}}\left(\boldsymbol{X}_q\right) \tag{4.4}$$

with

$$P^{(3)}_{\text{KRR}}\left(\boldsymbol{X}_q\right) := \sum_{i_3=1}^{N^{(3)}_{\text{train}}} \alpha^{(3)}_{i_3} k\left(\boldsymbol{X}_{i_3}, \boldsymbol{X}_q\right) \tag{4.5a}$$

$$P_{\text{KRR}}^{(3,4)}\left(\boldsymbol{X}_q\right) := \sum_{i_4=1}^{N_{\text{train}}^{(4)}} \alpha_{i_4}^{(3,4)} k\left(\boldsymbol{X}_{i_4}, \boldsymbol{X}_q\right) \tag{4.5b}$$

$$P_{\text{KRR}}^{(4,5)}\left(\boldsymbol{X}_q\right) := \sum_{i_5=1}^{N_{\text{train}}^{(5)}} \alpha_{i_5}^{(4,5)} k\left(\boldsymbol{X}_{i_5}, \boldsymbol{X}_q\right) \tag{4.5c}$$

The number of training samples used for each fidelity in the standard setup of MFML differs by a *scaling factor*, denoted by γ, of 2 based on research on sparse grid combination methods [37, 36, 35]. Hence, for the model in Eq. (4.4), assuming to have $N_{\text{train}}^{(5)} = 32$ training samples for fidelity 5 leads to $N_{\text{train}}^{(4)} = 64$ training samples for fidelity 4 and $N_{\text{train}}^{(3)} = 128$ training samples on fidelity 3. Thus, if the number of training samples at the target fidelity are set to be N_{train}^{F}, then the next lower fidelity uses $2 \times N_{\text{train}}^{F}$ of training samples and so on. Section 4.4 presents more details on the concept of scaling factors.

In ref. [32] it has been shown mathematically that the data difference MFML is equivalent to taking the difference of models built on the two different levels while ensuring a nested data structure. That is, $P_{\text{KRR}}^{(f,f+1)} \equiv P_{\text{KRR}}^{(f+1)} - P_{\text{KRR}}^{(f)}$ where $P_{\text{KRR}}^{(f)}$ is built on the training set $\left\{\left(\boldsymbol{X}_i, y_i^{(f,f+1)}\right)\right\}_{i=1}^{N_{\text{train}}^{(f+1)}}$ with conformations restricted to those found in the training set used for fidelity $f+1$. This result is further numerically verified in the supplementary material in Appendix A in Table A.3 for the excitation energies of arenes. Models of the type $P_{\text{KRR}}^{(f+1)}$ and $P_{\text{KRR}}^{(f)}$ are herein referred to as *sub-models* of MFML. A sub-model of MFML is built for a specific choice of a training set. For the current work, it implies selecting a fidelity, f, and the number of training samples at this fidelity, $N_{\text{train}}^{(f)}$ for $f = 1, \ldots, F$. This formulation of sub-models, represents a 2-dimensional multifidelity structure, that is, the fidelity, and the number of training samples. In such a structure, it is assumed that increasing the fidelity results in a more accurate (and therefore, a costlier) QC calculation. This in turn translates into a more accurate (and costlier to train) sub-model. In principle, there is no limit on the dimensions of MFML as long as a clear hierarchy can be established in each dimension [32]. For the specific case of the 2-D structure, one can identify a sub-model with an ordered pair, or index, $\boldsymbol{s} = (f, \eta_f)$ where f is the fidelity and the number of training samples chosen from this fidelity are given as $N_{\text{train}}^{f} = 2^{\eta_f}$. A standard KRR model built for the index $\boldsymbol{s}$ is then denoted as $P_{\text{KRR}}^{(\boldsymbol{s})}$.

With this development, one arrives at the MFML method written as the linear combination of the various sub-models. To this end, some notations are introduced. The set of indexes of all available sub-models is denoted by $\mathscr{S}$. A standard KRR model for a query molecule represented as $\boldsymbol{X}_q$ is built as $P_{\text{KRR}}^{(\boldsymbol{s})}\left(\boldsymbol{X}_q\right)$ for $\boldsymbol{s} \in \mathscr{S}$. Further, define the set of in-

dexes of sub-models used for a MFML model with target fidelity F, for $N_{\text{train}}^{F} = 2^{\eta_F}$, and a baseline f_b, as follows:

$$\mathscr{S}^{(F,\eta_F;f_b)} := \Big\{(f,\eta_f) \in \mathscr{S} | f \in \{f_b,\ldots,F\}, \eta_f \in \{\eta_F,\ldots,2^{F-f_b}\cdot\eta_F\}, \\ F+\eta_F-1 \leq f+\eta_f \leq F+\eta_F\Big\}, \tag{4.6}$$

where $\mathscr{S}^{(F,\eta_F;f_b)} \subseteq \mathscr{S}$. The motivation is to combine various sub-models such that only a few expensive training samples are required, which, when combined with cheaper training samples, yield a high-accuracy low-cost model for the target fidelity. This is achieved by the linear combination of the sub-models from $\boldsymbol{s} \in \mathscr{S}^{(F,\eta_F;f_b)}$. This is denoted by

$$P_{\text{MFML}}^{(F,\eta_F;f_b)}\left(\boldsymbol{X}_q\right) := \sum_{\boldsymbol{s}\in\mathscr{S}^{(F,\eta_F;f_b)}} \beta_{\boldsymbol{s}} P_{\text{KRR}}^{(\boldsymbol{s})}\left(\boldsymbol{X}_q\right), \tag{4.7}$$

where $\beta_{\boldsymbol{s}}$ are the coefficients of the linear combination. These coefficients can be interpreted as a measure of how much each sub-model contributes to the final MFML model. Based on work in MFML for atomization energies [32] and excitation energies [142], the coefficients are set in such a manner that each sub-model contributes in equal magnitude to the final MFML model. For a model of the form $P_{\text{MFML}}^{(F,\eta_F;f_b)}$, the $\beta_{\boldsymbol{s}}$, are set in conventional MFML as follows:

$$\beta_{\boldsymbol{s}}^{\text{MFML}} = \begin{cases} +1, & \text{if } f+\eta_f = F+\eta_F \\ -1, & \text{otherwise} \end{cases}, \tag{4.8}$$

where the terms are as discussed previously.

A hypothetical 2-dimensional multifidelity structure is shown in Figure 4.1 with the dimension of fidelity on the y-axis and the dimension of the number of samples on the x-axis. One can now identify various sub-models in this hypothetical structure. For instance, $P^{(\boldsymbol{s})}$ with $\boldsymbol{s} = (6-31G,5)$, represents a sub-model built at the 6-31G fidelity with $2^5 = 32$ training samples. In this scheme, the cost (and therefore, the accuracy to target fidelity) of the training data of the sub-models increases with increase in either of f or $s\eta_f$. That is, $\boldsymbol{s}$ is more accurate (and more expensive) than a sub-model built with $\boldsymbol{s}' = (3-21G,5)$. At the same time, a sub-model built with $\boldsymbol{s}'' = (6-31G,6)$ is more accurate (and expensive) than $\boldsymbol{s}$ from this example.

Example 4.2 (Model Difference MFML)**.** *Consider the set of sub-models for MFML being built for target fidelity $F = 4$, with 2^2 (that is, $\eta_F = 2$) training samples at this fidelity, and with a baseline fidelity of $f_b = 1$. The set of MFML sub-model indexes is then given by $\mathscr{S}^{(4,2;1)} = \{(4,2),(3,2),(3,3),(2,3),(2,4),(1,4),(1,5)\}$. The MFML model is built as the*

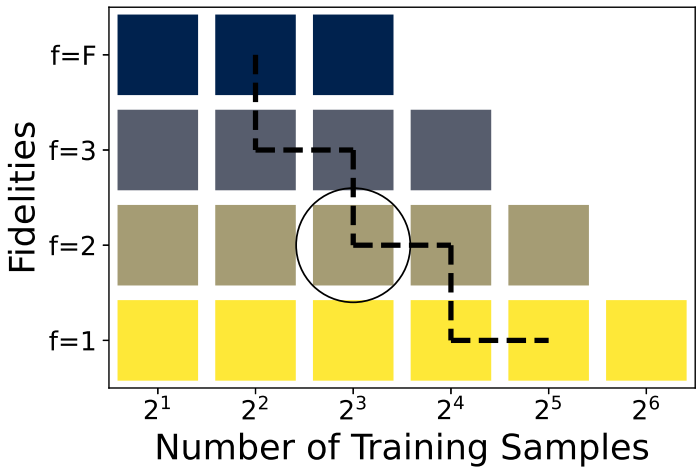


Figure 4.1: A hypothetical structure of sub-models for 4 fidelities is depicted here. Each sub-model can be identified with an index pair $\boldsymbol{s} = (f, \eta_f)$ representing the fidelity with $N^f_{\text{train}} = 2^{\eta_f}$. Thus, the circled sub-model can be denoted as $\boldsymbol{s}' = (2,3)$. Within this formulation, the MFML model is built by combining the sub-models as shown with the dotted black line. The contribution of sub-model $\boldsymbol{s}'$ is given by the coefficient denoted by $\beta_{\boldsymbol{s}'}$. In conventional MFML, this one in particular would equal to -1.

linear combination of the individual KRR sub-models with indexes $s \in \mathscr{S}^{(4,2;1)}$. *The coefficients are as defined by Eq.* (4.8)*, i.e., explicitly,*

$$P^{(4,2;1)}_{\text{MFML}}\left(\boldsymbol{X}_q\right) := P^{((4,2))}_{\text{KRR}}\left(\boldsymbol{X}_q\right) - P^{((3,2))}_{\text{KRR}}\left(\boldsymbol{X}_q\right) + P^{((3,3))}_{\text{KRR}}\left(\boldsymbol{X}_q\right) - P^{((2,3))}_{\text{KRR}}\left(\boldsymbol{X}_q\right) + P^{((2,4))}_{\text{KRR}}\left(\boldsymbol{X}_q\right) - P^{((1,4))}_{\text{KRR}}\left(\boldsymbol{X}_q\right) + P^{((1,5))}_{\text{KRR}}\left(\boldsymbol{X}_q\right) .$$

One can readily see that this is the very same model $P^{(4;1)}_{\text{MFML}}\left(\boldsymbol{X}_q\right)$ as would be arrived at by using Eq. (4.1) with 2^2 training samples used at the target fidelity. The conventional MFML model built with coefficients set by Eq. (4.8) is simply denoted as $P^{(F;f_b)}_{\text{MFML}}$ since it is identical to the MFML model built in ref. [142].

Let us connect back the development of MFML to the recursive multifidelity model presented in Eq. (3.9) from Chapter 3. As the example above indicates, there is clearly a recursive approach employed in making predictions at a target fidelity. The multiplicative correction term in Eq. (3.9) are signed unitary weights indicated in Eq. (4.8). The additive correction term itself, is the difference in predictions between two sub-models trained at fidelity f and $f-1$ with the same number of training samples. What remains to be seen is the orthogonality requirement from Eq. (3.9) being satisfied. Note first, that the orthogonality requirement essentially means that there is no further in-

formation that can be learned in the surrogate model at $f-1$ other than what is already present in the additive correction term. Consider that the training data used at different fidelities is nested (as presented above). With this setup, take the sub-models $P_{\text{KRR}}^{(f,\eta_f)}$ and $P_{\text{KRR}}^{(f,\eta'_f)}$ such that $\eta_f < \eta'_f$. That is, $P_{\text{KRR}}^{(f,\eta_f)}$ uses less training data than $P_{\text{KRR}}^{(f,\eta'_f)}$. But due to the nested nature of the training samples (see also section 4.5), the training samples in sub-model $P_{\text{KRR}}^{(f,\eta_f)}$ are also used in training the sub-model $P_{\text{KRR}}^{(f,\eta'_f)}$. In other words, the information contained in the sub-model $P_{\text{KRR}}^{(f,\eta_f)}$ is also contained in the sub-model $P_{\text{KRR}}^{(f,\eta'_f)}$. Next, it is observed that the additive correction term arising in the MFML model is the difference between the prediction of sub-models $P_{\text{KRR}}^{(f-1,\eta'_f)}$ and $P_{\text{KRR}}^{(f,\eta'_f)}$. Therefore, implicitly, the orthogonality condition is satisfied. Chapter 8, where the nestedness of multifidelity training data is relaxed, investigates the effect of the loss of this orthogonality condition.

4.2 Optimized Multifidelity Machine Learning

Having written the MFML model in terms of the individual sub-models of multifidelity, one can consider formulations of the coefficients, which are different from Eq. (4.8). The *optimized MFML* (o-MFML) method optimizes the coefficients, which are model parameters, this can be seen as a *hyperparameter* optimization of the different $\beta_{\boldsymbol{s}}$ to yield a multifidelity model, which has an improved accuracy. In most ML methods, a hyperparameter is a variable parameter which controls various aspects of the learning procedure. In KRR, for instance, the regularization strength and kernel width are hyperparameters which control different aspects of the learning, including penalizing overfitting.

For such an optimization, the validation set is defined as $\mathcal{V}_{\text{val}}^F := \left\{(\boldsymbol{X}_q^{\text{val}}, y_q^{\text{val}})\right\}_{q=1}^{N_{\text{val}}}$. To evaluate the accuracy of the model, define a test set $\mathcal{V}_{\text{test}}^F := \left\{(\boldsymbol{X}_q^{\text{test}}, y_q^{\text{test}})\right\}_{q=1}^{N_{\text{test}}}$ such that the two are mutually exclusive. That is, $\mathcal{V}_{\text{val}}^F \cap \mathcal{V}_{\text{test}}^F = \phi$, where ϕ denotes the empty set. The split of the validation and test sets is a common approach in ML techniques, wherein the optimization/ hyperparameter-tuning is performed on the former and the error of the final model is reported on the latter. It is to be noted that the test set is never used in any stage of the training process.

One can explicitly define a o-MFML model for a target fidelity F, with $N^{(F)} = 2^{\eta_F}$ training samples at the target fidelity, for a baseline fidelity f_b, as

$$P_{\text{o-MFML}}^{(F,\eta_F;f_b)}\left(\boldsymbol{X}_q\right) := \sum_{\boldsymbol{s}\in\mathscr{S}^{(F,\eta_F;f_b)}} \beta_{\boldsymbol{s}}^{\text{opt}} P_{\text{KRR}}^{(\boldsymbol{s})}\left(\boldsymbol{X}_q\right) \tag{4.9}$$

where $\beta_{\boldsymbol{s}}^{\text{opt}}$ are optimized coefficients, and $\boldsymbol{X}_q$ is the representation of a query molecule. In general, one is interested in solving the optimization task:

$$\beta_{\boldsymbol{s}}^{\text{opt}} = \arg\min_{\beta_{\boldsymbol{s}}} \left\| \sum_{v=1}^{N_{\text{val}}} \left(y_v^{\text{ref}} - \sum_{\boldsymbol{s} \in S^{(F,\eta_F;f_b)}} \beta_{\boldsymbol{s}} P_{\text{KRR}}^{(\boldsymbol{s})}(\boldsymbol{X}_v) \right) \right\|_p$$

where one minimizes some p-norm on the validation set $\mathcal{V}_{\text{val}}^F$. This is equivalent to solving

$$\boldsymbol{\beta}^{\text{opt}} = \arg\min_{\boldsymbol{\beta}} \left\| \boldsymbol{M}_{\mathscr{S}'} \boldsymbol{\beta} - \boldsymbol{y}^{\text{val}} \right\|_p \tag{4.10}$$

where $\boldsymbol{M}_{\mathscr{S}^{(F,\eta_F;f_b)}} = \left(P_{\text{KRR}}^{(\boldsymbol{j})}(\boldsymbol{X}_i) \right)_{i=1,\ldots,N_{\text{val}};\boldsymbol{j} \in \mathscr{S}^{(F,\eta_F;f_b)}}$ is a $N_{\text{val}} \times |\mathscr{S}'|$ matrix, $\boldsymbol{\beta}$ is the vector of coefficients with respect to $\mathscr{S}'$ as depicted in Eq. (4.9), and $\boldsymbol{y}^{\text{val}}$ is the vector of reference energies from $\mathcal{V}_{\text{val}}^F$. This work utilizes the ordinary least squares optimization (OLS) procedure to solve Eq. (4.10) with $p = 2$. In the results, the OLS o-MFML model is reported as $P_{\text{o-MFML}}$. However, it must be noted that any method that can solve the minimization problem in Eq. (4.10) can be used to optimize the coefficients.

Thus, the complete process of building an o-MFML model can be written as follows:

1. Identify the set of sub-models for a given MFML model, $\mathscr{S}^{(F,\eta_F;f_b)}$.
2. Build the various KRR sub-models for sub-models $\boldsymbol{s} \in \mathscr{S}^{(F,\eta_F;f_b)}$.
3. Optimize the coefficients, $\beta_{\boldsymbol{s}}$, on $\mathcal{V}_{\text{val}}^F$ using an optimizer of choice.
4. Evaluate the final model $P_{\text{o-MFML}}^{(F,\eta_F;f_b)}$ on $\mathcal{V}_{\text{test}}^F$ for some error metric (Section 2.7).

Chapter 6 discussed the use of o-MFML method in predicting excitation energies and atomization energies.

4.3 Multifidelity Δ-Machine Learning Method

Given the development of MFML for an ordered hierarchy of fidelities, $f \in \{1, 2, \ldots, F\}$, one can consider a case where all the training energies are 'centered' by the energies of the lowest fidelity, $f = 1$. This approach essentially creates a Δ-ML model over the MFML model and can be termed as *multifidelity Δ-machine learning* (MFΔML) method. The prediction using this method for a query representation $\boldsymbol{X}_q$ is given as:

$$P_{MF\Delta ML}^{(F,\eta_F;f_b,QC_b)}(\boldsymbol{X}_q) := \sum_{\boldsymbol{s} \in \mathscr{S}^{(F,\eta_F;f_b)}} \beta_{\boldsymbol{s}} P_{\Delta}^{(\boldsymbol{s})}\left(\boldsymbol{X}_q\right) . \tag{4.11}$$

Here, $P_{\Delta}^{(s)}$ are Δ-ML models identified by Eq. (3.16) where QC_b would be the fidelity $f = 1$, the target fidelity for these Δ-ML models would be fidelity f. The MFΔML model is built for some baseline fidelity $f_b > 1$ for a target fidelity F. Thus, with the MFΔML approach, the multifidelity model predicts the difference in energies of some fidelity f and the QC-baseline, f_b^{QC}. With this definition one can also readily extend the concept of o-MFML to optimized MFΔML (o-MFΔML).

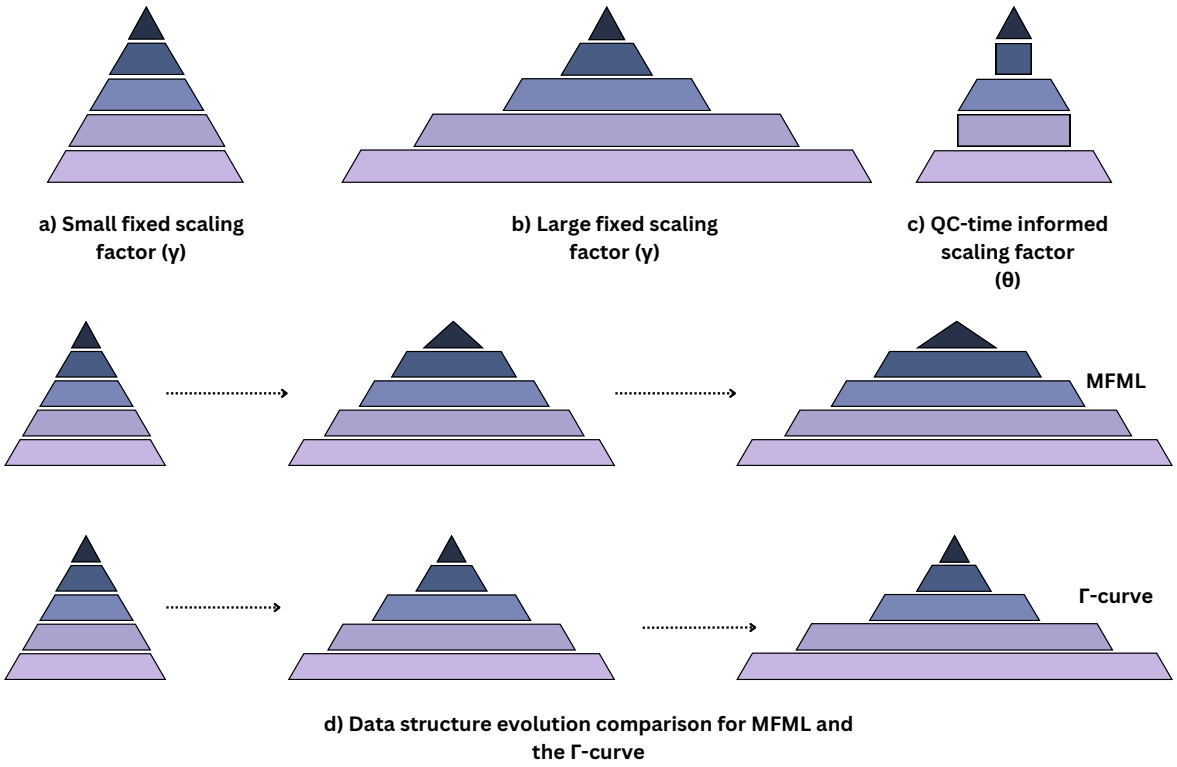


Figure 4.2: A hypothetical comparison of training data used across fidelities for the different kinds of scaling factors used in this chapter. a) The multifidelity training data structure used in MFML with a small fixed scaling factor (γ). b) Multifidelity training data structure for a large fixed scaling factor (γ) results in a larger number of training samples being used at the cheaper fidelities. c) The structure of multifidelity training data used for scaling factors that are decided based on the QC-time cost, explained in section 4.4 as θ_f^F and θ_{f-1}^f. d) Comparison of training data structure evolution for conventional MFML and the Γ-curve introduced in section 4.7. Notice how the number of training samples used at the target (the costliest) fidelity remain same across the data structure for the Γ-curve while they increase for the conventional MFML method.

Chapter 4.3 discussed the use of MFΔML method in comparison the Δ-ML approach, the MFML method, and the o-MFML method over a benchmark multifidelity dataset

developed in Chapter 7. Furthermore, the MFΔML method is employed in the prediction of CCSD(T) energies of several monomers in Chapter 11.

4.4 Scaling Factors

Thus far, in both MFML and o-MFML, the number of training samples used for each fidelity are scaled by a *scaling factor* of γ =2 based on research on SGCT [37, 36, 35]. For instance, if one has $N_{\text{train}}^{(F)} = 32$ training samples for fidelity F then it is that $N_{\text{train}}^{(F-1)} = 2 \times 32 = 64$ training samples for fidelity $F-1$, $N_{\text{train}}^{(F-2)} = 2 \times 64 = 128$ training samples on fidelity $F-2$, and so on. The scaling up of the training samples as one decreases the fidelities can be thought of intuitively from a perspective of sparseness of data. As one increases the fidelity, the cost of QC calculation increases. This results in lower number of point-calculations that need to be made at this fidelity. The scaling factor can itself be varied to assess its effect on the model error. For each value of these scaling factors, the training set size increases exponentially as one goes down the fidelities. If one starts with N_{train}^{F} samples at the target fidelity, then at each lower fidelity, $f < F$, the number of training samples would be $N_{\text{train}}^{f} = \gamma^{F-f} \times N_{\text{train}}^{F}$.

Example 4.3 (MFML Training Data Structure)**.** *If* $F = 5$, $N_{\text{train}}^{F} = 2$, *and* $\gamma = 3$, *then for* $f_b = 1$ *the training set size for each fidelity, in increasing order of the fidelity, would be* $\{3^4 \times 2, 3^3 \times 2, 3^2 \times 2, 3 \times 2, 2\}$.

The variation of the training set sizes with increasing values of γ is represented pictorially in Figure 4.2(a)-(b). For a smaller γ, the scaling is not as drastic as it is for larger values of the scaling factor. In Chapter 10, the effect of varying γ on model accuracy is studied while also assessing the cost effectiveness of such a data scaling.

This dissertation studies two additional approaches for QC-cost adapted selection of scaling factors. This approach takes into account the compute times for each fidelity before adaptively selecting the ratio of training samples between two consecutive fidelities. Choosing the scaling factors using the cost incurred in making the calculations for the fidelity can be motivated as follows: the MFML method builds sub-models at the different fidelities by training on the data from that specific fidelity. These sub-models are then combined to give the MFML model as expressed in Eq. (4.7). The sub-models are defined not just by the fidelity but also by the number of training samples used for each sub-model. The number of training samples in turn are chosen based on the scaling factor chosen between two subsequent fidelities. Therefore, if one incorporates some information of the time-cost

of the fidelity to the scaling factor, one could implicitly influence the number of training samples based on this cost.

While there could be different ways to determine these time-informed scaling factors, the most trivial approach is to take the nearest integer value of the ratio of the compute times for the subsequent fidelity. That is, one can define $\theta^f_{f-1} := \lfloor T^f / T^{f-1} \rceil$, where $\lfloor \cdot \rceil$ denotes integer rounding. This specific choice of scaling factors is made to take into account the relative time-cost of the fidelities used in the MFML model. It is reasonable to assume that the number of training samples used at consecutive fidelities should be based on the ratio of the cost of those fidelities. Hereon, the MFML models built with this approach of scaling factors are referred by θ^f_{f-1}.

A second approach of implicitly incorporating the time-cost of the fidelities is to take the ratio of the compute times with respect to the target fidelity. This approach is motivated by posing the question, what amount of training data used at a specific fidelity would cost the same as the training data used at the target fidelity. Once again, the nearest integer value is considered. This leads to the definition of $\theta^f_F := \lfloor T^F / T^f \rceil$ for all $f < F$. This formulation of scaling factors is hereon associated with θ^F_f. These different scaling factors are further assessed in Chapter 10.

4.5 Cross-Validation Scheme for Nested Multifidelity Data

In all results herein, the learning curves are averaged over a 10-run random shuffling of the MFML training set while ensuring the nestedness of the training samples. Due to the nested structure of the training data used in MFML, conventional cross-validation methods cannot be used. In its place, the algorithm discussed below is used. This flavor of validation set approach for the multifidelity data structures will ensure to catch any under or over fitting that might arise due to choice of training set. This also ensures that the results reported are robust to any variation that might arise due to the choice of the training set. For each of the 10 runs, the procedure is as follows:

1. Randomly select $N^F_{\text{train}} = 2^{\eta_F}$ training samples from $\mathscr{T}^F$. Define this as a new sampled training set, $\mathscr{D}^F \subseteq \mathscr{T}^F$.

2. Train the sub-model $P^{(F,\eta_F)}_{\text{KRR}}$ on training data from $\mathscr{D}^F$.

3. For the conformations $\boldsymbol{X}_i$ such that $\left(\boldsymbol{X}^F_i, y^F_i\right) \in \mathscr{D}^F$, train the sub-model $P^{(F-1,\eta_F)}_{\text{KRR}}$ with properties $y^{F-1,F}_i$, that is, the energies at fidelity $F-1$ for the conformations, which are also found in $\mathscr{D}^{(F)}$.

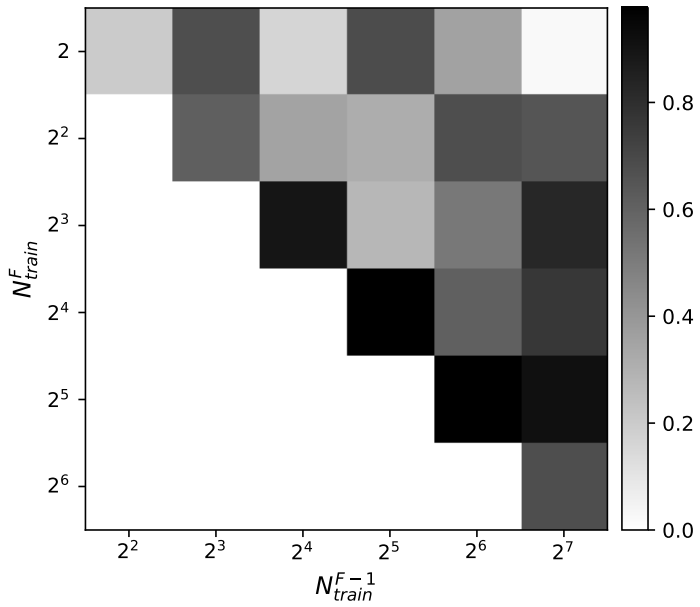


Figure 4.3: A hypothetical MFML error contour for fidelity f and $f-1$.

4. At the next lower fidelity, $f = F-1$, build the sampled training set

$$\mathscr{D}^{F-1} := \left\{\left(\boldsymbol{X}_i^F, y_i^{F-1}\right)\right\}_{i=1}^{N^F_{\text{train}}} \cup \left\{\left(\boldsymbol{X}_j^{F-1}, y_j^{F-1}\right)\right\}_{j=1}^{2\cdot N^F_{\text{train}} - N^F_{\text{train}}},$$

such that $\left\{\left(\boldsymbol{X}_i^F, y_i^F\right)\right\} \in \mathscr{D}^F$ and $\left\{\left(\boldsymbol{X}_j^{F-1}, y_j^{F-1}\right)\right\} \in \mathscr{T}^{F-1} \setminus \mathscr{D}^F$ is randomly sampled.

5. Train the sub-model $P^{F-1,\eta_{F-1}}_{\text{KRR}}$ on $\mathscr{D}^{F-1}$. Similar to step 3, train $P^{(F-2,\eta_{F-1})}_{\text{KRR}}$.

6. Repeat steps 4 and 5 recursively until baseline fidelity, $f = f_b$.

The cross-validation scheme that is developed above forms the essential backbone of evaluating the multifidelity models from this dissertation. In Chapter 8, since there is a relaxation on the assumption of nestedness of multifidelity data, for the case of non-nested MFML and o-MFML, this scheme is not used. Apart from that, in all error metric analysis of multifidelity methods, this form of cross-validation is used.

4.6 Error Contours of MFML

The prediction error of the different ML and MFML models can be computed using MAE or RMAE as presented in Eq. (2.27) and Eq. (2.28) respectively. In this section, the RMAE is considered as an indicator of error. All developments in this section can be extended to

MAE without loss of generality. As a further development of assessment methods for multifidelity methods in ML-QC, this section introduces *error contours of MFML* using RMAE. As a conceptual extension of learning curves, error contours of MFML report MFML model error as a function not simply of training samples chosen at a single fidelity but as a function of training samples, N^{f}_{train} and N^{f+1}_{train}, chosen at two consecutive fidelities, f and $f+1$. In other words, the error contour is the RMAE of the MFML model by varying the training samples at two fidelities simultaneously.

As a hypothetical example, consider the error contour of MFML presented in Figure 4.3. The rows correspond to a fixed number of training samples used at fidelity f. The columns of the contour plot depict the number of training samples used at fidelity $f-1$. The color of each block indicates the RMAE of an MFML model built with the training set consisting of N^{f}_{train} and N^{f-1}_{train} with the number of training samples at other fidelities scaled accordingly.

Example 4.4 (MFML error contour data structure)**.** *Consider $N^{F}_{\text{train}} = 2^2$, $F = 5$, and $f_b = 3$, that is a total of 3 fidelities, and $\gamma = 2$. The RMAE of the MFML model built with the training structure $\{N^{5}_{\text{train}}, N^{4}_{\text{train}}, N^{3}_{\text{train}}\} \equiv \{2^2, 2^6, 2^7\}$ would correspond to the second row,fourth column of the error contour plot in Figure 4.3. Similarly, the MFML model corresponding to the third row and sixth column has the training set sizes $\{2^3, 2^7, 2^8\}$. Notice that by definition of the training set size scaling in MFML, the lower triangle of the error contour is not defined.*

The error contours give a better view into the contribution of a multifidelity data structure to model accuracy for a given target fidelity. The investigation of error contours for each fidelity pair indicates, in some sense, the weighted contribution of those fidelities to the overall model. A better understanding of this contribution will aid the choice of N^{f}_{train} for each fidelity that constitutes the MFML model. The analysis of the learning curves for the different values of γ (see section 10.2.1 and section 10.2.2) indicate that the MFML training data structure needs only very little training samples at the higher fidelity. The contribution of each fidelity and the number of training samples at each fidelity is more complex than just the QC-time cost of the fidelity. The error contour effectively helps analyze this contribution.

4.7 The Γ-Curve

The study of the error contours in section 10.2.3 indicates that the multifidelity data structure can provide a high-accuracy model with a much lower number of costly training samples than the conventional MFML data structure approach. Coupled with the results of studying the learning curves for different scaling factors (see sections 10.2.2 and 10.2.3), a new multifidelity approach is proposed: the Γ-*curve*. As Figure 4.2(d) indicates, in the Γ-curve approach, one successively builds MFML-like models with a fixed number of training samples at the costliest fidelity, that is the target fidelity F. The data structure is different from the conventional MFML approach in that one increases, systematically, the number of training samples at the cheaper fidelities on the basis of the scaling factor. Formally, consider a fixed number of training samples at F, that is N^F_{train}. The standard MFML model would be built with a data structure as presented in Example 4.3. This would be followed by increasing N^F_{train} for a fixed γ and thereby the number of training samples at the other fidelities to generate the learning curve. However, in the Γ-curve N^F_{train} is fixed and only γ is varied. This results in the flattened appearance of the training data structure as seen in Figure 4.2(d) with a sharp peak. The collection of the model error versus time-cost of generating the training data for these models is taken together to form the $\Gamma(N^F_{\text{train}})$-curve. This approach is further studied in Chapter 10 for the prediction of vertical excitation energies of several molecules.

Example 4.5 (Γ-curve training structure)**.** *Consider* $N^F_{\text{train}} = 4$*,* $F = 5$*, and* $f_b = 3$*. Then the* $\Gamma(4)$*-curve would be built with the first multifidelity training data structure (in increasing fidelity) as* $\{2^2 \times 4, 2^1 \times 4, 4\}$*. The next point would be built with a training data structure of* $\{3^2 \times 4, 3^1 \times 4, 4\}$ *and so on.*

The study on scaling factors, error contours, and the use of Γ-curve are discussed in Chapter 10. The Γ-curve MFML method is further used to predict excitation energies of 90-atom porphyrin molecules in Chapter 11.

CHAPTER 5

MULTIFIDELITY MACHINE LEARNING FOR MOLECULAR EXCITATION ENERGIES

This chapter is taken from the work published as ref. [142] in the Journal of Chemical Theory and Computation.

Excited states form the basis for understanding various photo-induced processes in physics, chemistry, and the life sciences. A detailed knowledge of their energies and properties is key in uncovering the secrets of the intricate working of many systems. Moreover, in many technical applications such as photovoltaics or light-emitting diodes, excited states play a key role as well. As one particular example, we would like to mention the collection of solar energy by light-harvesting complexes, not only in plants and algae but also in some bacteria [145]. To be more specific, a recent model of a light-harvesting organelle, a chromatophore, includes more than 2000 pigment molecules [16]. In order to determine the flow of excitation energy in such a system or to study its spectroscopic properties, one usually needs to determine the time evolution of the excited states for this large number of molecules [146, 147, 148]. At the same time, however, the determination of the excitation energies needs to be rather accurate since small differences in energy can influence the direction of energy flow, which might be crucial for a proper functioning of the biological process. Hence, for an accurate description of such systems, each single-point calculation usually comes with a high computational cost, which is amplified by the large number of those calculations.

In recent years, machine-learning (ML) techniques have been applied to this area of excitation and excited state calculations, resulting in predictive models that are faster than conventional computational methods [149, 150, 151, 152, 153, 22, 154]. It is a well known fact that it is harder for ML models to learn excited state properties in comparison to ground state properties due to the complex chemistry that arises in excited systems [22, 155, 154]. This requires a wider training data set for excited systems of interest for ML models to achieve reasonable accuracy. Thus, the computational effort to calculate molecular properties is shifted from an on-line calculation during the quantum chemistry computational run to an off-line phase, in which only the training data is generated, and ML models are trained. In such models, it has been commonly observed that the larger the number of training samples, the better the accuracy of prediction [22, 154]. Since excitation energy calculations at high accuracy are expensive to perform, the cost of generating the training data imposes a demanding obstacle to train accurate ML models. Therefore, methods to reduce the necessity for numerous highly accurate but costly calculations to generate the necessary training data are needed.

The primary motivation in this work is thus to reduce the cost involved in generating the training data, without compromising on the accuracy of prediction. The number of training samples and the time to calculate individual training samples jointly contribute to this total cost. Currently, various ways exist to reduce the total cost of generating training data. Some of them fall under the category of selecting optimal molecular conformations for training. For instance, active learning approaches shift the training data generation back to the on-line phase, where training samples are adaptively added to the training set based on estimators of the prediction error or the variance of the constructed model [156, 157]. In contrast, sampling techniques like the "*de novo* exploration" of a potential energy surface [158] or an *ab initio* random structure searching [159] select well distributed molecular samples in the off-line phase. The Δ-ML [29, 160] approach follows a different idea. It is another off-line phase method, but adds a second training set for the same molecular conformations with either the same or a different chemical property, which is typically cheaper to compute. By only learning the difference between the cheaper and more expensive property, the approach results in a prediction error comparable to that of conventional ML methods but for a smaller training set size. It should be noted that the method still requires the same number of samples to be computed for the numerically cheaper and the more costly properties. The Δ-ML method has been used for the prediction of various quantum chemical properties such as potential energy surfaces [134], band gaps [161], and excited state energies [162].

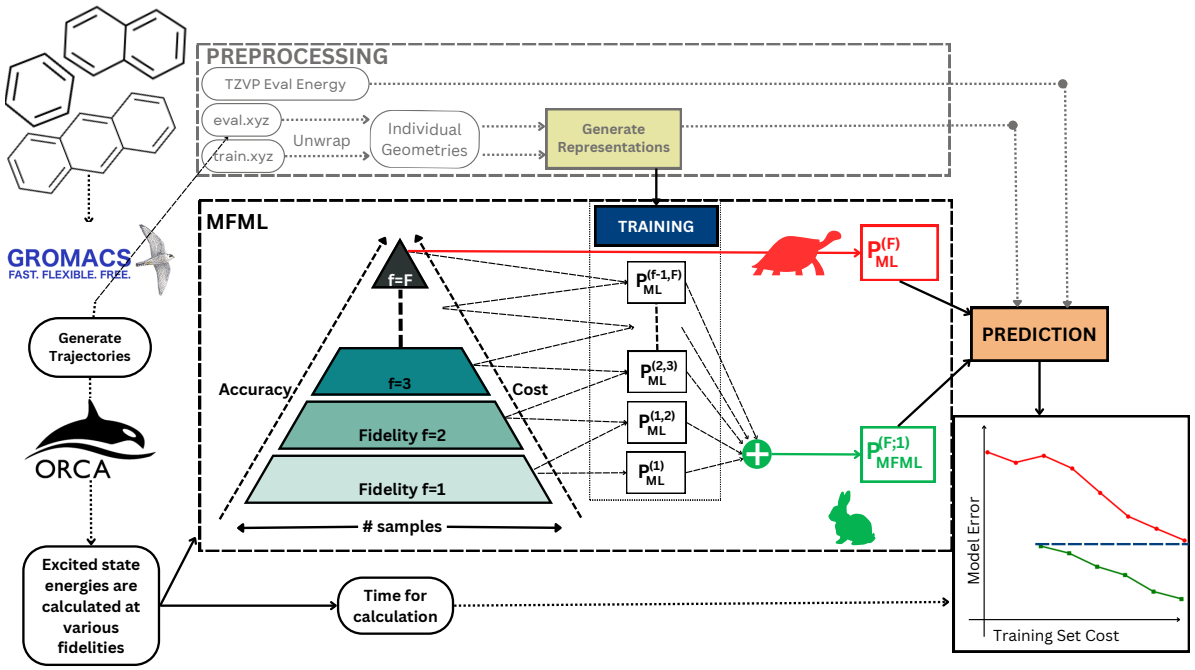


Figure 5.1: Multifidelity machine learning (MFML) based on kernel ridge regression significantly reduces the cost of training an ML model for the prediction of quantum chemistry properties, here, excitation energies. In contrast to the conventional single-fidelity ML method, the discussed method uses data from multiple fidelities with a few highly accurate (and costly) data samples and a growing number of less accurate (hence usually cheaper) data samples, thereby reducing the overall computational cost for the generation of the training data.

In the multifidelity machine learning (MFML) method, also termed Combination Technique Quantum Machine Learning (CQML) [32], it is possible to combine sub-models that utilize a few training samples of the highest fidelity, while using more samples from the cheaper fidelities to achieve the accuracy of a certain target fidelity. The MFML approach is a systematic generalization of the Δ-ML method and exploits the correlations across multiple levels or fidelities in order to determine a certain property. In contrast to the two-level Δ-ML approach, the MFML approach discussed in the present study uses several and not just two levels of calculations in order to enhance the gain in numerical efficiency. Moreover, the numerical efficiency is enhanced by decreasing the number of required training samples at higher levels of accuracy. Previously, the here discussed MFML method has been used for the prediction of atomization energies [32]. Moreover, in Refs. [163, 131] the related two-level multifidelity co-kriging approach is applied for the prediction of band gaps. In contrast, the hML formalism [139] was used for high-accuracy PES reconstructions using multiple Δ-ML models. The number of training samples for each of these models is

optimized by a semi-automatic procedure and has been reported to reduce the numerical cost of the training set generation by a large factor over a final model built with 8 different Δ-ML models [139]. In this respect, the hML approach can be seen as a specific case of the MFML method by fixing the training samples using a semi-automatic optimization. Apart from MFML being built with decreasing sizes of training samples at the higher fidelities, data on the lowest fidelity level is also replaced by an ML model, which eliminates the need to re-calculate the properties on this level during predictions.

The main aim of this work is to further develop and evaluate the MFML method for excitation energy calculations. Earlier work based on the discussed approach [32] further did not quantify the achieved speedup by using MFML. Instead, below, it will be shown that the MFML approach allows to drastically reduce the cost of the training data generation, while achieving the same prediction errors as classical ML models in the field. Thus, in the off-line phase, the number of costlier calculations is substantially reduced, as the numerical results in this work show. As discussed before, MFML combines sub-models that utilize a few training samples of the highest accuracy or *fidelity* with sub-models using more samples from cheaper fidelities to achieve the accuracy of a certain target fidelity. The sub-models are all built using classical ML models. Although various models exist for ML in quantum chemistry, kernel ridge regression (KRR) and neural networks (NNs) are the two most predominantly used methods [22]. The choice of the ML method for this research is KRR. One point is that KRR models are often considered easier to optimize for the prediction [164, 165]. Furthermore, KRR is considered to be less prone to overfitting in comparison to NNs, where external steps like early-stopping and k-fold cross-validation are implemented to prevent overfitting [22]. The regularly reported drawback of the KRR approach lies in the cubic scaling of solving a system of linear equations, but is less threatening to the present application since the desired low error is already achieved for a maximum kernel matrix size of $2^{13} = 8192$.

The use of ML for the prediction of quantum chemistry properties such as the excitation energy to the first excited state requires that the Cartesian geometry of the molecules be transformed into some machine-learnable features. This transformation is achieved by *representations* or *molecular descriptors*. The descriptors which encode the chemical and physical properties of the molecule [44, 166, 167] become the input to the ML model and are then used by the model to find a map between the descriptor and the property to be predicted. For this work, unsorted Coulomb Matrices (CMs) are used. This choice is due to their simplicity and robustness for the type of data used in this work. In the present case, the CMs are not row-sorted since (i) this work does not require invariance under permu-

tations for the implementation as the models are built for individual types of molecules and not across the chemical space, (ii) the ordering of atoms is identical across individual frames of the trajectories for all molecules studied, and (iii) row-sorting the CMs is known to introduce discontinuities that are undesirable [44, 50]. The analysis presented in Figure A.2 of Appendix A further shows empirically that unsorted CMs outperform their sorted counterparts on the data considered in this study.

While the long-term interest is, for example, on light-harvesting complexes containing chlorophyll molecules in complex environments, it is useful to first establish the present method for smaller molecules in gas phase. To be able to do proper benchmarking, one needs a large amount of comparison data at the highest fidelity to be able to compare the results with and without the use of the MFML approach. The calculation of many single-point calculations using a high-level quantum chemistry formalism can, however, get numerically very expensive, and thus gaining experience with the MFML approach for excitation energy calculation using smaller molecules is the way to go. Moreover, we restrain ourselves to the prediction of the excitation energies to the first excited state. From the machine learning perspective, an extension to other excitation energies should be straightforward, though from the view point of quantum chemistry accurate training data will likely be harder to obtain, e.g., a proper identification of the desired state can be cumbersome. As already mentioned above, the very specific application scenario we have in mind are large arrays of porphyrin or chlorophyll complexes, which can be present in artificial or biological light-harvesting systems. To understand the energy flow in such aggregates, the excitation energies to the first excited state need to be known accurately due to the shallow energy landscape in some of these systems. Moreover, some systems contain more than 2000 pigments [16] and dynamical simulations for the systems are envisioned to obtain the (time-dependent) spectroscopic properties [146, 150, 147, 153, 168, 148]. The present work is a first step in the direction of these applications and benchmarks MFML and its effectiveness on three molecules of growing size, namely benzene, naphthalene, and anthracene. The challenge to viably cover the chemical space for such an application is beyond the scope of this work but can include various methods such as active learning approaches [156, 157]. It must be noted that this is a general challenge in the field of ML for quantum chemistry and is not restricted to the work presented herein.

The training data is based on time-dependent trajectories calculated by classical molecular dynamics (MD) and density functional tight-binding (DFTB) theory (see section 5.1). As part of this work, preliminary analyses have been performed on the training data to understand the multifidelity structure. This allows to preempt any possible issues with the

MFML models being built. This is carried out in section 5.2.1 for all the molecules used in this work. The constructed ML models are validated by analyzing learning curves derived from the evaluation of prediction errors on a distinct molecular trajectory. In particular, the work first discusses how the prediction error decreases for a growing number of training samples on the most accurate but also numerically most expensive target fidelity level. As a second analysis, the actual reduction in computation time is quantified. In a first instance of such an analysis, the time cost to generate the training data required to achieve a certain accuracy is studied for a systematic increase in the number of training samples (and thereby the complexity of the model). This allows for clear benchmarks in the case of excitation energies to the first excited state. Depending on the application, the various plots of the 5.2 section show a drastic numerical gain in computational efficiency by over a factor of 30 achieved by the current method compared to classical single-fidelity KRR models. This outcome clearly shows that MFML is a viable choice for much more complex systems and larger data sets, where it is expected to show even stronger performance improvements and time benefits. A general workflow and the multifidelity structure of the MFML method used in this work is depicted in Figure 5.1.

5.1 Calculating Excitation Energies of Arenes

The training data sets for the excitation energies were generated in gas phase for the three arenes, namely, benzene, naphthalene, and anthracene along classical MD and DFTB trajectories mainly based on TD-DFT calculations with various basis sets. In the case of the classical MD simulations, the GAFF force field prepared by the ACEPYPE interface [169] and the GROMACS-2022.3 package [170] were employed to perform the molecular dynamics simulations. First, an energy minimization was performed followed by a 100 ps-long equilibration at 300 K. Subsequently, a 100 ps run was performed. Finally, a 15 ps-long unbiased simulation in gas phase was carried out in which the geometries were stored at every time step, i.e., every 1 fs. This procedure yielded a total of 15,000 frames, which were then utilized for excited state calculations to be used as training as well as evaluation data sets. In case of the DFTB simulations, the 3OB parameter set [171] was employed as chosen in the DFTB+ package version 21.1 [172] and 15 ps-long NVT simulations were carried out for the three molecules. Again, the trajectories were stored using a 1 fs time step, producing 15,000 frames for the excited state calculations. Subsequently, the first excited states of the molecules were determined along the trajectories using the TD-DFT formalism with the CAM-B3LYP functional as implemented in the ORCA package version

4.1.2 [6]. In all cases, five different basis sets according to their quantum chemical hierarchy were employed, i.e., STO-3G, 3-21G, 6-31G, def2-SVP and def2-TZVP. The exact hierarchy of excited state calculations across the 15,000 frames is discussed in section A.1 of Appendix A. In case of benzene, the larger basis set def2-QZVP was tested as well. During these calculations, the Tamm-Dancoff approximation (TDA) approximation was employed together with the Resolution of Identity approximation (RIJCOSX) in order to speed up the calculations. In addition, the computationally cheap semi-empirical methods ZINDO/S-CIS(10,10) and time dependent LC-DFTB [173] were employed to determine the excitation energies of the benzene molecule along both the MD and the DFTB trajectories. In shorthand notation, ZINDO/S-CIS(10,10) and time dependent LC-DFTB are described as ZINDO and LC-DFTB in the respective parts of section 5.2. The ZINDO calculations were performed using the ORCA package, whereas the LC-DFTB calculations were conducted using the DFTB+ package.

5.2 Results

In MFML, cost-efficient models for a given target fidelity of the excitation energy are built. For a large part of this study, the fidelities are given by different basis sets for the excited states using the TD-DFT approach with the CAM-B3LYP functional (see section 5.1). Therefore, the different fidelities are simply named after the basis set (or even a shorthand version thereof). Single-fidelity ML for the most accurate target fidelity, F, with def2-TZVP basis set hence leads to a model, which is denoted by $P_{\text{KRR}}^{(TZVP)}$. Please note that the accuracy of the data increases with the fidelity f, i.e., $f = 1$ denotes the least accurate and $f = F$ the most accurate data. The MFML approach replaces the model $P_{\text{KRR}}^{(\text{TZVP})}$ by a cheaper-to-train model that still targets the same fidelity but contains data from a sequence of fidelities starting from the target fidelity (TZVP) down to a *baseline fidelity*, f_b, e.g., 3-21G. As discussed in section section 4.1, this is mathematically realized by first constructing a single-fidelity model on the level of the baseline fidelity and adding up several Δ-ML type intermediate models for $f_b \leq f < F$, $P_{\text{KRR}}^{(f,f+1)}$ between fidelities, e.g., 6-31G, def2-SVP, leading to

$$P_{\text{MFML}}^{(\text{TZVP;3-21G})} := P_{\text{KRR}}^{(\text{3-21G})} + P_{\text{KRR}}^{(\text{3-21G,6-31G})} + P_{\text{KRR}}^{(\text{6-31G,SVP})} + P_{\text{KRR}}^{(\text{SVP,TZVP})} .$$

Typically, the prediction errors for the three studied molecules benzene, naphthalene, and anthracene are discussed via learning curves. The main objective of this analysis is to show, how the additional cheaper fidelities enhance the prediction. Improvements are

reported with respect to the number of the most expensive training samples (see section 5.2.2) and with respect to the projected total time to generate the training data as discussed in section 5.2.4. The learning curves also indicate how additional training data provides a better prediction accuracy. Since the models were trained individually for separate molecules, no tests of transferability were carried out. The task of transferability across molecule sizes is beyond the scope of this work since this is a first study on MFML for excitation energies.

Since all calculations were performed along both MD and DFTB trajectories, the main manuscript in most cases shows only the results arising from the MD trajectories of the molecules. For each trajectory, the training is performed with the data set structured as follows: on the target fidelity, that is TZVP, $1.5 \times 2^9 = 768$ excitation energies are determined. The factor of 1.5 ensures that the training data is sufficiently different for each random shuffling needed in the model evaluation. For each subsequent lower fidelity, this number is scaled by a factor of 2 thus resulting in $1.5 \times 2^{13} = 12288$ excitation energy calculations at the lowest fidelity, that is STO-3G. The evaluation of the MFML models is performed on a separate holdout set, also called the evaluation set, with energies calculated at the TZVP fidelity (see section 5.1). These conformations from the evaluation set are never used in the training of the model. The plots for the results along the DFTB trajectories are shown in section A.1 of Appendix A. Some final results shown here, however, include results of both MD and DFTB trajectories as evidence of the method implementation across trajectory types. For all analyses performed in this work, the excitation energies at the different fidelities are mean centered. The TZVP energies of the evaluation set are centered by the mean of the training TZVP energies. As a result, the MFML model predicts the centered energies of the target fidelity, TZVP. To get the actual TZVP energies of the evaluation set, one must add the mean of the TZVP energies from the training set $\mathscr{T}^{(F)}$. Centering the energies ensures that the MFML technique does not simply learn the offsets of the different levels of theory employed. This allows the multifidelity models to truly learn the underlying structure of the energies from the various levels of theory.

5.2.1 Multifidelity Structure Analysis

Before using the training data for the MFML approach, some of its characteristics need to be analyzed to understand its structure. Shown in Figure 5.2A are the energy distributions, i.e., the kernel density plots of the energy values, at different fidelities (basis sets) for the three molecules based on the classical MD trajectory. The equivalent results for the DFTB trajectory are shown in Figure A.3. In several cases, the energy distributions have a Gaussian

shape, while in some cases, such as the MD based benzene and DFTB based anthracene, the distributions are bimodal. This bimodal shape of some distributions might be due to the finite number of samples, and could be the result of a limited coverage of the conformation space arising from the use of a single trajectory. It is important to note again that the training set for each fidelity has been individually centered by its mean energy value, which is why the distributions are all centered around an energy of 0 EV. For comparison, the distributions of the uncentered energies are shown in Figure A.4 for both MD and DFTB trajectories of the molecules.

The second type of plots shown in Figure 5.2B are scatter plots between the target fidelity TZVP and the other fidelities, which are present in the training data. For this stage of the analysis, only those molecular conformations were considered, which belong to $\mathscr{X}^{\text{TZVP}}$ (see section 4.1). The conformations from the evaluation set are not considered. This plot helps to understand how the fidelities included in the MFML model deviate from the target fidelity. For the approach to work, one anticipates that the lower fidelities have a systematic distribution with respect to the target fidelity. In the data based on the MD but also the DFTB trajectories of benzene, the points are closely packed for each fidelity and show a nearly linear dependence between the excitation energies. The same is observed for the data based on the DFTB trajectory for naphthalene (see Figure A.3B) and the MD trajectory for anthracene. For the MD-based naphthalene data, the SVP and 6-31G points are relatively close to a line, while the 3-21G and STO-3G results show a much larger spread. The same is the case for the DFTB-based anthracene excitation energies determined using the STO-3G basis set when plotted against the TZVP energies, as can be seen in the third frame of Figure A.3B. Thus, not always the same amount of improvement seems to be present when increasing the basis set size for these cases. For certain molecular conformations, the increase in accuracy is larger than for others. This effect might have to do with the ability to describe the ground and/or excited state molecular orbitals with small basis sets better for some conformations than for others. The relatively large spread in the relationship between the target fidelity and some other fidelities is a first hint that for some combinations of trajectory and basis set, the hierarchy in the accuracies of the different fidelities might be slightly problematic.

To further understand the training data, the mean absolute differences

$$\Delta y_f^{\text{TZVP}} = \frac{1}{N^{\text{TZVP}}} \sum_{i=1}^{N^{\text{TZVP}}} \left| y_i^{\text{TZVP}} - y_i^f \right|$$

were calculated on the centered energies. In Figure 5.2C this quantity is depicted on the vertical axis and the results are shown as a function of the different fidelities while the error

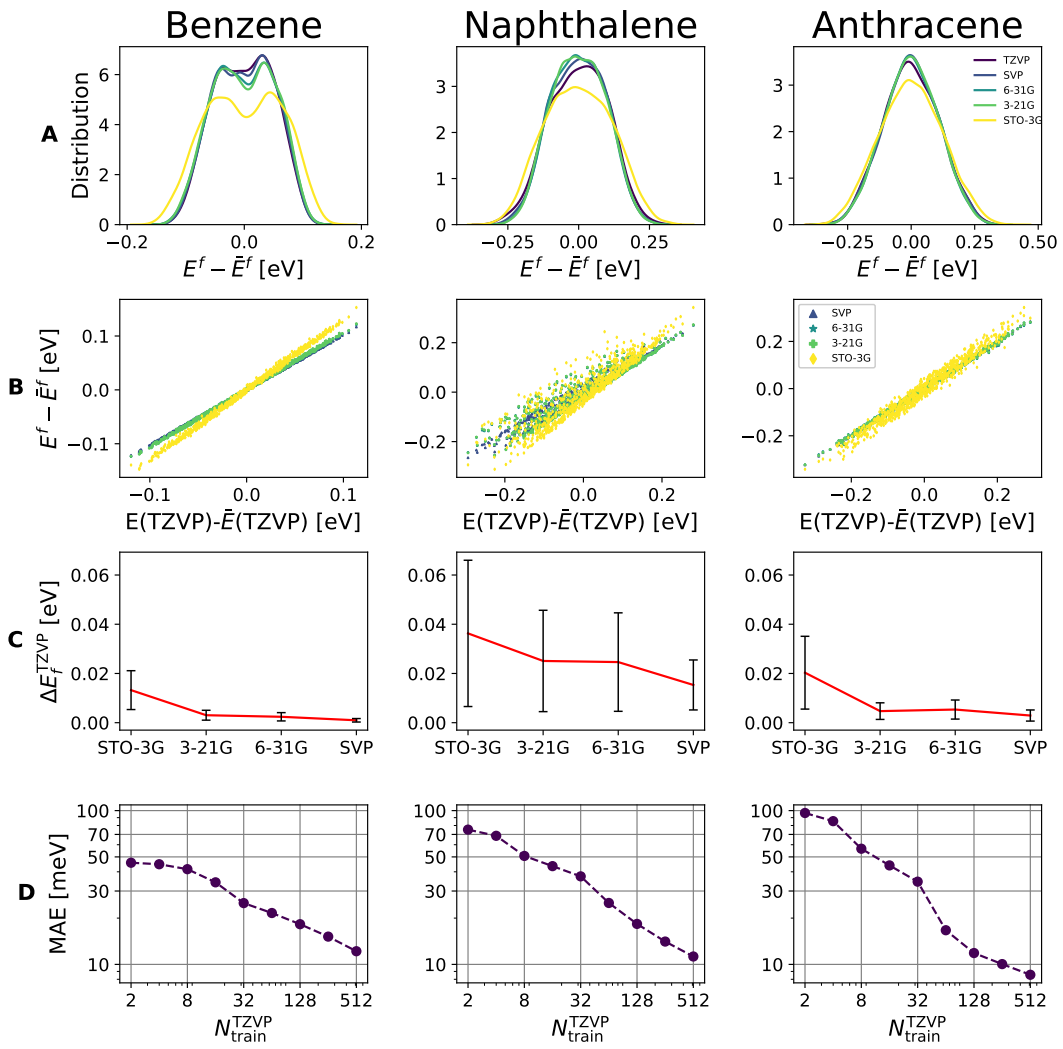


Figure 5.2: A) Energy distributions of the different fidelities (basis sets) in the training sets based on the MD trajectories of benzene, naphthalene, and anthracene. The complete training data for each fidelity is represented in terms of the density plot obtained using the kernel density estimation. B) Scatter plots comparing the excitation energies using the TZVP basis set to the excitation energies at the other fidelities (basis sets) for the conformations in the training data. C) Energy differences (including standard deviations) between the different fidelities and the target fidelity (TZVP) for the conformations in the training data. D) Learning curves for the single-fidelity KRR model presented on a double-logarithmic scale. All results are given for mean centered training sets, thus potential constant shifts between the energies of the various the fidelities were removed.

bars correspond to the standard deviations of the absolute differences. The values of the absolute differences varies across the molecules. In general, it is expected that the differences decay at least monotonically for growing fidelity. Except for the move from the 3-12G to the 6-31G basis set for anthracene, this is always the case. Hence, the ordering of the

different fidelities is appropriate for an MFML scheme. For the case of anthracene, the ordering of 3-21G as a lower and 6-31G as a higher fidelity is still considered, since a larger basis set should usually lead to more accurate results, especially for these rather small basis sets.

For the standard deviations over the differences, denoted by the error bars, it would again be plausible to expect an at least monotonic decay for growing fidelity. Anthracene with the 3-21G and 6-31G fidelities is again the only exception. In general, the standard deviations are small for the MD and DFTB trajectories of benzene. This is also the case for the MD trajectory of anthracene and the DFTB trajectory of naphthalene. In all these cases, the standard deviation of the SVP level of theory is smaller in comparison to those of the other fidelities. For MD-naphthalene, the error bars are large across the various fidelities and thus the error bars span a larger range on the y-axis. The large error bars are numerical indicators of the corresponding wide spread of the predictions found in the scatter plot, as discussed previously. This could be a preliminary indicator that the multifidelity method may not give the expected results for the MD-based naphthalene molecule. A similar observation is made for the DFTB-based anthracene molecule in Figure A.3C, where a large error bar is observed for the fidelity based on the STO-3G level of quantum chemical theory. It could be an indicator that the use of the STO-3G fidelity in the multifidelity structure, might not provide sufficient benefit in this case.

As a final part of the preliminary analysis, the learning curves for single-fidelity KRR models, that is $P_{\mathrm{KRR}}^{(\mathrm{TZVP})}$, are reported. The models are built on the training sets using only the TZVP fidelity and target the same fidelity. Learning curves, averaged over ten randomly shuffled training sets (see section 2.7) have been generated for these models and are presented in Figure 5.2D. It can be seen that these learning curves decay algebraically regardless of the molecule or on which ground state trajectory the excited state results are based. For the DFTB-based benzene, a range of low improvement for smaller training set sizes is observed. This is, however, the pre-asymptotic region and for larger training set sizes the learning curve clearly depicts a reduction in the MAE. For the number of training samples, $N_{\mathrm{train}}^{\mathrm{TZVP}} = 512$, the model for the DFTB-based benzene (see Figure A.5D) reached an MAE comparable to that of the benzene data based on the MD conformations. With 512 training samples at the TZVP level, the models for all molecules reached an MAE of the order of 10 meV. The negative slopes of the learning curves for large $N_{\mathrm{train}}^{\mathrm{TZVP}}$ indicate that further addition of training samples can potentially improve the accuracy of the predictions even further.

5.2.2 Multifidelity Results

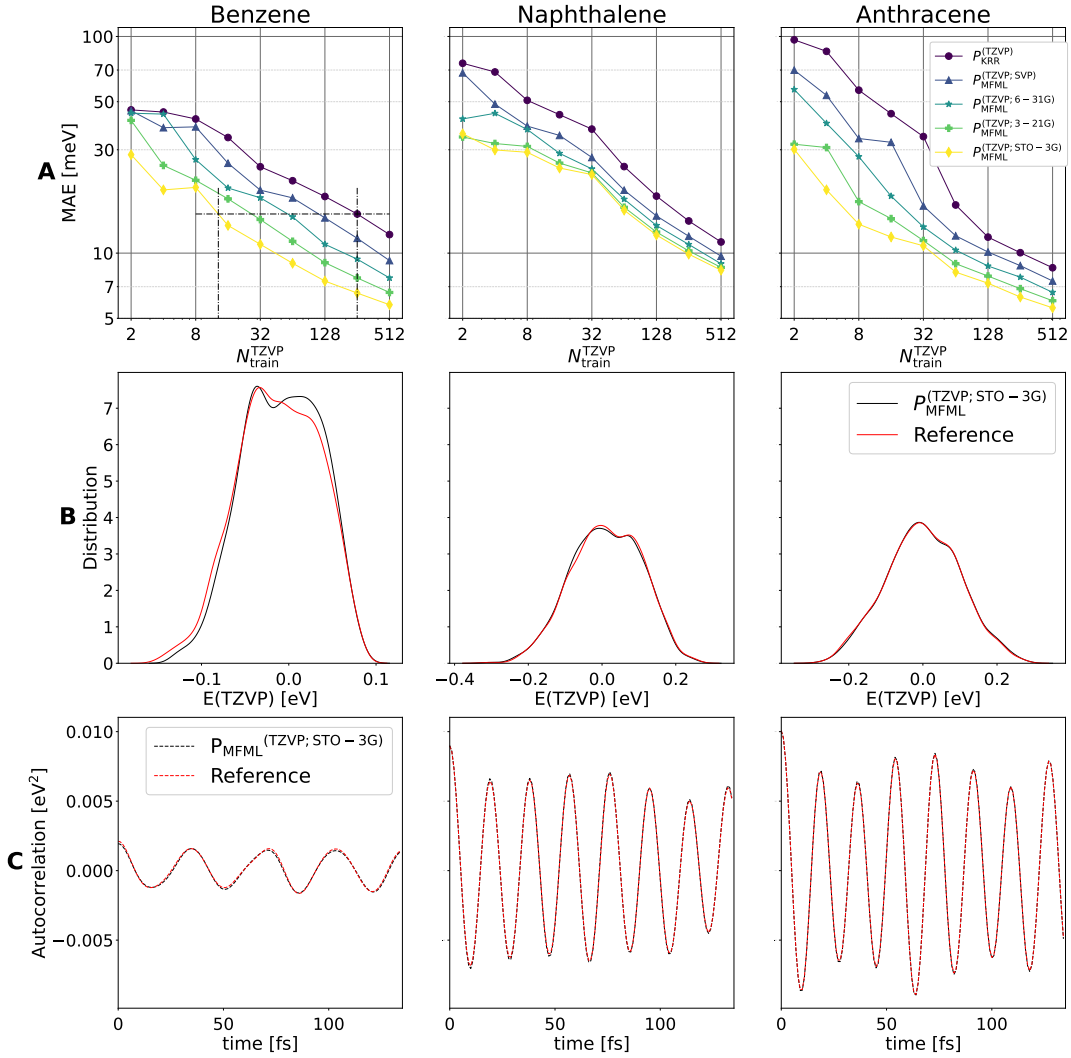


Figure 5.3: The effectiveness of the MFML method is represented through learning curves, while the results for the evaluation set are also analyzed in the time and energy domains. A) Multifidelity learning curves based on the excitation energies along the MD trajectories for benzene, naphthalene, and anthracene. With the addition of lower fidelities, the prediction error decreases, as can be seen in the difference between the standard KRR model (blue) and the MFML model using data from all five fidelities (yellow). B) Energy distributions based on the holdout sets using the TZVP reference calculations (red) and the predictions from the MFML model $P^{(TZVP;STO-3G)}_{MFML}$ (black) for $N^{TZVP}_{train} = 512$. For all molecules, it can be observed that the predictions from the MFML model matches the reference energy distributions accurately. C) The corresponding time autocorrelation functions (ACFs) of the excitation energies. The red lines correspond to the ACFs of the TZVP reference calculations from the holdout set, while the black lines report the ACF of the excitation energies predicted from the MFML model for the conformations belonging to this set.

Continuing with the present study, the results of the MFML approach are shown and discussed next. In Figure 5.3A the MFML learning curves for the MD trajectories are delineated. The top blue line in each of these panels refers to the standard single-fidelity KRR model, as already shown in Figure 5.2D. The other lines correspond to MFML models built using an increasing number of fidelities in the models. In detail, this means that the first MFML model includes the SVP data ($f = 4$) in addition to TZVP. The subsequent models each include an additional fidelity, i.e., 6-31G ($f = 3$), 3-21G ($f = 2$), and STO-3G ($f = 1$). Thus, the most elaborate MFML model includes data from five different excited state calculations, where the number of data points increases by a factor of two when going down in the fidelity. Horizontal and vertical dashed lines are included in Figure 5.3A for the case of benzene to highlight that the addition of cheaper fidelities does in fact reduce the MAE for a given training set size at the target fidelity. The horizontal line is drawn at the MAE value resulting from the single fidelity standard KRR model $P_{\text{KRR}}^{(\text{TZVP})}$ at $N_{\text{train}}^{\text{TZVP}} = 256$. At the position where this line intersects with the line for the MFML model $P_{\text{MFML}}^{(\text{TZVP;STO}-3\text{G})}$, the left vertical dashed line is depicted. The horizontal position of this perpendicular line corresponds to a value of about 16 training samples at the TZVP fidelity. Thus, the error for the MFML model $P_{\text{MFML}}^{(\text{TZVP;STO}-3\text{G})}$ with about 16 training samples at the TZVP level is roughly the same as the one for the single fidelity model $P_{\text{KRR}}^{(\text{TZVP})}$ with 256 data points at the same theory level. Certainly, for the MFML approach, calculations at the other fidelities were involved, i.e., the model is built with the number of training samples $N^f = (2^8, 2^7, 2^6, 2^5, 2^4) = (256, 128, 64, 32, 16)$ at the fidelities $f = (1, 2, 3, 4, 5)$ which are ordered as explained in section 4.1. This finding shows a significant reduction in the number of samples in the numerically costly training data set required to achieve a certain MAE of prediction. In this specific example of the benzene molecule and an MD trajectory using a value of 512 for $N_{\text{train}}^{\text{TZVP}}$, the MAE were 12.2 meV and 5.7 meV for $P_{\text{MFML}}^{(\text{TZVP;STO}-3\text{G})}$ and $P_{\text{KRR}}^{(\text{TZVP})}$, respectively.

The learning curves for the excited states of naphthalene along the MD trajectory also show a clear and systematic offset between the standard KRR model and the MFML models, as can be seen in Figure 5.3A. The addition of the STO-3G fidelity and to some extent of the 3-21G basis set, however, did not improve the model significantly. Despite the offsets between the learning curves, the MAE values for $P_{\text{MFML}}^{(\text{TZVP;3}-21\text{G})}$ and $P_{\text{MFML}}^{(TZVP;STO-3G)}$ are very similar, i.e., 8.6 meV and 8.4 meV, respectively. Already in section 5.2.1, based on the scatter plots in Figure 5.2B and the plot of the absolute differences in Figure 5.2C, it was anticipated that the MFML scheme might not provide perfect results. From the learning curves, it can now be seen that the method did actually work for the present case. However,

the improvement for some fidelity levels was only marginal, while including most of the other fidelity levels did result in an increase in accuracy. For $N_{\text{train}}^{\text{TZVP}} = 512$ the standard KRR model $P_{\text{KRR}}^{(\text{TZVP})}$ yields an MAE of 11.2 meV while $P_{\text{MFML}}^{(\text{TZVP;SVP})}$ and $P_{\text{MFML}}^{(\text{TZVP;6-31G})}$ reach smaller MAE values of 9.6 meV and 8.9 meV, respectively. It is worthwhile to mention that in spite of the irregularities in the data, the multifidelity model $P_{\text{MFML}}^{(\text{TZVP;STO-3G})}$ still results in lower error values than the preceding models. This finding again indicates the robustness of the present MFML method.

For the MD trajectory of anthracene, the learning curves are reported in Figure 5.3A as well. The addition of each cheaper fidelity shows a clear and distinct reduction in the MAE indicating the effectiveness of the approach. For $N_{\text{train}}^{\text{TZVP}} = 512$, the averaged MAE for the standard KRR model, $P_{\text{KRR}}^{(\text{TZVP})}$, was 8.6 meV. In comparison, the multifidelity model, $P_{\text{MFML}}^{(TZVP;STO-3G)}$ resulted in an averaged MAE of 5.5 meV. In addition, Figure A.5A shows the MFML learning curves for the DFTB-based trajectories of the various molecules. For benzene, these show a trend similar to the MD trajectory results. The averaged MAE for $P_{\text{KRR}}^{(\text{TZVP})}$ and $P_{\text{MFML}}^{(TZVP;STO-3G)}$ were 10.8 meV and 6.7 meV, respectively. In the learning curves for DFTB-based naphthalene shown in Figure A.5A, the MFML models built with each additional less accurate fidelity, show lower offsets for various training set sizes. That is, if one considers a vertical line drawn at some $N_{\text{train}}^{\text{TZVP}}$, then the learning curves with the less accurate fidelities fall below the learning curves of the preceding models. There is a jump observed for all learning curves between $N_{\text{train}}^{\text{TZVP}} \approx 16$ and ≈ 32, which is carried forward due to the jump observed in the conventional KRR model. The subsequent multifidelity models were built including the TZVP data, and thus this jump is also included in their results. For $N_{\text{train}}^{\text{TZVP}} = 512$, for example, the averaged MAE for $P_{\text{KRR}}^{(\text{TZVP})}$ is 8.9 meV and for $P_{\text{MFML}}^{(\text{TZVP;STO-3G})}$ 6.5 meV. Therefore, the addition of training samples from the less accurate but numerically cheaper fidelities does in fact reduce the error of the models built for DFTB naphthalene.

For the DFTB-based anthracene, it can be observed that the MFML model does not provide an improvement with the addition of STO-3G fidelity. This is shown in the MFML learning curves on the right-hand side of Figure A.5A. For smaller training set sizes, the learning curve corresponding to the model $P_{\text{MFML}}^{(\text{TZVP;STO-3G})}$ crosses above that of the MFML model built on the 3-21G baseline. However, for larger training set sizes, the robustness of the MFML approach succeeds, resulting in a comparable MAE for $P_{\text{MFML}}^{(\text{TZVP;STO-3G})}$ as can be seen for $N_{\text{train}}^{\text{TZVP}} = 512$. This issue was identified with the high spread of the energies in the scatter plot from Figure A.3 and pointed out in section 5.2.1. The details of this specific case are further elaborated in Appendix A under section A.1.4.

For ready reference, the MAEs of the MFML model $P_{\text{MFML}}^{(\text{TZVP;STO-3G})}$ are listed in Table 5.1.

This table also includes the MAEs resulting from the reference KRR model. It is evident from these numerical values that the MFML models built across molecule size result in errors that are very similar. Even the problematic cases of MD-based naphthalene and DFTB-based anthracene show MAEs which are close to that of the rest of the molecules. This is an indicator that the multifidelity method performs independent of molecule size.

MOLECULE	**MAE [meV]** - $P^{\text{TZVP}}_{\text{KRR}}$	**MAE [meV]** - $P^{(\text{TZVP;STO3G})}_{\text{MFML}}$
MD benzene	12.179	5.778
MD naphthalene	11.257	**8.343**
MD anthracene	8.561	5.592
DFTB benzene	10.822	6.688
DFTB naphthalene	8.982	6.586
DFTB anthracene	8.150	**5.952**

Table 5.1: MAEs of the MFML model built with the STO-3G level of theory as the baseline fidelity. The values are reported for $N^{\text{TZVP}}_{\text{train}} = 512$ with the remaining training samples scaled appropriately across the fidelities. For the special cases of MD-naphthalene and DFTB-anthracene, the MAE of the MFML model are presented in bold.

5.2.3 Predictions in the Energy and Time Domains

Before analyzing the computational costs of the MFML scheme, the results of this approach are analyzed in ways different from learning curves and slightly closer to applications, e.g., in the calculation of spectral densities [148]. Such properties might not really be relevant for molecules in the gas phase, but will become essential once similar calculations will be performed for molecules in non-trivial environments. To this end, Figure 5.3B compares the distributions of the excited states along the MD trajectory. This comparison is performed between the TZVP reference energies from the holdout set with those predicted for the conformations in this evaluation set by the MFML formalism using five fidelities, i.e., the $P^{(\text{TZVP;STO}-3\text{G})}_{\text{MFML}}$ model for $N^{\text{TZVP}}_{\text{train}} = 512$. A visual comparison yields basically no differences for the molecules naphthalene and anthracene, while for benzene, small differences in the peak structure are visible. For most applications, this level of accuracy is certainly more than necessary. Looking again at only the training data at the different fidelities for benzene in Figure 5.2A, it becomes evident that the training data has a bimodal distribution, which translates to the $P^{(\text{TZVP;STO}-3\text{G})}_{\text{MFML}}$ model. If the models were to be trained on a larger dataset, this likely would be smoothed out since a larger section of the conformation space would be covered and the bimodality in the training set most likely would disappear, being an artifact of the small number of training data along a trajectory. A similar agreement is

observed for the MFML model for the DFTB-based trajectories of the molecules, as can be seen in Figure A.5B.

After having analyzed the data in the energy domain, the next step was to have a closer look at the time domain. Instead of looking at individual arbitrary pieces of the trajectory, the autocorrelation function (ACF), which can also be averaged in a meaningful way, was analyzed. The ACF for a discrete time series can be determined as [174]

$$C_m(t_l) = \frac{1}{N-l} \sum_{k=1}^{N-l} \Delta E_m(t_l + t_k) \Delta E_m(t_k), \tag{5.1}$$

where ΔE_m denotes the difference between the excitation energies E_m and the time average $\langle E_m \rangle$, i.e., $\Delta E_m(t) = E_m(t) - \langle E_m \rangle$. Moreover, N represents the number of frames present in the respective part of the trajectory. The initial 2700 frames from the evaluation data set were taken into account for each molecule. These trajectories were divided into ten independent windows, each with 270 conformations. Since a time step of 1 fs was employed, an ACF of a length of 135 fs was constructed using this data. The correlation functions were averaged over the ten windows. In Figure 5.3C the reference data, i.e., excitation energies along the MD trajectory determined using the TZVP fidelity, is compared to the predictions from the $P_{\text{MFML}}^{(\text{TZVP;STO-3G})}$ model built with $N_{\text{train}}^{\text{TZVP}} = 512$. It is clearly visible that the predictions from the MFML model in the case of this averaged ACF reproduce the results obtained at the TZVP level with high accuracy. In Figure A.5C, a similar agreement can be observed for the excitation energies along the DFTB trajectories.

5.2.4 Reduction of Computation Time for Generating Training Data

Finally, as the most important part of the present study, the decrease in the computation time needed to generate the training data when using MFML is studied. To this end, the MAE will not be studied as a function of the number of training samples on the highest fidelity, but as a function of the computation time to generate the complete training data on all hierarchy levels. The average computation times of single point calculations at the different fidelities are reported in Table A.1 for the three studied molecules. Based on this data, the total time to generate the training sets for a given model can be determined as $\sum_{f=f_b}^{F} N^{(f)} \cdot \overline{T}^{(f)}$, where the sum runs from the baseline fidelity f_b up to the target fidelity, F, which in this case is TZVP. In this expression, $N^{(f)}$ denotes the number of training samples used for fidelity f and $\overline{T}^{(f)}$ denotes the corresponding average computation time for the respective single point calculation, as reported in Table A.1. For example, the computational time to generate the training set for benzene to construct the model

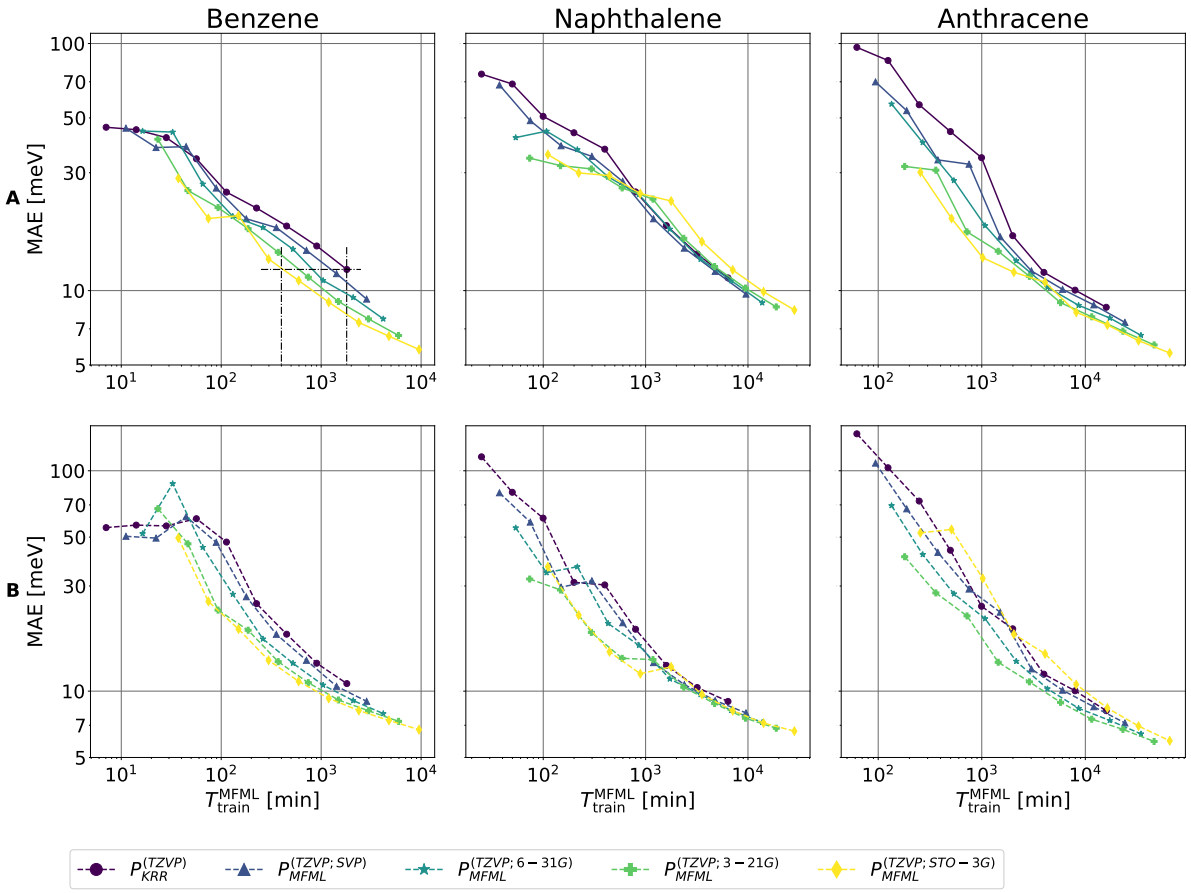


Figure 5.4: Computation times to generate the training data sets versus the MAE of the MFML models, verifying the computational benefits of the MFML models. A) Results for the MD trajectories: With addition of each numerically cheaper fidelity, the training time decreases for a specific MAE, i.e., prediction accuracy. B) Findings for the DFTB trajectories: The time benefits are clearly visible for the various molecules across the fidelities. The efficiency of the multifidelity method is numerically visible for the DFTB trajectories.

$P_{\text{MFML}}^{(\text{TZVP;6-31G})}$ with $N_{\text{train}}^{\text{TZVP}} = 4$ training samples at the target fidelity can be estimated to be $T_{\text{train}}^{\text{MFML}} = 4 \times 3.53\ \text{min} + 8 \times 1.02\ \text{min} + 16 \times 0.65\ \text{min} \approx 32\ \text{min}$.

Shown in Figure 5.4A are the MAEs as a function of the the projected computation times for generating the training data sets for conformations of the three molecules benzene, naphthalene, and anthracene along the classical MD trajectory. For the case of benzene, dashed lines have been included again to help to interpret the data. For $N_{\text{train}}^{\text{TZVP}} = 512$, the projected computation time to generate the training set only at the target fidelity is approximately 1800 minutes. Using this data, one can generate the standard KRR model $P_{\text{KRR}}^{(\text{TZVP})}$. The horizontal line shows that a similar MAE can be achieved for the MFML model $P_{\text{MFML}}^{(\text{TZVP;STO-3G})}$ for which the projected time to generate the training set is about 400 min-

utes. Thus, the MFML method provides a factor of roughly 4.5 in reducing the computation time required for the generation of the training set. The plot of time to generate the training data set versus the MAE for the MD-based naphthalene conformations in Figure 5.4A shows again that the MFML model is affected if the distribution of the fidelities with respect to the target fidelity have a wide spread and if the absolute differences are unexpectedly large as was explained in section 5.2.1. Thus, it becomes all the more important to ensure that the employed training data follows the assumed hierarchy of basis set sizes or quantum chemistry methods and does not show any anomaly in their respective distributions. While the MFML approach still remains robust, the cost of the training data generation might not always follow suit. In addition, the time to generate the training data set versus MAE for anthracene based on the MD trajectory given in Figure 5.4A reflects the results of the corresponding learning curves in Figure 5.3A. A computational cost reduction in the training data generation time is observed across the multifidelity models. The standard KRR model $P_{\text{KRR}}^{(\text{TZVP})}$ at $N_{\text{train}}^{\text{TZVP}} = 512$ yields a projected time of about 16000 minutes while the multifidelity model $P_{\text{MFML}}^{(\text{TZVP;STO-3G})}$ gives a similar error with a training set generation time of roughly 7000 minutes, which results in a cost reduction by a factor of about 2.3 across the multifidelity model. The model $P_{\text{MFML}}^{(\text{TZVP;3-21G})}$ results in a similar error for a training set with a projected time of generation roughly 6000 minutes, which corresponds to a time benefit factor of about 2.7 for the training data generation cost.

Similarly, for the molecules based on the DFTB trajectory, the time to generate the training data sets versus the error in prediction is shown in Figure 5.4B. For benzene, the benefit of the MFML is evident. If one draws reference lines again for this plot, one observes that the projected time to generate the training set for benzene to train the model $P_{\text{KRR}}^{(\text{TZVP})}$ is about 1800 minutes, whereas the time to generate the training set for $P_{\text{MFML}}^{(\text{TZVP;STO-3G})}$ to result in a similar MAE is roughly 600. This corresponds to a saving in the computational time by a factor of 3.

For DFTB-based naphthalene in Figure 5.4B, one observes that in the case of $N_{\text{train}}^{\text{TZVP}} = 32$ a jump occurs for the $P_{\text{MFML}}^{(\text{TZVP;STO-3G})}$ model. This jump occurs due to a jump which is already present in the MAE values for $P_{\text{KRR}}^{(\text{TZVP})}$ as shown in Figure A.3D. It can be understood, since the TZVP data is contained in both models. One has to notice, however, that the computational time to generate 512 TZVP training samples is close to 6400 minutes. The $P_{\text{MFML}}^{(TZVP;STO-3G)}$ model results in a similar error for a computational time for generating the training data set of about 3000 minutes. The MFML model thus reduces the time cost for the generation of the training set by a factor of about 2 for DFTB-based naphthalene.

As can be seen in Figure 5.4B, for the DFTB-based trajectory of anthracene, the MFML

model $P_{\text{MFML}}^{(\text{TZVP;STO}-3\text{G})}$ performs poorly and therefore does not provide any cost reduction. As discussed in section 5.2.1, this is due to the scattered nature of the STO-3G data, which is depicted in Figure A.3B. The effect of such a wide scatter on the difference models is further explained in section A.1 of Appendix A. For the other MFML models of the same system, a reduction in computational training time is still clearly visible. The time to generate 512 training samples at the TZVP fidelity is about 16000 minutes. The prediction error achieved by $P_{\text{KRR}}^{(\text{TZVP})}$ for this training size can be achieved by the $P_{\text{MFML}}^{(\text{TZVP};3-21\text{G})}$ MFML model for a training data set with a computational time of roughly 7000 minutes. This corresponds to cost reduction by a factor of 2.3 in the training data generation time resulting while achieving a similar accuracy. This example once more shows the need for a clear distribution of the energies of different levels of theory with respect to the target fidelity for the multifidelity method to be effective.

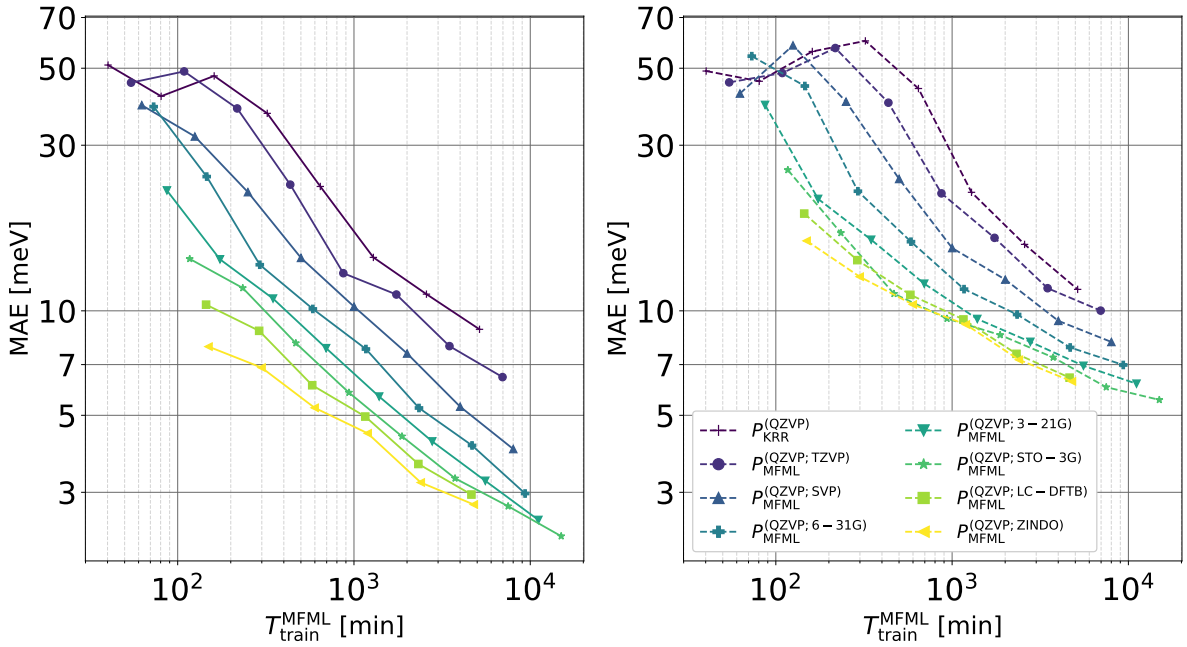


Figure 5.5: Computational time to generate the MFML training set versus the MAE for benzene. The results for the MD-based trajectory are presented on the left-hand side, while the right-hand side shows the results for the DFTB-based trajectory. The target fidelity is set to QZVP. Additionally, two semi-empirical methods, ZINDO and LC-DFTB were employed. For each numerically cheaper fidelity that is added into the model, clear offsets of the learning curves can be observed.

5.2.4.1 Additional levels of fidelity

Calculating excited state energies for naphthalene and anthracene along trajectories using TD-DFT with basis sets larger than TZVP, becomes numerically quite expensive. Thus, the analysis was only furthered for the two trajectories of benzene by training the MFML model to predict the excitation energy to the first excited state using the def2-QZVP basis set for the TD-DFT calculations, which is considerably larger than the def2-TZVP. Thus, it is numerically costlier to calculate the excited state energy for this fidelity. In addition to these more accurate calculations for the excited states of benzene, two semi-empirical methods, namely, LC-DFTB and ZINDO were used. It is assumed that ZINDO is the least accurate approach, followed by LC-DFTB and then TD-DFT with the CAM-B3LYP functional and the diverse basis sets as studied above. Figure 5.5 shows the plots for the computational time for the generation of the training data versus the MAE for benzene along the MD and the DFTB trajectories. In this case, the models are built to target the numerically most expensive QZVP fidelity.

First, we consider the plot for MD-based benzene. Already with two training samples at QZVP, the MFML model $P_{\text{MFML}}^{(\text{QZVP;ZINDO})}$ outperforms the standard KRR model $P_{\text{KRR}}^{(\text{QZVP})}$ with 256 training samples. On comparing the time required to generate the training set, one observes that the standard KRR model required about 5000 minutes while the MFML model achieves a lower MAE for about 150 minutes. This represents a time benefit of over a factor of 30. If the trend of the MAE for the KRR model were to continue, i.e., if the curve for $P_{\text{KRR}}^{(\text{QZVP})}$ would be slightly extended, we extrapolate a time benefit of over 50 while using the MFML model.

For the benzene conformations along the DFTB trajectory, although the MAEs are slightly larger than those for the data based on the MD trajectory, the time benefit is about a factor of 17 with roughly 5000 versus 300 minutes for the $P_{\text{KRR}}^{(\text{QZVP})}$ and the $P_{\text{MFML}}^{(\text{QZVP;ZINDO})}$ models, respectively. Even the MFML model $P_{\text{MFML}}^{(\text{QZVP;STO-3G})}$ yields a computational time benefit of larger than 10-fold for the DFTB and MD trajectories of benzene. Thus, it becomes evident that for the presented MFML approach, the savings in computational time for the training data set generation tend to be larger when the most accurate fidelity is numerically much more expensive than the lower fidelity levels.

5.3 Conclusion

This chapter has introduced the utilization of MFML models for the prediction of excited states, here with a focus on the first excited state. With molecules of various sizes, it has

been shown that if the multifidelity data structure has a clear distribution with respect to the target level of theory, this method does in fact reduce the computation time for training the models while improving the accuracy of the predictions. The work presented herein shows that the various ML methods can be made more efficient with the use of data in a multifidelity structure. While in this work, the method is applied to the vertical excitation energies to the first excited state, the overall method can be applied to any property where a hierarchy of training data can be established. MFML is certainly not restricted to the dimension of the basis set and can be generalized to multiple dimensions [32]. The numerical gain using the MFML approach increases with the difference in numerical effort for the single-point calculations for the individual levels. It is easy to foresee that when using high-level quantum approaches for excited states like multi-configurational approaches, equation-of-motion coupled-cluster models, or quantum Monte Carlo schemes [175] as the target fidelity, the numerical gain in using the MFML approach for the training data generation can be tremendously.

Optimizing the factor that scales the number of data points between the fidelities is another interesting point for future research in MFML. Here, one should systematically assess the effect of the ratio $\gamma = N_{f+1}/N_f$ for all $f_b \leq f < F$ on the prediction errors. Understanding this relationship can potentially lead to an approach that further reduces the time required to generate the training sets. Chapter 10 analyses this relationship in detail and shows that the use of different values of γ does in fact reduce the cost of generating training sets.

Overall, this chapter has numerically shown that for unseen data, the MFML method can predict the excitation energies with a high level of accuracy as made evident by the learning curves. In tests, distributions of the respective excitation energies and their time correlations along dynamical trajectories have been accurately reproduced, thereby being a strong contender to high-accuracy low-cost ML models for excited state properties. Specifically, for the excitation energy to the first excited state, this method has achieved a time reduction by a factor of 30 and more. In case one wants to achieve highly accurate excitation energy gaps by electronic structure methods which might scale with higher powers in the number of atoms such as coupled cluster theories, the MFML approach very likely will lead to even much larger numerical gain factors. Combining this with the wish to do such calculations along trajectories, e.g., of a chromatophore with more than 2000 pigment molecules, gives an idea how large the reduction in numerical cost reduction might become. The same will be true if one wants to determine excited state potential energy surfaces or perform non-adiabatic dynamics.

CHAPTER 6

OPTIMIZED MULTIFIDELITY MACHINE LEARNING FOR QUANTUM CHEMISTRY

This chapter is taken from the work published as ref. [143] in the Journal of Machine Learning: Science and Technology.

Fast and accurate calculations of chemical properties have become increasingly accessible to the community of QC in recent years with the accelerated development of ML for QC [155, 176, 154, 31] as well as improvements in computer hardware. Various supervised and unsupervised learning approaches have seen widespread application in the field of QC. These applications include areas of material design and discovery [177, 178, 179, 180, 181, 182, 22, 30, 31] excitation energies [165, 154, 162, 183, 142], potential energy surfaces [184, 185, 186, 157, 139, 134, 187], and even the prediction of chemical reactions [188] as well as ML molecular dynamics for the simulation of infrared spectra [189]. The usually numerically expensive QC calculations are gradually being replaced by ML models or hybrids of ML and QC resulting in a drastic reduction of the compute cost associated with chemical design and discovery. The core principle common to the various ML techniques is the aim to reproduce some implicit mapping between the geometry of the molecules to some property of interest such as atomization or excitation energies and even complete potential energy surfaces. These quantities are usually targeted at some level of theory which is relevant to the area of application.

The general ML-QC pipeline for such applications begins with the generation of raw

data consisting of the Cartesian geometries of the molecules of interest and the QC calculation property to be predicted at a specific level of theory, e.g. MP2 or CCSD(T) [190], that is deemed accurate for the application. The Cartesian coordinates are then transformed into some input feature format, called *representations* or *molecular descriptors*, that the ML models can map to the property of interest. In the recent past, much work has been dedicated to the development of such representations. These include molecule-wise descriptors which encode the entire molecule, e.g. inverse distance representations and their extensions such as the CM [44, 49, 49, 125, 166, 29, 191] and BoB [192, 193, 194]. The other category of representations are referred to as atom-wise descriptors and they encode the atoms of each molecule in their respective environment. Commonly used atom-wise descriptors include SOAP [195, 192], SLATM [51], permutationally invariant polynomials (PIP) [134], the PaiNN representation [47], and the Faber-Christensen-Huang-Lilienfeld (FCHL) representation [44, 49, 196, 176]. Significant research has also been performed on using other types of representations such as SMILES strings [197, 198], graph-based representations [19], and representations that are either generated with neural network (NN) models such as the Deep Tensor NN [199, 191, 200] or are generated *ad hoc* [201, 202]. Once machine interpretable features are generated, any of the various ML methods such as kernel ridge regression (KRR), Gaussian Process Regression (GPR), or NN models such as ANI [203, 204], SchNet [199, 191] and PhysNet [205], can be used to map the input features to their respective QC properties.

Within such frameworks, it has been a common observation that the higher the number of training samples, the better the accuracy of the prediction. However, a high cost is associated with generating this training data, since conventional electronic structure calculations with at a high level of accuracy are expensive to generate. Thus, the compute cost associated with discovery in QC is shifted from conventional QC calculations to the cost associated with generating the training data sets for these ML models. While any of the aforementioned ML methods is a promising candidate to replacing the time-consuming conventional calculations, only rather recently the cost of the training data generation for the various ML models has been investigated [206, 32, 139, 142]. Previously, various techniques and models have been implemented to reduce this cost. Among these are methods such as the Δ-ML [29, 131, 163, 134, 207, 208], and active learning approaches [156, 158]. An *ad hoc* optimization procedure for the Δ-ML method has been implemented for the ground state potential energy surface reconstruction of CH_3Cl, termed hierarchical-ML (h-ML) [139]. Based on the CPU compute time of single point calculations, the training samples to be used at various fidelities are selected by minimizing an objective function. This reduces the

number of electronic structure calculations needed to generate the multifidelity data set for some user-defined target error.

The previous chapter already dealt with the application of the systematic generalization of Δ-ML method, called MFML, to the prediction of excitation energies of arenes. The MFML method exploits the existence of varying levels of accuracy of conventional QC methods, thereby resulting in a hierarchy of methods for properties such as excitation energies. MFML reduces the number of expensive training samples needed by training on the difference of various *fidelities* between a *baseline fidelity* and the *target fidelity*. The MFML model is built by iteratively adding models built on the difference between the excitation energies calculated at the various fidelities. In MFML, the number of training samples is decreased by a factor of two for each subsequent more costly fidelity [142]. Thus, there is an inherent decrease in the number of costly training samples. For each fidelity and training set size at this corresponding fidelity, a *sub-model*, for a given training set size, is trained [32]. This is recursively performed from a *baseline* fidelity (cheaper and less accurate) up to the target fidelity (expensive and more accurate). The various sub-models are combined to give the final MFML model. This combination was performed based on the sparse-grid combination technique [34, 209, 36, 35, 37, 38, 38] as has been discussed in ref. [32] and in section 4.1.

This chapter furthers the methodological research in MFML by introducing a novel method of optimally combining the various sub-models built on the different fidelities. The novel approach is inspired by Refs. [210, 33] where an optimized sparse-grid combination technique is introduced and discussed for the solution of partial differential equations. In contrast to that work, here, it is applied to ML for QC where the optimal combination of the sub-models is performed with respect to a validation set of the property of interest, not based on intrinsic approximation properties of the given problem. This results in a multifidelity model that predicts the property at the target fidelity with improved accuracy (see Section 6.2). Thus, the optimized MFML (o-MFML) presents an optimal linear combination of the sub-models. The present study benchmarks this novel method on the QM7b dataset with the prediction of atomization energies at the CCSD(T) level of theory with the cc-pVDZ basis set [49, 32]. Further benchmarking is carried out on the excitation energy of arenes from the previous chapter. The results indicate that the o-MFML approach is indeed superior to the conventional MFML scheme.

While the core methodological concepts of the o-MFML method are derived and presented in section 4.2, this chapter discusses its implementation. The chapter is structured as follows. A brief overview of the data used for this study is reported in Section 6.1.1. Sub-

sequently, various results of the comparison of MFML and o-MFML for the two datasets are delineated. In Section 6.2.1 the results for the benchmark on the QM7b dataset [49, 49] are reported, while in Section 6.2.2 the corresponding results for the excitation energy predictions are delineated.

6.1 Dataset

6.1.1 Atomization energies of QM7b

The effectiveness of the optimized MFML method is benchmarked on two independent datasets. Firstly, it is employed for the prediction of atomization energies of the QM7b dataset [49], which consists of a total of 7211 molecules with up to seven heavy atoms. The atomization energies for each of these molecules were calculated in units of kcal/mol as mentioned in Ref. [32]. The atomization energy of a molecule is the energy required to completely dissociate all the bonds of the molecule. That is, the energy required to break a molecule into its constituent unbound atoms. From the original dataset given in Ref. [32], only the MP2 [211, 123, 212] and CCSD(T) [121, 122, 15] levels of theory are considered in this work. The fidelity structure was formed by evaluating these with three varying basis set sizes, namely: STO-3G, 6-31G, and cc-pVDZ (with increasing size). While the original use of this dataset in Ref. [32] considers a 3-dimensional multifidelity structure, in this work these are flattened into a 2-dimensional multifidelity structure. In this work, the multifidelity structure is built with the assumption that by using the basis sets in the order of their size, for a choice of basis set, the CCSD(T) level of theory is more accurate. Thus the order of the fidelities in the assumed hierarchy was taken as MP2–STO-3G, MP2–6-31G, MP2–cc-pVDZ, CCSD(T)–STO-3G, CCSD(T)–6-31G, and CCSD(T)–cc-pVDZ. The CCSD(T)–cc-pVDZ combination is set as the target fidelity, i.e., as the highest level of accuracy. Out of the total set of 7211 molecules, $1.5 \times 2^7 = 6144$ molecules were randomly chosen as the training set. Of the 1067 molecules which remained after separating the training data, 367 were randomly sampled and used as the validation set along with their atomization energies calculated at the CCSD(T)–cc-pVDZ fidelity. The remaining 700 molecules and their atomization energies at the target fidelity were utilized as the test set. For the form of multifidelity models used in this work, the work in Chapter 5 recommends performing a preliminary analysis to verify hierarchy structures. It is to be noted that such analysis can only be made with respect to computational methods, and the use of experimental data in such multifidelity structures is not considered therein. Since the atomization energy dataset is

taken from Ref. [32] where the MFML method was already established for a similar hierarchy, the hierarchy assumed above is used. Certainly exceptions may arise in accuracy to the experimental data should they be considered, as has been shown in Ref. [213].

6.1.2 Excitation energies

Secondly, o-MFML is shown to be effective for the prediction of excitation energies on a separate and independent dataset carried forward from Chapter 5. This data is either based on density functional tight-binding (DFTB) or on classical molecular dynamics (MD) simulations for benzene, naphthalene, and anthracene. For each, a 15 ps-long trajectory was generated after energy minimization and equilibration. The trajectories were saved every 1 fs giving 15,000 frames which were subsequently employed as input for the excitation energy calculations using time-dependent density functional theory (TD-DFT) together with the CAM-B3LYP functional. The resulting excitation energies served for training and evaluation. For training, the first $N_{\text{train}} = 1.5 \times 2^{13} = 12288$ frames were used with excitation energies calculated at five fidelities, i.e., basis sets: def2-TZVP, def2-SVP, 6-31G, 3-21G, and STO-3G. The sampling and calculations are identical to those discussed in section 5.1. This work uses the same number of fidelities provided in the original dataset. These fidelities are calculated with the TD-DFT level of theory with the CAM-B3LYP functional. Five basis sets of increasing quantum chemical hierarchy were used for these calculations to give the hierarchy. Namely, STO-3G, 3-21G, 6-31G, def2-SVP, and def2-TZVP. For the rest of this work, the fidelities of the excitation energies are simply referred to by the basis sets or their short-hand notations such as TZVP for def2-TZVP. No further calculations are performed to increase the number of fidelities or achieve higher accuracy for the excitation energies, since the QC calculations for excitation energies is computationally expensive and often scales in polynomial order with number of atoms [22, 155, 154]. Therefore, not only is excitation energy data scarce, it is also extremely tedious to add to existing data. For each molecule, the 2712 samples with the target fidelity of TZVP were randomly split into 712 and 2000 samples for the validation and test set respectively. The random sampling was performed using the Scikit-learn package [214].

6.2 Results

The effectiveness of the optimized MFML method is benchmarked on two independent datasets. Firstly, it is employed to the prediction of atomization energies of the QM7b dataset.

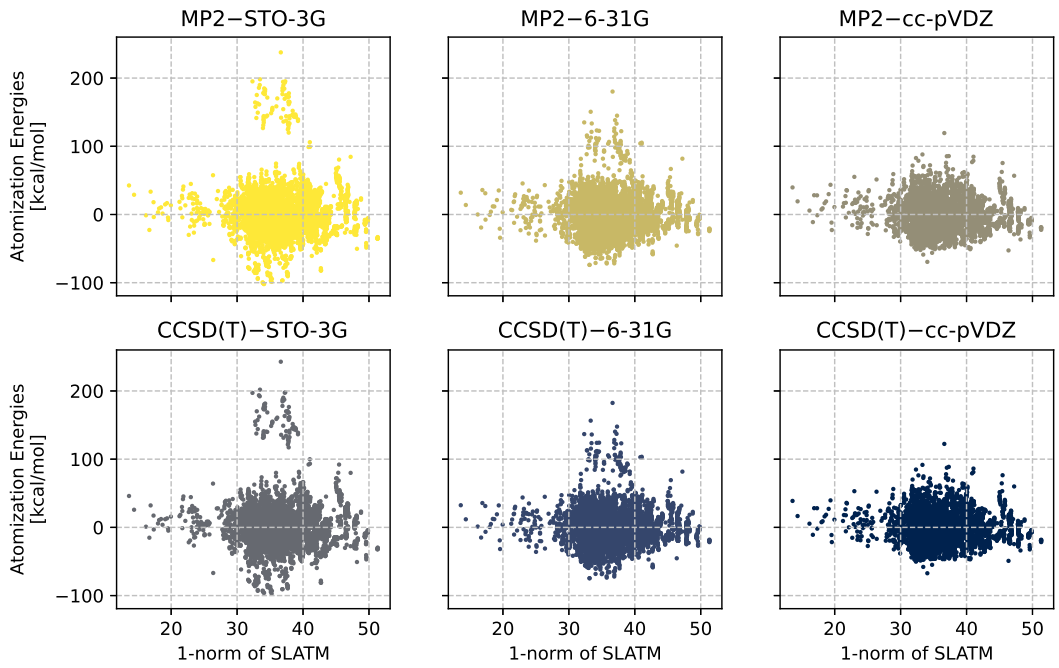


Figure 6.1: Scatter plot of the various fidelities from the training data with respect to the 1-norm of the corresponding SLATM representation [51]. The SLATM representation serves as a proxy to the chemical space. Thus, these scatter plots represent the spread of the atomization energies across the chemical space. The first row corresponds to the MP2 level of theory for increasing basis set sizes. Similarly, the second row displays the scatter plots for the CCSD(T) level of theory.

In particular, this work reports the prediction of atomization energies for the QM7b dataset as calculated in Ref. [32], and the prediction of the first excitation energies for the data used in Chapter 5. The process of the kernel generation and training of the KRR for the work recorded here are carried out with the QML package [66].

6.2.1 Atomization Energy Prediction on QM7b

Previous work by Zaspel *et al.* already provided a benchmark for the MFML method in predicting atomization energies for various molecules in the QM7b dataset [32]. The same values for the Laplacian kernel width of 400, and for the regularization of 10^{-10} have been used in this work to maintain uniformity for the comparison between MFML and o-MFML. The updated o-MFML models are benchmarked on the same dataset with modifications as reported in Section 6.1.1. The hyperparameters of KRR are chosen to be identical to the values reported in the previous work. In total, six fidelities are considered with the target fidelity of CCSD(T)–cc-pVDZ being numerically most costly and the MP2-STO3G fidelity

being the cheapest one. In essence, the order of the fidelities in the ascending order of accuracy is MP2–STO-3G, MP2–6-31G, MP2–cc-pVDZ, CCSD(T)–STO-3G, CCSD(T)–6-31G, and CCSD(T)–cc-pVDZ.

As a preliminary analysis, the scatter plot between the 1-norm of SLATM representations [51] and the atomization energies of the molecules from the training set is depicted in Figure 6.1. This plot assists in understanding the layout of the chemical space by studying the proxy of the chemical space, which in this case is the SLATM representation. On comparing the distribution across the basis sets, that is, row-wise, one observes that increasing basis set size results in clearer separation of the atomization energies across the proxy chemical space. The higher energy clusters become clearer. A similar comparison for an increasing level of theory shows visible differences only for the cc-pVDZ basis set. Here, the CCSD(T) level of theory further separates clusters of molecules in comparison to the MP2 level of theory, especially for those with atomization energies in the region of -100 Kcal/mol. For increasing accuracy to the target fidelity of CCSD(T)–cc-pVDZ, one observes that the scatter plot of the energies with respect to the chemical space gets closer to that of the target fidelity. The smallest basis set STO-3G does not show any atomization energies higher than 100 kcal/mol for both MP2 and CCSD(T) levels of theory. One observes that each increasing fidelity results in a clearer, more distinct categorization of the molecules in the QM7b dataset, which was previously discussed in Ref. [32] with respect to the 1-norm of the CM. The STO-3G basis sets fail to provide any form of information of the separation of the clusters of molecules. The scatter plot of the fidelities with this basis set show a strong clustering around the 0 kcal/mol mark. For the larger basis sets, one observes that higher atomization energies show two distinct clusters. A large one around the 0 kcal/mol mark and another around the 150 kcal/mol mark. As identified in Ref. [32], these correspond to the largest molecules of the QM7b dataset. Since this information is missing from the smaller STO-3G basis set, one anticipates that the use of the fidelities MP2–STO-3G and CCSD(T)–STO-3G in the conventional MFML would provide little to no benefit in predicting the atomization energies at the target fidelity of CCSD(T)–cc-pVDZ where the clustering is all the more distinct.

The resulting learning curves of the multifidelity analysis on the QM7b data are shown in Figure 6.2. All the sub-models for the MFML and o-MFML methods were built with KRR using the Laplacian Kernel, a regularization strength of 10^{-10} and a kernel width of 400 as prescribed in Ref. [32]. The left panel of the figure depicts the learning curves for the conventional MFML method with default coefficients for the sub-models. The learning curves for o-MFML method are depicted in the right panel of the same figure. The conventional

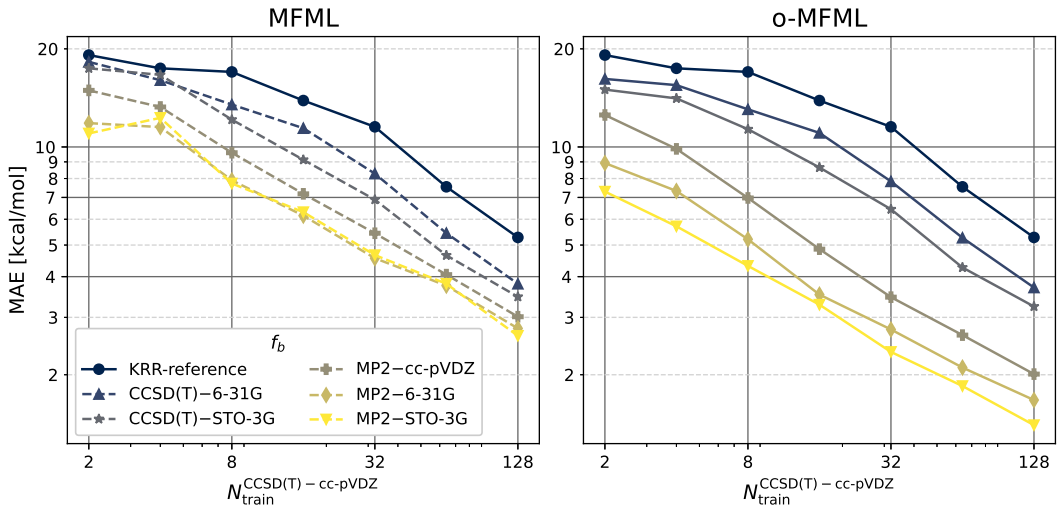


Figure 6.2: Various learning curves for the prediction of atomization energies of molecules in the QM7b dataset. The left panel corresponds to learning curves built with the conventional MFML method, that is $P_{\text{MFML}}^{(F;f_b)}$. The o-MFML models optimized with OLS, referred to as $P_{\text{o-MFML}}^{(F;f_b)}$, are delineated in the right panel. In both cases, each curve corresponds to a model where the target fidelity F is CCSD(T)–cc-pVDZ. The various baseline fidelities f_b are as shown in the figure legend. The learning curve for the conventional KRR model (KRR-reference) is also shown for reference.

reference KRR learning curve is presented in both panes for reference. The horizontal axis denotes the number of training samples used at the target fidelity in training the various models. For conventional MFML learning curves, one observes distinct lowered offsets of the learning curves with decreasing baseline fidelities. As preemptively discussed in the preliminary analysis, the addition of MP2-STO-3G fidelity does not provide any perceivable benefit to the MFML model. The model built on the CCSD(T)–STO-3G baseline, however, does show improvement. A possible reason for this could be that the test set includes molecules with larger atomization energies, the information for which might be absent from the training set. This indeed appears to be the case as is seen in the various distribution plots of Figure 6.1. In other words, the MFML model does not see any new information to learn in the addition of the MP2–STO-3G fidelity.

The learning curves for the o-MFML models are presented on the right-hand side of Figure 6.2. Firstly, one observes that even for smaller training set sizes, the o-MFML model does not show any pre-asymptotic fluctuation. The MAE of the various models always decreases for increasing training samples. This is contrasted to the conventional MFML method where a pre-asymptotic region exits wherein the MAE of the model built with f_b = MP2–STO-3G fluctuates before settling down. In other words, there is a constantly

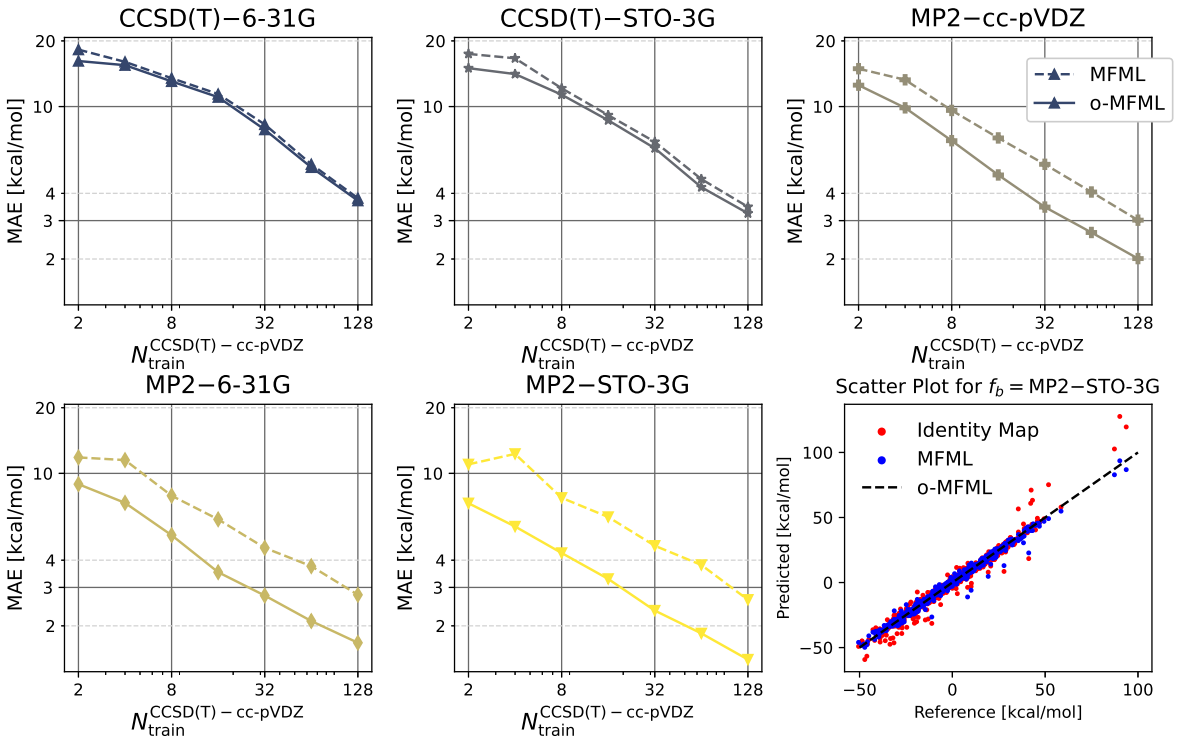


Figure 6.3: The learning curves of the MFML (dashed lines) and o-MFML (solid lines) models are contrasted for varying baseline fidelities f_b. Moreover, a scatter plot comparing the predictions and the CCSD(T)–cc-pVDZ reference for the two models is presented for f_b = MP2–STO-3G.

lowered offset with the addition of each cheaper fidelity, even for very small training set sizes for the o-MFML models. The same sub-models are used for both MFML and o-MFML models. The combination of these models is optimized resulting in an increased accuracy of prediction. Secondly, one also notices that the addition of the MP2 level of theory even with the largest basis set size results in a significant decrease in the error of the model. The addition of the MP2–STO-3G fidelity further improves the capability of predictions of the o-MFML models resulting in a lower error of prediction. For $N_{\text{train}}^{\text{CCSD(T)}-\text{cc-pVDZ}} = 128$ and the baseline of MP2–STO-3G, the MFML method results in an MAE of 2.73 kcal/mol while the MAE corresponding to the o-MFML method is 1.40 kcal/mol.

The improvements offered by the o-MFML method become more evident when one compares individually the learning curves of the MFML and o-MFML models for each baseline fidelity. This is done in Figure 6.3, where, for each baseline fidelity f_b the learning curves of the MFML and o-MFML are compared for decreasing fidelity. Already for the baselines from the CCSD(T) level of the theory the improvement of the o-MFML method is visible, but not significantly. The stark decrease of the MAE with the addition of MP2-cc-

pVDZ fidelity becomes evident in the right-most pane in the first row of this figure. Subsequent addition of cheaper fidelities further reduces this offset in comparison to the conventional MFML method. For the case of the MP2–STO-3G baseline, the o-MFML method for the predictions of atomization energies results in an MAE that is almost twice as low compared to the MFML method. It becomes evident that for each case of the baseline, the o-MFML models are superior predictors in comparison to the conventional MFML methods.

A closer look at the MFML and o-MFML models for $N_{\text{train}}^{\text{CCSD(T)-cc-pVDZ}} = 2^7 = 128$ clarifies this interpretation. The last pane in the second row of Figure 6.3 shows the scatter plot between the reference atomization energies of the molecules from the test set and the prediction of the two multifidelity methods on the same molecules. Identical training data was used to build the various sub-models for both the MFML and o-MFML methods. One immediately observes that the spread for the o-MFML model in the scatter plot is closer to the identity mapping than that for the MFML model. Of particular interest are the areas around -50 kcal/mol and beyond 50 kcal/mol. The MFML model consistently underestimates the atomization energies at the lower end while over-estimating those at the upper end of the energy range. The over-estimation in particular begins as early as about 40 kcal/mol and becomes evident as one goes in energy. The systematic issue in estimation of higher and lower energies could also be an artifact arising from a poor choice of hierarchy of methods, as concluded by Ref. [213]. The o-MFML on the other hand manages to predict these higher atomization energies with enhanced accuracy, thus bringing the distribution closer to the diagonal.

6.2.1.1 Coefficient Study

As discussed in Section 4.2, the o-MFML method optimally combines the various sub-models to result in a superior multifidelity method. The coefficients are optimized on the validation set with the OLS method. In order to further understand the o-MFML method, the analysis of these coefficients is performed as seen in Figure 6.4. The default coefficients used in the MFML methods are depicted in the last column of the second row. Notice that this corresponds to the discussion in Section 4.1 wherein the MFML model is built with the differences between the sub-models. For the different o-MFML models, one observes that the coefficients of each sub-model, $P_{\text{KRR}}^{\boldsymbol{s}}$, vary with varying baseline fidelities. This change signifies the optimization of the MFML model with respect to the validation set.

A meaningful analysis of the different cases is the comparison of the magnitude of the coefficients $\beta_{\boldsymbol{s}}^{\text{opt}}$ to $\beta_{\boldsymbol{s}}^{\text{MFML}}$. For the o-MFML models built for baseline fidelities from the

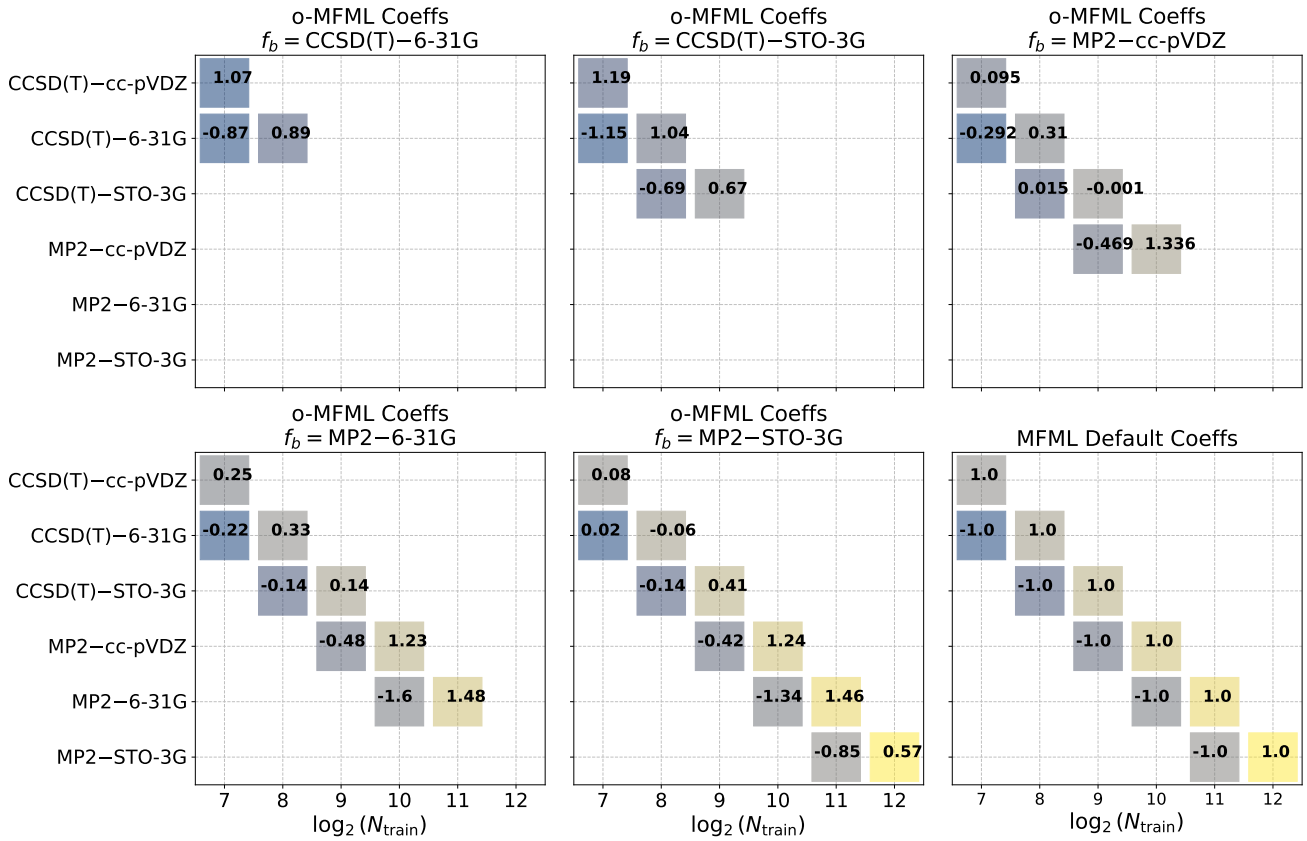


Figure 6.4: Values of the o-MFML coefficients for $N_{\text{train}}^{\text{CCSD(T)−cc-pVDZ}} = 2^7 = 128$. For varying baseline fidelities, the final values of the coefficients are shown. For reference, the default coefficients used in MFML are shown for the MP2–STO-3G baseline.

CCSD(T) level of theory, the coefficients are close in magnitude to those of the conventional MFML. This could imply that the MFML method was already nearly optimized for these fidelities. With the addition of the MP2 fidelities, however, the coefficient landscape changes. The optimization of the coefficients results in values that are significantly different from the conventional β_s^{MFML} values. This flexibility in combining sub-models rather than simply adding the differences (as done in MFML) allows o-MFML to be a superior method. The middle and right-hand side plots of the second row in Figure 6.4 assist in comparing the values of β_s^{MFML} and β_s^{opt} for the case of the MP2–STO-3G baseline. There is significant difference in the optimized coefficients and the default MFML coefficients for almost all sub-models. This finding shows that the conventional MFML method was not optimized in combining the different fidelities.

In particular, one observes that the values of β_s^{opt} for the CCSD(T)–6-31G fidelity are small in comparison to those of the other fidelities in the central plot of the second row. This

fact could indicate that the optimization method identified this fidelity to be less useful. In order to verify this, an experiment was carried out by separately building two models. The first was the usual complete model with all six fidelities with $N_{\text{train}}^{\text{CCSD(T)−cc-pVDZ}} = 2^7$ and the training samples at the other fidelities scaled by 2. The second model was built without the CCSD(T)–6-31G fidelity, but the training samples at the other fidelities were kept to be identical to that used in the first model, that is, $(2^7, 2^9, 2^{10}, 2^{11}, 2^{12})$. For these two models, the o-MFML was generated and the MAE evaluated. The original model resulted in an MAE of 1.421 kcal/mol while the second model resulted in an MAE of 1.431 kcal/mol which is a difference of only 0.72%. This result is a strong indicator for the robustness of the o-MFML method and how it can be a tool to detect whether a particular fidelity benefits the overall multifidelity structure or not. More details on the effectiveness of the coefficient analysis are reported in Appendix A in section A.3.3.

6.2.2 Excitation Energy Prediction

The dataset for excitation energies consists of excitation energies calculated along MD and DFTB-based trajectories for the molecules benzene, naphthalene, and anthracene as calculated in Chapter 5. A total of five fidelities were calculated and ordered as discussed in Section 6.1.1. In brief, the target fidelity is set to be TD-DFT–def2-TZVP and the cheapest fidelity is considered to be TD-DFT–STO-3G. All sub-models used in both the MFML and o-MFML method are built with KRR using the Matérn Kernel of first order and the l_2 norm. A regularization strength of 10^{-9} is employed. The kernel widths for each molecule were chosen as recorded in Table A.2. Unsorted coulomb matrices are used as representations for all cases. For the case of excitation energies, the multifidelity structure was built with varying basis sets. In increasing order of accuracy, these are STO-3G, 3-21G, 6-31G, SVP, and TZVP. Previously, various preliminary analyses of this dataset have been discussed, and two problematic data structures were thereby identified in Chapter 5. For MD-based naphthalene, there was no clear multifidelity structure. For DFTB-based anthracene, a high spread of the STO-3G energies with respect to the target fidelity of TZVP was also identified to be problematic. From these, it was shown that the MFML method would not provide favorable results for these two cases.

The learning curves of the conventional MFML method for the MD-based trajectories of benzene, naphthalene, and anthracene are shown in the top row of Figure 6.5. At the same time, the bottom row displays the learning curves resulting from the novel o-MFML method. Various baselines fidelities for the multifidelity models are as shown in the legend. Of particular interest in these findings is the case of naphthalene. The MFML results re-

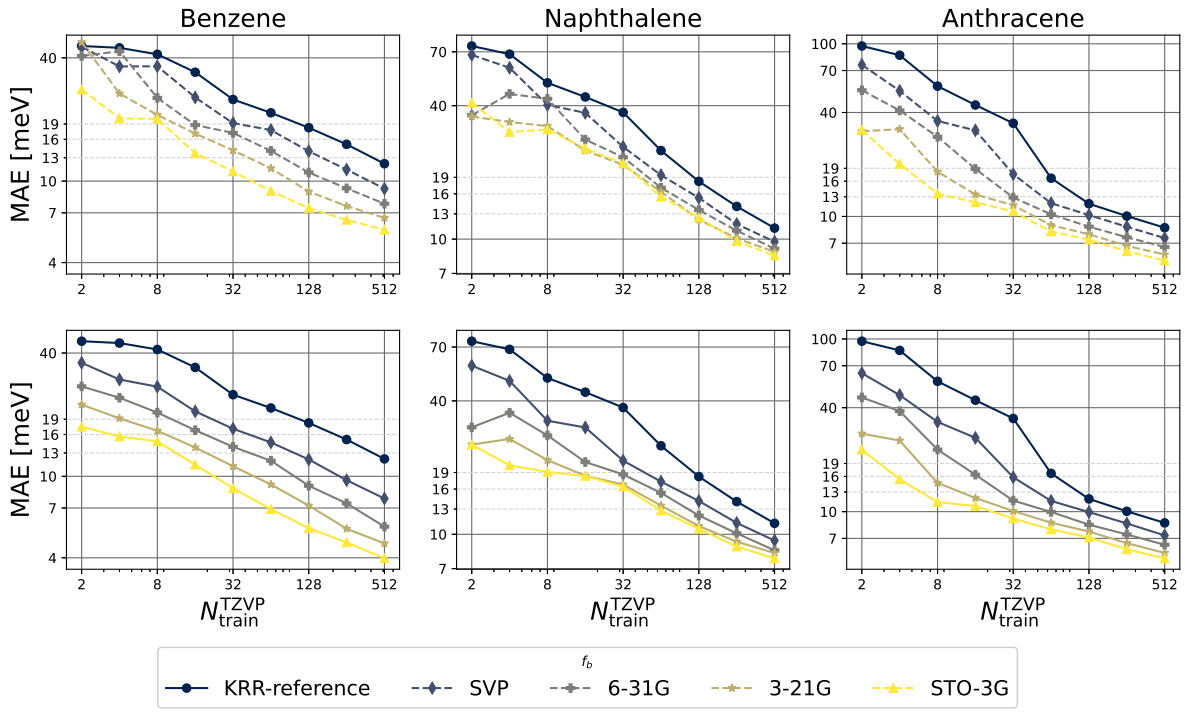


Figure 6.5: Learning curves for the MFML (top row) and o-MFML (bottom row) models for MD-based trajectories of various molecules for the prediction of excitation energies. The various baselines fidelities used are delineated in the legend. For each case, the KRR reference (black curve) is provided for a single-fidelity training on TZVP. The y axes are scaled identically for the MFML and o-MFML methods, but are different for each of the different molecules.

flect the issue of the wide spread of the scatter, as previously identified in section 5.2.1. However, with the o-MFML method, one observes that the model built with the 3-21G and 6-31G fidelities still results in constant lowered offsets as opposed to the conventional MFML method where these models do not provide much improvement. Thus, the o-MFML method provides a robust multifidelity method even if the data distribution of the quantum chemistry methods is not as anticipated for MFML. For benzene and anthracene, the improvement in the MAEs is perceptibly small. This fact could indicate that the original MFML model already was properly optimized for these cases.

Similarly, the learning curves for the excitation energies along DFTB-based trajectories for the three different molecules are given in Figure 6.6. As for the case of the MD-based trajectories, the use of the o-MFML method results in models which perform consistently better across the various training set sizes. That is, even for smaller training set sizes, the benefit of the multifidelity structure becomes evident. Across the molecules and for smaller training set sizes, the learning curves for the MFML methods have various crossing points

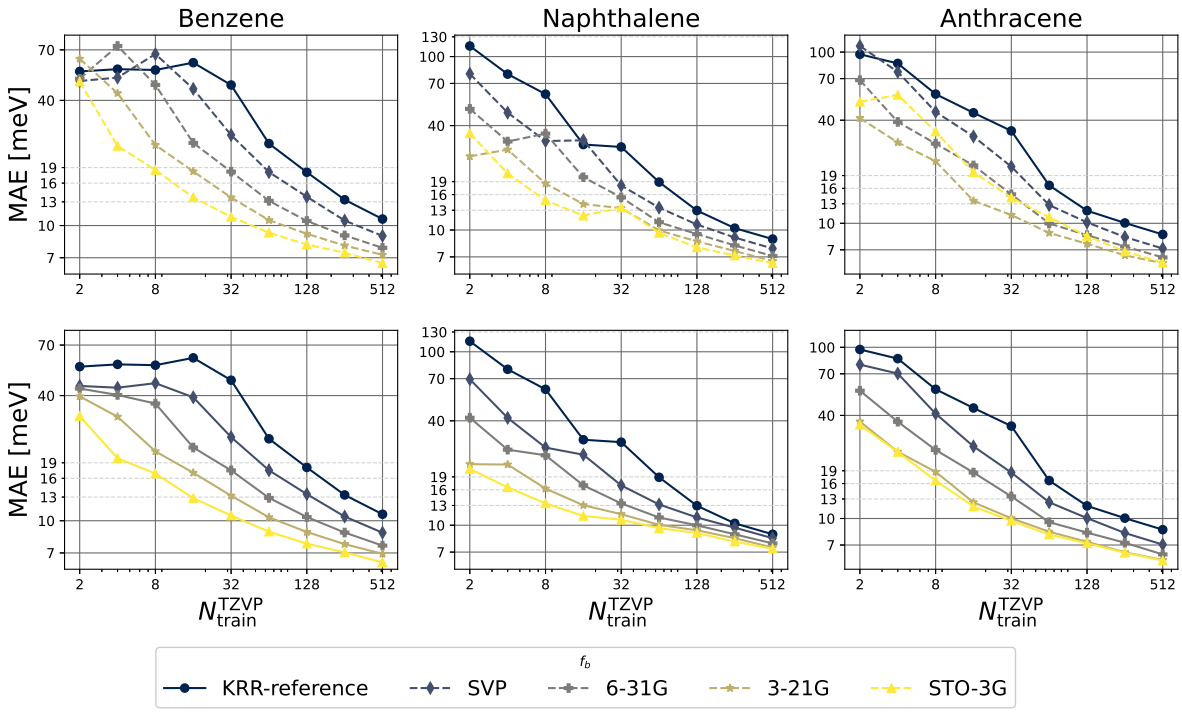


Figure 6.6: Learning curves for MFML (top row) and o-MFML (bottom row) for excitation energies along DFTB-based trajectories for three different molecules. The MAE is reported for the prediction of first excitation energies. The single-fidelity (TZVP) KRR learning curve (black line) for the prediction on the same test set as the other models is provided for reference. For the individual molecules, the scaling of the y axes is the same for the MFML and o-MFML models to ease comparison.

indicating a distinct region of pre-asymptotics for smaller training set sizes. That is, the benefit of using cheaper baselines does not become evident until a sufficiently large number of training samples is used. In contrast, the o-MFML method appears to have smoothed out any pre-asymptotics. Even for very small training set sizes, the addition of each numerically cheaper baseline shows an immediate decrease in the MAEs of the models. Next, consider the case of DFTB-based anthracene. For the MFML method, the addition of the STO-3G fidelity results in a decrease in the performance of the model, as discussed in the previous chapter, where it was argued that the wide-spread distribution of the STO-3G fidelity with respect to the target fidelity of TZVP results in a poorer improvement with the conventional MFML method. With the o-MFML method, the optimization of the coefficients results in a model that performs much better. The learning curve indicates that the o-MFML model for the STO-3G baseline is now comparable to that of the model built with the 3-21G baseline. The o-MFML method results in a better model in spite of the poor distribution of the STO-3G with respect to the target fidelity. Further results and analyses of the

o-MFML employed for the prediction of the excitation energies are discussed in Appendix A in section A.3.

Finally, a short note on the computational complexity for the OLS method employed in this work is to be made. The OLS shows cubic scaling with respect to the number of sub-models of multifidelity that are used. The number of sub-models only increases linearly with the number of fidelities being used. Nonetheless, the time cost of making such an optimization is minimal with respect to the time cost of a conventional QC calculation. As a particular example, it took 0.09 seconds to run the optimization method for MD-based benzene with 9 sub-models. To put this into perspective, conventional quantum chemistry calculations for the TD-DFT–def2-TZVP fidelity in the case of MD benzene take roughly 4 minutes *per* training sample calculation (see Table A.1). In the opinion of the authors, the time cost of the optimization procedure itself is negligible in this case.

6.3 Conclusion

This chapter has numerically established an improvement of the conventional MFML, termed o-MFML, by optimally combining the various multifidelity sub-models. For the prediction of atomization energies of molecules from the QM7b dataset, and the prediction of excitation energies for three molecules of growing sizes, o-MFML has been shown to unconditionally improve the prediction capabilities of the multifidelity method. The use of o-MFML was shown shown to be especially beneficial for cases where the hierarchy or distribution of the cheaper fidelities is not optimal. The learning curves indicate that the use of o-MFML yields low errors for the prediction of both atomization energies and excitation energies.

In the chapters that follow, the MFML and o-MFML method will be used in several applications such as prediction of excitation energies and ground state SCF energies. The efficiency of each of these methods will be discussed later in Chapter 9 in terms of model error versus time-cost incurred for the models.

CHAPTER 7

QeMFi: Multifidelity Dataset of Quantum Chemical Properties of Molecules

This chapter is taken from the work published as ref. [41] published in Scientific Data.

Recent developments in the field of machine learning (ML) for quantum chemistry (QC) have significantly changed the landscape of research and discovery in QC properties [22, 30, 155, 154] with significant reduction in the time to predict QC properties once an ML model has been trained. For such models, the protocol often involves testing them against some benchmark datasets such as the MD17 [185], QM7 [138], or the QM9 dataset [126, 127]. Recently, the WS22 database was released with a collection of Wigner Sampled geometries of 10 diverse molecules [215, 216]. With varied chemical complexity and number of atoms, the WS22 datasets provides a collection of QC properties for these molecules calculated at one level of theory, or *fidelity*. It was also shown that for this collection of molecules the use of ML methods is indeed challenging due to the wider chemical space that the geometries cover [215, 217].

Multifidelity methods harnessing inherent QC hierarchies to cancel out errors across different numerical QC methods have since superseded the single fidelity ML methods. These methods include Δ-ML [29] based models such as hierarchical machine learning [139], MFML [32], and o-MFML (Chapter 6). Certain other flavors of ML using multifidelity data have been proposed and tested, including multi-task GP treating the different fidelities

as interdependent tasks [110, 218]. Multifidelity methods have been applied to predicting diverse QC properties such as band gaps in solids, excitation energies, and atomization energies of various molecules [131, 163, 32, 142, 143].

Several QC datasets have been generated for the general work of ML for QC, some consisting of multiple fidelities of data. Some of these include ground state energies and electronic spectra data computed at DFT level of theory and some semi-empirical levels of theory hosted in the bigQM7ω [219] dataset. The PubCheMQC project presented a large database of electronic structure properties calculated with the DFT formalism using two distinct basis sets for use in training ML models [220]. The QM8 [130] dataset records electronic spectra properties with DFT formalism for over 20k geometries of small organic molecules sampled from the larger QM9 dataset [126, 127]. Ref. [163] introduced a multifidelity dataset of 358 polymer bandgaps for benchmarking use with multifidelity co-kriging methods. The ANI-1x and ANI-1ccx datasets provide a rich multifidelity dataset of around 5 million data points with HF, MP2, and NNPO-CCSD(T) energies and forces primarily created for the training of the ANI-1x potential [219, 24]. VIB5 is yet another multifidelity dataset with ab initio quantum chemical properties including PES for five molecules with MP2, HF, and CCSD(T) levels of theory for different basis set sizes [221]. The QM7b dataset consists of multifidelity atomization energies computed at the MP2, HF, and CCSD(T) fidelities for three distinct basis set sizes with 7,211 small-to-medium sized molecules [138, 49]. Ref. [222] offers orbital energies of 134k molecules for PBE and GW fidelities. The MultiXC-QM9 is an elaborate dataset of diverse QC properties such as reaction energies which are computed with the DFT formalism using 76 different functionals for three basis sets [223]. All of these above mentioned datasets have important use-cases for multifidelity machine learning methods and related benchmarks. However, none of these datasets offer the QC compute time for the different fidelities present. That is, although multifidelity models can be created and benchmarked in terms of model error, it is not possible to use these datasets to perform time-cost benchmarks for multifidelity models. Since the entire conceptualization of multifidelity methods such as Δ-ML [29] or MFML [32] is to reduce the cost of generating training data, this key factor is necessary to meaningfully benchmark these methods. A mere model accuracy benchmark is insufficient.

To unify the research in this rapidly developing field of multifidelity methods, it becomes necessary to present to the community a diverse collection of multifidelity data over a range of molecular complexity which also includes the time-cost of the QC calculations. Building up on existing datasets is preferred in such a scenario to prevent redundant calculations and geometry generation. After all, the entire point of a multifidelity method is

to reduce compute cost and resource usage in discovery and research. In interest of such an approach, the WS22 database [216] was chosen to be the collection of geometries. In addition to being a collection of molecules that are chemically complex with distinct conformers, the molecule in this dataset also cover a wide range of the quantum chemical configuration space in contrast to other datasets such as MD17. The presence of flexible functional groups make the geometries, and by extension, the QC properties, of this dataset challenging for ML models to learn [215]. These features make this collection the preferred choice to generate multifidelity data. For each of the molecules of increasing size and chemical complexity, this dataset offers 120,000 geometries. This creates a vast dataset collection of diverse geometries covering various conformers of the different molecules. In total there are around 1 million geometries in the WS22 database. Performing multifidelity QC calculations for such a vast number of geometries is not feasible. It is more realistic and computationally feasible to produce a multifidelity dataset for a portion of the geometries of the WS22 database. Therefore, for each of the molecules in the WS22 database, 15,000 geometries were evenly sampled, for a total of $9 \times 15,000 = 135,000$ geometries, and the multifidelity QC calculations performed for these.

This dataset is provided to the ML-QC community under the name QeMFi (**Q**uantum Ch**e**mistry **M**ulti**Fi**delity) dataset [224]. A detailed description of the geometry sampling, data generation procedure, the fidelities, and the technical details of the QeMFi dataset are provided in the following section. In addition, scripts to generate two multifidelity models from Chapter 6, namely, MFML and optimized MFML (o-MFML) are provided. Scripts to assess time benefit of multifidelity methods are also included in the code repository. This makes it easy for future research in the multifidelity methods to establish a clear time benefit for these models over standard single fidelity ML methods. The diverse collection of molecules in QeMFi along with their multifidelity properties, provides a challenging dataset for the domain of ML in QC. Due to the large number of multifidelity data points along with their QC time-costs and easily usable associated scripts, we believe that QeMFi is a significant collection that will help push the boundaries of multifidelity methods for ML in QC properties enabling meaningful time-cost assessments for these methods.

The original WS22 database includes the following molecules (in increasing order of number of atoms):

1. urea
2. acrolein
3. alanine

4. 2-(methyliminomethyl)phenol (SMA)
5. 2-nitrophenol
6. urocanic acid
7. 4-(dimethylamino)benzonitrile (DMABN)
8. thymine
9. 4-(2-hydroxybenzylidene)-1,2-dimethyl-1H-imidazol-5(4H)-one (o-HBDI)

In addition to these molecules, toluene is also included to compare with the MD17 [185] database. Since toluene consists of a single conformer and was only introduced in WS22 for comparison to existing datasets such as MD17, this molecule was not included while generating the QeMFi dataset. The original WS22 database was first generated as reported extensively in ref. [215]. The pipeline involves optimized equilibrium geometries identification for the different conformations of the molecule with DFT [211, 119]. Following this, the respective Wigner Sampling is carried out from ground state (S0) and/or excited state (S1) minima. For these, the geometries are subsequently interpolated by finding on a Riemann manifold, an optimized geodesic curve. The metric for this is defined by a redundant internal coordinate functions [225]. In the original WS22 database provided in ref. [216], this results in a little over 1 million samples across 9 molecules with various properties calculated at the TD-DFT level of theory using the PBE0/6-311 G* functional and basis set combination [215].

7.1 Data Sampling and Quantum Chemistry Calculations

To build the QeMFi dataset from the WS22 database, 15,000 geometries were sampled from the original 120,000 geometries for each of these molecules. For each of the nine molecules from the WS22 database, 15,000 geometries were evenly sampled from the original 120,000 geometries. To achieve this, every 8^{th} geometry for each molecule was selected from the WS22 database resulting in a total of $9 \times 15,000 = 135,000$ point geometries for QeMFi. An even sampling of the original dataset ensures that there are sufficient geometries from all conformations of the molecule. Once these geometries were sampled, they were used to perform point calculations for the QC properties.

All QC calculations were performed with the ORCA(5.0.1) QC package [7]. From these calculations, a diverse set of QC properties were extracted including information of the

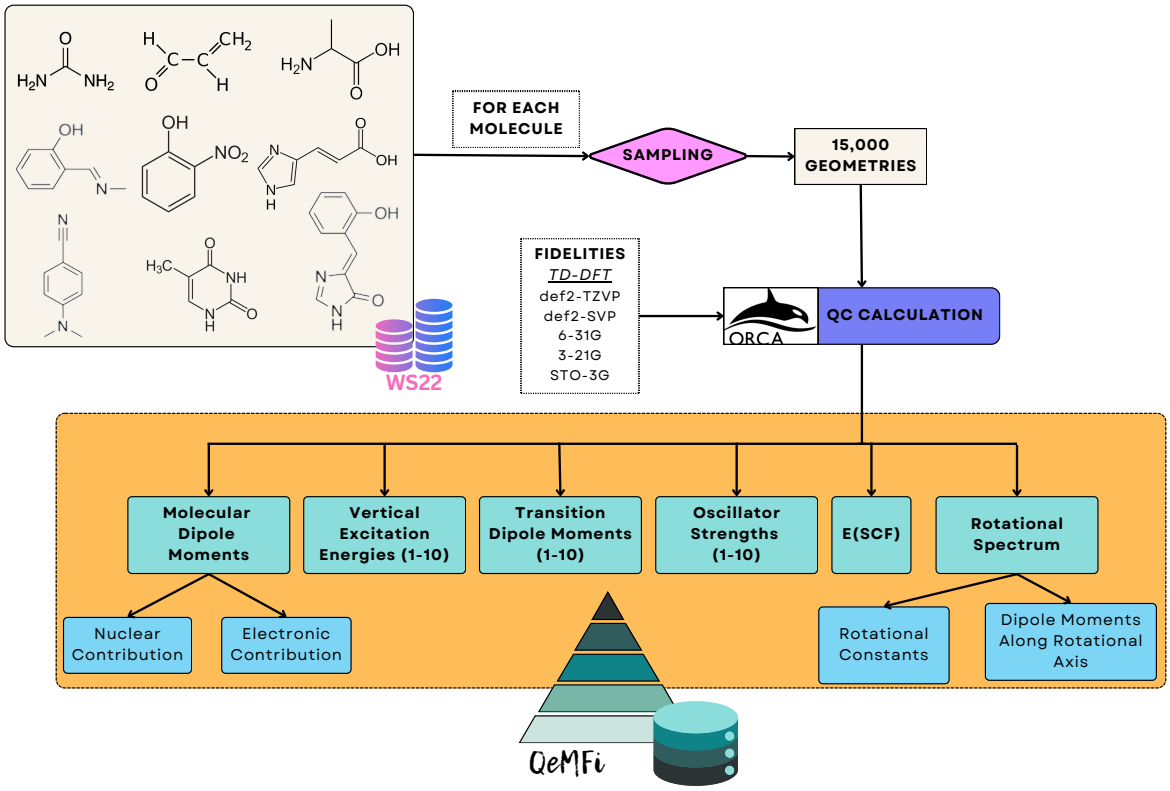


Figure 7.1: The workflow of generating the QeMFi dataset by sampling from the WS22 database. 15,000 geometries are used for each molecule resulting in a total of 135,000 single point geometries. For each of these, multiple QC properties are calculated at DFT level of theory with varying basis set sizes to create the diverse multifidelity dataset.

vertical excitation states such as energies and oscillator strengths. QC calculations were performed at the TD-DFT level of theory with the CAM-B3LYP functional. For each geometry, five fidelities were calculated. These fidelities are the basis set choice of increasing size. In increasing hierarchy of the fidelity, these are: STO-3G, 3-21G, 6-31G, def2-SVP, and def2-TZVP. In the rest of the document, for the most part, these are referred to by their short-hand, i.e., STO3G up to TZVP. The `TightSCF` keyword was employed to ensure energy convergence of the order of 10^{-9} a.u. for each calculation. Resolution of Identity approximation (RIJCOSX) was employed in order to speed up the excitation energy calculations. For any calculation, the maximum memory usage was limited to 2.0 GB. In practice, the ORCA calculations did not use this amount of memory. A total of 10 vertical excitation energies were calculated with each fidelity for each geometry. The complete workflow of the dataset generation process is pictorially depicted in Figure 7.1.

A note on the calculations being restricted to DFT methods is to be made here. Since

QeMFi is a benchmark dataset and not a high accuracy model training dataset, the cost of generating a costlier dataset, say at coupled cluster level of theory, was considered to be excessive. The aim of this dataset is to present a diverse collection of QC properties based on a set of complex molecules which can be used to uniformly assess MFML methods. Therefore, the QC properties are calculated only with DFT methods and higher accuracy methods such as the gold standard CCSD(T) are not considered.

PROPERTY	**DIMENSIONS/UNITS**	**npz ID**
Atomic Numbers†	(n_atoms,)	'Z'
Cartesian Coordinates†	(n_atoms,)/Å	'R'
Ground State Energies (SCF)	(15000, 5)/hE	'SCF'
Vertical Excitation Energies	(15000,5,10)/cm^{-1}	'EV'
Transition Dipole Moments	(15000,5,10,3)/a.u.	'TrD'
Oscillator Strength	(15000,5,10)	'fosc'
Molecular Dipole Moment (electronic)	(15000,5,3)/a.u.	'DPe'
Molecular Dipole Moment (nuclear)	(15000,5,3)/a.u.	'DPn'
Rotational Constants	(15000,5,3)/cm-1	'RCo'
Dipole Moment Along Rotational Axis	(15000,5,3)/a.u.	'DPRo'
QC Calculation Times	(5,)/seconds	't'

Table 7.1: List of properties available in the QeMFi dataset. The corresponding dimension(s) and units of the properties are also given with the npz file key. †From the WS22 database [215, 216]

The list of available multifidelity properties is given in Table 7.1. The Cartesian coordinates and atomic numbers are taken from the WS22 database. The SCF ground state energies are reported in Hartree units. The first 10 vertical excitation energies are provided in cm^{-1} with their corresponding oscillator strengths and transition dipole moments (in a.u.). The molecular dipole moments are also a property included in the QeMFi dataset with both the nuclear and electronic contributions being separately cataloged in atomic units (a.u.). The rotational spectrum data is also included in the form of Rotational Constants (in cm^{-1}) and the total molecular dipole moments (in a.u.) aligned along rotational axes.

As an important contribution to the multifidelity research, the average time to run the QC computations are also provided for each molecule for each fidelity. This information can be further used to benchmark multifidelity models across QC properties as was shown in Chapter 5 for excitation energies of arenes. The notion here is to assess the error of a model with respect to the time it takes to generate a training set for that specific model. Such an analysis of the model (see Chapter 5) shows the true time benefit of multifidelity models, in this specific case, for MFML. In the QeMFi dataset, the time to run an ORCA cal-

culation for a specific fidelity is provided for each molecule in units of seconds. The parallelization of numerical codes is a challenge unto itself and usually comes with artifacts that arise due to the form of parallelization. This results in non-uniform calculations schemes across basis set sizes and molecules. In order to enforce consistency in the calculation times, ORCA calculations were run on a single compute core for 10-evenly sampled geometries of each molecule. The calculation times as returned by the ORCA software are then averaged over these 10 geometries and reported for each fidelity for each molecule. Thus, the time for a single-core calculation of each fidelity for each molecule is provided in units of seconds to benchmark the time-benefit of multifidelity models against single fidelity models. This diverse collection of QC properties is made available for $9 \times 15,000 = 135,000$ geometries across five different fidelities providing ample room for development and benchmarking of MFML methods and models.

7.2 Data Records

The various QC properties of the QeMFi dataset are stored in separate NumPy (v 1.26.4) `npz` files for each molecule. These `npz` files have a dictionary-like format allowing for each property to be accessed via its corresponding key denoted in Table 7.1. Each property itself is stored as a NumPy `ndarray` with the first dimension being 15,000 corresponding to the number of geometries. Thus, the QC properties can be accessed by querying the right ID. For example, the SCF ground state energies can be accessed with the key `'SCF'` returning a NumPy `ndarray` of size $15,000 \times 5$ where the second dimension of the array corresponds to the five fidelities used. Similarly, one can access the QC computation times using the key `'t'` which results in a NumPy array of shape (5,) corresponding to the five fidelities used. The compute times are stored in units of seconds. An example script to accessing the QC properties is shown in Listing 7.1.

The dataset itself is hosted on Zenodo at https://doi.org/10.5281/zenodo.13925688 with a detailed README file documenting the key aspects of the data. The README also provides information on how to access the different properties using Python. For the QeMFi dataset, the various scripts involved in generating the data, including ORCA input files and shell scripts to extract properties from the ORCA log files, are stored in the code repository that can be accessed at https://github.com/SM4DA/QeMFi. In addition to these scripts, the code repository also contains Python scripts to perform multifidelity benchmarks on this dataset. These can be launched using the CLI and are a handy tool in setting benchmarks for this dataset using current state of art multifidelity methods.

```
import numpy as np

#load the dataset for alanine
data = np.load('QeMFi_alanine.npz')
#query for the vertical excitation energies
EV = data['EV']
#Select the second vertical state for SVP (4th fidelity)
EV_SVP = EV[:,3,1]

#load QC compute times
QC_time = data['t']
```

Listing 7.1: Python example to extract the SVP fidelity values of second vertical excitation state of alanine from QeMFi.

7.3 Technical Validation

In order to verify that this form of sampling did in fact evenly cover the conformation space of each molecule, Uniform Manifold Approximation and Projections (UMAPs) [226] are studied herein. To further validate the QeMFi dataset and its use in benchmarking multifidelity methods, the MFML and o-MFML models presented in Chapter 6 were tested in predicting ground state energies and the first vertical excitation energies. The multifidelity models are built for different baseline fidelities, which refers to the cheapest fidelity included in the model. For example, a baseline fidelity f_b =631G implies that the multifidelity model is built up of the fidelities 631G, SVP, and TZVP (see section 4.1). In addition to benchmarking the multifidelity models on properties of individual molecules, the models are also tested on using data from all the molecules of the dataset. For this purpose, the ground state energy of all molecules are used to train one single MFML and o-MFML model. This is then tested on predicting the ground state energies of all molecules.

While these broad tests serve as a benchmark for multifidelity models on this dataset, the benchmarks of the other properties and molecules are not reported here. However, it is to be pointed out that the scripts provided can be readily used to generate benchmarks for these cases using standard ML methods such as learning curves. All learning curves are reported for a 10-run average, that is, for 10 random shuffling of the training set as directed in section 2.7.

7.3.1 Conformation space coverage

In order to ensure that the 15,000 geometries that are sampled from the WS22 database do cover all of the conformation space as spanned by the complete 120,000 geometries,

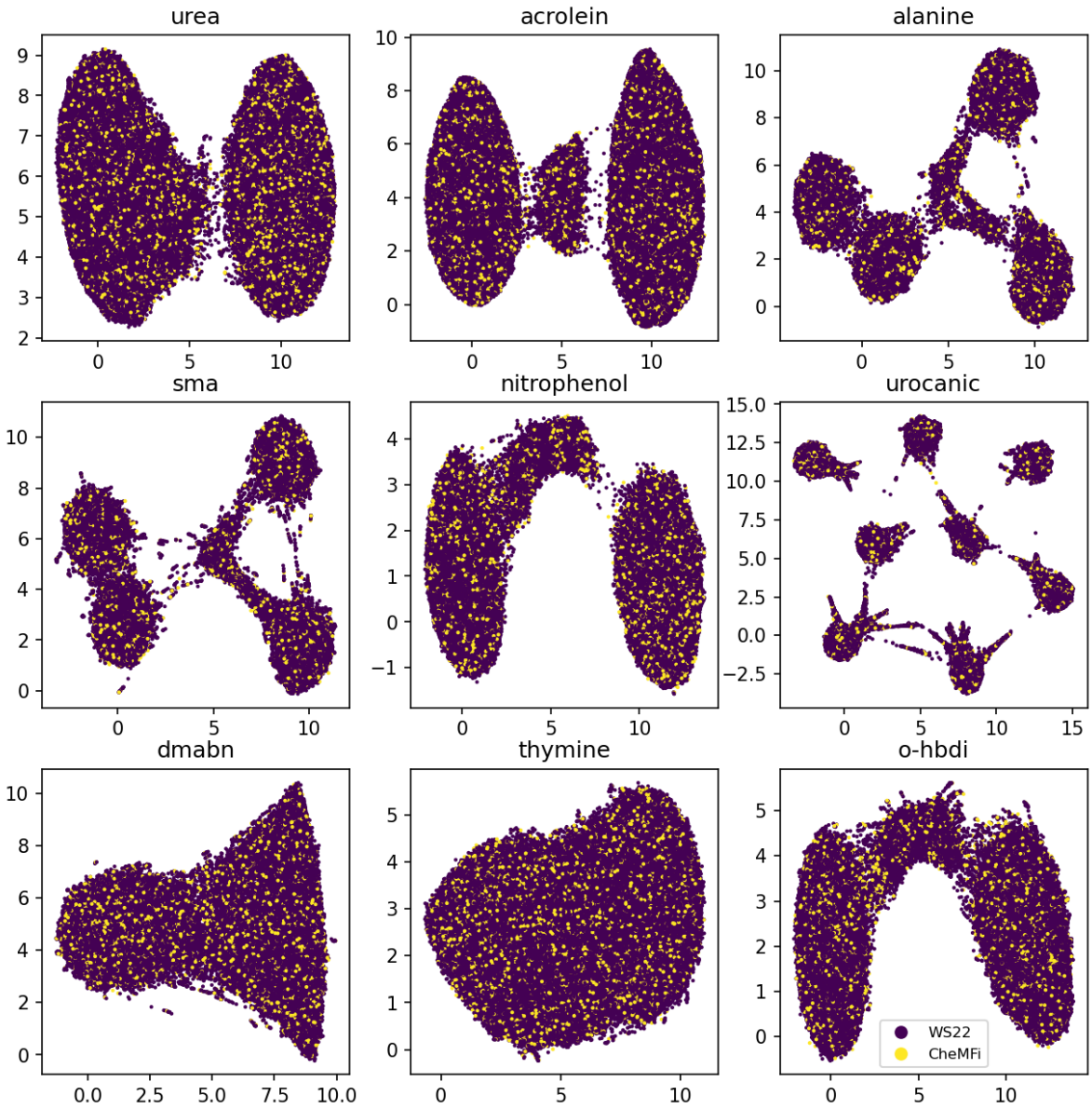


Figure 7.2: Scatter plots of UMAPs for the various molecules which compose the WS22 database. The UMAPs were generated for the unsorted Coulomb Matrix (CM) molecular descriptor for each molecule. The legend key indicates the geometries which are part of the WS22 and the QeMFi dataset respectively. For all molecules, it can be observed that the QeMFi dataset traverses the entirety of the configuration space that WS22 also covers.

a short study is performed. UMAPs are a powerful tool to perform dimensionality reduction of data. In addition to this, they are useful tools to visualize the multidimensional feature space in ML [226]. To assess the chemical space coverage achieved with this form of sampling, UMAPs of the molecules can be studied. UMAPs are a dimensionality reduction

method that is useful to visualize multidimensional data such as the molecular representations used in ML for QC. The resulting 2D embedding can then be used to visualize the complexity of the conformation space based on the coverage that is observed. To this end, molecular descriptors were first generated for the various molecules. The choice of descriptors for this case was the unsorted Coulomb Matrix (CM) which is calculated as:

$$C_{i,j} := \begin{cases} \frac{Z_i^{2.4}}{2}, & i = j \\ \frac{Z_i \cdot Z_j}{\|\boldsymbol{R}_i - \boldsymbol{R}_j\|}, & i \neq j \end{cases}, \tag{7.1}$$

where, Z_i is the atomic charge of the i^{th} atom of the molecule and $\boldsymbol{R}_i$ is its Cartesian coordinate. For this proxy of the conformation space that geometries cover, a 2D UMAP was generated and the resulting plots are shown in Figure 7.2 for all 9 molecules. It is observed that the UMAPs for QeMFi uniformly cover the conformation space spanned by UMAPs of WS22. From this plot it becomes clear that the geometries sampled for the QeMFi dataset from the WS22 database uniformly cover the entire chemical space of WS22. This is true even in cases of multiple localized clusters as seen in the case of alanine or urocanic acid. Therefore, it becomes evident that even though only 15,000 geometries are sampled from the WS22 database, these do uniformly cover the conformation space of WS22 and should therefore offer the same level of chemical complexity for ML models.

7.3.2 Single molecule benchmarks

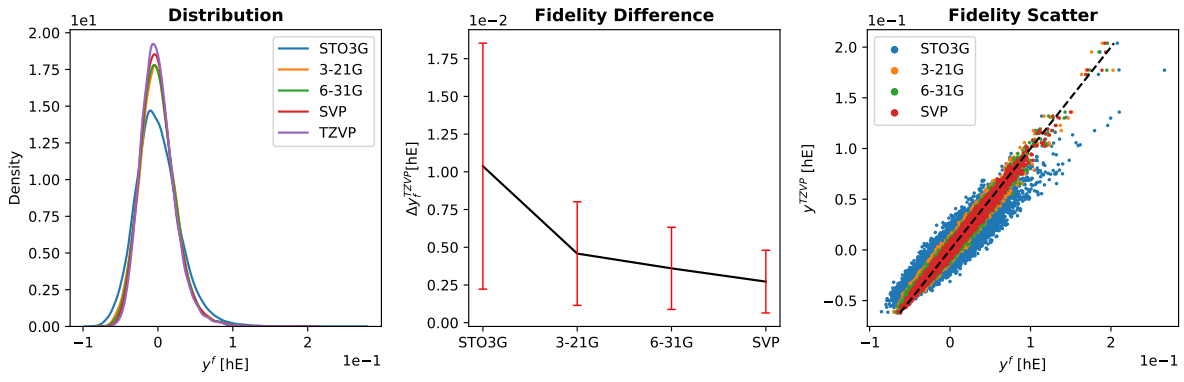


Figure 7.3: Preliminary analysis of multifidelity structure of SCF ground state energies for the SMA molecule. The three different preliminary tests for the hierarchy are performed as prescribed in Chapter 5. The ground state energies show a normal distribution centered around 0 hE. The scatter plot of the energies of different fidelities with respect to the TZVP fidelity show a compact distribution for the most part. With STO3G there is a wider deviation from the identity map (dashed black line).

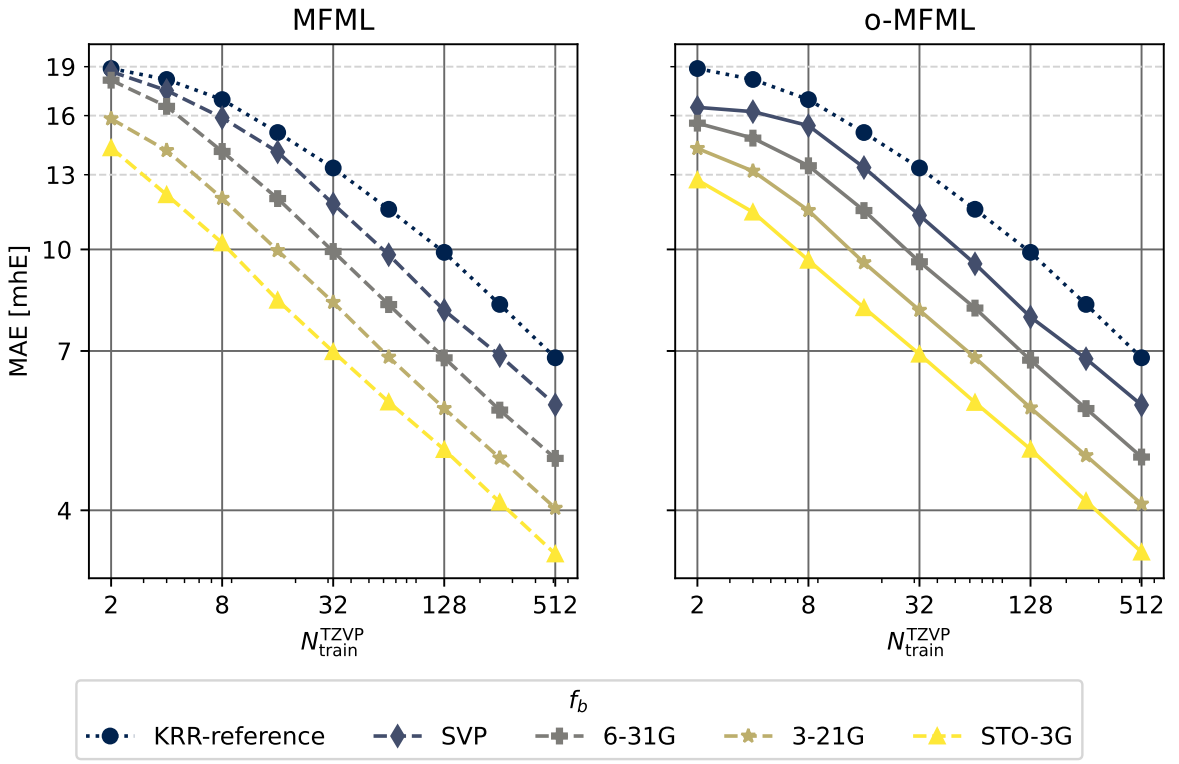


Figure 7.4: Learning curves for MFML and o-MFML for the SCF ground state energies of SMA as recorded in the QeMFi database. The reference single-fidelity KRR is also shown by training on TZVP only. The Laplacian kernel was used with a kernel width of 200.0 and regularization of 10^{-10}. The Global SLATM [51] molecular descriptors were used.

The technical validation carried out individually for SMA and O-HBDI molecules is in line with the experimental set-up of Chapter 5. A total of 12,288 training samples were chosen to build the multifidelity models. 712 samples were set aside as a validation set for the o-MFML model (section 4.2), and the remaining 2,000 samples were used as a test set. The accuracy of the models are gauged with mean absolute error (MAE) in the form of learning curves. Learning curves display the MAE with respect to increasing number of training samples, here, at the highest fidelity, that is, TZVP. In addition, a special kind of learning curve as seen in Chapter 5 are also shown. These are MAEs versus the total time to generate the training set for the MFML model. These special learning curves provide a better picture of the time-benefit of using MFML over conventional single fidelity methods. The o-MFML method additionally requires a validation set computed at the target fidelity, F. In these benchmarks presented herein, this is not accounted for since this work is meant as a data descriptor and not a comprehensive comparison of the MFML and o-MFML method. For the benchmarks that are reported here, the MFML and o-MFML methods perform similarly

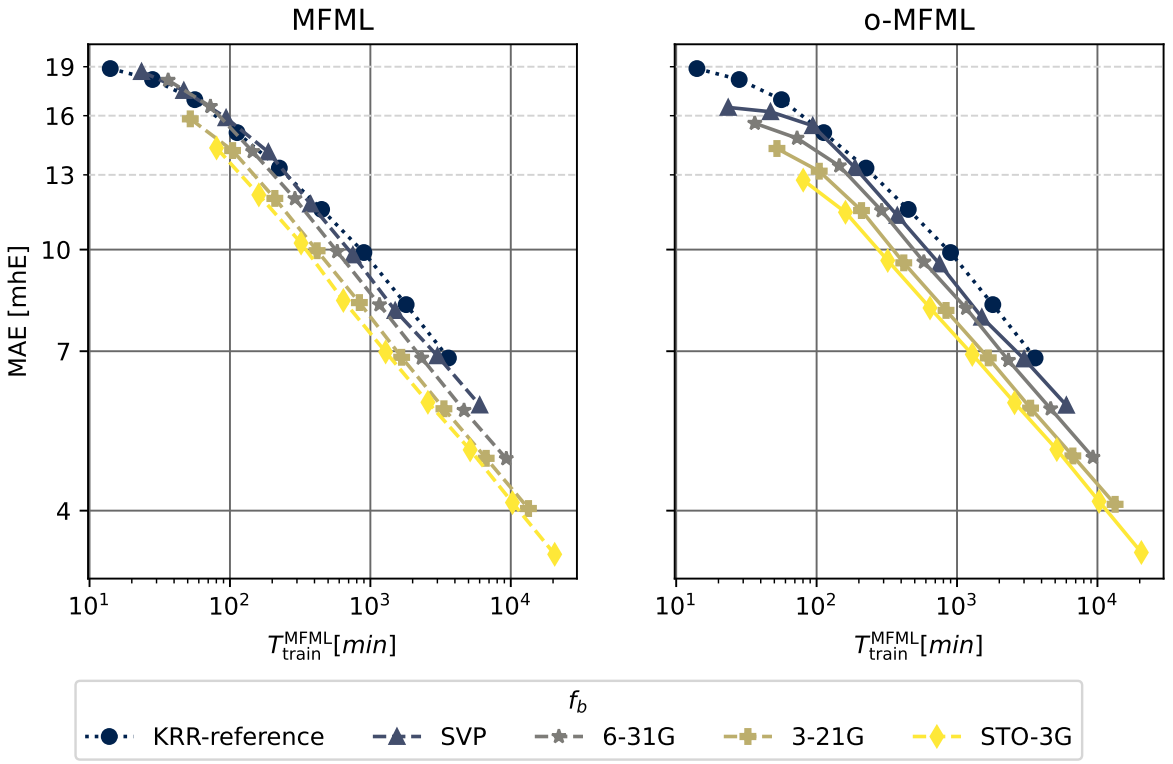


Figure 7.5: Time versus MAE plots for MFML and o-MFML models predicting the SCF ground state energies of the SMA molecule. The time to generate the training set for MFML is a comprehensive measure of the cost of a multifidelity model as prescribed in Chapter 5.

in terms of MAE. In such scenarios, it is to be concluded that MFML is better suited than o-MFML due to the lack of the cost associated with a validation set.

Figures 7.3-7.5 report the multifidelity benchmarking results for SCF ground state energies for SMA. The SLATM molecular descriptor [51] was used with a Laplacian kernel to perform kernel ridge regression. First, the preliminary analyses as recommended in Chapter 5 are shown in Figure 7.3. The first of these is to study the distribution of the multifidelity data. The second analysis is the study of mean absolute differences between each fidelity and the target fidelity (that is, the most accurate fidelity, here, TZVP). Generally, it is anticipated that these differences decay monotonically for increasing fidelity. Thirdly and finally, a scatter plot of the energies of the different fidelities with respect to the energies of the target fidelity is generated to study how these deviate with respect to the target fidelity. The three different preliminary analyses show a systematic ordering of the fidelities which confirm the assumed hierarchy as seen in the fidelity difference plots. The fidelity scatter plot also shows a systematic distribution of the energies when compared to the target fidelity of TZVP. In Chapter 5 this was identified as a good preliminary indicator of favorable results

in the MFML models. In Figure 7.4, the learning curves of MFML and o-MFML (with OLS optimization) are depicted for the prediction of the ground state energies. The multifidelity learning curves can be understood as follows: the addition of a cheaper fidelity systematically decreases the error (here reported as mean absolute error, MAE) which is reported in units of MhE. With each cheaper fidelity being added, one notices that the corresponding learning curve from Figure 7.4 has a lower offset. The continuing negative slope indicates that further addition of training samples could decrease the MAE of these models. There is no significant difference in MAE between the MFML and o-MFML models and they both perform similarly for the prediction of ground state energies. As a final assessment for SMA, an MAE versus time to generate the MFML training data is also shown in Figure 7.5 for the MFML and o-MFML models. The total time cost for a given MFML (or o-MFML) model is the total time taken to generate all the training samples at the different fidelities. In other words, if one picks the model $P_{\text{MFML}}^{(\text{TZVP;631G})}$ with $N_{\text{train}}^{\text{TZVP}} = 2$, then the total time to generate the training set for this model would be given as $T_{\text{MFML}} = 2 \times t_{\text{TZVP}} + 4 \times t_{\text{SVP}} + 8 \times t_{\text{631G}}$, where t_f is the computation time corresponding to the fidelity f. Since the QeMFi dataset comes with the average compute times for each fidelity of each molecule, this allows for a meaningful benchmarking of multifidelity models with this form of analysis. In Figure 7.5 one observes that with addition of cheaper baseline fidelities, the curves show an increased offset along the time axis. Consider the very last data point of the curve corresponding to the reference KRR for o-MFML (right-hand side plot), which corresponds to $\sim 4 \times 10^3$ minutes. If one were to draw an imaginary line parallel to the horizontal axis, it would intersect the curve corresponding to the STO3G baseline MFML model around $\sim 10^3$ minutes. This implies that one could use the o-MFML model with STO3G baseline to achieve similar accuracy as the reference KRR model with a reduction in time cost by a factor of $4 \times 10^3 / 1 \times 10^3 = 4$. This indeed shows the effectiveness of such forms of multifidelity models over conventional single fidelity methods.

A similar benchmarking procedure was carried out for the prediction of first vertical excitation energies of o-HBDI. In this case, the unsorted CM were used with the Matérn Kernel. As for the case of SMA, a preliminary analysis study was performed with the resulting plots shown in Figure 7.6. The difference in fidelities plot in the center indicates that the assumed hierarchy holds true for the fidelities. However, the fidelity scatter plot on the right hand side shows two distinct clusters. These correspond to the two main conformers of o-HBDI, namely the cis and trans conformers. The scatter plot also shows some cases where the STO3G fidelity covers a wider range of values and is less localized than the other

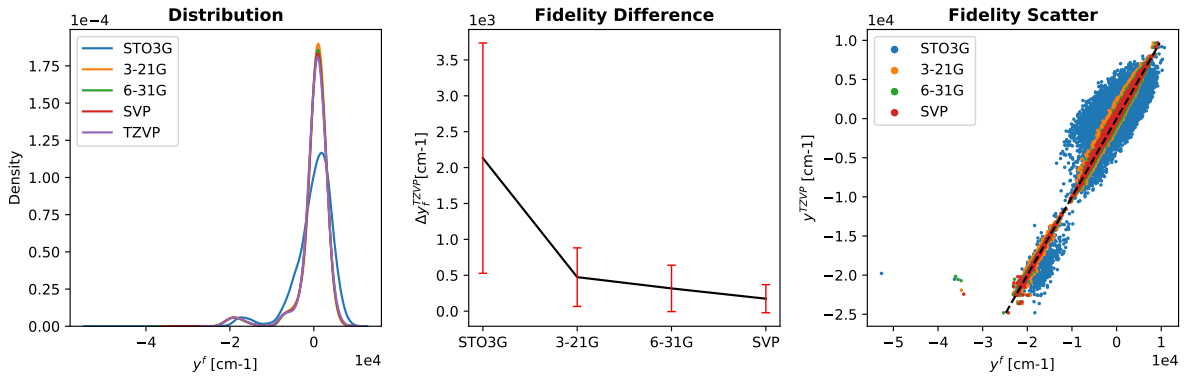


Figure 7.6: Preliminary analysis of multifidelity structure of the first vertical excitation energies for the o-HBDI molecule. The different fidelities show distinct peaks around 0 cm^{-1} with a small bump around 18000 cm^{-1}. The difference in the fidelities shows a distinct hierarchy with reducing difference with increasing fidelity.

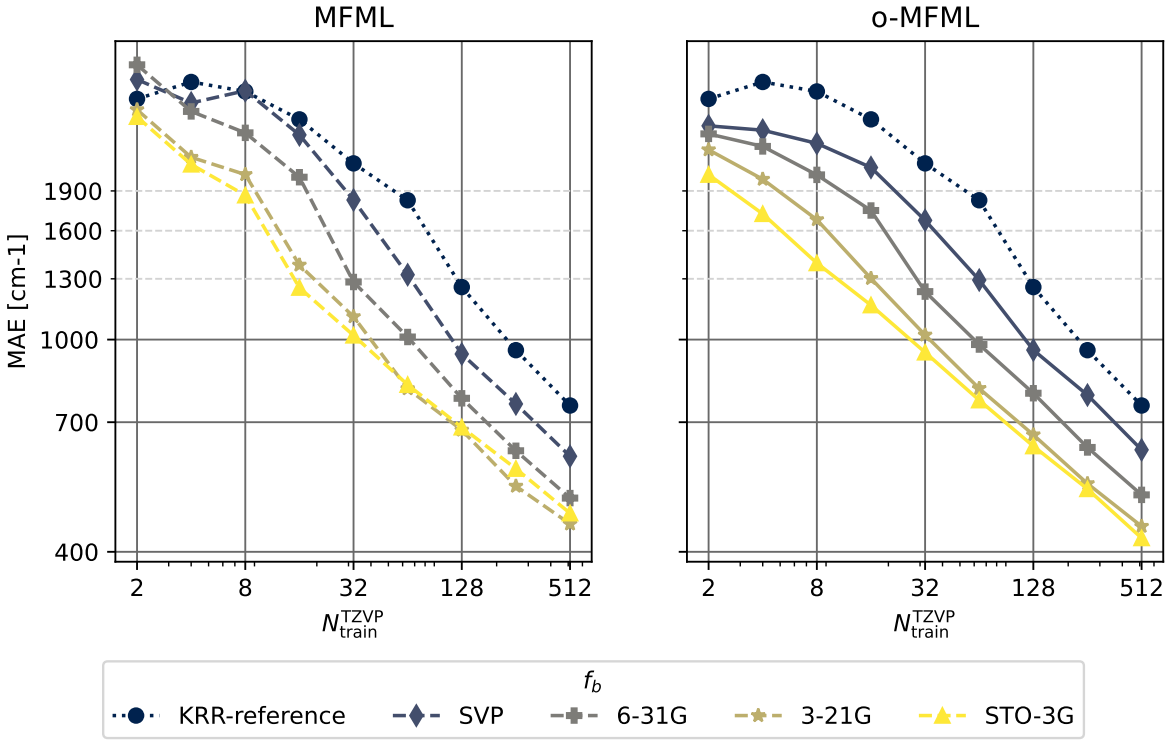


Figure 7.7: Learning curves for MFML and o-MFML for the first vertical excitation energy of o-HBDI from the QeMFi database. The reference single-fidelity KRR is also shown by training on TZVP only. The Matérn kernel of first order with L_2-norm was used with a kernel width of 150.0 and regularization of 10^{-10}. Unsorted CM descriptors were used for these cases.

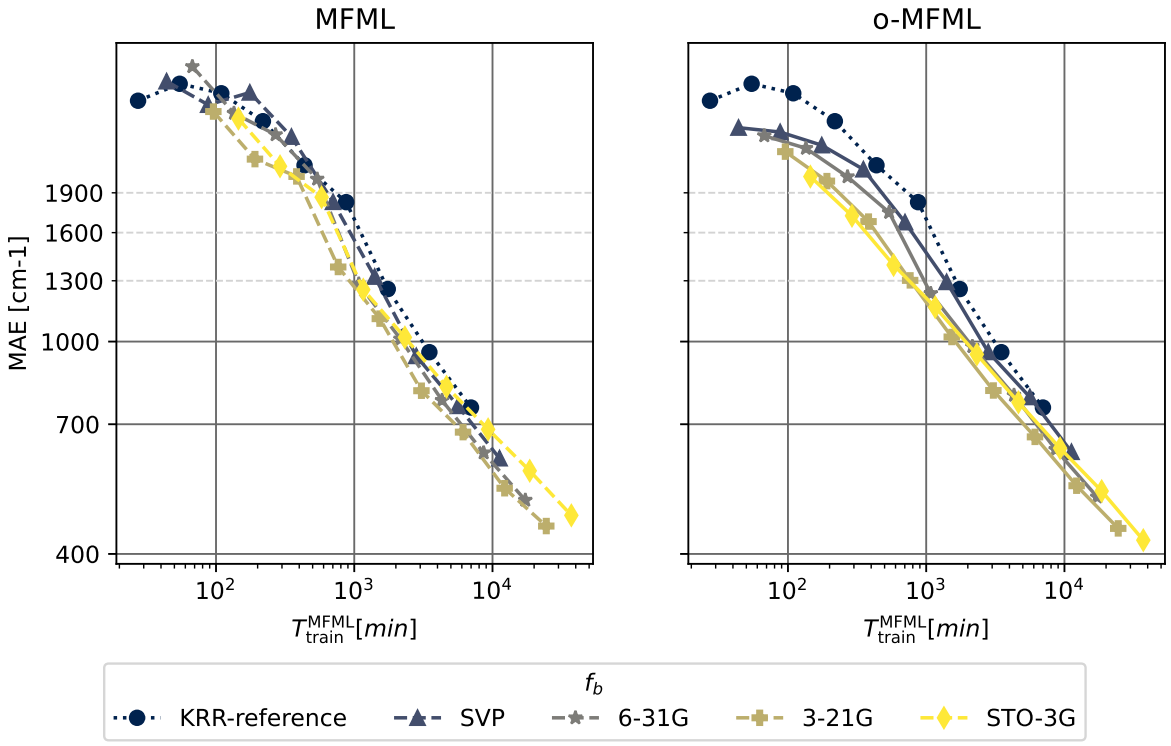


Figure 7.8: Time versus MAE plots for the prediction of first vertical excitation energy for the o-HBDI molecule. The time to generate the MFML training set is calculated as described in Chapter 5.

fidelities. This could indicate that the use of STO3G in the MFML models would result in a lower improvement of the accuracy of the model.

The learning curves for the prediction of the first vertical excitation energies of o-HBDI from QeMFi are shown in Figure 7.7. The MAE are reported in cm^{-1} with the axes identically scaled for both MFML and o-MFML. With the addition of cheaper fidelities, the learning curves show a constant reduction in the offset of the MAE as seen in the near parallel learning curves of the different baselines fidelities. As anticipated from the preliminary analysis, the addition of the STO3G fidelity does not provide significant improvement especially for larger training samples. However, this is rectified, as expected, by the o-MFML method which was indeed shown to fix this very issue in Chapter 6. Indeed, in the MAE versus time to generate training data plots seen in Figure 7.8 for o-HBDI, one observes that the STO3G baseline model fails to provide a reasonable improvement in the time benefit unlike observed for the case of SMA in Figure 7.5. However, for models built with with the other baseline fidelities, a time improvement is still visible. These results are a strong indicator towards the possibility of further research in the field of multifidelity methods for

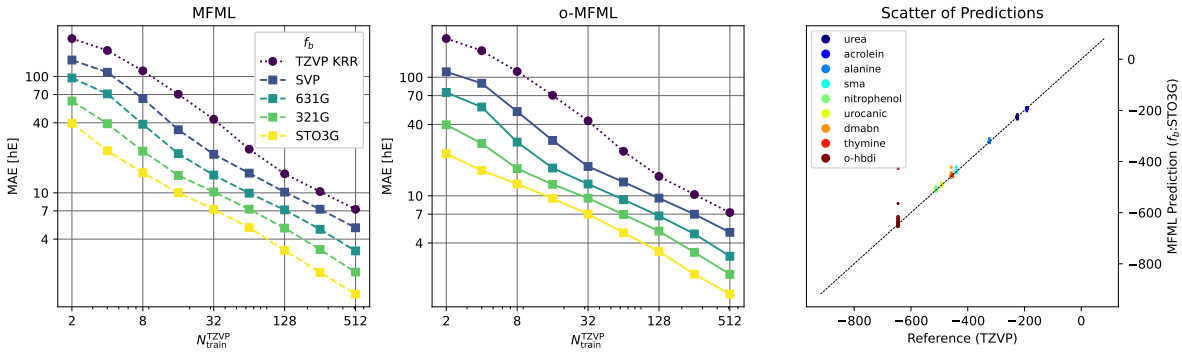


Figure 7.9: Learning curves for MFML and o-MFML for SCF ground state energies based on the cumulative use of the QeMFi dataset. 1,500 samples were randomly chosen from each molecule to perform this example test. The single fidelity KRR is also shown for comparison. The scatter plot of MFML predicted versus reference ground state energies is also shown. The MFML and o-MFML models perform well on the cumulative dataset showing errors in the range of a few hE for the ground state energies.

QC.

7.3.3 Cumulative use of the dataset

The QeMFi dataset contains multifidelity QC properties of 9 molecules for 15,000 geometries. This totals to $9 \times 15,000 = 135,000$ point calculations of the QC properties. This is therefore the largest collection of multifidelity dataset which can be used in various benchmarking processes. To demonstrate this form of cumulative use of the dataset, multifidelity models from Chapter 6 were tested against this in predicting the ground state energies of the molecules. From each molecule of the QeMFi dataset, 1,500 geometries were randomly chosen and compiled into a total of $9 \times 1,500 = 13,500$ data points. From the 13,500 samples, a random set of 11,000 samples were used as the multifidelity training data. Of the remaining, 500 samples were used as a validation set and 2,000 as the holdout test set. With this setup learning curves were generated for the different multifidelity models in the same fashion as prescribed in Chapter 6.

The results of this test are shown in Figure 7.9 for MFML and o-MFML models. The learning curves show a decreasing slope for both cases for the different baseline fidelities. The addition of each cheaper fidelity results in a lower offset of MAE. The constant slope on the log-log axis indicates that addition of training samples can further decrease the MAE. On the right-hand side plot the scatter of reference TZVP versus MFML predicted SCF ground state energies are delineated. Across the energy ranges the MFML model predicts the SCF ground state energies accurately as can be inferred from the scatter of the

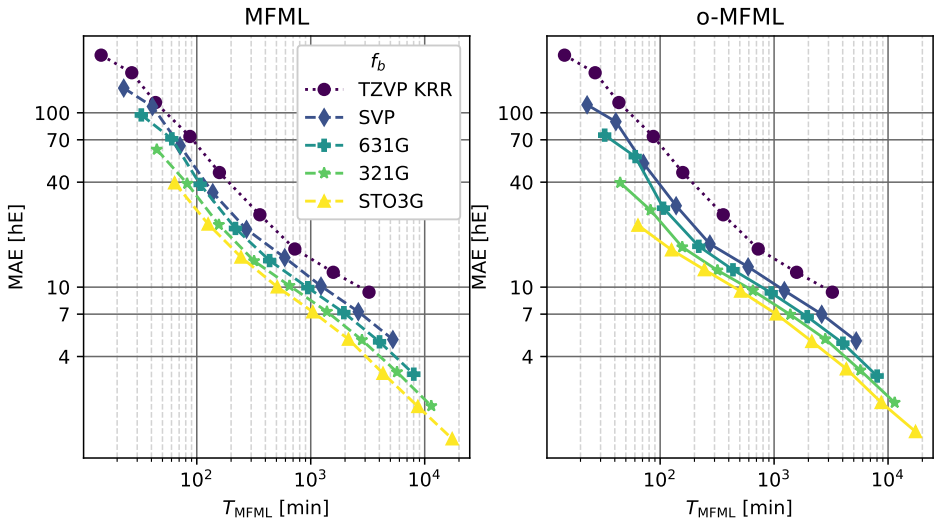


Figure 7.10: MAE versus time taken to generate a training set for MFML and o-MFML for the cumulative use of the QeMFi dataset. Here one observes the time-benefit of using MFML over single fidelity KRR for each cheaper baseline fidelity.

values being close to the identity mapping line.

For completeness, the MAE versus time to generate the multifidelity training data are reported in Figure 7.10 for MFML and o-MFML for this cumulative use of the QeMFi dataset. In this specific case for the cumulative use of the QeMFi dataset, the time to generate the multifidelity training data is calculated based on the molecular geometry that is included in the model. It is observed in Figure 7.10, that the addition of each baseline fidelity results in a distinct reduction in the time cost of generating training data. The resulting curves indicate a time benefit factor of about 6 for the STO3G baseline fidelity.

7.4 Usage Notes

In addition to the multifidelity dataset, various tools to assess and benchmark multifidelity methods are also provided. These include scripts to perform preliminary analysis of the data based on the property of choice as recommended in Chapter 5, and the scripts to produce learning curves. Further, scripts to generate unsorted Coulomb Matrices, and the global SLATM descriptors are provided which are built upon the qmlcode package [66]. The scripts are easy to use and well documented allowing for a streamlined benchmarking process with an example shown in Listing 7.3 for the preliminary analysis. Listing 7.2 shows an example to generate the multifidelity learning curves.

```
# this code is running in an activated conda environment

#this creates the SLATM representations
(conda) $ python GenerateSLATM.py -m='sma' \
-d='../dataset/'

#this runs the code to generate learning curves of \gls{MFML}
(conda) $ python LearningCurves.py -m='sma' \
-d='../dataset/' \
-p='SCF' \
-n=10 \
-w=200.0 \
-rep='CM' \
-k='laplacian' \
-r=1e-10 \
--seed=42 \
--centeroffset

#this creates the various MAE files in npy format
#the following command will plot the learning curves as a PDF file
(conda) $ python LC_plots.py -m='sma' \
-p='SCF' \
-u='hE' \
-rep='CM' \
--centeroffset \
--saveplot
```

Listing 7.2: Python example to generate the global SLATM representations, multifidelity learning curves, and the corresponding plot of SCF ground state energies for the SMA molecule.

```
# this code is running in an activated conda environment
(conda) $ python PrelimAnalysis.py -m='sma' \
-d='../dataset/' \
-p='SCF' \
-u='hE' \
--centeroffset \
--saveplot
```

Listing 7.3: Python example to perform preliminary analysis of SCF ground state multifidelity data for the SMA molecule.

7.5 Code availability

All scripts needed to assess the QeMFi dataset are hosted at https://github.com/SM4DA/QeMFi. This includes scripts to run ORCA calculations, extract properties from the output log files, generating CM and SLATM molecular descriptors, and generating learning curves.

7.6 Conclusion

In this chapter the details of a new multifidelity dataset generated have been discussed with detailed information about accessing and utilizing it to benchmark multifidelity models. In particular, the MFML and o-MFML model have been benchmarked in this chapter on the QeMFi dataset. A diverse collection of QC properties are provided in the QeMFi dataset for 9 different molecules. It was shown that what sets the QeMFi dataset apart from the others is the compute-times for each fidelity and molecule are included allowing for time-cost benchmarks as presented in sections 7.3.2 and 7.3.3.

The QeMFi dataset will be used extensively in the following chapters. For instance, the nestedness of training data will be assessed for MFML and o-MFML using the QeMFi data in the next chapter. The efficiency of the MFML and o-MFML models will be evaluated alongside Δ-ML and its variants in the Chapter 8. In Chapter 10 the QeMFi dataset will be used to study the effect of the data hierarchy on the model accuracy for multifidelity models.

CHAPTER 8

ASSESSING NON-NESTED CONFIGURATIONS OF MULTIFIDELITY MACHINE LEARNING

This chapter is taken from the work published as ref. [42] in the Journal of Machine Learning: Science and Technology.

Machine learning (ML) for QC has become a rapidly developing field of research for the prediction of various QC properties [30, 154, 31]. With a focus on reducing the time taken for new computation with such ML-QC methods, research has recently begun focusing on reducing the time it takes to generate a training set, for such ML models. The introduction of the Δ-ML method [29] has allowed for ML models to be trained on the difference of two properties, one computationally expensive and other other computationally cheaper, sometimes even semi-empirical.

Multifidelity Machine Learning (MFML) following refs. [32, 142] was introduced to be a systematic methodological improvement of Δ-ML. The term fidelity is used to refer to the accuracy of the QC method used to train the model. In general, a high accuracy QC method is also cost-expensive. Here, multiple QC methods are used to create multiple sub-models, which are finally summed up to predict QC properties such as excitation energies at the most expensive QC method, or target fidelity. MFML was shown to produce low-cost high-accuracy models. A methodological improvement of MFML was introduced in Chapter 6 and shown to be superior to MFML in prediction atomization energies of molecules and vertical excitation energies along molecular trajectories.

Research in multifidelity methods has increased to cover a wide range of formulations including those that are different from Δ-ML based methods. Variations include hierarchical ML (h-ML) model which was shown to significantly reduce training data generation time in predicting the potential energy surface (PES) of CH_3Cl [139]. The h-ML method used more than two QC methods of calculations and implemented an *ad hoc* optimization procedure to select the number of training samples to be used at each level or *fidelity* of QC calculation. Multi-task GPR (MTGPR) have also begun to utilize the different fidelities to train one ML model simultaneously for multiple fidelities [110, 218]. Here, each fidelity is treated as inter-related to the others and a surrogate MTGPR model is created. Further, a diverse multifidelity dataset consisting of 135,000 point geometries has recently been made available [224, 41] with various QC properties, such as vertical excitation energies, calculated with DFT formalism. The QeMFi dataset is described in Chapter 7. The fidelities are differentiated by the choice of basis set used in the calculation. This dataset will be used in all benchmarks of this study.

In most variations of multifidelity methods, which have been used in predicting a range of QC properties [201, 32, 163, 139, 132, 142], there has been one similarity: the use of nested training data. Simply put, it refers to using the same molecular geometries across the different fidelities. If a molecular geometry is used at the highest fidelity, then it is also used at the subsequent lower fidelities. This approach was recommended in ref. [32] based on research from sparse-grid combination techniques [37, 36, 35] and carried forward in the works presented in Chapter 5 and Chapter 6. In ref. [83], multifidelity physics informed neural networks (MPINNs) are investigated for their ability in functional approximations of PDEs. Here as well, nested data is employed in the sense that the high and low fidelities are evaluated along the same inputs for most of the presented examples. However, it must be noted that ref. [83] does not assume nested data for the method proposed therein to work. In the cases where nested data is presented, it is shown that the MPINN model for the lower fidelity can be eliminated. Recently, ref. [110] investigated building multitask surrogate models with heterogeneous data. However, it is to be pointed out that the data used did not ensure complete non-nestedness since some training data at the lower fidelity included at least a few samples which were used as test data in the final surrogate model. While most ML-QC approaches with multifidelity methods use nested configurations of training data, this is not necessarily the case for the broader field of multifidelity methods. For instance, ref. [108] discusses the use of Gaussian processes (GP) with multifidelity data that is taken from different domains in order to solve fluid flow around an airfoil. this is achieved by learning a heterogeneous mapping function across the domains based on het-

erogeneous transfer learning [109]. The comparison of this to the QC application would be not just the use of non-nested geometries, but also different molecular descriptors at each fidelity. A similar multifidelity method has recently been employed to model laser directed energy deposition with GP [227]. Refs. [141, 109] provide a detailed review of non-nested data configurations of multifidelity methods with applications to fields such as image detection [228, 229] and drug efficacy [230].

A note is to be made on the motivation for the need of a non-nested configuration of training data for MFML in QC. The current state of MFML models usually trains them on nested datasets restricting their ability to be transferred across unrelated datasets. On the other hand, having non-nested configurations of MFML methods could enable the use of disparate datasets resulting in more flexible multifidelity methods without necessitating calculations at costlier fidelities. Thus, it becomes relevant to inquire the effectiveness of fully non-nested configurations of MFML for QC. In addition to reducing training data generation for multifidelity methods, it would result in more versatile MFML models which can combine across the molecular space without restrictions. While Chapter 6 introduced the o-MFML method, it was restricted to nested configurations of training data for excitation energies. In addition, therein, the excitation energies were studied for molecular trajectories of arenes. In contrast, this work uses a collection of diverse molecules from the QeMFi dataset for the study of both of ground state and excitation energies. Furthermore, the use of non-nested data for training multifidelity models is thoroughly analyzed here for both excitation and ground state energies. Therefore, even though the multifidelity methodology used in this work is similar to that from Chapter 6, the entire set-up of dataset, multifidelity data structure, and the applications thereof are entirely different.

This chapter compares the use of fully non-nested configurations of training data versus the nested configurations for multifidelity methods. The assessment is performed for the MFML and optimized MFML (o-MFML) models introduced in Chapter 4. These models are built to predict the ground state energies and first vertical excitation energies for molecules of the QeMFi dataset from Chapter 7.

8.1 Non-Nested Multifidelity Data from QeMFi

QeMFi contains various QC properties calculated at 5 fidelities for nine diverse molecules, namely: acrolein, alanine, thymine, urea, urocaninc acid, 2-nitrophenol, DMABN, SMA, and o-HBDI. For each of these molecules 15,000 geometries are provided with properties such as ground state energies and excitation energies calculated at different fidelities.

These are all TD-DFT calculations with varying basis sets constituting the different fidelities. The hierarchy of fidelities is taken as follows in increasing order: STO3G, 321G, 631G, def2-SVP, def2-TZVP. For the remainder of this work, these are referred to by their shortened nomenclature such as TZVP and SVP.

The QeMFi dataset contains a total of 135,000 point geometries of 9 diverse molecules. To ensure that the data chosen would indeed be non-nested the following strategy was employed: $1.5 \times 2^9 = 768$ samples were randomly chosen from the 135,000 for the TZVP fidelity. Of the remaining 134,288 samples, $1.5 \times 2^{10} = 1,536$ samples were chosen for the SVP fidelity. In this way, the STO3G fidelity contains $1.5 \times 2^{13} = 12,288$ training samples in total. Thus the total training set spans $768 + \ldots + 12,288 = 23,808$ training samples with the respective sampling for each fidelity. For the case of nested training data, across five fidelities, the corresponding number of training samples as mentioned above were chosen.

For the validation set to be used in o-MFML, 1,000 samples were chosen at random from the QeMFi dataset after removing all the training data. Similarly, a holdout test set was chosen consisting of 2,192 samples. In other words, the test set is never used in any stage of training the multifidelity models. The validation set and the test set are fixed and not changed during the course of the experiments in this work.

8.2 Results

To numerically study the effect of nestedness of training data for multifidelity methods, the two multifidelity models from Chapter 6 were built with details as reported in section 4.2. These were built to predict two QC properties from the QeMFi dataset [224, 41], namely, the ground state energies, and the first vertical excitation energies. In this section, a preliminary analysis as recommended in Chapter 5 is performed for the multifidelity data for ground state and excitation energies. Following this, the multifidelity learning curves for these two properties are analyzed for nested and non-nested training data set-ups with MFML and o-MFML models.

8.2.1 Preliminary data analysis

An assessment of using different descriptors for single fidelity KRR was performed between the Spectral London and Axilrod-Teller-Muto (SLATM) representation [51] and sorted and unsorted CM. Unsorted CM are simply molecular descriptors as returned by Eq. (2.1). In some application cases using CM matrices that are sorted by row-norm have been studied.

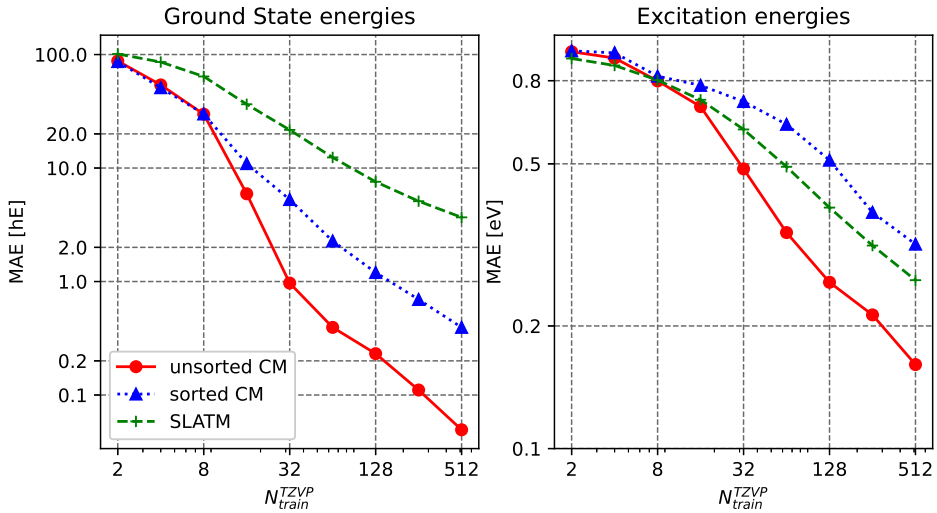


Figure 8.1: Comparison of the use of unsorted and row-norm sorted CM [44] and SLATM [51] representations for the prediction of ground state and excitation energies with single fidelity KRR at the TZVP fidelity. For both ground state and excitation energies, the unsorted CM outperforms the other representations.

This is usually noticed to introduce undesirable discontinuities in the representations [44, 50]. To assess the best molecular descriptor for this work, a short test was performed on the use of unsorted CM and SLATM representations for single fidelity KRR models. The TZVP fidelity properties were predicted with these models. The resulting learning curves are shown in Fig. 8.1 for both ground state and excitation energies. The horizontal axis on the left-hand side is scaled for the ground state energies, while the one on the right-hand side corresponds to the excitation energies. It becomes evident from the learning curves that the unsorted CM outperforms the SLATM representations for both ground state and excitation energies. Based on these results, the unsorted CM are used for the remainder of this work. The results are in favor of unsorted CM and therefore, the molecular descriptor used for this work is the unsorted CM. All multifidelity and single fidelity models hereon are built with unsorted CM representations.

Chapter 5 on MFML for excitation energies recommends that preliminary analysis be performed for multifidelity data to determine clear hierarchy of the fidelities and a systematic distribution of the fidelities to the target fidelity. The first preliminary analysis is to observe the multifidelity data distribution of the properties of interest, that is, to look at how the values are distributed across the energy domain. The second analysis measures the absolute difference of each fidelity to the target fidelity (in this case, TZVP). In this work, this specific analysis is depicted by the use of mean-marked box-plots. The final analysis for the

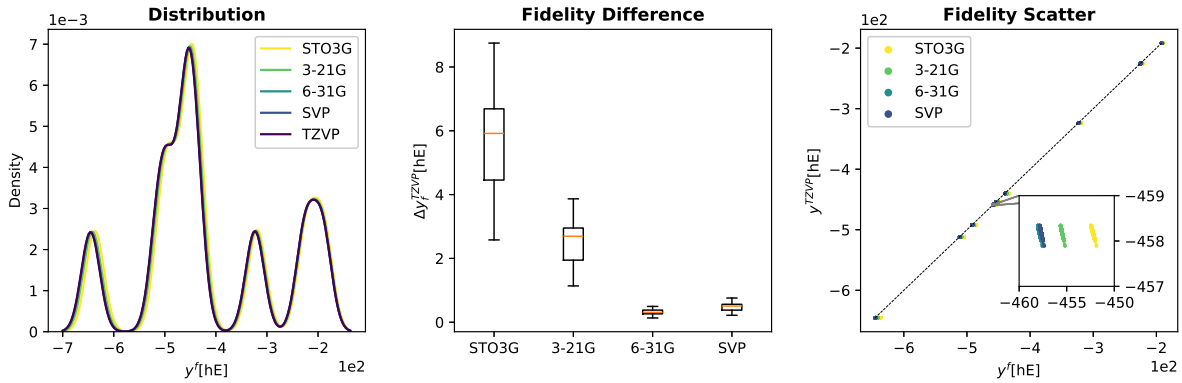


Figure 8.2: Preliminary analysis for the multifidelity structure of ground state energies. The distribution of the ground state energies shows that it covers a wider range of values. The absolute difference of the various fidelities to the target fidelity of TZVP shows that for the most part this decreases with increasing fidelity. A scatter plot of the various fidelity energies with respect to TZVP shows a systematic distribution of the energies as can be seen in the inset image.

preliminary assessment of the multifidelity data is a scatter plot of the property calculated at different fidelities with respect to the target fidelity.

This form of analysis for the ground state energies is depicted in Fig. 8.2 with the energies being in HE. Since the ground state energies belong to a conglomerate of molecules, the different peaks of the values are visible in the distribution plot. These peaks appear since the dataset consists of multiple molecules, each with a significantly different ground state energy on average. It is also to be noted that the energy distribution is also occasionally zero at certain locations on the energy domain for all fidelities. This will not be a challenge since the test set is also sampled from the QeMFi dataset and would lack energies within these valleys of the density plot. The difference in values for various fidelities, as seen in the center pane of the figure, shows a decreasing difference to the target fidelity for the most part, with the exception of SVP which is slightly more than its preceding fidelity of 6-31G. This minor deviation is not anticipated to cause any break down on the MFML model since Chapter 5 also identified such cases with MFML models still showing reasonable error reductions for nested configurations. The analysis recommended in Chapter 5 proposes a multifidelity data structure where the distribution of the energies of a fidelity are systematic with respect to the target fidelity. The exact values of the energy difference are not shown to make a significant difference to the MFML model. The box plot seen in the center pane of Fig. 8.2 shows the absolute difference in the energies of a given fidelity to the target fidelity of TZVP. The mean-line of the box plots are at different values for the different fidelities indicating that the MFML model could learn the energy differences with-

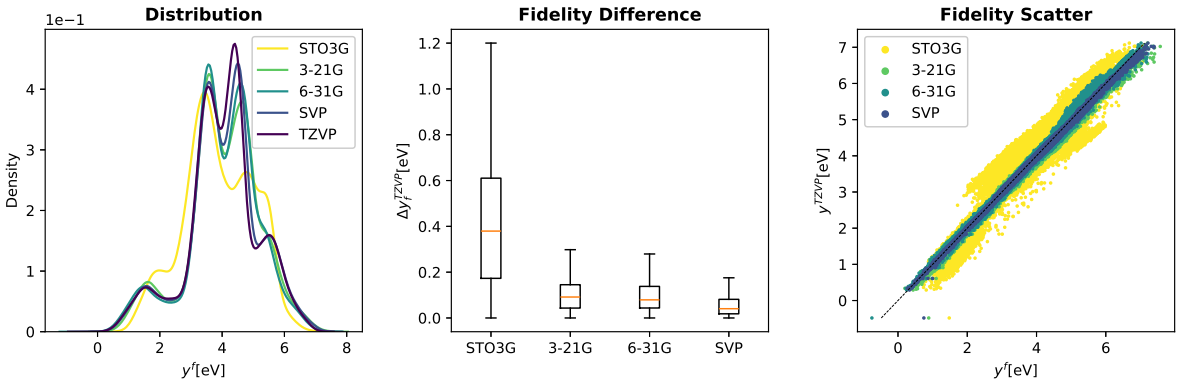


Figure 8.3: The multifidelity structure of the first vertical excitation energies are analyzed to confirm the assumption of hierarchy. The distribution of the energies on the left-most plot shows distinct peaks which correspond to the different molecules. The difference in fidelities to the TZVP fidelity decreases on average with STO3G having a larger ranges as can be seen in the box plot in the center. This is confirmed in the scatter-plot from the left-most plot as well where the STO3G energies show a wide distribution with respect to TZVP energies.

out having to struggle with the degeneracy of the fidelities. The right-hand side scatter plot does not show any clear distributions since the energies are spread over a large range of values. However, the inset shows that there is indeed a systematic distribution of the fidelities with respect to the target fidelity of TZVP. This systematic distribution of energies across different fidelities generally results in a meaningful MFML model as shown in Chapter 5. Therefore, one anticipates that the conventional MFML model for nested configuration will indeed show decreasing errors with addition of cheaper fidelities.

A similar analysis for the excitation energies sampled from QeMFi was performed and the results are shown in Fig. 8.3 for the energies in eV. Since the excitation energies are not as large as the ground state energies, in this case, one is able to better observe the different fidelities. The distribution plot shows various local variations for each fidelity. This is simply an indicator of the diversity of the data. Here one observes that most of the fidelities show a predominantly bimodal distribution. However, STO3G shows a slightly different distribution of energies. The central pane of Fig. 8.3 shows the fidelity difference mean marked box-plot where one observes that STO3G shows a larger range in contrast to the other fidelities. The fidelity difference depicts a decreasing value as one increases the fidelity. This means that the assumed hierarchy of methods is indeed correct. Finally, the scatter plot of energies calculated at various fidelities with respect to the TZVP fidelity shows a systematic distribution for all fidelities. Since STO3G has a wider spread in comparison to the other fidelities, it could potentially be less effective in a multifidelity model. However, as the work

in Chapter 6 has shown, the o-MFML method can still result in a fairly accurate model superseding conventional MFML.

8.2.2 Ground state energies

The learning curves for the diverse multifidelity models for the prediction of ground state energies are shown in Fig. 8.4 for the MFML and o-MFML models. The first row of the plots corresponds to the case, where the training data across various fidelities is from a nested data setup. Multifidelity learning curves are interpreted a little different from the conventional ML learning curves. Consider the case of nested MFML as seen in the top left pane of Fig. 8.4. The top learning curve corresponds to the standard KRR single fidelity method. Here, addition of training samples directly corresponds to the values of the x-axis. The next line is a MFML learning curve for f_b =SVP. In this case, if the x-axis shows $N_{\text{train}}^{\text{TZVP}} = 4$, then it also includes $2 \times 4 = 8$ samples at the SVP fidelity. Similarly as one goes down the baseline fidelities, the number of training samples used in the different fidelities are indicated by the number of training samples used at TZVP. For instance, the point on the learning curve of the STO3G baseline fidelity with 8 training samples at TZVP implies that the MFML model has $[8, 16, 32, 64, 128]$ training samples at TZVP, SVP, 631G, 321G, and STO3G respectively.

The learning curves for decreasing baseline fidelity for ground state energies are shown in Fig. 8.4 and show clearly lowered offsets. Consider the learning curve of single fidelity KRR with 128 training samples. If one were to draw a horizontal line at the corresponding MAE, it would intersect the multifidelity learning curve corresponding to STO3G at around $N_{\text{train}}^{\text{TZVP}} = 8$. This implies that MFML with STO3G baseline can be built with a lower number of expensive training samples and achieve the same error as a standard KRR model. For $N_{\text{train}}^{\text{TZVP}} = 512$, both the models $P_{\text{MFML}}^{\text{STO3G}}$ and $P_{\text{o-MFML}}^{\text{STO3G}}$ report an error 0.010 hE ($\sim$ 6.2 kcal/-mol) which shows that these two models are close in performance. One reason these models perform nearly same could be that the default MFML combination of the sub-models is already optimized. Such results have been previously reported in Chapter 6 for some cases. In such nested training data configurations, where MFML and o-MFML show similar performance in terms of model accuracy, it is better to opt for the less computationally expensive MFML method. The o-MFML would incur the cost of an additional validation set at the target fidelity.

The second row of Fig. 8.4 shows the MFML and o-MFML learning curves for the case of non-nested training data of ground state energies, as explained in 4.1. One immediately notices that the conventional MFML model breaks down with a non-nested multifidelity training dataset. It fails to provide any reasonable improvement for the different baseline

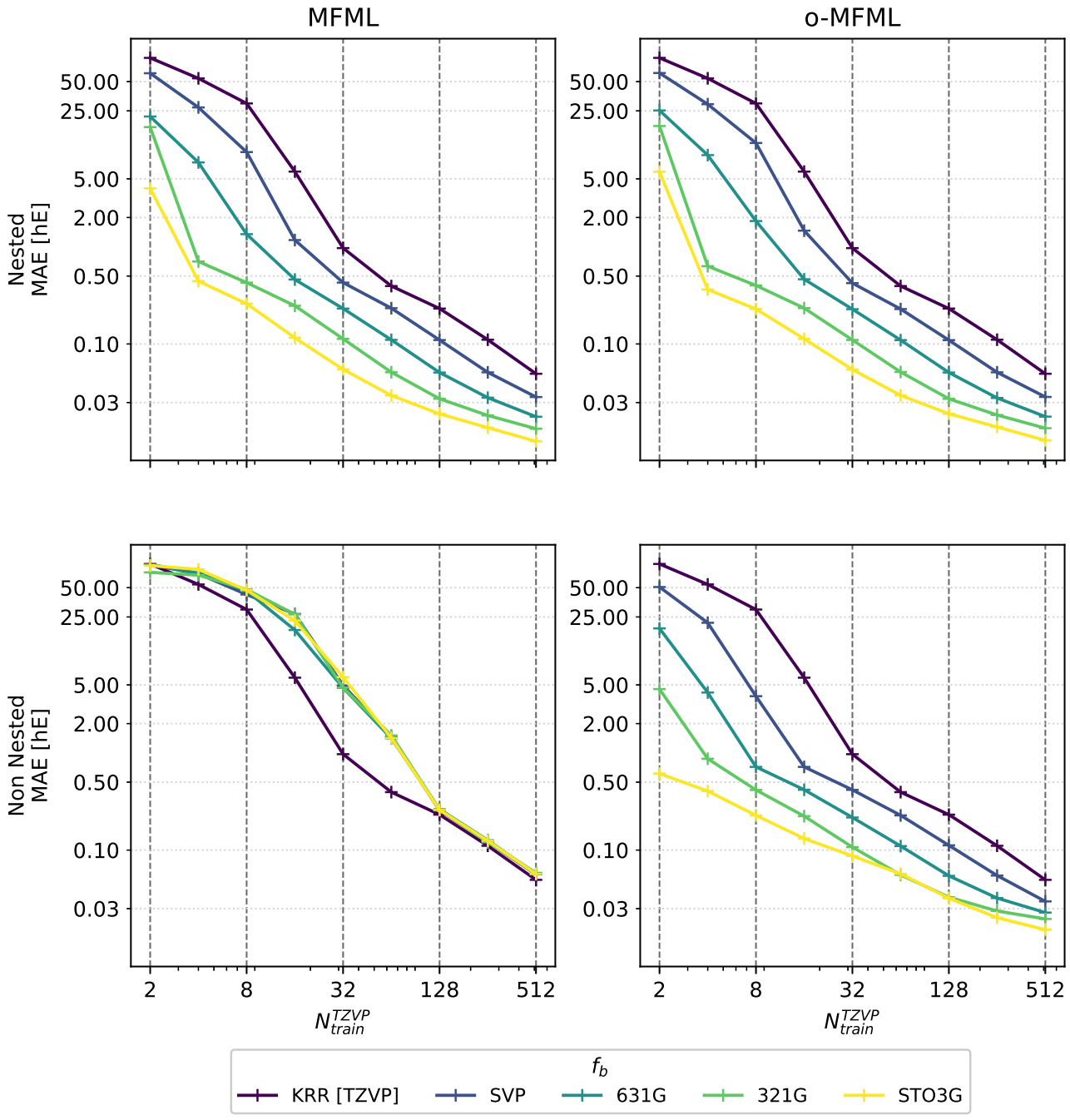


Figure 8.4: Learning curves of the MFML and o-MFML models built for ground state energies. The top row corresponds to nested training set case while the bottom row shows the results when non-nested training sets are used to build multifidelity models. Both conventional MFML and o-MFML are assessed here with the help of learning curves. The reference single fidelity KRR is also shown.

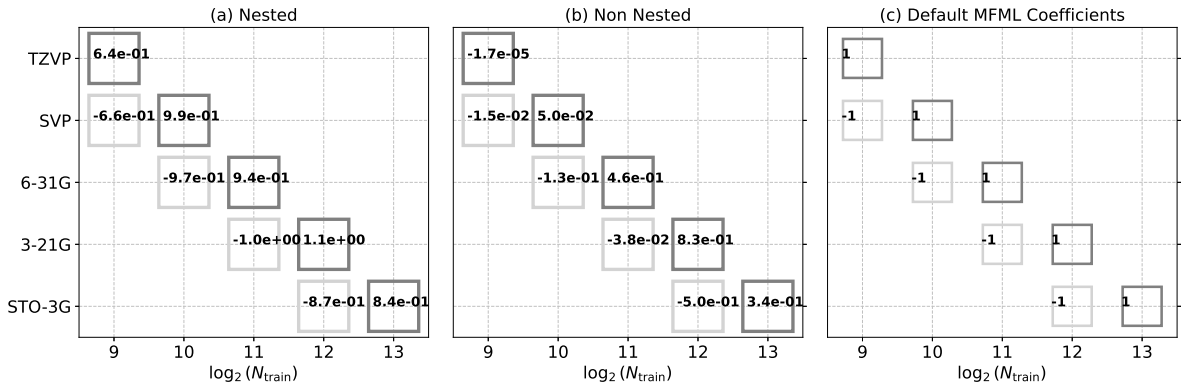


Figure 8.5: A study of the optimized coefficient values of o-MFML in predicting the ground state energies of the QeMFi dataset. (a)Nested configuration of training data. (b) Non-nested configuration of training data. (c) The default coefficients of MFML are also reported for comparison.

fidelities. Regardless of the training set sizes, the conventional MFML models fail to reduce the MAE in comparison to the single fidelity KRR model. On the other hand, for the o-MFML models built with varying baseline fidelities shows improvement similar to the MFML and o-MFML models built with nested training data. For $N_{\text{train}}^{\text{TZVP}} = 512$, the MAE of $P_{\text{o-MFML}}^{\text{STO3G}}$ is 0.015 hE ($\sim$ 9.4 kcal/mol) which is only negligibly larger than it was for the case of the nested training data. For $N_{\text{train}}^{\text{TZVP}}$ up to about 2^4, the non-nested configuration of o-MFML results in model error that is comparable to the nested configuration of o-MFML. However, it must be noted that for the non-nested case of o-MFML, the reduction in error with addition of cheaper fidelities is not as pronounced as it is for the nested configuration for larger number of training samples. The learning curves for baseline fidelities of 321G and 631G appear to be converging. This goes to show that non-nested configurations are indeed a challenging task.

To better comprehend the behavior of o-MFML and the corresponding results, one can study the optimized coefficients, $\boldsymbol{\beta}_{\boldsymbol{s}}^{\text{opt}}$, of o-MFML. For the ground state energies, these are shown in Fig. 8.5 with the default MFML coefficients shown in Fig. 8.5(c) for reference. The values of $\boldsymbol{\beta}_{\boldsymbol{s}}^{\text{opt}}$ are also reported in Table 8.1 for the nested and non-nested configurations of o-MFML to aide easy comparison. For each plot in Fig. 8.5, the x-axis implicitly depicts the value of training samples used at the fidelities, which are denoted on the y-axis. Each box, therefore, represents a sub-model used to build the final multifidelity model. The values of the coefficients are shown inside the square boxes and correspond to $\beta_{\boldsymbol{s}}^{\text{opt}}$ for o-MFML and $\beta_{\boldsymbol{s}}^{\text{MFML}}$ for conventional MFML.

Fig. 8.5(a) shows the values of the coefficients for the case of nested training data. One

$\boldsymbol{s} = (f, \eta_f)$	**Nested**	**Non-nested**
(TZVP,9)	0.64	-1.7×10^{-5}
(SVP,9)	−0.66	0.15
(SVP,10)	0.99	0.05
(6-31G,10)	−0.97	−0.13
(6-31G,11)	0.94	0.46
(3-21G,11)	−1.0	−0.38
(3-21G,12)	1.1	0.83
(STO-3G,12)	−0.87	−0.50
(STO-3G,13)	0.84	0.34

Table 8.1: Coefficient values of o-MFML for nested and non-nested configurations for the prediction of ground state energies for the QeMFi dataset.

observes that the values of the coefficients of o-MFML for the nested configuration, both magnitude and sign, are close to the default MFML coefficient values. This could be due to the conventional MFML model already being optimal. Furthermore, the magnitudes of the coefficients are similar implying that the each sub-model contributes almost similarly to the overall multifidelity model.

For the non-nested configuration of o-MFML, the resulting values of $\boldsymbol{\beta}_{\boldsymbol{s}}^{\text{opt}}$ are shown in Fig. 8.5(b). One notices a significant deviation from the the default coefficient values of MFML. Further, the values of the coefficients cover a wider range of values in comparison to the nested configuration of o-MFML.

In Fig. 8.5(b) it is observed that the magnitude of the coefficient for the sub-model built with TZVP fidelity training data is small, of the order of 10^{-5}. This is about 3 orders of magnitudes smaller than the coefficients of the sub-models for other fidelities. Chapter 6 did identify a similar case except for an intermediate fidelity. Following this, the fidelity with exceedingly small coefficients were removed and o-MFML models were rebuilt. The chapter reported an MAE comparable to having included the fidelity. In line with this test, for this current work, o-MFML models were built with non-nested configurations. The first was for the complete multifidelity structure for $N_{\text{train}}^{\text{TZVP}} = 512$ which results in 1024 samples at SVP. This is the same non-nested multifidelity structure that is used across the majority of this work. This model results in a model MAE of 0.015 hE. A second o-MFML model is built by ignoring training data at TZVP altogether. That is, the non-nested multifidelity structure now only contains STO3G, 321G, 631G, and SVP fidelities. This mode is now built with $N_{\text{train}}^{\text{SVP}} = 1024$ for uniform comparison. It is to be noted that this model too is optimized on the same validation set as before and evaluated on the same test set as for the previous model. This results in a MAE of 0.468 hE. These results indicate that although the

o-MFML model results in a very small magnitude of the coefficient for the sub-model built with the TZVP fidelity, this fidelity is still pertinent for the model to accurately approximate this target fidelity.

This could further confirm, in addition to the learning curves, that the non-nested configuration does in fact pose a challenge to the multifidelity model. A strong deviation of $\boldsymbol{\beta}_s^{\text{opt}}$ from the values of $\boldsymbol{\beta}_s^{\text{MFML}}$ indicates a significant change in the contribution of the corresponding sub-models in the final multifidelity model. This could imply that with the non-nested configuration, each sub-model being trained on distinct training data, cannot be combined as in Eq. (4.7) but perhaps requires something different. However, the flexibility of o-MFML in optimizing the coefficients allows it to optimally combine the sub-models even if the samples are non-nested resulting in a multifidelity model that still reduce errors with addition of cheaper fidelities. The optimization over the validation set for this model makes it superior over the conventional MFML model in the non-nested configuration.

8.2.3 Excitation energies

The prediction of excitation energies is in general considered to be more challenging than predicting ground state energies [22, 154]. The multifidelity learning curves for the prediction of excitation energies is shown in Fig. 8.6 for both nested and non-nested configurations of MFML and o-MFML models. The nested configuration for both categories of multifidelity models shows promising results for the prediction of excitation energies. A constantly lowered offset is observed with addition of cheaper baselines in addition to a negative slope of the learning curves. Similar to the case of ground state energies, the negatively sloped learning curves indicate that the addition of further training samples could potentially decrease the error of prediction. It is also observed that MFML and o-MFML have similar model errors. In nested configurations of training data for multifidelity methods, it is then recommended to use the MFML method since it is computationally less expensive. Although the o-MFML model reports an error of 0.05 eV ($\sim$ 1.1kcal/mol), it is to be pointed out that this is the ML model error. The overall error further includes the error of the method the ML model was trained on. Therefore, we refrain from arguing that the o-MFML model may provide predictions for excitation energies at chemical accuracy.

However, for the non-nested configuration for MFML, the learning curves indicate a poor performance of the models. For MFML, similar to the case of ground state energies, the entire multifidelity structure seems to have broken down with no meaningful model being formed for any baselines fidelity being added. In fact, the addition of cheaper baselines worsens the model. It is a ready conclusion that the non-nested configuration of MFML

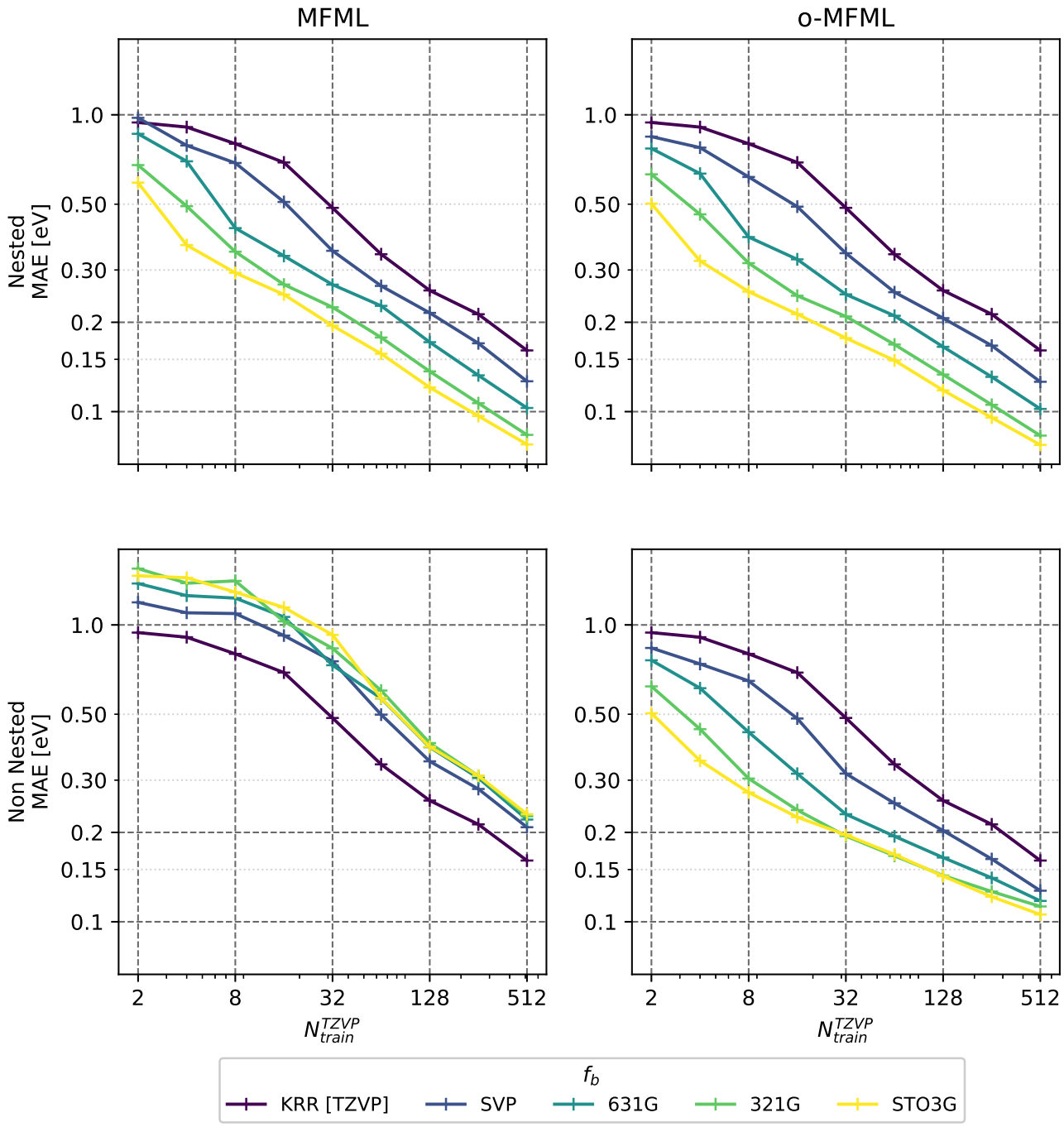


Figure 8.6: Multifidelity learning curves for the prediction of first vertical excitation energies. The first row shows the results for MFML and o-MFML with nested training set data. Similarly, the second row delineates the learning curves for the case of non-nested training data. The learning curve for a single fidelity KRR model is also shown for reference.

fails for the prediction of excitation energies similar to the ground state energies. On the

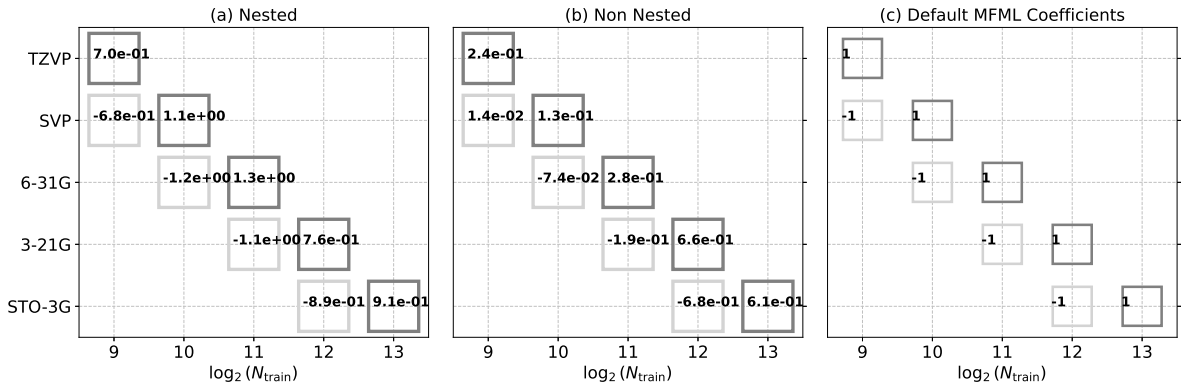


Figure 8.7: o-MFML coefficient values for the prediction of excitation energies of the QeMFi dataset. (a)Nested configuration of training data. (b) Non-nested configuration of training data. (c) The default coefficients of MFML are also presented for ready reference.

other hand, with o-MFML, the non-nested configuration performs noticeably better as in the case for ground state energies. The addition of SVP and 631G fidelities in the multifidelity structure improves the model albeit not as well as seen in the nested configuration of o-MFML. The learning curves show lowered offsets for these baseline fidelities, although for SVP, the difference is not as significant as it was for the nested configuration of o-MFML. With the 321G and STO3G fidelities, the o-MFML model shows improvements for small to medium training set sizes. But with $N_{\text{train}}^{\text{TZVP}} = 512$, the learning curve for the 321G baseline fidelity converges to MAE values close to that corresponding to f_b =631G. This is also observed for the STO3G baselines, where even for medium training set sizes, the multifidelity learning curve converges to that with f_b =321G. One possible reason for this could be that as larger training samples are used at TZVP, the number of training samples at the lower fidelities scales by 2. For large enough TZVP training samples, therefore, the multifidelity model has a larger amount of non-nested data to combine and the OLS optimization struggles to optimize these large and seemingly unrelated (due to non-nestedness) submodels. For the most part the learning curves show a constant slope which could indicate that the addition of further training samples might reduce the prediction error of the model resulting in more optimal values of $\boldsymbol{\beta}_s^{opt}$.

A further analysis of the o-MFML models can be performed with the study of the value of $\boldsymbol{\beta}_s^{opt}$ for nested and non-nested configurations of o-MFML for the prediction of excitation energies. These are shown in Fig. 8.7 where Fig. 8.7(c) also shows the default coefficients of MFML for reference. The coefficients for nested configuration of o-MFML are delineated in Fig. 8.7(a). Here, one observes that most of the coefficients lie in the same range of values and closer to the default values of conventional MFML coefficients. As ar-

$s = (f, \eta_f)$	**Nested**	**Non-nested**
(TZVP,9)	0.7	0.24
(SVP,9)	−0.68	0.14
(SVP,10)	1.1	0.13
(6-31G,10)	−1.2	−0.074
(6-31G,11)	1.3	0.28
(3-21G,11)	−1.1	−0.19
(3-21G,12)	0.76	0.66
(STO-3G,12)	−0.89	−0.68
(STO-3G,13)	0.91	0.61

Table 8.2: Coefficient values of o-MFML for nested and non-nested configurations for the prediction of excitation energies of the QeMFi dataset.

gued in Chapter 6, this could indicate that the combination of the sub-models was already optimized with the default values of the coefficients. The values of $\boldsymbol{\beta}_s^{opt}$ for the non-nested configuration of o-MFML are shown in Fig. 8.7(b). The values of the coefficients for the three cheapest fidelities change significantly in comparison with the nested configuration. For 321G and STO3G the change is not very significant. This could be due to the additional noise that these fidelities include into the multifidelity model due to the non-nested training data. As noted in the case for ground state energies, the large number of training samples which are unrelated due to the non-nested configuration might prove challenging to the OLS optimizer. Thus, it becomes additionally difficult to discard the noise from this training data structure to optimally combine the sub-models to provide a multifidelity model. The values of the coefficients are also reported in tabular format for this case in Table 8.2.

8.3 Conclusion

Through the various numerical tests employed in this chapter, the effect of nestedness of training data has been evaluated for multifidelity models. It is seen that nested configurations of MFML and o-MFML generally out perform their non-nested counterparts. However, the use of o-MFML with non-nested training data shows promising outlooks. A focus on improving the optimization routine and possibly including steps to account for the noise incorporated by the non-nested training data could potentially make this a vital tool in ML for QC. Future work on non-nested configurations of training data could include the use of other multifidelity methods such as h-ML or multi-task methods. Improved multifidelity models for non-nested configurations would allow for a more flexible use of these

models.

Overall, the work presented here opens up areas for research in the use of non-nested configurations of multifidelity models. The scope of using methods such as o-MFML to tackle non-nested and heterogeneous multifidelity data has become evident through the numerical examples in this chapter. Although using non-nested training data seems to be a bottle neck for current multifidelity models, further research in this direction can certainly improve them.

CHAPTER 9

BENCHMARKING DATA EFFICIENCY IN Δ-ML AND MULTIFIDELITY MODELS

This chapter is taken from the work ref. [144] hosted at the time of this submission as a pre-print on arXiv.

Simultaneous progress in both quantum chemistry (QC) theory and machine learning(ML) methods has resulted in a wide range of applications ranging from molecular dynamics to alloy design [231, 231, 232, 30]. Such ML models are often optimized to predict singular molecular properties such as atomization energies [138, 32] or molecular dipole moments [46], however also covering potential energy surfaces at both ground and excited states [139, 22, 181].

Once trained, ML models for QC are capable of reducing the time-cost of making new predictions for unseen data tremendously in comparison to conventional QC computation [31, 30, 22]. However, it is a common observation in ML-QC that the more samples a ML model is trained on, the better the accuracy of the model [22, 165, 154]. Recent research in developing ML methods for QC has therefore begun posing a slightly different question to this entire approach: how long does it take to generate training data for such ML models? Various methodological improvements have since been developed to tackle this specific aspect of the ML-QC approach. These include Δ-ML [29], MFML [32], and hierarchical-ML [139]. The Δ-ML approach trains an ML model on the difference between two *fidelities* of QC data. The term fidelity refers to the accuracy of the QC method. It is generally the

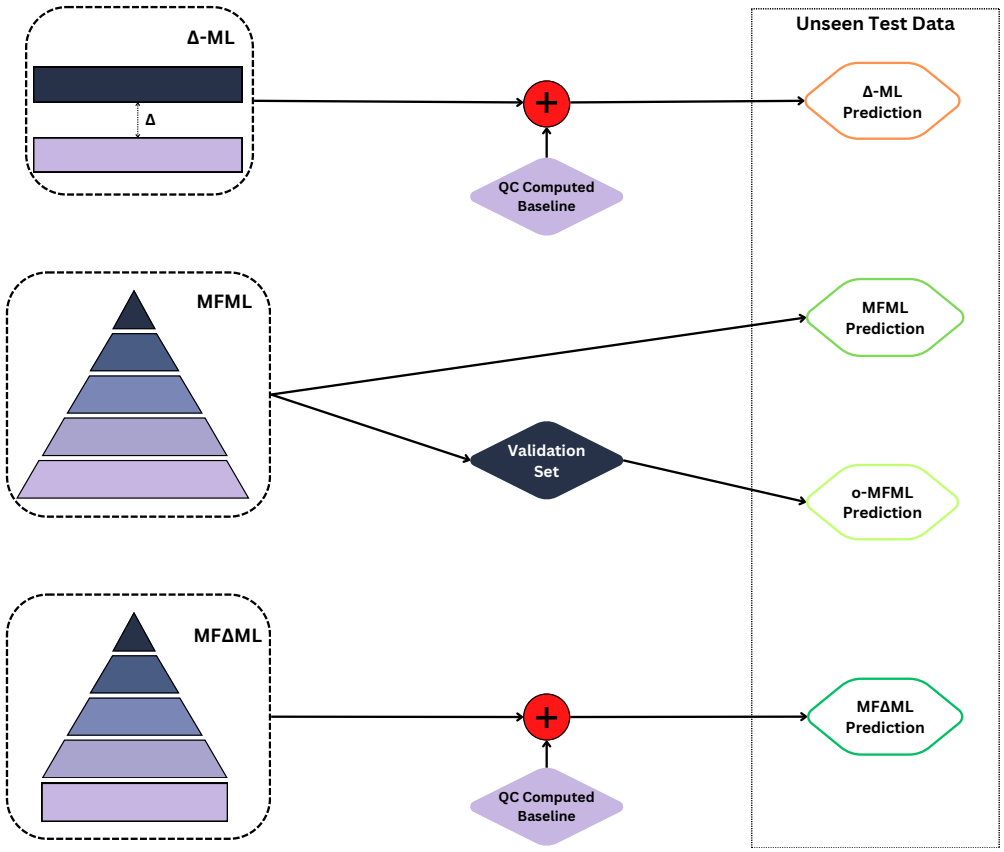


Figure 9.1: A visual depiction of the different ML methods benchmarked in this chapter. The MFML and o-MFML models do not need any further QC-calculations for the unseen test set once they have been trained. In contrast, Δ-ML method and the MFΔML method that is introduced herein, both require additional QC computations at the QC-baseline that is used in these models. This work benchmarks these different models to understand the time-cost versus model accuracy efficiency.

case that a higher fidelity is associated with a higher compute-cost. Thus, Δ-ML trains on a costly fidelity, called the *target fidelity*, and a cheaper fidelity, referred to herein as the *QC-baseline fidelity*. This is pictorially depicted at the top of Figure 9.1. The final predictions involve making the QC-baseline fidelity calculations and adding to that the difference predicted by the Δ-ML model. The h-ML approach builds several Δ-ML like models for more than two fidelities where the training samples at each fidelity are decided by minimizing a cost function based on user defined error and compute-cost budget in generating training data. This model was shown to be effective in predicting the potential energy surface of the ground state energies in the CH_3Cl molecule [139].

Multifidelity machine learning (MFML) was introduced as a systematic generalization

of the Δ-ML method [32] with several ML models, called *sub-models*, being built with different fidelities and number of training samples. The method provides several methodological improvements over the Δ-ML approach in that the cheaper fidelity is no longer calculated but are predicted within the MFML method. In both MFML and Δ-ML, the training samples needed at the cheaper fidelities include the training samples used at the target fidelity. This property of the training data is referred to as the nestedness of training data [32], that is, the training data is homogeneous. A methodological development over the MFML method is the optimized MFML (o-MFML) method (see section 4.2) which uses a validation set computed at the target fidelity to optimally combine the sub-models of MFML. This method was shown to be superior in prediction of excitation energies and atomization energies in Chapter 6 and shown to be better in use cases where the training data is heterogeneous, or non-nested, in Chapter 8. The MFML and o-MFML methods are depicted in the middle of Figure 9.1. Note that the conventional MFML method only needs the training data while the o-MFML method requires the validation set computed at the target fidelity. The prediction is made without any further computation of the baseline fidelities. Another flavor of ML methods that has recently been introduced is the use of multitask Gaussian processes (MT-GPR) to predict molecular properties such as ionization potentials with a training data cost reduction of almost an order of magnitude when coupled with the Δ-ML approach [110]. Yet another MF approach that was recently introduced is the minimal multi-level scheme which optimizes a loss function in association with a loss function to arrive at optimal cost-error balance of the MF model [233].

This chapter provides a time-cost versus model accuracy benchmark of the Δ-ML, MFML, and o-MFML methods. A uniform assessment requires that these models be evaluated on the same dataset. For this purpose, the QeMFi dataset from Chapter 7 is used which contains five fidelities of QC properties for nine diverse molecules. Specifically, the ground state energies, the first and second vertical excitation energies, and the magnitude of the electronic contribution to molecular dipole moments are used from the QeMFi dataset in order to make time-cost efficiency benchmarks for the MF models. Since the compute cost of each fidelity for each molecule is given in the QeMFi dataset, a uniform assessment of the data efficiency, that is, the cost of generating training data for a certain model accuracy. In addition to the benchmarks for the there above stated models, this chapter introduces another MF method in interest of reducing the training data cost for the ML-QC pipeline. This is the multifidelity Δ-ML (MFΔML) method from section 4.3, wherein, a MFML model is built with several Δ-ML models which predict the QC property for different fidelities. This concept of this method is depicted at the bottom of Figure 9.1. The QC-baseline fidelity is

used to create the Δ-ML models of the higher fidelities in the training data structure. The final prediction requires the calculation of the QC-baseline fidelity to which the prediction from the MFΔML model are appended. This method is also benchmarked alongside the Δ-ML, MFML, and o-MFML methods for the time-cost incurred in generating the training data.

9.1 Dataset and Machine Learning Details

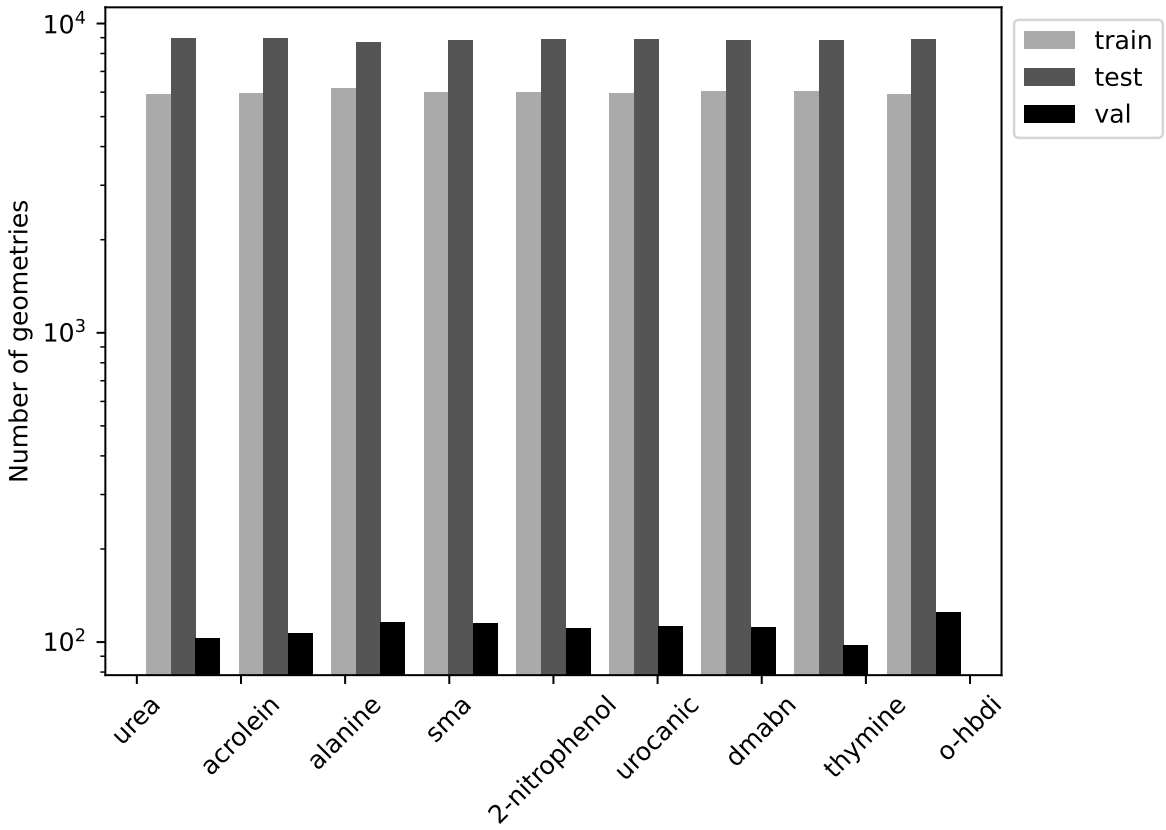


Figure 9.2: Distribution of training, validation, and test sets from the QemFI dataset used in this chapter. All the nine molecules of the QeMFi dataset are evenly present in each of the sets. The same train/test/validation split is used for all QC properties studied in this chapter.

In this chapter, the different ML models are evaluated for the following QC properties of the QeMFi dataset:

- ground state energies,

- first vertical excitation energies ($E_{(1)}$),
- second vertical excitation energies ($E_{(2)}$),
- magnitude of electronic dipole moments ($|\boldsymbol{\mu}_e|$)

Although ground state energies are considered easier than excitation energies for ML models to predict [22], both of these are studied in this work to ensure that the efficiency analysis of the MF methods is less dependent of the QC property. The magnitude of the electronic contribution to molecular dipole moments is studied instead of a component-wise vector prediction of the property since the latter requires specialized molecular representations which are equivariant under rotation and translation [46] unlike for the case of energies where invariance is mandated [21]. This is since the vector quantity of dipole moments is dependent on the orientation of the molecule itself. On the other hand, the magnitude is a rotation and translation invariant QC property which can be modeled with most conventional molecular descriptors such as the Coulomb Matrices used in this work (see section 2.1). The use and development of specialized descriptors for dipole moments lies outside the aim of this work, which is focused on the efficiency of the MF methods. Furthermore, only the electronic contribution is studied since the nuclear contribution to the molecular dipole moment is identical for a molecule regardless of the fidelity of data used.

The split of data into training, validation, and test sets was uniform across these different QC properties. Of the 135,000 geometries, a random collection of 54,000 data points were removed to be used for the necessary training data sets. From the remaining geometries, 1,000 samples were chosen at random to be the validation set to be used for the optimization procedure in o-MFML. The remaining 85,000 samples were set aside as the test set. It is to be noted that the test set is not used at any stage of training the models and is therefore a true proxy for unseen data. Since the QeMFi dataset consists of nine molecules of different sizes and chemical complexities, one should make sure that the different molecules are sufficiently represented in the training and test sets. In interest of this sanity check, the composition of the training, validation, and test sets is shown in Figure 9.2 based on this selection choice. It can be seen that the nine molecules are uniformly represented in all three sets. This form of sampling ensures that there is no separate influence of the composition of the dataset itself.

The molecular descriptor used in this assessment is the unsorted CM discussed in section 2.1. The Matérn kernel of first order with L_2 norm is employed with a kernel width, σ, of 150.0 for ground state energies, 20.0 for excitation energies, and 3000.0 for the magni-

tude of electronic contribution to molecular dipole moments based on a hyper-parameter grid search. For KRR, the value of the Lavrentiev regularizer, λ was set to 10^{-10}.

9.2 Results

In order assess the various MF models, the models were trained on a pool of MF data taken from the QeMFi dataset, partitioned as explained above. The models are evaluated on a holdout test set sampled as explained in the previous section. Learning curves are studied to understand model accuracy as a function of the number of training samples used at the costliest fidelity, that is TZVP. Next, the cost of training the MF models is calculated including any additional costs such as the validation set cost for o-MFML, or QC-baseline computation for the Δ-ML variants. This is the benchmark that is presented in this chapter. The time-cost of using the different MF methods are studied with recommendations of which method to use when based on the results.

9.2.1 Learning curves

Figure 9.3 reports the learning curves for the Δ-ML method with varying baseline fidelities for the (a) prediction of ground state energies, (b) first excitation energies ($E_{(1)}$), (c) second excitation energies ($E_{(1)}$), and (d) the magnitude of electronic contribution to the molecular dipole moment ($\boldsymbol{\mu}_e$). For all QC properties, one notices that the use of cheaper QC_b offsets the learning curves upwards. In other words, the closer QC_b is to the target fidelity of F, the better the Δ-ML model is. The difference in MAE between the STO-3G baseline and the SVP baseline for Δ-ML is observed to be almost an order of magnitude. This observation is similar to what is made in ref. [29], where the Δ-ML approach was first introduced. In Appendix A, the results of similar experiment for the QM7b dataset [49] are reported in section A.4.1. In that experiment, different QC levels of theory are considered as fidelities, as opposed to the different basis sets that are considered as fidelities in this chapter. The results once again confirm the trend of the Δ-ML model resulting in lower error for a QC-baseline that is closer to the target fidelity as seen in Figure A.13.

Figure 9.4 reports the learning curves for MFML and o-MFML models. Note that the vertical axis are scaled differently for each QC property with the units of the energies being reported in Kcal/mol while using Debye for dipole moments. For each cheaper f_b, one observes that the learning curve is lowered by a constant offset. For ground state energies, as seen in Figure 9.4(a), there is a region of pre-asymptotics that is observed for small training

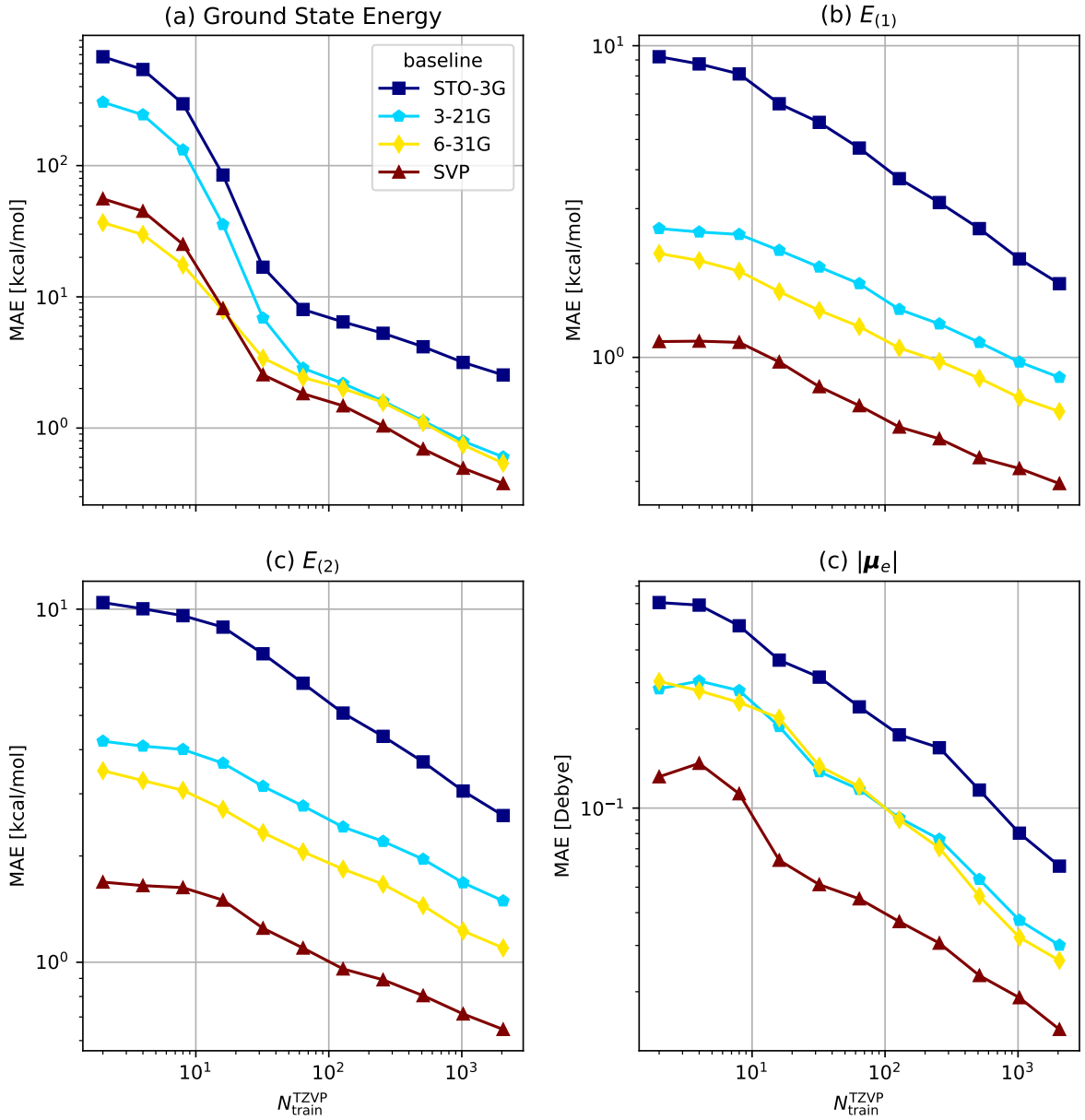


Figure 9.3: Learning curves for Δ-ML with varying QC_b. These are shown for the prediction of ground state energies, first vertical excitation energies ($E_{(1)}$), second vertical excitation energies ($E_{(2)}$), and the magnitude of electronic contribution to molecular dipole moments ($|\boldsymbol{\mu}_e|$). Across the QC properties, it is observed that the closer the QC_b is in hierarchy to the target fidelity, the better the model accuracy, as also observed in ref. [29].

set sizes up to $N_{\text{train}}^{\text{TZVP}} = 16$. However, for larger training set sizes, the learning curves show constant lowered offsets and nearly constant slope on the log-scaled axes. For both $E_{(1)}$ and $E_{(2)}$, this behavior of the learning curves is observed even for small training set sizes. As seen in Figure 9.4(d), the learning curves for the prediction of $|\boldsymbol{\mu}_e|$ too show lowered offsets

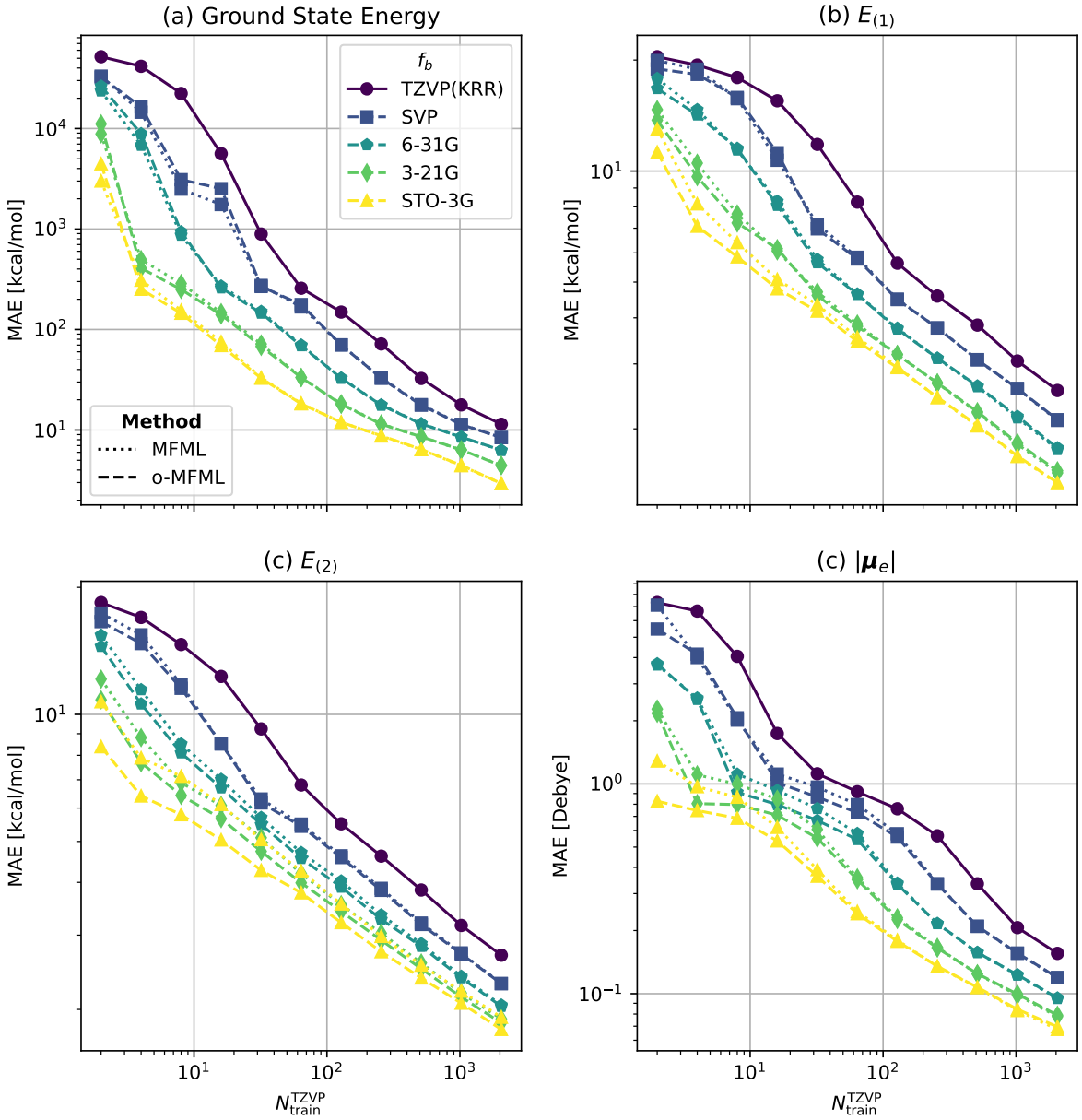


Figure 9.4: MFML and o-MFML learning curves for different QC properties studied in this chapter. The different baselines used in the MFML and o-MFML models are reported in the legend. It is seen that the o-MFML model does not provide a significant improvement over the conventional MFML model in terms of MAE. This could indicate that the MFML combination of sub-models is already sufficiently optimized.

with the addition of cheaper baseline fidelities. In all cases, the learning curves continue to have constant negative slope for large training set sizes. This indicates that further addition of training samples could indeed reduce model error. The learning curves for MFML and o-MFML do not show much difference in the model accuracy for any of the QC properties.

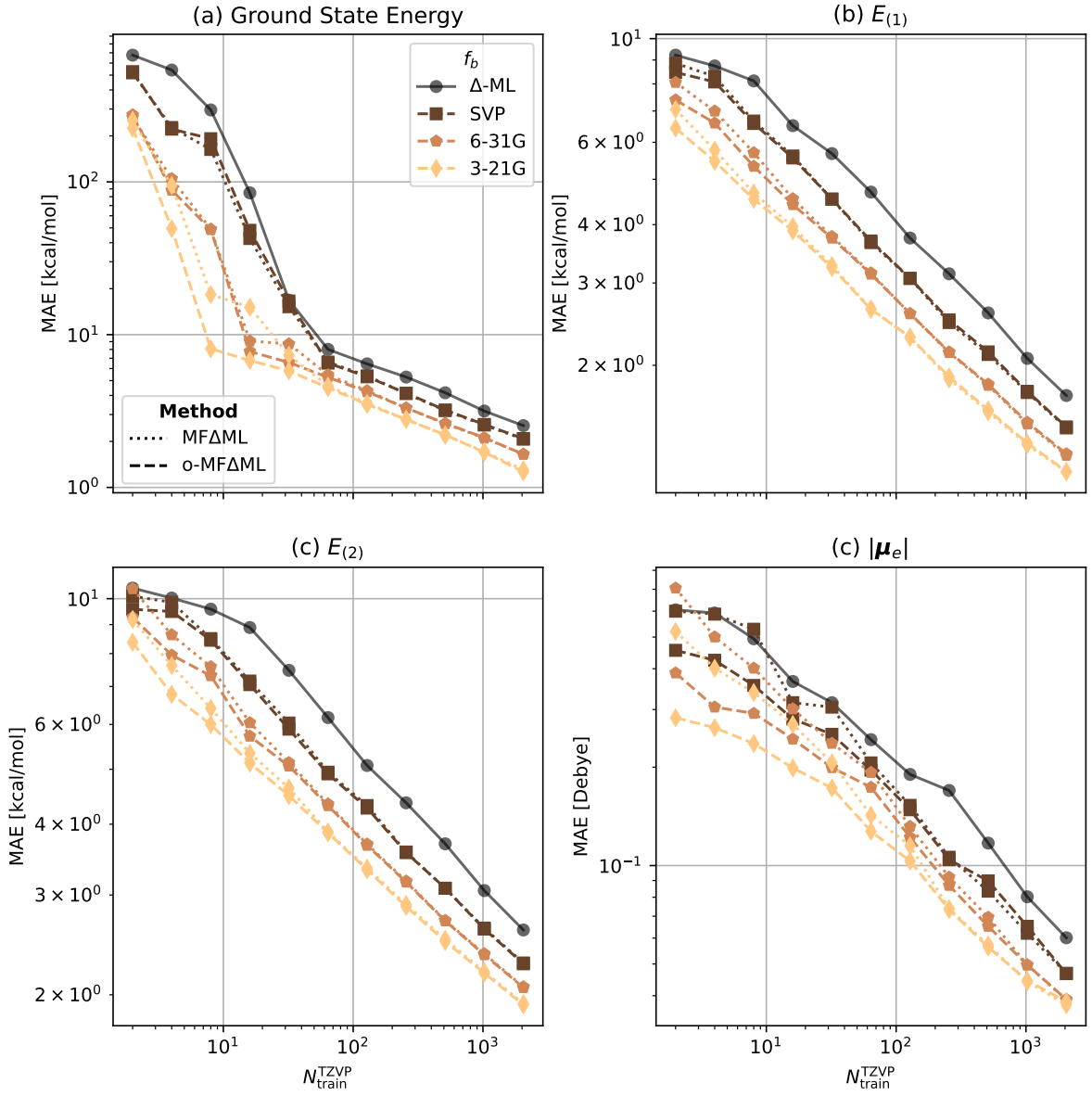


Figure 9.5: Learning curves for the MFΔML and o-MFΔML with differing baseline fidelities for the prediction of ground state energies, first vertical excitation energies ($E_{(1)}$), second vertical excitation energies ($E_{(2)}$), and the magnitude of electronic contribution to molecular dipole moments ($|\boldsymbol{\mu}_e|$) of the QeMFi dataset.

This could be due to the small size of the validation set in comparison to the large training and test set sizes that are considered. In Chapter 6 it was argued that such a behavior could also be due to the conventional MFML model being already optimized for the combination of the multifidelity sub-models. In such outcomes, it is to be noted that the MFML model is computationally more efficient over the o-MFML model, since there is no added cost of

generating a validation set. More about this is discussed in section 9.2.2 in relation with the time-cost results.

Finally, the learning curves for the MFΔML and o-MFΔML methods as introduced in section 4.3 are presented in Figure 9.5 for the different QC properties studied in this chapter. In this case, the energies are offset by the STO-3G fidelity. That is, the STO-3G energies are subtracted from the energies at the other fidelities as explained in section 3.4.1. Thus the lowest baseline that is used for the MFΔML model is the 3-21G fidelity. The learning curves are contrasted with the Δ-ML model built for F =TZVP and f_b=STO-3G. All learning curves show a region of pre-asymptotics for training set sizes up to $N_{\text{train}}^{\text{TZVP}} = 64$. In this region of pre-asymptotics, the o-MFΔML method provides some improvement for the 3-21G baseline fidelity. Beyond that, the MFΔML and o-MFΔML methods result in very similar MAE values. With MFΔML, it can be seen that the addition of cheaper fidelities to the model results in a constant lowered offset with a negative slope. The learning curves for the cheaper baselines of MFΔML are seen to be below the conventional Δ-ML model.

Property/Model	**KRR**	**MFML**	**Δ-ML**	**MFΔML**
Ground state energies [kcal/mol]	11.41	2.93	2.54	1.3
$E_{(1)}$ [kcal/mol]	2.54	1.42	1.72	1.18
$E_{(2)}$ [kcal/mol]	2.69	1.92	2.6	1.93
$\lvert\boldsymbol{\mu}_e\rvert$ [Debye]	0.16	0.07	0.06	0.04

Table 9.1: MAE in appropriate units for single fidelity KRR and multifidelity models with $N_{\text{train}}^{\text{TZVP}} = 2^{11}$ for different QC properties for STO-3G as the cheapest fidelity included. MAEs for o-MFML and o-MFΔML are not shown since they are very close in value to the conventional MFML and MFΔML values.

Table 9.1 reports the MAEs of single fidelity KRR and the different multifidelity models for comparison of the model accuracies for different QC properties. The MAEs correspond to models built with $N_{\text{train}}^{\text{TZVP}} = 2^{11}$. The MFML model is built with f_b=STO-3G, the Δ-ML model with QC_b =STO-3G, and the MFΔML model with QC_b =STO-3G and f_b =3-21G. These correspond to the last data point in the learning curves presented in this section. One observes in Table 9.1 that the MFΔML model has the lowest error regardless of the QC property that is studied. However, for $\lvert\boldsymbol{\mu}_e\rvert$ the difference is not as pronounced with respect to MFML and Δ-ML. The largest difference in the errors is seen for ground state energies, while both the excitation energies show considerable difference in the MAEs between single fidelity KRR and other multifidelity models. Thus, if only the model error is considered, the MFΔML method is seen to be a methodological improvement over both MFML and Δ-ML approaches.

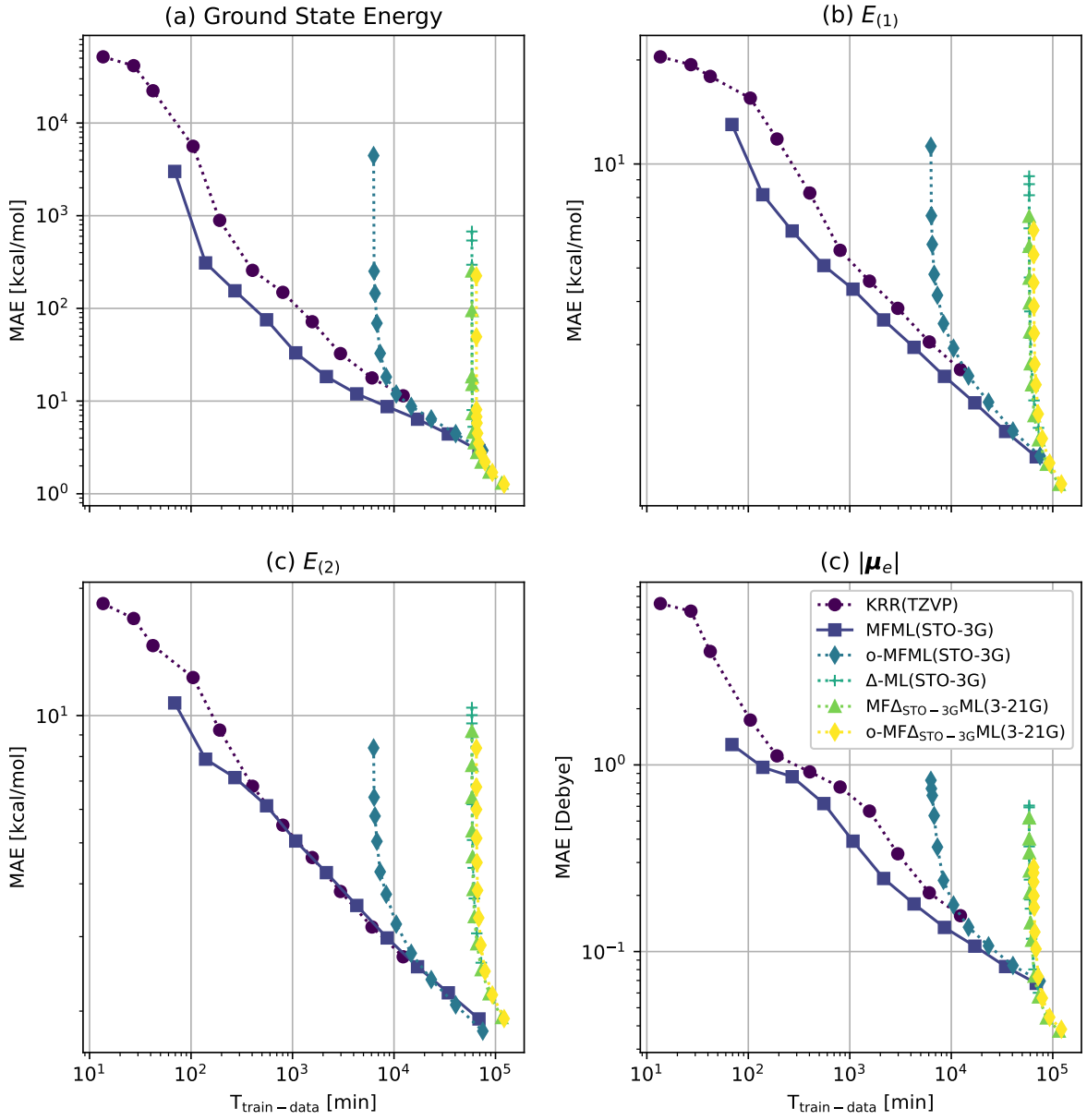


Figure 9.6: Time-cost assessment of the different multifidelity models for diverse QC properties. The x-axis reports $T_{\text{train-data}}$ in minutes which is the time taken to generate training data. For Δ-ML models this also includes the cost of the QC-baseline calculations.

9.2.2 Time-Cost assessment

The various ML methods studied in this chapter have promising results when the MAE is presented as a function of the training samples required at the target fidelity. However, as the reader is probably familiar by now in reading this dissertation, the total cost to generate the training data is to be studied. For the MFML models, this corresponds to all the training

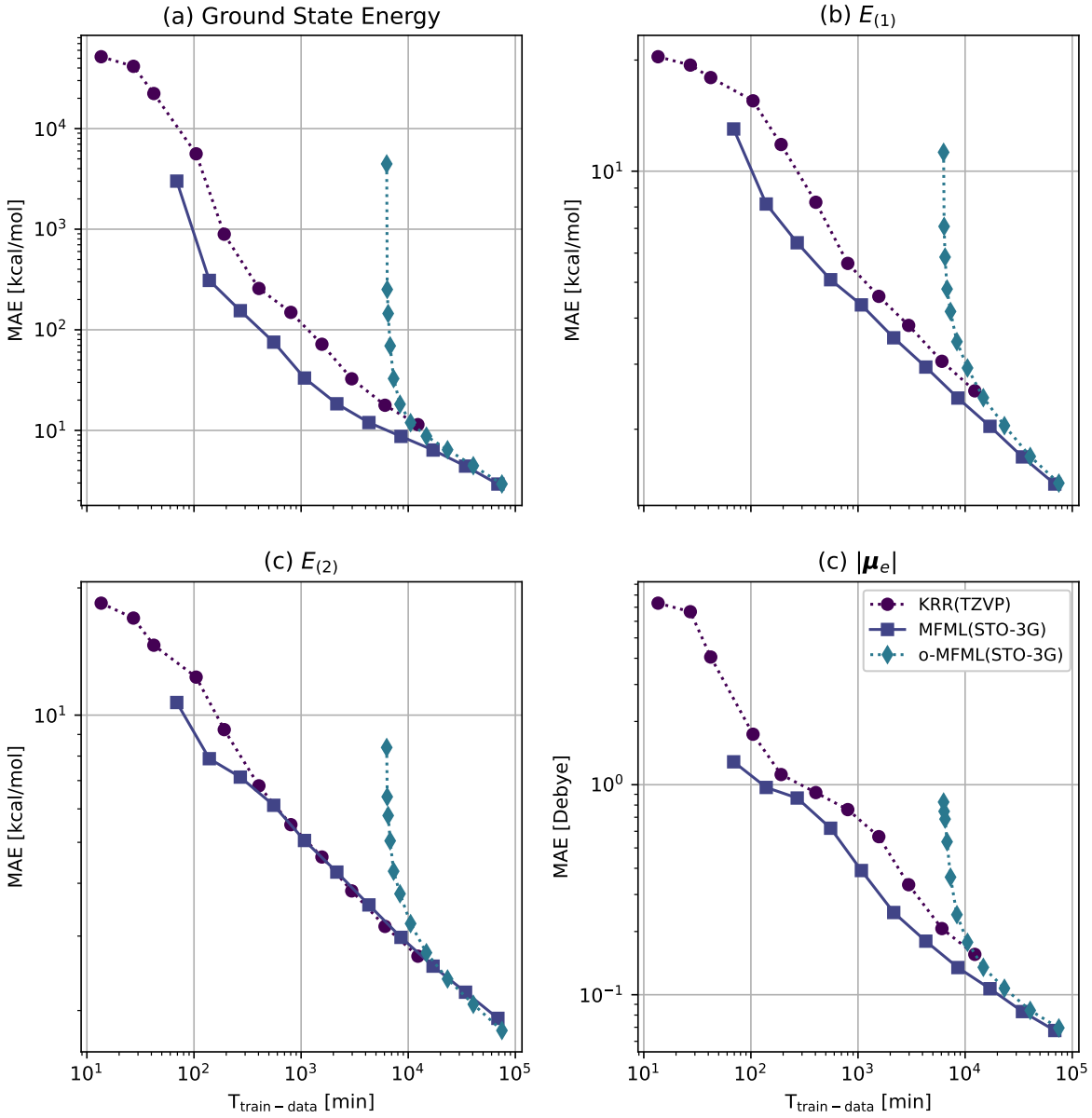


Figure 9.7: Time-cost versus MAE for MFML and o-MFML in comparison with single fidelity KRR for the prediction of diverse properties of the QeMFi dataset.

data used in the multifidelity structure. This is computed as $T_{\text{MFML}} = \sum_f N_f \cdot T_f$, where T_f is the time to perform the QC computation for fidelity f and N_f is the number of samples used by the MFML model at this fidelity. For o-MFML this calculation would further include the cost of the validation set, $N_{val} \cdot T_F$. For Δ-ML, the cost of the model for the prediction of some property for a molecule would be given as $T_{\Delta-ML} = N^F_{\text{train}} \cdot (T_{f_b} + T_F) + T_{f_b}$. In case several predictions are made, that is N_{test} number of predictions, then this cost becomes $T_{\Delta-ML} = N_F \cdot (T_{f_b} + T_F) + N_{\text{test}} \cdot T_{f_b}$. Note that T_F and T_{f_b} are different for different molecules

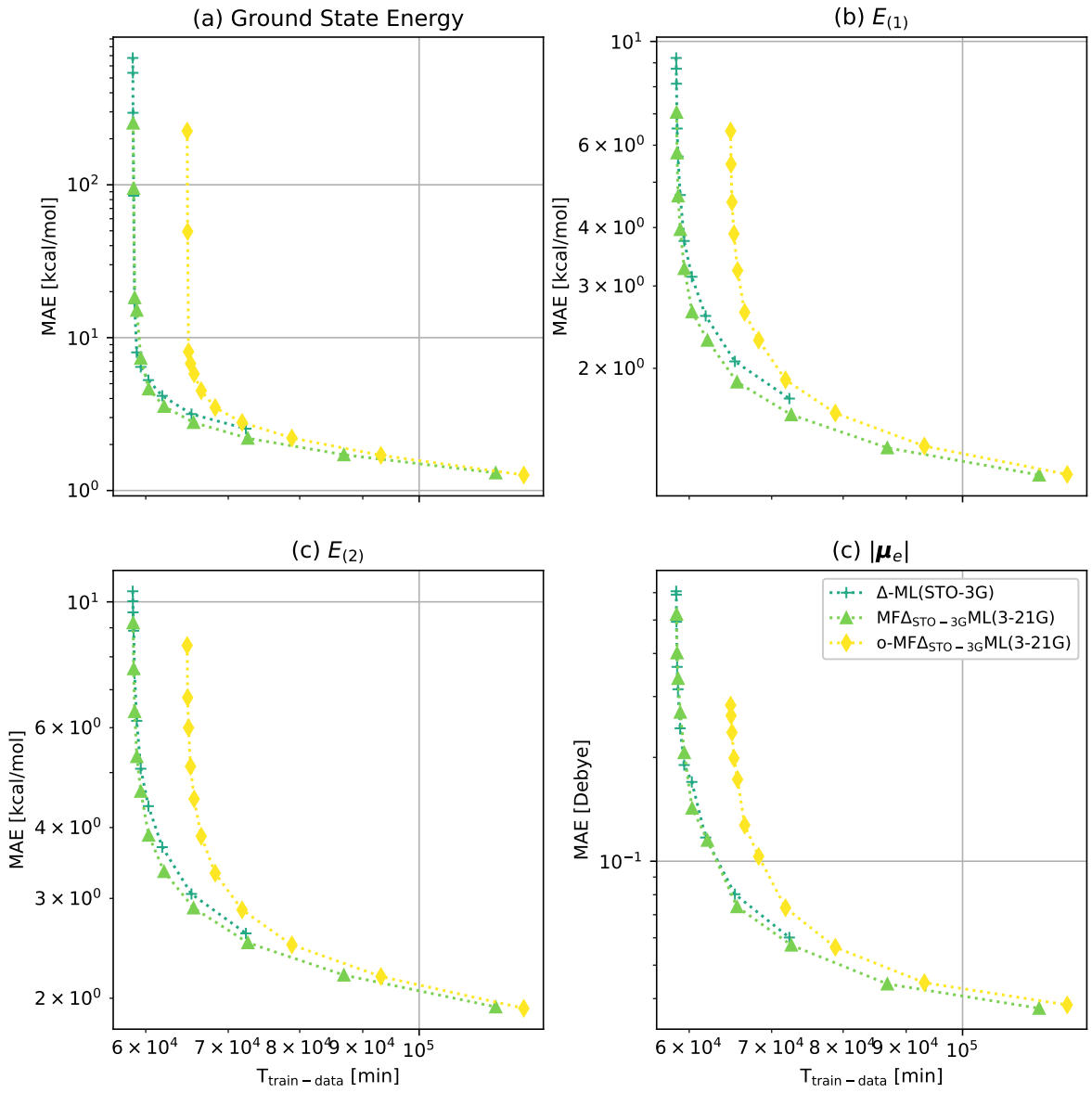


Figure 9.8: Time versus MAE for Δ-ML, MFΔML, and o-MFΔML models in prediction of different QC properties of the QeMFi dataset.

from the QeMFi dataset. However, since the the dataset provides the compute costs for each molecule, this can readily be incorporated into the computation. In short, for the Δ-ML variants, the cost of the model includes the cost of making the calculations of the QC_b for each geometry of the test set. Note that the term 'test set' here is referred not to the small collection of molecules which would be used to assess the ML models before deployment. On the other hand, the large number of samples in this set allow it to be a meaningful proxy of actual use-case scenario where the ML model is used to predict the QC property

of interest several times for several different geometries. Note here, again, that for larger amounts of evaluations of the model, i.e. for larger test sets, the total time for the required QC calculations grows as will be observed in the following parts of this chapter.

The comparison of model MAE and time to generate training data for the diverse models studied in this work are shown in Figure 9.6. The horizontal axis reports $T_{\text{train-data}}$ which is the cost of generating training data for the ML models. For the Δ-ML variants, this also includes the cost of the QC-baseline calculations. Figure 9.6 displays the results for all the models studied in this work, namely,

1. KRR with a single fidelity, in this case, TZVP,
2. MFML and o-MFML with f_b: STO-3G,
3. Δ-ML with QC_b: STO-3G,
4. MFΔML and o-MFΔML with QC_b: STO-3G and f_b: 3-21G.

In the cases of the Δ-ML variants, the QC-baseline is the QC method that is subtracted from all the fidelities which is not to be confused with f_b, which is the *baseline fidelity* for the multifidelity methods. The MAE is reported in appropriate units of the QC property, while the time-cost is reported in minutes. While this figure serves the purpose of overall comparison of methods, each multifidelity method is compared separately in separate figures.

Consider the plot for the MFML curves seen in Figure 9.7. For ground state energies as seen in Figure 9.7(a), with around 10^3 min time-cost, the MFML model results in MAE of ~ 30 kcal/mol. The o-MFML model is shifted along the time-cost axis and results in a similar error for a cost of $\sim 8 \cdot 10^3$ min. This is due to the the additional cost incurred to compute the validation set used in the optimization procedure involved in o-MFML. This offset is more pronounced in the Qc properties studied here since the o-MFML method did not additionally provide any improvement to the multifidelity model, possibly due to the conventional MFML combination already being optimal. In cases, where the o-MFML method does provide an improvement over conventional MFML method, such as that reported in Chapter 6, o-MFML might result in a better MAE versus time-cost trade-off. For example, consider Figure 9.7(b) and Figure 9.7(c) for the excitation energies. Here, although MFML initially shows better efficiency in terms of reaching a certain error for a lower time-cost that o-MFML, close to $5 \cdot 10^4$ min, the o-MFML method results in a lower, albeit marginally, MAE for the same time cost as compared to MFML.

Figure 9.8(a)-(d) shows the time-cost versus MAE for the Δ-ML, and MFΔML variants for the prediction of diverse QC properties. These correspond to the furthest cluster of curves seen in Figure 9.6. Due to the large test set size of 80,000 samples, the cost of the QC calculation of the baseline outweighs the plausible benefit of the Δ-ML and its variants. Since fully trained ML models are generally used for a large number of predictions, it becomes evident that the use of Δ-ML models for such cases becomes costly. Even so, the MFΔML model performs better than the conventional Δ-ML model, although not by a large margin. The cost of generating the QC baseline for the Δ-ML models contributes the most to the overall cost of the models. The MFML (and by extension, o-MFML) method does not incur this cost since the baseline fidelity is predicted as opposed to QC computed. For each QC property, it is seen that the MFΔML model is more efficient than the conventional Δ-ML model. The optimized version, the o-MFΔML model, is less efficient due to the additional cost of the validation set that is incurred.

As one additional assessment, the time taken to merely train the models, that is given a training dataset, return a ML model that can be used for predictions, was also studied for the different ML models. This is reported in the section A.4.4 of Appendix A. The curve of time taken to train the model as a function of the number of training samples used at the TZVP fidelity is shown in Figure A.16. The explicit values of the time taken are reported in Table A.4. For $N_{\text{train}}^{\text{train}} = 2^{11}$, the time taken to train a single fidelity KRR model, and by extension, the standard Δ-ML model was 0.8 seconds. For MFML with f_b:STO-3G, that is total of 5 fidelities, this time was ~ 4300 seconds, and for MFΔML with f_b :3-21G, that is 4 total fidelities, this time was ~ 440 seconds. These time-costs are marginal in comparison to the training data generation cost. Thus, this cost can be neglected as a contributing factor in the efficiency analysis of the MF models.

9.2.3 Large Test Set Sizes

The above time-cost results reveal crucial aspects of using multifidelity methods for QC. When employing the ML-QC pipeline for large test set size, such as in the case of molecular trajectories, the MFML and o-MFML methods supersede the Δ-ML variants. As the test set size increases, at the model accuracy versus cost trade-off for the Δ-ML variants becomes difficult to justify. To really put in perspective the time-cost incurred in Δ-ML, Figure 9.9 reports single fidelity KRR, MFML, and Δ-ML MAE versus time-costs for the different QC properties for a hypothetical test set size of 1 million geometries. The single fidelity and MFML models are unaffected by the size of the test set and therefore do not show any change, the Δ-ML model, however, shifts significantly to the right due to the cost incurred

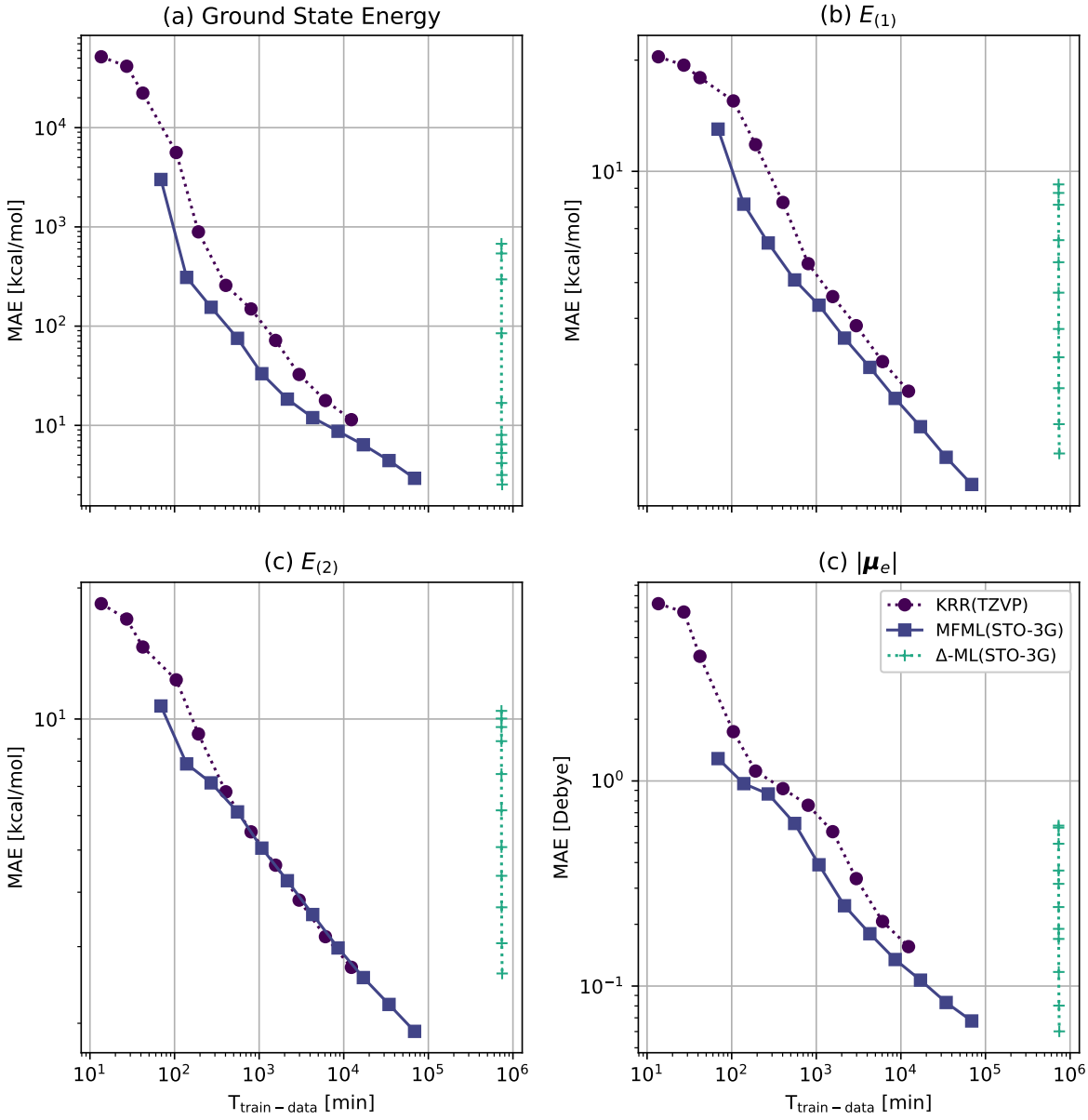


Figure 9.9: Time-cost versus model error for MFML and Δ-ML for a hypothetical test set size of 1 million geometries for the prediction of diverse QC properties.

in the calculation of QC_b. The cost of using the Δ-ML variants becomes a lot more evident here once again delineating the possible benefit of using the MFML method. As for the question of using a cheaper QC-baseline for Δ-ML, the results of Figure 9.3 and Figure A.13 inform that the use of cheaper QC-baselines comes at the cost of model accuracy as is also evident from ref. [29]. The use of MFML is therefore recommended for cases where one is generally interested in predicting QC properties for a very large test set size. However, if the

use-case only demands a small number of predictions, the Δ-ML methods would still be a useful method. In particular, as shown in this work in Figure 9.5 and Figure 9.8, the MFΔML method would then be preferred.

9.3 Conclusion

This chapter set up time-cost benchmarks for the MF models developed in this dissertation for the prediction of several QC properties from the QeMFi dataset. These were assessed alongside the commonly used Δ-ML method and shown to be more efficient. In addition, the MFΔML method was introduced and shown to be more effective than Δ-ML. The MFΔML method could be preferred over MFML in areas of application that require only a small number of model evaluations. However, since such cases are rare in the field, and one generally predicts QC properties for a large collection of molecules or a long trajectory of a molecules, the MFML or o-MFML method are preferred.

CHAPTER 10

INVESTIGATING DATA HIERARCHIES IN MULTIFIDELITY MACHINE LEARNING

This chapter is taken from the work published as ref. [43] in the Journal of Chemical Theory and Computation.

Machine learning (ML) and quantum chemistry (QC) have become increasingly interlinked over the recent times. Both have seen rapid development in tandem allowing for quick prediction of QC properties in place of the costly conventional calculations [156, 181, 22, 234, 30, 235]. This has allowed researchers to perform preliminary examination of complex QC problems with much speed. The ML-QC pipeline first identifies a QC property of interest. Next, a training set is calculated for a desired QC method, say Density Functional Theory (DFT); this is also referred to as a *fidelity*, that is, the level of accuracy of the method with respect to what would be considered ground truth. Once a training dataset is computed, a ML model of choice is trained.

The bottleneck in such a pipeline is often the high cost of generating training data. A ML model can only be as good as the data it is trained on. It is often noted that a larger number of training samples results in a more accurate ML model [22, 154]. This observation implies that either one needs to use a less accurate, and thereby less expensive QC method to train the ML model, or have less training samples at a higher fidelity thereby, resulting in a less accurate ML model, when it comes to the prediction error relative to the data. Several methodological improvements over the single fidelity ML methods for QC have been

proposed to overcome this hurdle in the ML-QC pipeline, including Δ-ML [29], where one trains on the difference between two fidelities. This method has been shown to reduce the number of training samples needed at the expensive fidelity and has since been modified in various flavors including hierarchical ML [139], multifidelity ML (MFML), and optimized MFML (o-MFML). MFML and its variant of o-MFML, systematically combine several Δ-ML like models with more than two fidelities. This method has been shown to be superior in predicting excitation energies along molecular trajectories in Chapter 5. A recent work has also introduced the use of multitask Gaussian processes to harness heterogeneous multifidelity data in order to predict three-body interaction energy in water trimer with coupled cluster accuracy [110]. MFML differs from the conventional Δ-ML method not just in terms of the number of fidelities that are used but also in the number of training samples used at each fidelity. Conventionally, the Δ-ML method uses the same number of samples at both the fidelities. In MFML, these training samples are scaled down as one increases the fidelity of the training data. This further reduced the number of costly training samples needed at the highest fidelity, also called the *target fidelity*. This *scaling factor*, in previous studies was set to be 2, meaning that at each subsequently lower fidelity, the number of training samples would be scaled up by a factor of 2 [32] which was decided based on previous work related to sparse grid combination techniques (SGCT) [37, 36, 35].

The *scaling factor*, the ratio of training samples used at two consecutive fidelities, or levels, directly controls the total number of training samples used for MFML and thereby the cost of generating a training set for the approach. Understanding the effect of this parameter in the efficiency and accuracy of the MFML approach would potentially provide opportunities to further improve the overall multifidelity approach for QC. The *scaling factor*, the ratio of training samples used at two consecutive fidelities, or levels, directly controls the total number of training samples used for MFML and thereby the cost of generating a training set for the approach. Understanding the effect of this parameter in the efficiency and accuracy of the MFML approach would potentially provide opportunities to further improve the overall multifidelity approach for QC. Previously, ref. [163] has studied a two-fidelity MFML model with varying the number of training samples at the cheaper fidelity. By increasing the number of training samples at the lowest fidelity in an additive manner, the model error has been shown to decrease for the prediction of polymer bandgaps. A similar study has been performed in ref. [135] for the study of bandgaps in solids. However, these studies lack any systematic assessment of the scaling factor itself but rather loosely study the effect of training data size within a two-fidelity data structure. This work assesses scaling factors that are different from those used thus far in literature. Several fixed scal-

ing factors, that is, the same scaling factors across the different fidelities are systematically tested. These are evaluated on the recently released benchmarking multifidelity dataset, QeMFi [224, 41], which consists of 135,000 geometries of nine complex molecules. Since the QeMFi dataset also provides the compute time for each fidelity for each molecule type, two time-cost informed scaling factors are also assessed.

Studying model accuracy in relation to the cost of generating the training set for the model also provides a robust measure of how the diverse MFML models behave with respect to the single fidelity models as has been shown in refs. [142, 41]. Therefore, assessment of model accuracy and time-cost of generating corresponding training data is made for the diverse scaling factors. In interest of a complete investigation not only into the scaling factors but also into better understanding the multifidelity data structure, this work further introduces a new error metric for multifidelity methods for QC, namely *error contours* of MFML. Error contours describe the model error with respect to training samples used at two fidelities thereby giving a more comprehensive analysis of the contribution of each fidelity to the overall accuracy of the MFML model. The study of the error contours of MFML indicates that using much lower training samples at the costlier fidelity while increasing the number of training samples at the lowest fidelity results in an MFML model of high accuracy at a much lower cost than the conventional MFML approach with a fixed scaling factor. To systematically assess this, this work studies multifidelity models built with a small number of fixed training samples at the target fidelity and increasing the scaling factor. This gives rise to the notion of the Γ-curve as delineated in secction 4.7. The models that are build in such a manner are shown to be superior to the conventional MFML approach in terms of model error for a given cost of generating the training data.

This chapter assesses scaling factors that are different from those used thus far in literature. Several fixed scaling factors, that is, the same scaling factors across the different fidelities are tested. These are evaluated on the recently released benchmarking multifidelity dataset, QeMFi from Chapter 7, which consists of 135,000 geometries of nine complex molecules. Since the QeMFi dataset also provides the compute time for each fidelity for each molecule type, two time-cost informed scaling factors are also assessed.

The concepts of scaling factors and the tools used in this chapter have already been presented in section 4.4 in addition to the theoretical development of the error contours and Γ-curve in sections 4.6 and 4.7 respectively. However, a small refresher is provided below for the sake of continuity of this chapter. This is followed by the results of MFML and o-MFML models for the prediction of excitation energies with the different scaling factors in section 10.2. In addition to the time-cost of the different MFML models in section 10.2.2, the error

contours of MFML and the Γ-curves are studied in sections 10.2.3 and 10.2.4 respectively. Inferences on these results are made followed by a discussion and outlook of this chapter.

10.1 Methodological Refresher

10.1.1 Scaling factors

This chapter studies two the effect of varying scaling factors, γ, on the overall MFML model accuracy and cost incurred to generate training data. In particular, five values of scaling factors $\gamma \in \{2,3,4,5,6\}$ are studied. Two additional approaches for QC-cost adapted selection of scaling factors are also studied in this chapter. The latter takes into account the compute times for each fidelity before adaptively selecting the ratio of training samples between two consecutive fidelities. The QeMFi dataset, introduced in Chapter 7, provides the QC compute time in seconds for each of the five fidelities when computed on a single core. This information can be used to determine a time-informed scaling factor for each fidelity as opposed to setting a single scaling factor for all the fidelities. The two time-cost informed scaling factors were already introduced in section 4.4 as $\theta^f_{f-1} := \lfloor T^f / T^{f-1} \rceil$ and $\theta^f_F := \lfloor T^F / T^f \rceil$. Here, these are explained in light of the QeMFi dataset that is used.

Since QeMFi is a collection of different molecules, this approach was carried out with reference to the compute times for the largest molecule in the database: o-HBDI. This results in scaling factors $\theta^f_{f-1} = \{3,1,2,1\}$ for increasing fidelity. That is, at SVP, the same number of training samples as TZVP are used while at the 631-G fidelity, it is twice, and so on. However, MFML models are built in such a way that subsequently cheaper fidelities have some more training samples than the previous fidelity so that the difference between the sub-models can be taken. In order to achieve this, after the number of training samples are decided by the scaling factors, if fidelity $f-1$ has the same number of training samples as fidelity f, then one additional sample is added to the sub-model at fidelity $f-1$. As an example, if $N^{\text{TZVP}}_{\text{train}} = 2$, then the training samples for the different fidelities would be $\{12, 4+1, 4, 2+1, 2\}$.

As for the case of θ^f_{f-1}, the reference molecule was chosen to be o-HBDI. This leads to scaling factors $\theta^F_f = \{9,3,2,1\}$ for increasing fidelity. Since the SVP fidelity is scaled by a factors of 1, as discussed earlier, one additional training samples was added each time to maintain the multifidelity structure required for MFML. As an example, consider the case for $N^{\text{TZVP}}_{\text{train}} = 2$. Then the training samples at the different fidelities would be $\{108, 12, 4, 2+1, 2\}$.

10.1.2 Error contours

The learning curve is used as an indicator of the ability of the ML model to predict over unseen data. Learning curves form a major part of the analysis offered in this chapter. In addition to the usual RMAE versus training samples learning curves, this chapter also studies the RMAE versus cost of generating training data for the multifidelity model as first proposed and implemented in Chapter 5 for excitation energies. Error contours of MFML introduced in section 4.6 are also implemented in this chapter. Since the QeMFi dataset from Chapter 7 is used, the error contours of MFML are studied for the following fidelity pairs: TZVP-SVP, SVP-631G, 631G-321G, and 321G-STO3G. Since the error contours are a function of two variables, N_{train}^{f} and N_{train}^{f+1}, they are reported as contour plots. Herein, error contours of MFML are discussed only for $\gamma = 2$.

10.1.3 Γ-curve

As expressed in section 4.7, in this approach, a fixed number of training samples are chosen at the highest fidelity, $N_{\text{train}}^{\text{TZVP}}$. Notice that since the QeMFi dataset is used, the highest fidelity is TZVP. With the fixed number of training samples at TZVP, an o-MFML model is built with $\gamma = 2$. The cost of the training data is noted along with the model error over the holdout test set. For the next step of this curve, instead of varying $N_{\text{train}}^{\text{TZVP}}$, γ is increased by an integer value. This curve is identified as $\Gamma(N_{\text{train}}^{\text{TZVP}})$-curve and is a measure of MAE versus time-cost of training data of the multifidelity model for varying γ.

Example 10.1 (Γ-curve training structure)**.** *If one were to set* $N_{\text{train}}^{\text{TZVP}} = 2$, f_b : *321G, then the* $\Gamma(2)$*-curve would be built with the first multifidelity training data structure (in increasing fidelity) as* $\{2^3 \cdot 2, 2^2 \cdot 2, 2^1 \cdot 2, 2\}$. *The next point would be built with a training data structure of* $\{3^3 \cdot 2, 3^2 \cdot 2, 3^1 \cdot 2, 2\}$ *and so on.*

In general, for a $\Gamma(N_{\text{train}}^{\text{TZVP}})$-curve, the training data structure for $f : b$ 321G is given as $\{\gamma^3 \cdot N_{\text{train}}^{\text{TZVP}}, \gamma^2 \cdot N_{\text{train}}^{\text{TZVP}}, \gamma^1 \cdot N_{\text{train}}^{\text{TZVP}}\}$. For reasons explained in section 10.2.2 and section 10.2.3, the STO3G baseline is not considered. In this chapter, the $\Gamma(\cdot)$-curve is studied for $N_{\text{train}}^{\text{TZVP}} \in \{2, 8, 64\}$ to further assess the multifidelity structure of training data and evaluate the limits of the multifidelity approach. Since there is no trivial way to express the number of training samples used at a certain fidelity and relate it to the MAE of the model, the $\Gamma(\cdot)$-curve is plotted only as MAE versus multifidelity training data generation cost.

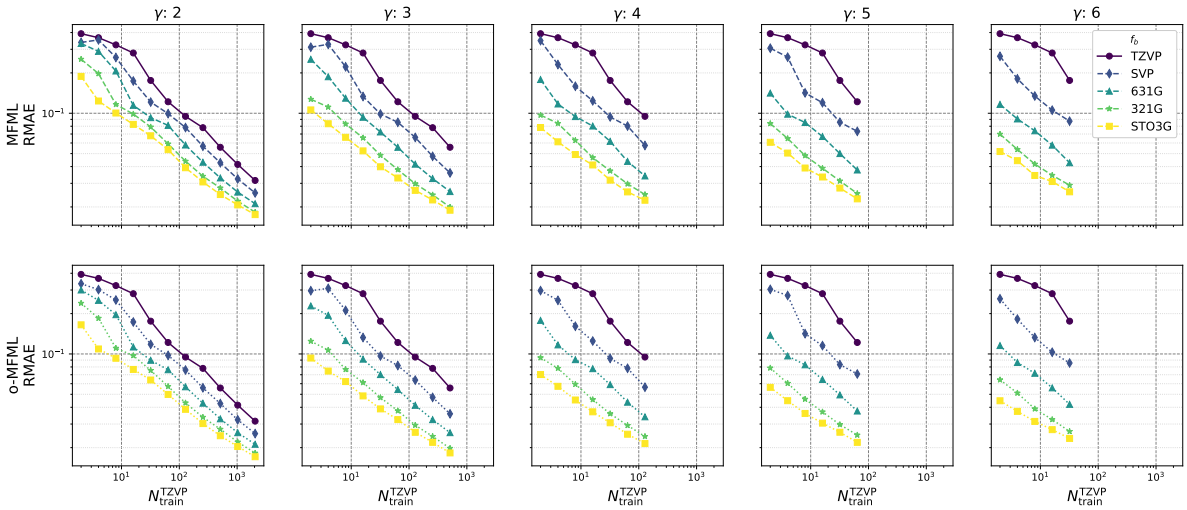


Figure 10.1: Multifidelity learning curves for the prediction of excitation energies taken from the QeMFi dataset. The top row corresponds to the MFML models while the bottom row is for the o-MFML models. Different fixed scaling factors are used to scale the data across each fidelity in the multifidelity models as explained section 4.4. The scaling factors are reported on the top of each column.

10.2 Results

This section presents the analysis of varying the scaling factor for MFML and o-MFML. The results are presented in two major formats. First, standard learning curves of MAE versus number of training samples used at the highest fidelity of TZVP are presented. Following this, the model error is assessed a function of the time-cost of generating the training data for the model. This assessment informs of the effectiveness of the diverse models that are studied in this chapter. Once these results are interpreted, error contours of MFML as described in section 4.6 are studied.

10.2.1 Learning curves

The primary assessment of the effect of different scaling factors is carried out using learning curves for the resulting MFML and o-MFML models. These learning curves are shown in Figure 10.1 for different scaling factors. In all cases, the scaling factors are constant across the different fidelities as explained in section 4.4. The top row of the figure depicts the learning curves for MFML while the bottom row corresponds to the o-MFML models. The learning curves are shown for varying baseline fidelities. A single fidelity KRR learning curve is also shown for reference. The RMAEs are reported as unitless quantities.

In Figure 10.1, the first column shows the learning curves for the scaling factor of 2.

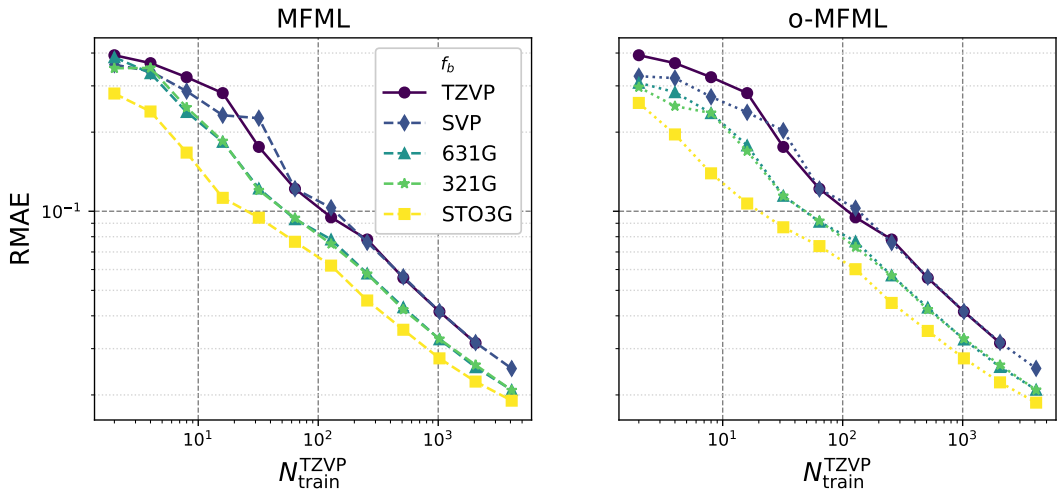
This is the original scaling factor used in ref. [32] and the preceding chapters of this dissertation. This scaling factor is used as a reference to evaluate the other scaling factors against. Within these reference results, one observes that the addition of cheaper baselines results in a constantly lowered offset of the learning curves. The interpretation from the lowered offsets is that similar models errors can be achieved with lower number of training samples at the target fidelity with the addition of cheaper fidelities. With the cheapest fidelity, STO3G, being added to the multifidelity model, one observes an one observes RMAE of 0.4 with around 200 training samples at TZVP.

In comparison to this, an increase of of the scaling factor, γ, provides MFML models that achieve similar errors for lower number of training samples at TZVP. A comparison of the MFML models built with different values of γ for the STO3G baseline fidelity are shown in Figure 10.3. For example with a scaling factor of 3, the STO3G baseline MFML model achieves an RMAE of 0.4 with $N_{\text{train}}^{\text{TZVP}} = 32$. With a scaling factor of 6, the number of training samples at TZVP needed to achieve this same error is lowered further to about 4. The learning curves for o-MFML also indicate the same across varying scaling factors. There is little difference between the learning curves for MFML and o-MFML. This could be due to the MFML combination being already optimal for this multifidelity data structure as has been argued in Chapter 6. There is however, slight improvement in all cases of o-MFML and it does result in reduced RMAEs across the different scaling factors.

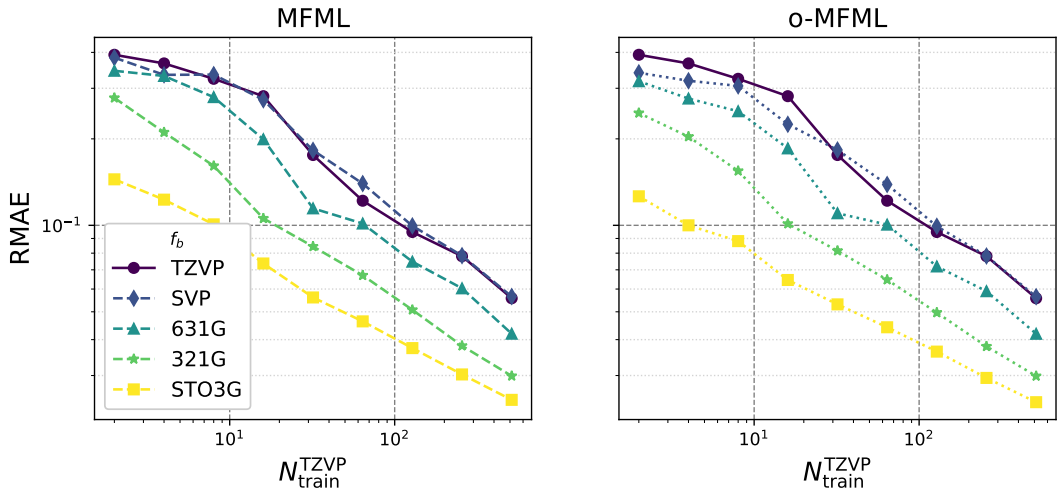
From these results it appears that a higher number of training samples with cheaper fidelities improves the predictive capabilities of the MFML and o-MFML models. One possible reason for this could be that the use of larger data at the cheaper fidelities results in more information about the overall multifidelity structure being included into the MFML models.

In addition to the fixed scaling factors across fidelities, two special cases of scaling factors were introduced in section 4.4 based on the QC compute time of each fidelity. These were denoted by θ_{f-1}^{f} and θ_{f}^{F} as explained in section 4.4 in detail. Learning curves were generated for both MFML and o-MFML models for both these cases. The results are shown in Figure 10.2 for both scaling factor cases with various baseline fidelities. The single fidelity KRR learning curves is also depicted for reference.

Figure 10.2(a) depicts the results for θ_{f-1}^{f} with the left pane for MFML and right pane for o-MFML. As explained in section 4.4, these scaling factors are based on the ratio of QC-compute times of subsequent fidelities. Between some fidelities - namely between TZVP and SVP, and between 631G and 321G - this scaling was observed to be 1. It is anticipated that these fidelities will not significantly improve the MFML models since there is very little



((a)) Scaling factors, θ_{f-1}^{f}, between fidelities chosen as ratios of the QC compute time of subsequent fidelities. Single fidelity KRR at TZVP is also shown for reference.



((b)) Scaling factors, θ_{f}^{F}, between fidelities selected as ratios of the QC compute time of that fidelity to the compute time of TZVP, that is the target fidelity.

Figure 10.2: MFML and o-MFML learning curves time informed scaling factors (see section 4.4). The two time informed scaling factors are described in detail in section 4.4. Single fidelity KRR learning curves are also provided for reference. The legend describes the baseline fidelity, f_b, of the multifidelity model.

additional information that is being added to the model. Indeed, as seen in Figure 10.2(a), the multifidelity model built with SVP baseline does not provide any improvement over the single fidelity KRR. This is due to the fact that the number of training samples at both fidelities are nearly identical, only different by 1 sample, due to the scaling factor. This same observation can be made for the learning curves with 321G as baseline fidelity. With 631G and

STO3G baselines, however, one observes improvement of the MFML and o-MFML models. With the STO3G baseline, MFML and o-MFML reach an RMAE of 0.03 with roughly 500 training samples at TZVP.

Similarly, Figure 10.2(b) reports the learning curves for θ_f^F. The left-pane shows the results for MFML, while the right pane shows those for o-MFML. Similarly, Figure 10.2(b) reports the learning curves for θ_f^F. The left-pane shows the results for MFML, while the right pane shows those for o-MFML. The SVP baseline fidelity once again shows very little improvement over the single fidelity KRR due to the scaling factor being unity (see section 4.4). However, each additional cheaper baseline fidelity, results in lowered offsets of the corresponding learning curves. With $N_{\text{train}}^{\text{TZVP}} = 256$, the multifidelity models with the STO3G baseline result in RMAE of ~0.03.

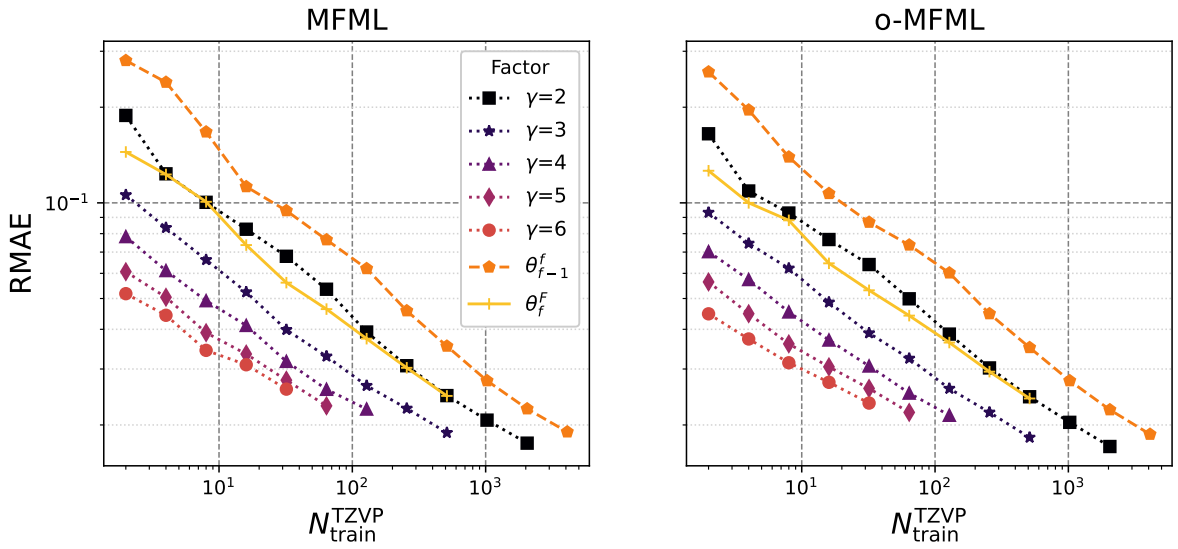


Figure 10.3: Comparison of learning curves for fixed scaling factors γ, θ_{f-1}^f, and θ_f^F with f_b: STO3G. The x-axis reports the number of training samples used at the highest fidelity, that is, TZVP. Both MFML and o-MFML models are compared. Increasing values of γ result in a constant lowered offset of the learning curves. The cost informed scaling factors show a higher value of MAE.

To aid comparison of the different scaling factors discussed so far, Figure 10.3 depicts the learning curves for the MFML and o-MFML models built with the STO3G fidelity as baseline. The various factors are delineated in the legend of the plot. This plots shows that increasing values of γ result in a lowered constant offset of the learning curves. In contrast, the multifidelity models built with time-informed scaling factors, θ_{f-1}^f and θ_f^F both show the highest model error. This observation is consistent for both MFML and o-MFML models as can be seen from the two plots shown in Figure 10.3. Furthermore, the o-MFML

Factor	MFML	o-MFML
2	0.0790	0.0754
3	0.0486	0.0472
4	0.0371	0.0297
5	0.0312	0.0264
6	0.0290	0.0264
θ^f_{f-1}	0.1201	0.1143
θ^F_f	0.0844	0.0854

Table 10.1: RMAE rounded off to 4 decimal points for MFML and o-MFML models built with the STO3G baseline fidelity for $N^{\text{TZVP}}_{\text{train}} = 2^5$. This allows for a uniform comparison of the model accuracy not just between MFML and o-MFML but also across the scaling factors that are studied in this work. Notice that the learning curve for $\gamma = 6$ only goes up to $N^{\text{TZVP}}_{\text{train}} = 2^5$ and therefore this is chosen as a comparison point for all other curves.

models show lower errors than the MFML counterparts for all the cases as seen in table 10.1 which reports the RMAEs for the MFML and o-MFML models with various scaling factors for the STO3G baseline for $N^{\text{TZVP}}_{\text{train}} = 2^5$ for ready reference. This training set size is chosen so that there is uniform comparison between the different scaling factors. The behavior of the MFML models with increasing γ is in some sense expected since an increasing value of the scaling factor implies an increased amount of training data, albeit only at the cheaper fidelities. This could be one potential reason to explain the lowered offsets that are observed. An increased amount of training samples at the lowered fidelities, due to the nested structure of the multifidelity training data, could impart meaningful information about the conformational phase-space and its relation to the excitation energies. The limited improvement that is seen from the learning curves of θ^f_{f-1} and θ^F_f, which both had a much larger number of training samples at the cheapest fidelity in comparison to the other fidelities, could be due to the lack of sufficient training data in the fidelities that lie between the baseline and target fidelities. Furthermore, the value of θ^f_{f-1} for the SVP and 321G fidelities was 1 which did not provide any additional information to the MFML model as was pointed out in the discussion for Figure 10.2(a). This in turn affects the overall model that is built with the STO3G baseline fidelity. A similar argument can be made for why the MFML model with θ^F_f has limitations. Regardless, the MFML model built with θ^F_f does in fact achieve model RMAE that are comparable to the MFML model built with $\gamma = 2$.

The results for fixed scaling factors, γ, indicate that a higher γ results in a lower model error, or, smaller number of training samples at the costly target fidelity are needed. For the time-informed scaling factors, it was seen that these do not perform as well as was anticipated. However, one must be cautious about the results from Figure 10.1 and Figure

10.2 before considering them to be improvements over the conventional MFML method with $\gamma = 2$. Since one uses much more training samples at the cheaper fidelities as one increases the value of γ, the cost of generating training data needs to be assessed to better understand the cost-accuracy trade-off in these multifidelity models. In interest of such an analysis, the time-cost of generating training data versus model MAE are discussed in the next section. Although the MFML and o-MFML models show similar RMAEs, only o-MFML models are discussed hereon. This is due to the observation of Chapter 6 that the o-MFML method provides a superior model even in cases of poor data distribution of the cheaper fidelities. Since all o-MFML models use the same validation set, the cost of generating the validation set is not included in the time-analysis plots.

10.2.2 Time benefit analysis

As presented in the preceding chapters, a good assessment of multifidelity methods is the study of model error versus the time to generate the training samples for the model. In interest of such a study, the RMAE versus training data generation time are studied for the just discussed test cases. The time to generate data for a multifidelity model is the sum over the times for generation of all the training samples used at all fidelities that form the multifidelity model. That is, $T^{MFML}_{train-data} := \sum_{f_b \leq f \leq F} N^f_{\text{train}} \cdot T^f_{QC}$ where N^f_{train} is the number of training samples used at some fidelity f, and T^f_{QC} is the corresponding single-point QC-compute time for that fidelity. The QC-compute times recorded in the QeMFi dataset are those for a single-core computation and are provided for each fidelity for each molecule type as discussed in Chapter 7.

The RMAE versus $T^{MFML}_{train-data}$ plots for the various scaling factors are shown in Figure 10.4 for o-MFML. Only o-MFML is shown since it has a lower error compared to MFML for all the cases (see Figures 10.1,10.2(a), and 10.2(b)). The RMAE and the time axes are both presented in log-scaled values. The axes of the plots are scaled identically for easy comparison among the different scaling factors. The bottom-right corner plot compares the time-cost based scaling factors to the case of $\gamma = 2$ for the MFML model built with f_b: 321G and not for the cheapest STO3G baseline for reasons discussed below.

For the different cases of scaling factors shown in Figure 10.4, it can be seen that the addition of cheaper baselines helps achieve a specific model accuracy with less time cost to generate the training data. In general, fixing a specific MAE, one can see that the curves of the cheaper baseline achieve this error earlier with respect to the time axis. Alternatively, if one were to set a time budget and draw a vertical line at that value (as on the x-axis), then the cheaper baseline models result in lower RMAEs than the single fidelity KRR model. The

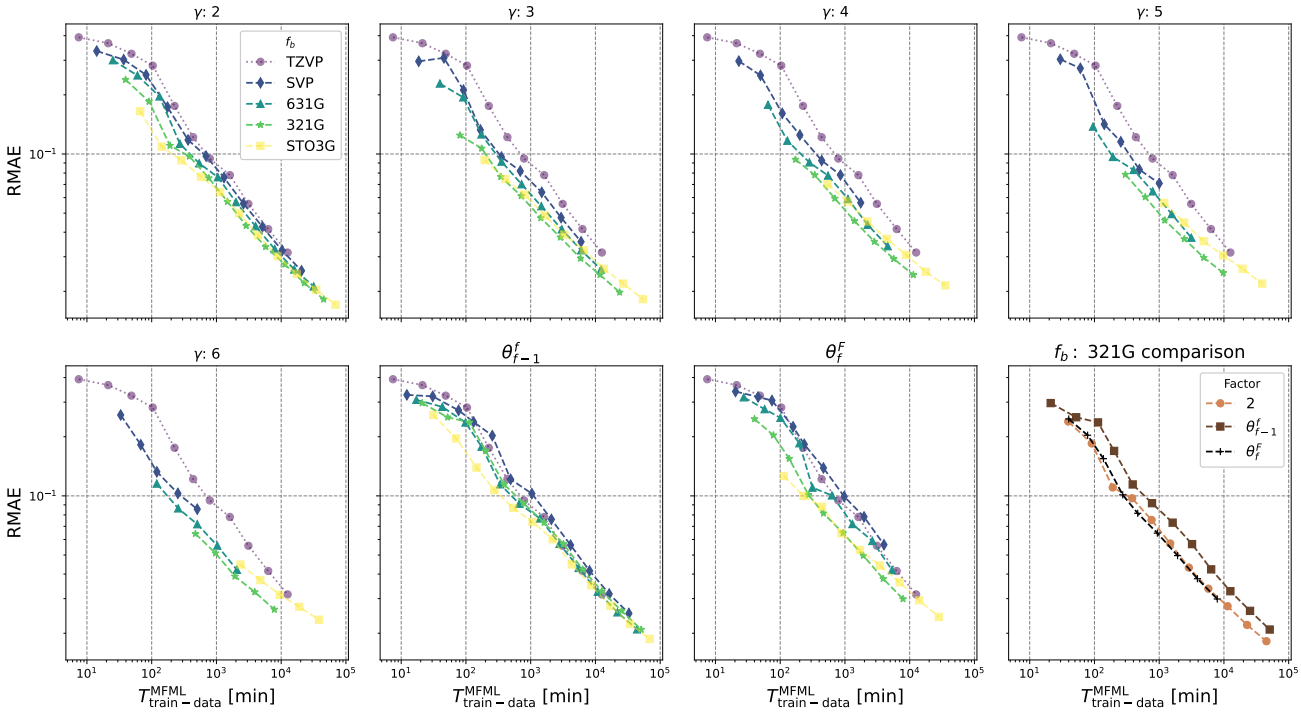


Figure 10.4: Time to generate training data versus RMAE of the corresponding o-MFML model for the diverse scaling factors studied. The different scaling factors used are denoted as sub-titles. The RMAE is unitless while the time-cost is in minutes. The single fidelity KRR case is also depicted for reference. As one increases the scaling factors across the fidelities, one observes that the learning curves of the MFML models shifts further due to the larger amount of training samples used. The two cases of θ^f_{f-1} and θ^F_f are explained in Scaling Factors. The bottom-right corner plot compares the o-MFML curves for the 321G baseline for the two time-informed scaling factors and the case of $\gamma = 2$.

case of STO3G baseline is an exception. For all scaling factors, the addition of the STO3G baseline does not provide significant improvement of the model. In fact, it increases the training data generation cost. The STO3G energies do not provide major improvement to the o-MFML model over the 321G baseline. This could be due to poor data distribution that has previously been noted for the STO3G fidelity for excitation energies of molecules (see Chapter 5). The time-cost versus RMAE plots make this evident. Although the analysis of conventional learning curves from section 10.2.1 indicated that the STO3G baseline fidelity improved the MFML model, these time-cost plots indicate that this comes at a cost which supersedes the RMAE improvement that is observed. However, consider the case of the special scaling factors θ^F_f, which are decided by the ratio of the QC-compute times of a fidelity f to the QC-compute time of the target fidelity F. For some portions of learning curve for f_b = STO3G, o-MFML does provide lower errors as is expected from such multifidelity

models This could indicate that the use of o-MFML could improve the model accuracy even for the cases of poor data distribution as seen in the STO3G fidelity. For the o-MFML models that are built with the 32G baseline fidelity, the time benefit of the multifidelity approach becomes all the more perceptible across the various scaling factors. For instance, in the case of $\gamma = 3$, the o-MFML model results in an RMAE of 0.1 with a time cost of ~ 200 minutes. The KRR model achieves a similar error with a time-cost of $\sim 1,000$ minutes. This indicates a time benefit of about 5 times with this baseline fidelity for $\gamma = 3$. Similarly, for $\gamma = 5$, the KRR model achieves an RMAE of 0.07 with a time cost of $\sim 2,000$ minutes while the o-MFML model achieves a similar error for a time cost of ~ 300 minutes resulting in a time benefit of about 6 times. Similar observations can be made for the other values of γ. The time benefit is less pronounced for the cases of θ^f_{f-1} and θ^F_f but is still present for f_b : 321G.

While each scaling factor does improve the time-cost needed to achieve a certain RMAE vis-á-vis the single fidelity KRR, it is also important to see which scaling factor performs better with respect to the others for a given baseline fidelity. The bottom-right plot of Figure 10.4 compares the time-cost versus RMAE curves of MFML models for $\gamma = 2$, θ^f_{f-1}, and θ^F_f for the baseline fidelity of 321G. The STO3G baseline is not considered due to its poor distribution. These specific scaling factors are chosen to better understand the standing of the time-informed scaling factors with respect to the fixed scaling factors. The time-cost versus RMAE of different γ are compared in Figure 10.6 in light of the discussion about the $\gamma(\cdot)$-curves. This comparison in Figure 10.4 for the 321G fidelity shows that the fixed scaling factor of $\gamma = 2$ performs better than both the time-cost informed scaling factors (see section 4.4). The MFML model built with θ^F_f does perform only as well as that built with $\gamma = 2$, which is the default set-up for MFML. This could indicate that just the QC-compute time-cost information might not suffice to select the training samples at each fidelity. It could be that the model accuracy and multifidelity training structure relation is more complex than just accounting for the QC-compute cost. To better understand how each fidelity and the number of training samples at each fidelity contribute to the overall model error, the next section studies a new error metric, error contours (see section 4.6 for details). This is intended to give a better view into the inner mechanisms of the multifidelity data structure in building a MFML model.

10.2.3 Multifidelity error contours

The time versus RMAE results for different scaling factors hint that one might not necessarily need many training samples at the target fidelity. This would imply that one could build

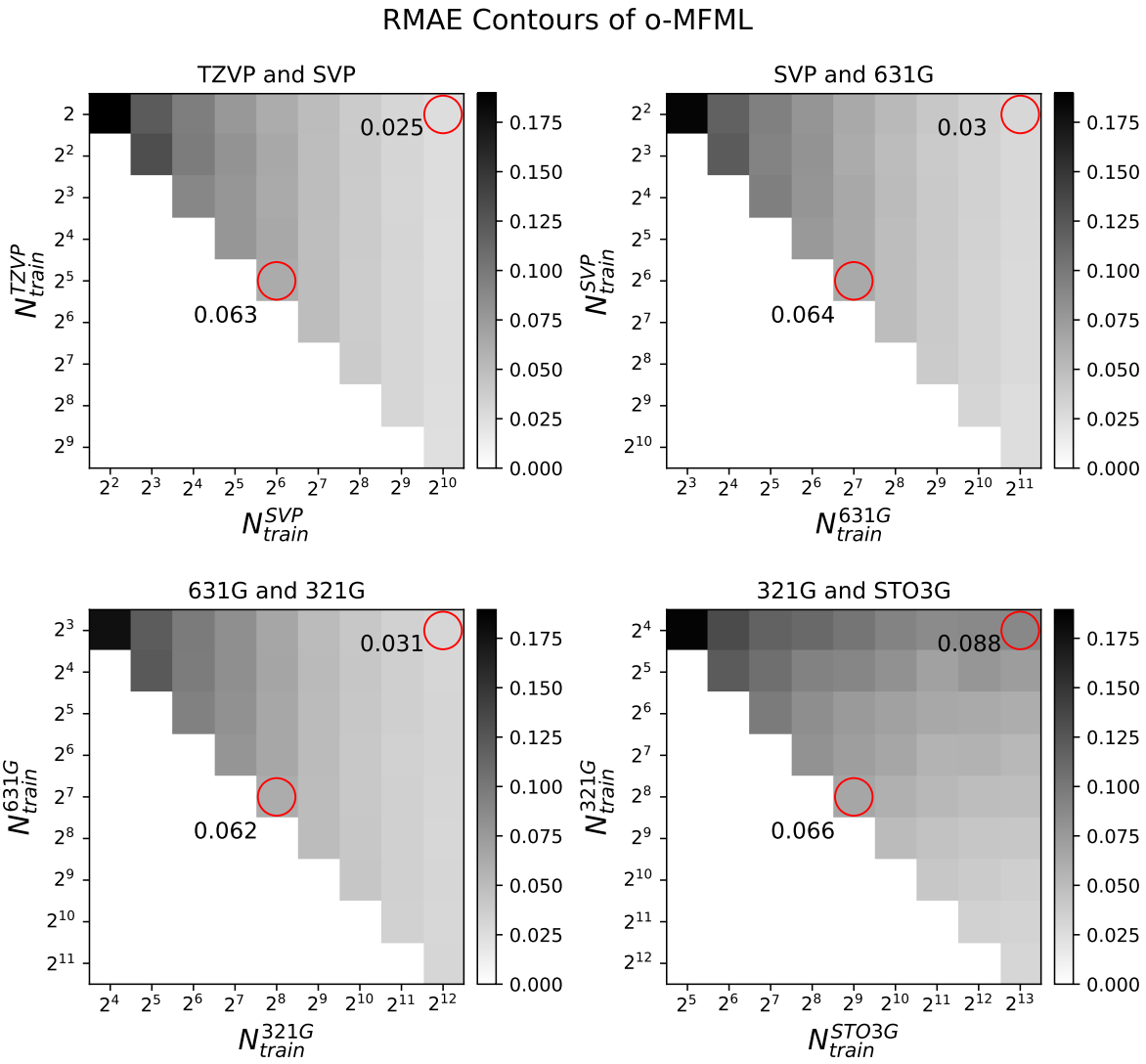


Figure 10.5: Excitation energy prediction error contours with o-MFML for different training samples at different fidelities. The details of the method are explained in section 4.4 for each case. In each plot, the vertical axis depicts the number of training samples used at the costlier fidelity, f, while horizontal axis reports the training samples used at the cheaper fidelity $f-1$. The resulting error for the o-MFML model with the specific choice of training samples used at fidelity f and $f-1$ are depicted as the error contours. Here, the RMAE are depicted as contour plots for different training samples spanned across two fidelities. Two specific RMAEs are enumerated for all 4 cases: first, that for the smallest training set size at the higher fidelity, f, and the largest training set size at $f-1$; second, for the case where the training sample at f and $f-1$ have the scaling factor of 2.

a cheap multifidelity model with a large number of training samples at the cheaper fidelities and then 'raise' it to the target fidelity with an exceptionally small number of training samples at the target fidelity. This can be further studied with the error contours of multifidelity. These contours involve studying the model prediction error by varying the training sizes along fidelity f and $f-1$ for all $f \leq F$. As was discussed in section 4.4, this is performed for consecutive fidelity pairs TZVP-SVP, SVP-631G, 631G-321G, and 321G-STO3G.

Figure 10.5 illustrates the multifidelity error contours of o-MFML for different fidelity pairs for $\gamma = 2$. Consider the top-left plot corresponding to the TZVP-SVP fidelity pair. The y-axis denotes the number of training samples used at TZVP, while the x-axis depicts the number of training samples used at the SVP fidelity. The colors of the plot itself correspond to the MAE. In the usual o-MFML approach, the number of training samples used at SVP with respect to the number of training samples used at TZVP would be scaled by the factor γ (in this case by 2). However, for this set-up there is no trivial scaling of training data that is carried out. Instead, the multifidelity model is built with a specific selection of training samples. For example, take the case for $N_{\text{train}}^{\text{TZVP}} = 2$ and $N_{\text{train}}^{\text{SVP}} = 2^{10}$ which is marked by the top-corner red circle. The RMAE reported here, 0.025, is for a multifidelity model that is built with the following multifidelity training structure (with increasing fidelity): $\{2^3 \cdot 2^{10}, 2^2 \cdot 2^{10}, 2 \cdot 2^{10}, 2^{10}, 2\}$. In other words, the scaling factor is only applied for the fidelities that are not studied as part of the error contour. In contrast, in the usual o-MFML the training data structure would be $\{2^4 \cdot 2, 2^3 \cdot 2, 2^2 \cdot 2, 2 \cdot 2, 2\}$, this is the block that corresponds to $(N_{\text{train}}^{\text{SVP}} = 2^2, N_{\text{train}}^{\text{TZVP}} = 2)$ on the plot. The accompanying color-bar depicts that this regular o-MFML model results in a higher RMAE than 0.025. In general, the diagonal of the contour plot depicts the regular o-MFML model which is identified in the learning curves of Figure 10.1 for $\gamma = 2$. The RMAE for $(N_{\text{train}}^{\text{SVP}} = 2^6, N_{\text{train}}^{\text{TZVP}} = 2^5)$ is highlighted as well reporting an RMAE of about 0.063 which is over twice of what is observed for $(N_{\text{train}}^{\text{SVP}} = 2^{10}, N_{\text{train}}^{\text{TZVP}} = 2)$. This is a remarkable observation in that simply using two training samples at TZVP while increasing the training size at the lower fidelities results in a model that is more than twice as accurate. Furthermore, the RMAE for the block $(N_{\text{train}}^{\text{SVP}} = 2^{10}, N_{\text{train}}^{\text{TZVP}} = 2)$ is similar to the model block $(N_{\text{train}}^{\text{SVP}} = 2^{10}, N_{\text{train}}^{\text{TZVP}} = 2^9)$. In general, it is seen that a lower number of TZVP training samples with a larger training set size at the cheaper fidelities results in more accurate multifidelity model.

Similar observations and inferences can be made for the error contour for the SVP-631G fidelity pair as seen on the top-right plot of Figure 10.5. In this set-up, consider the top right corner which is marked with a circle. This is identified as $(N_{\text{train}}^{631G} = 2^{11}, N_{\text{train}}^{\text{SVP}} = 2^2)$ and has the following multifidelity data structure (with increasing fidelity): $\{2^{13}, 2^{12}, 2^{11}, 2^2, 2\}$. As in

the previous case, the training data scaling is only applied to the fidelities that are not studied as part of the error contour. This mode reports 0.030 as the RMAE. Once again, the diagonal of the contour plot corresponds to the regular o-MFML model. Consider then, the block identified by ($N_{\text{train}}^{631G} = 2^6, N_{\text{train}}^{\text{SVP}} = 2^7$) which has a training data structure (in increasing order of fidelity): $\{2^9, 2^8, 2^7, 2^6, 2^5\}$. This regular o-MFML model reports an RMAE 0.064, over twice as much as for the previous one. The overall contour plot reveals that the use of very few training samples at SVP paired with a larger number of training samples at the lower fidelities results in RMAEs that are comparable to the cases where one would use a lot more training samples at the SVP fidelity. In particular, this form of *flattening* out the multifidelity training structure by using few training samples at the top fidelities and increasing the training samples at the cheaper fidelities, outperforms the regular o-MFML model (which are the diagonal blocks of the error contour).

Similar observations are made for the 631G-321G and 321G-STO3G pairs of fidelities which are seen in the bottom row of Figure 10.5. It is interesting to note, however, that the 321G-STO3G error contours do not follow the same trend as the others. Using very little 321G training samples and increasing the training samples at the STO3G fidelity does not result in lower RMAE as seen from the top-corner red marker error being 0.088 while the center marker reporting RMAE of 0.066. This is once again explained by the poor data distribution that has previously been reported for the STO3G fidelity in Chapter 5, Chapter 6, and Chapter 8.

The error contours for the multifidelity model hint at an interesting mechanism in the MFML approach. Based on the behavior of the model error as discussed above, it appears that one does not necessarily need to use many training samples at the higher fidelities, in particular at the target fidelity. This is indeed something that has been previously been hinted at in ref. [139] using the optimization procedure for h-ML albeit with a larger number of training samples at the target fidelity. However, a thorough investigation of the multifidelity structure such as that performed in Figure 10.5 reveals that not only can a multifidelity model be built with low number of training samples at the costlier fidelities, but that this number is far smaller than what would be anticipated in the general MFML and similar methods. The error contours indicate that there is still a great deal of information available at the cheaper fidelities which only need to be 'raised up' to the target fidelity with a surprisingly small number of training samples. With such an understanding of the multifidelity training structure, one can begin to think of ways to select training samples at the different fidelities that need not necessarily follow the concept of a scaling factor between the fidelities. Furthermore, the results of varying γ from Figure 10.4 hint at a possible

approach which is pursued in the following section.

10.2.4 Γ-curves

The contour plots of Figure 10.5 provide an interesting observation about MFML. One can potentially build cheaper multifidelity models by limiting the training samples used at the expensive fidelities and then proceeding to add cheap fidelity data to the multifidelity model.

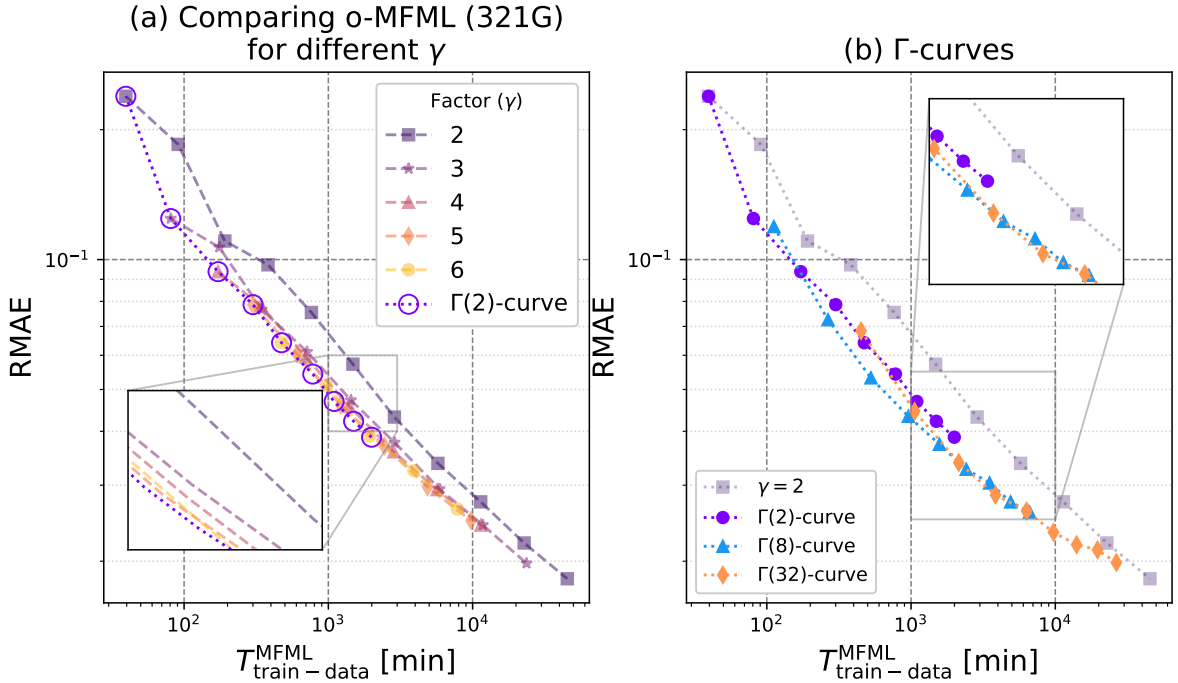


Figure 10.6: (a) Time to generate training data and corresponding o-MFML model error as RMAE for constant scaling factors, γ used in this study. An inset between 1,500-3,000 minutes is provided for the comparison of the curves for all γ studied in this work to readily compare in regions that are too crowded to be observed in the main plot. (b) RMAE versus time-cost for different $\Gamma(N^{\mathrm{TZVP}}_{\mathrm{train}})$-curves. Increasing the number of training samples at TZVP improves the model accuracies along the $\Gamma(\cdot)$-curves with a saturation observed towards the end of each curve.

Consider first the left-hand side plot of Figure 10.6 which shows the RMAE curves of o-MFML with f_b : 321G for the different γ studied in this work for comparison. An inset plot is provided which zooms into the region between 1,000-3,000 minutes to show the different curves clearly. In addition, a new curve as introduced in 4.7, the $\Gamma(2)$-curve is depicted in the plot. The $\Gamma(2)$-curve is essentially the case where the number of training samples at TZVP are constrained to 2 but the remaining multifidelity data structure is allowed to grow

as per the scaling factor γ. That is, the $\Gamma(2)$-curve is built with the first point of the curves for the different γ values. In some sense, this translates into it being a learning curve not as a function of $N_{\text{train}}^{\text{TZVP}}$ but rather of γ. From Figure 10.6, it becomes evident that even for as little as 2 training samples at the highest fidelity, if one adds cheaper data to the multifidelity model - which corresponds to increasing the value of γ without increasing $N_{\text{train}}^{\text{TZVP}}$ - the error of the o-MFML model decreases. For the same time-cost of a conventional o-MFML model built with $\gamma = 2$, if one were to chose the models along the $\Gamma(2)$-curve, a lower RMAE can be achieved. The $\Gamma(2)$-curve in Figure 10.6 shows data points for up to $\gamma = 10$ where the multifidelity training data structure (for increasing fidelity) is: $\{20000, 2000, 200, 20, 2\}$. In the inset of the plot, one observes that the $\Gamma(2)$-curve results in errors that are lower than the o-MFML learning curves for fixed γ. However, the $\Gamma(2)$-curve converges to the o-MFML model built with $\gamma = 6$. One potential reason for this could be the saturation of the multifidelity model built along the $\Gamma(2)$-curve. Due to a very large number of training samples at the cheaper fidelities (for instance, $2 \cdot 10^4$ at 321G for the last point on the $\Gamma(2)$-curve), the model is no longer able to clearly learn the correction between SVP and TZVP.

The right-hand plot of Figure 10.6 further investigates this saturation by comparing the $\Gamma(N_{\text{train}}^{\text{TZVP}})$-curves for $N_{\text{train}}^{\text{TZVP}} \in \{2, 8, 32\}$. The y-axis reports the RMAE while the x-axis denotes the time-cost in minutes to generate the training data used in the multifidelity models. An inset is provided for the interval between $10^3 - 10^4$ minutes for a better view of the $\Gamma(\cdot)$curves. The o-MFML learning curve for $\gamma = 2$ is provided for reference.

Since the model errors throughout this work were reported in unitless RMAE, in order to understand how this translates to actual energy prediction, predictions are made for the holdout test set. For this purpose, the multifidelity model corresponding to $\Gamma(32)$ is used with $\gamma = 10$. Using this model, the prediction of the first vertical excitation energies is made on the holdout test set. The absolute error values are then computed as $|y_{\text{ref}} - y_{\text{pred}}|$. The resulting values are reported in Table 10.2. This includes the mean of the absolute error, minimum absolute error, maximum absolute error, and the standard deviation of the absolute error. For all molecules, it can be seen that the model error is nearly identical indicating that the final multifidelity model built with the Γ-curve is not affected by the difference of the molecule. This is of course due to the fact that the training data consists of these molecules. In all cases, the model reports a mean absolute error close to 1 kcal/mol.

10.2.5 Transferability Assessment

Transferability in ML refers to the concept of ML models being trained on specific type of dataset and having it predict the QC property for an out of sample dataset. For example,

Molecule	mean	min	max	Std. Deviation
urea	1.0178	0.9940	1.0264	0.0035
acrolein	1.0145	0.9928	1.0220	0.0034
alanine	1.0145	0.9928	1.0222	0.0035
SMA	1.0147	0.9928	1.0224	0.0035
2-nitrophenol	1.0147	0.9928	1.0224	0.0035
urocanic	1.0146	0.9928	1.0224	0.0035
DMABN	1.0145	0.9928	1.0220	0.0034
thymine	1.0145	0.9928	1.0224	0.0035
o-HBDI	1.0146	0.9928	1.0224	0.0035

Table 10.2: Absolute difference in prediction and reference of excitation energies of molecules in QeMFi using $\Gamma(32)$ with $\gamma = 10$. The mean, range, and standard deviation of the absolute differences are also listed.

one could train ML models on the QeMFi dataset for excitation energies and use these trained ML models to predict the excitation energies of molecules that do not belong to the 9 molecules of the QeMFi dataset. The transferability of ML models in QC has long been a challenging issue [236, 22, 237]. In general, since ML is a statistical method, transferability is restricted by the type of data the model is trained on. That is, if a model is trained only on, say, benzene configurations, it is not expected to perform well in predicting for methanol or acrolein. Not only so, the type of molecular representation that is used in the model makes a major difference to the overall transferability of the model [22, 237]. However, in this subsection, the robustness of the Γ-curve method is investigated for transferability in order to complete the discussion on the development of this method.

The QUESTDB database is a collection of several small molecules for which high accuracy excitation energies are available [238, 239]. The energies for these molecules are computed with mostly CC levels of theory. In order to properly assess the transferability, which is challenging as is, the excitation energies of 90 molecules from QUESTDB were computed with the DFT method using CAMB3LYP functional with the def2-TZVP basis set in the exact same manner as was done for the QeMFi dataset. This in principle curtails artifacts arising by virtue of comparing two different QC fidelities as reference and ML predicted values of excitation energies.

The geometries from the QUESTDB database have an additional challenge in testing for transferability. This is the issue of index invariance in generating the unsorted CM molecular descriptor. Since the geometries of the molecules from QeMFi are arranged in such a way to ensure index permutation invariance, one can use unsorted CM to train and evaluate ML models. However, this is not the case with the QUESTDB database. To overcome

this fundamental issue, the following tests are performed using row-norm sorted CM representations wherein the regular CM representation is built and then the rows are sorted based on L_2 norm [44, 19]. That is, all models are trained and tested using sorted CM representations unlike the unsorted CM used in the preceding sections. The protocol followed to demonstrate the performance of the Γ-MFML models is as given below:

- Calculate the TD-DFT CAM-B3LYP def2-TZVP energies for the 90 molecules from QUESTDB
- Generate row-norm sorted CM for QeMFi
- Generate row-norm sorted CM for 90 molecules from QUESTDB
- Train single fidelity KRR model on QeMFi with 2^{10} training samples, Matérn kernel from Eq. (2.24) using $\sigma = 200$, and $\lambda = 10^{-10}$
- Train Γ(8) and Γ(32) MFML models with $\gamma = 10$ on QeMFi as discussed in preceding sections
- Predict energies for the 90 molecules from QUESTDB using KRR and Γ-MFML models from step 3 and step 4
- Compare prediction to reference computed excitation energies from step 1

A scatter plot of reference excitation energies versus ML predicted excitation energies is shown in Figure 10.7. The scatter plot is shown for Γ(8) and Γ(32) MFML models. The best 10, that is those with the lowest relative error, are highlighted in green along with the worst 10 in red. For each of the Γ-curve models, for the best and worst predictions, the corresponding predictions of the single fidelity KRR model are also presented. This is done for two reasons. Firstly to indicate that it is a challenge even for single fidelity ML models to handle transferability tests. Secondly, it is interesting to assess how well the MFML models performed with respect to the single fidelity KRR models.

Consider the left pane of Figure 10.7 for Γ(8) MFML model. The overall prediction of the Γ(8) MFML model is poor for the QUESTDB database. There is a wide scatter of the points across the identity line (dashed black line). Admittedly, the best 10 predictions do lie on or close to this identity line. Consider the predictions made by the single fidelity KRR model for these very geometries which corresponds to the translucent green square markers. These are much further away from the identity line. In other words, the Γ-curve method is somewhat better than the single fidelity KRR method. Even for the 10 worst predictions of

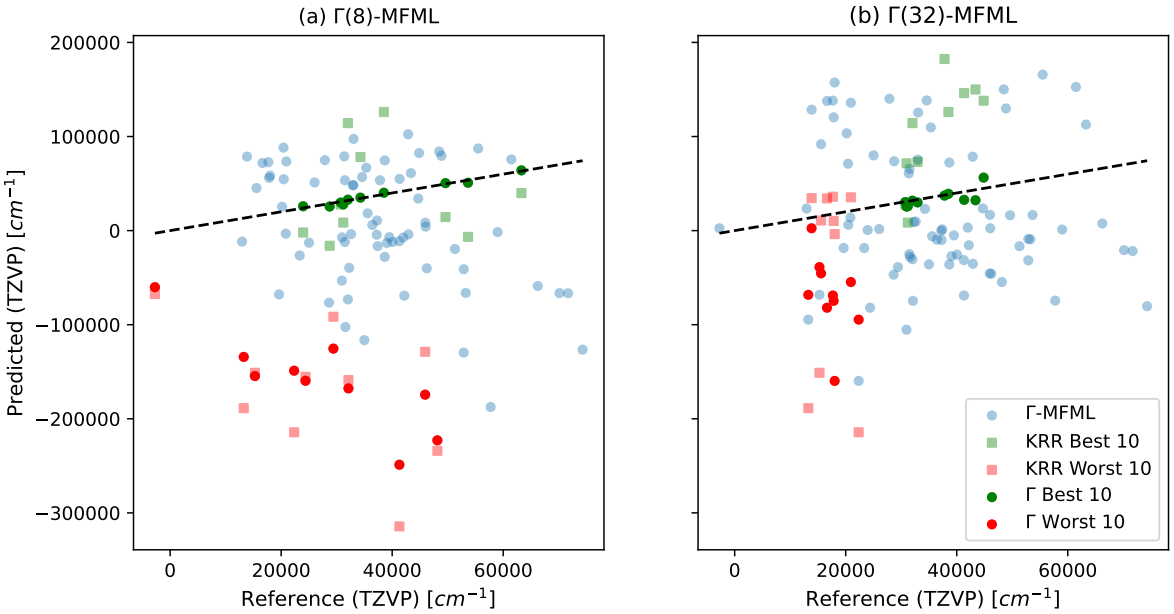


Figure 10.7: Scatter of reference and ML predicted excitation energies for transferability tests of Γ-curve on molecules from the QUESTDB database. The best 10 and worst 10 predictions are highlighted along with their predictions made using the single fidelity KRR model.

the Γ(8) model, the corresponding KRR predictions given as translucent red square markers, lie further away from the identity mapping.

Similar observations can be made for the case of Γ(32) on the right-hand side pane of Figure 10.7. The best 10 predictions of the Γ-curve model are close to the identity line while the predictions of the KRR model are more loosely scattered. However, in this case, the poorest predicted molecules for single fidelity KRR do have some points close to the identity mapping line. Certainly, the overall scatter plot for Γ(32) MFML looks closer to the identity mapping line as compared to the Γ(8) MFML model. Based on these results, one can hold on to what was stated before we started this assessment, the transferability of ML models remains a challenge, even for multifidelity approaches. Certainly the Γ(8) MFML model is more efficient, and more accurate as has been sufficiently established in preceding sections. The key takeaway from this discussion on transferability is not the trump of one model over the other but rather the fact that transferability is a difficult task for both single fidelity and multifidelity ML models.

To complete this discussion on transferability one can take a closer look into the errors of the Γ(8) MFML model. Figure 10.8 presents the molecules which comprise the 10 best predictions and 10 worst predictions along with the relative absolute error. It is unsurpris-

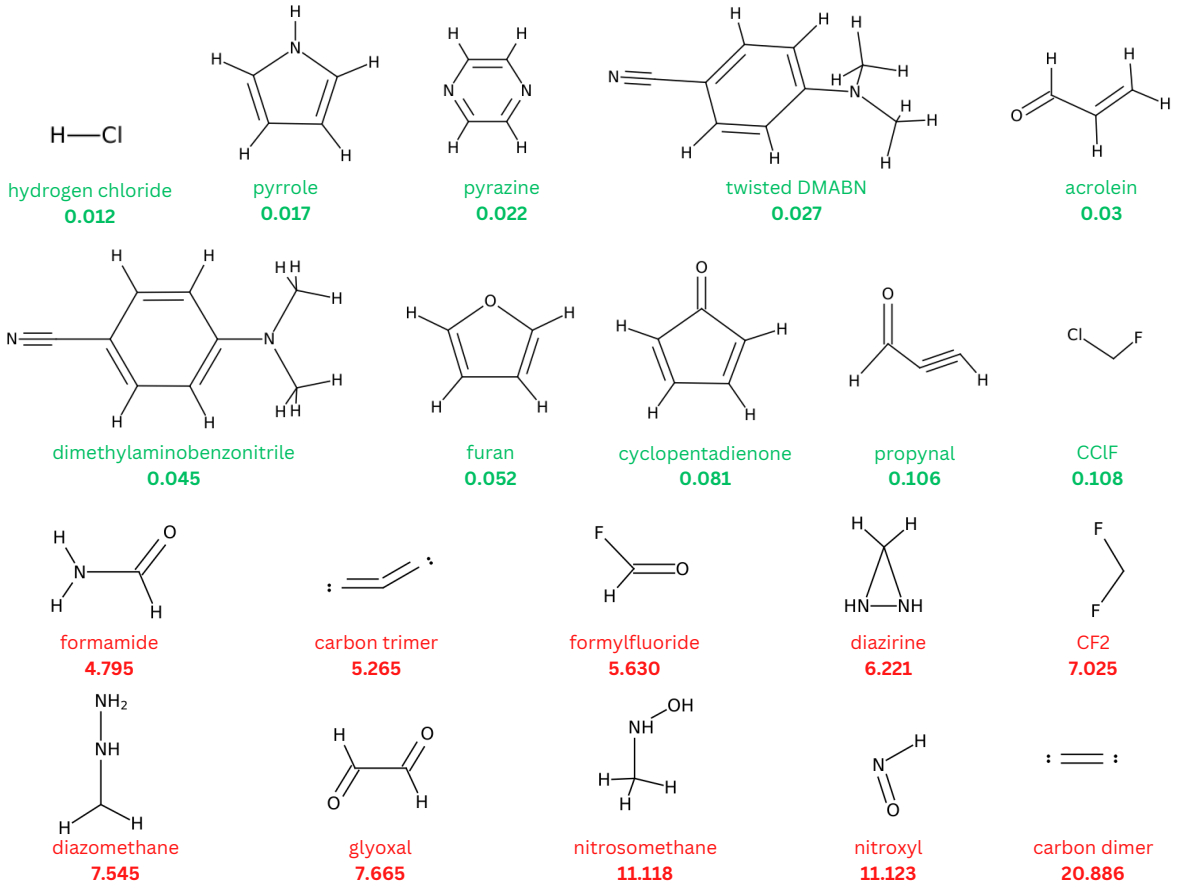


Figure 10.8: Best 10 (green) and worst 10 (red) predictions of the Γ(8)-MFML model over the QUESTDB dataset. The numbers under the name of the molecules indicate the relative absolute error.

ing to notice that the DMABN and acrolein geometries from the QUESTDB database have well predicted energies since the QeMFi dataset consists of geometries of these molecules. As argued previously, this is due to the fact that the ML model trained on specific geometries does well in predicting the energies of similar geometries from 'unseen' datasets. Consider the molecules that are not well predicted, the bottom two rows of Figure 10.8. Radicals such as carbon dimer and trimer along with molecules containing species which were not present in the QeMFi dataset are the primary inhabitants of this group. Once again, it becomes clear how important the initial training dataset is to the ability of a ML model in predicting QC properties.

For the Γ(32) MFML model, the 10 molecules with lowest and highest errors of prediction are shown in Figure 10.9 along with the relative absolute error under the names of the molecules. Again, excitation energies for DMABN and acrolein are well predicted. In addition, several other aromatic compounds such as naphthalene and phthalazine show low

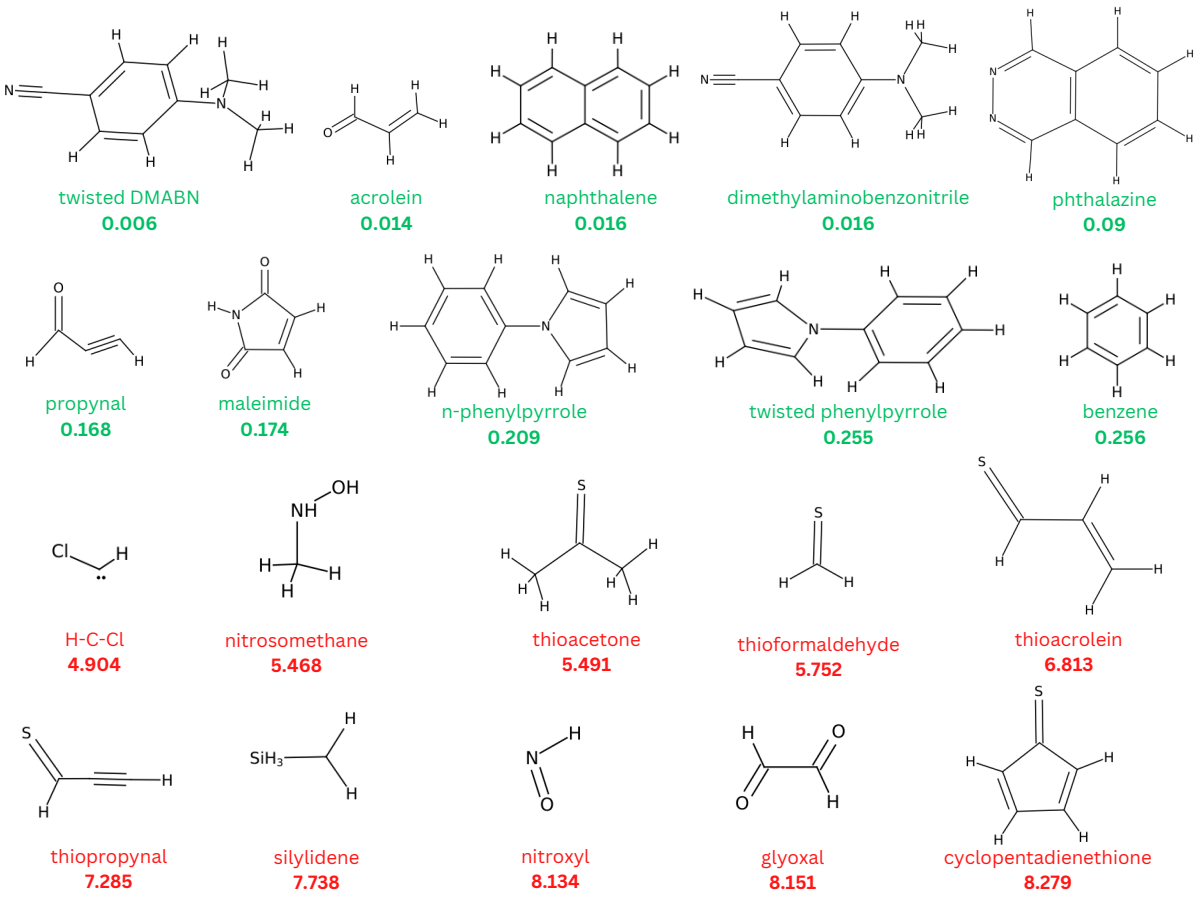


Figure 10.9: Best 10 (green) and worst 10 (red) predictions of the Γ(32)-MFML model over the QUESTDB dataset. The numbers under the name of the molecules indicate the relative absolute error.

relative absolute errors. In the bottom 2 rows depicting the molecules with the highest error in prediction, one observes radicals and molecules with elements such as sulfur and silicon which are not present in the QeMFi dataset.

Concluding this digression on testing the Γ-curve for transferability, some final remarks are made here. As has become evident, the task of transferability of a ML model is challenging and demands investigation in its own right [237]. The analysis of this short subsection is not indicative of an issue in the Γ-curve approach itself but rather ties into the larger picture of training general purpose ML models for QC [240, 241]. The reader is reminded that the work presented in this manuscript is intended to perform a training data hierarchy assessment for multifidelity ML methods in QC.

10.3 Conclusion

This chapter discussed the concept of scaling for the number of training data across different fidelities for MFML and o-MFML in the prediction of excitation energies of the QeMFi dataset. Constant scaling factors, γ, were studied along with QC-calculation time-cost informed scaling factors, θ^f_{f-1} and θ^F_f. It is seen in the results that the use of constant scaling factors,γ, is effective with a higher value of γ resulting in lower model errors for reasonable time-cost of generating training data. A new error metric, the error contour of MFML, was introduced and results discussed for the prediction of first vertical excitation energies of the QeMFi dataset. Such an analysis revealed that the data requirements for MFML-like methods is not as trivial as has been previously employed. In fact, one can achieve similar model accuracies with much less costly training samples if one increases the number of training samples at the lower end of the multifidelity data structure. The error contours revealed that one could potentially use as little as 2 training sample at the target fidelities and achieve exceptional model accuracy if the subsequent fidelities used a larger number of training samples in comparison to MFML models built with some γ. This was systematically studied with the newly introduced Γ-curve for a fixed number of training samples at the target fidelity and an increasing value of γ. The models built in this fashion were shown to be time-cost efficient over conventional MFML approach.

These results provide a window into the inner mechanisms of MFML-like methods allowing for a better understanding of how they can be employed for accurate predictions of excitation energies with low cost of training data generation. The development of the Γ-curve approach in this work in its current form is only benchmarked for a specific DFT functional and this could be a potential limitation of this work. A possible extension of this work could be the study of the Γ-curve approach for a wider range of DFT functionals. At the moment this is inhibited by the lack of compute cost times in most large-scale multifidelity datasets. Another interesting area of research can be the use of approaches developed in this work to assess the efficiency of multifidelity approaches for fidelity structures built on CC level of theory and would be of particular interest since CC is considered the gold standard in QC.

Part III

Complementary Applications and Conclusive Remarks

CHAPTER 11

MULTIFIDELITY MACHINE LEARNING IN PRACTICAL APPLICATIONS

Whatever it is you are seeking won't come in the form you are expecting.

— Haruki Murakami, *Kafka on the Shore*

The recurring theme of this dissertation was the development of MFML in order to reduce the overall cost incurred in generating training data for applications in the field of QC. The works presented in the preceding chapters dealt mostly with the development phase with some applications to small-scale QC molecules. These developed methods showed great improvement over existing single fidelity methods and the popular Δ-ML method. The methods developed in this dissertation have already been implemented extended applications to QC. Below, two such applications are discussed. These are adapted from refs. [1, 2] with only those portions which pertain to the contributions made by the author of this dissertation being presented. Dataset details when borrowed from the publications, are appropriately cited and contributions acknowledged. This chapter is a peek into realistic application of the MFML method and related approaches in the field of QC.

11.1 Molecular Energies of Monomers

This section is adapted from ref. [1] titled "Predicting molecular energies of small organic molecules with multifidelity methods". Only those sections which were contributed by the author of the dissertation are included here. Any further additions such

as description of data is appropriately cited. This application is based on the methods developed and presented in Chapter 9. In particular, the MFΔML approach and MFML method are compared for prediction of ground state energies at the DLPNO-CCSD(T) fidelity.

In order to understand day-to-day chemical processes such as atmospheric chemistry, it is pertinent that high accuracy thermochemical calculations be made. Coupled cluster QC theory is considered to be the gold standard method in computational Quantum Chemical methods. In particular, the Coupled cluster single, double, and a perturbative treatment of triple excitations (CCSD(T)), which accurately describes the electron correlation in molecules [15]. This increased accuracy in calculations comes with an increased cost of computation which scales non-linearly, approximately as $O^2(N^8)$, where N is the number of basis functions considered and O is occupied orbitals. Approximations such as the domain-based local pair natural orbital (DLPNO) method help reduce this cost without drastically altering the accuracy of the method [242, 243].

The use of ML in QC has significantly reduced the computational cost for large chemical systems [44, 49, 155, 244, 245, 22, 31]. The ML models learn a mapping between the Cartesian coordinates along with the respective atomic number, often converted to machine learnable input features called *molecular descriptors* or *representations*, and the QC property of interest such as ground state energies. This allows them to make predictions of the QC properties for molecules that the model has not previously been trained on. While ML in QC has provided a major respite to the cost of making costly calculations, a new overhead has since been presented to the use-case of ML-QC pipelines. This is the cost of generating the training data required for an ML model to achieve a certain accuracy. It is a common observation that the more training samples one uses, the better the model is able to predict the QC property of interest [22, 154].

Several methods to reduce the cost of training data have been introduced in this dissertation, including but not limited to the Δ-ML method [29]. In Δ-ML, training data from two different fidelities are used to train an ML model on the difference between the two fidelities. It is observed with the application of Δ-ML based methods that it is easier to learn the difference rather than the explicit value at the highest fidelity [29, 22, 31, 154]. The final prediction with an Δ-ML model involves the QC calculation of the cheap fidelity and the prediction of the difference. Since its introduction in the QC community, it has become a ubiquitous tool for a vast array of applications, including excitation energies, potential energy surfaces, electronic spectra, and isomerization enthalpies [29, 130, 206, 22, 31, 154, 134, 207, 208]. The method demonstrated that a smaller number of training samples could

be used to achieve a higher level of accuracy in the model. Previously, in ref. [208] some of the present authors used the Δ-ML approach to learn the CCSD(T) corrections over the CCSD energies for a collection of small organic molecules. In another related work, the Δ-ML was employed to predict the CCSD(T) energies of small organic monomers based on DFT results [246]. It is to be noted that Δ-ML is slightly different from transfer learning (TL)[247] which is another common approach used in ML-QC to reduce the use of costly data and has been employed in diverse applications such as thermochemistry and material analysis [248, 249, 250]. The key difference is that while Δ-ML trains on the explicit difference between two fidelities, TL first trains an ML model on the low fidelity and uses that to train for model parameters such as in the case of a neural network, the weights of the different hidden layers. The model parameters from this cheap-fidelity network are then 'transferred' to a new model, which is trained on the sparsely available high fidelity data.

Multifidelity Machine Learning (MFML) was introduced in Chapter 4 and shown to be superior in efficiency and accuracy to the single fidelity method. Alternative variations of the Δ-ML and MFML method have been introduced. Hierarchical-ML (hML) builds several Δ-ML like models for different fidelities in a manner similar to an MFML approach, however, with the number of training samples chosen to use an *ad hoc* optimization scheme [139]. The method has been shown to be effective in predicting ground state potential energy surfaces for CH_3Cl. In Chapter 4, optimized MFML (o-MFML), was developed as a methodological improvement over the conventional MFML approach by optimally combining the sub-models used for MFML. The o-MFML method uses a validation set computed at the target fidelity to optimize the combination of the sub-models and has been shown to provide better accuracy for the overall prediction for both excitation energies and atomization energies in Chapter 6 and in cases where training data might be heterogeneous as shown in Chapter 8.

Other ML methods have also been studied in their effect to reduce the computational cost associated with the generation of training data. Hierarchical-ML uses solves a minimization problem for a use defined target error and a number of training samples to be used at the different fidelities [139]. The method has been used to predict a full basis set approximation of the ground state potential energy surface for CH_3Cl. Multi-task Gaussian processes are yet another method introduced recently and have been seen to reduce the overall cost associated with a multifidelity model [110]. The model was seen to be effective in the prediction of many-body interaction terms for water and showed favorable results even in cases of heterogeneous training data. Another useful approach to reduce the cost of training data is the recently introduced minimal multilevel machine learning (M3L)

method, an update of the MFML method. In this method, the number of training samples to be used at each fidelity are optimally computed using Bayesian optimization of a cost function for a target model error [233].

This study of model efficiency for several multifidelity methods in Chapter 9 revealed that the use of MFML is beneficial when requiring large numbers of predictions. The multifidelity Δ-ML (MFΔML) method was also developed in Chapter 4. In this method, several Δ-ML like sub-models are combined in a manner similar to that in MFML. This method was shown to be superior to the conventional Δ-ML method in model error and overall efficiency. These benchmarks in Chapter 9 were performed for models that are trained and evaluated across different fidelities restricted to the DFT level of theory. In the application presented herein, single fidelity KRR, Δ-ML, MFML, and MFΔML methods are employed in the prediction of CCSD(T) accuracy energies of small organic molecules.

Dataset details taken from ref. [1]

The database from a previous study [246] was extended in ref. [1]. In ref. [246], around 8000 monomers were randomly selected from a public database which focuses on determining the enthalpies of radical reactions for small organic molecules [251], and then geometry optimized at the B3LYP-D3(BJ)/cc-pVTZ level of theory and then their single-point energies were computed using DLPNO-CCSD(T) theory. More than 12000 additional molecules from the same quantum chemistry database were geometry optimized at the B3LYP-D3(BJ)/cc-pVTZ level of theory. The free radicals in the database are important intermediates in combustion and atmospheric chemistry and their energies are essential to determine the thermodynamics and kinetics of reaction pathways. In order to save the time cost for advanced quantum chemical calculations, only a small molecules in the database (no more than ten heavy atoms) were selected. The molecular energy and weight distributions of the dataset are given in the supplementary information of ref. [1]. After checking for duplicates via the generated SMILES, 12340 molecules remained in the database (4606 data points with DLPNO-CCSD(T) single-point energies from the previous database and 7734 additional molecules) consisting of only hydrogen, carbon, nitrogen, and oxygen atoms. All these molecules were then subjected to DFT single-point energy computations using the B3LYP-D3(BJ) functional in conjunction with the STO-3G basis set. Subsequently, 1500 data points with DLPNO-CCSD(T) energies were randomly selected as the test set for the ML models, and all the rest were used for training.

The dataset generation process was carried out by the collaborative authors of ref. [1]

L. Dongyu, M. Ruth, P. Schreiner, and U. Kleinekathöfer.

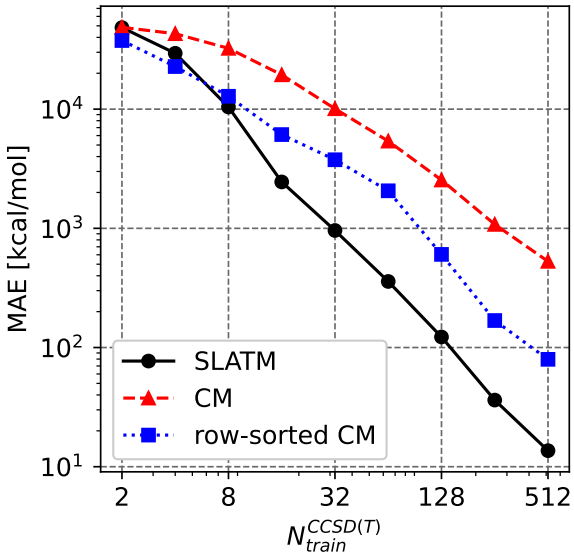


Figure 11.1: Comparing representations for single fidelity KRR at the DLPNO-CCSD(T) fidelity. Results are shown for an average of ten runs with shuffled training data. The SLATM representation performs the best out of the three, and the sorted Coulomb Matrices (CM) performs better than the unsorted CM.

A preliminary assessment of molecular descriptors was made to prepare for the use of multifidelity methods to the dataset. Unsorted CM, row-norm sorted CM, and SLATM molecular descriptors were tested since these are the most common descriptors for such applications. The results of the assessment are shown in Figure 11.1 for a single fidelity KRR model trained only on the target fidelity DLPNO-CCSD(T). The learning curves indicate that the SLATM representation performs the best out of the three. The sorted CM performs better than the unsorted CM. This could be due to the fact that the sorted CM and SLATM representations retain index invariance of the descriptor, which is missing in the unsorted CM descriptor. For a use case such as the one presented here where the models are trained and evaluated on different molecules as opposed to training on a trajectory of the same molecule as in ref. [142], the retention of indexing invariance is pertinent [44, 252, 22, 19]. At the same time, the sorted CM performs worse than the SLATM representation. This could be due to the fact that the sorting of the CM results in undesirable discontinuities [44, 19] which potentially deter the ML models from being able to learn anything meaningful. Based on this assessment, for the remainder of this work, the SLATM representation is used throughout for all ML models. The preliminary data assessment of the training data as prescribed in ref. [142] is given in Figure A.17. The analysis indicates that the chosen hier-

archy of the fidelities is indeed conducive to effective working of the multifidelity models. The mean absolute difference in the energy values of the fidelities shows a systematic decrease and is a first indicator of the abilities of MFML model in predicting the target fidelity with good accuracy.

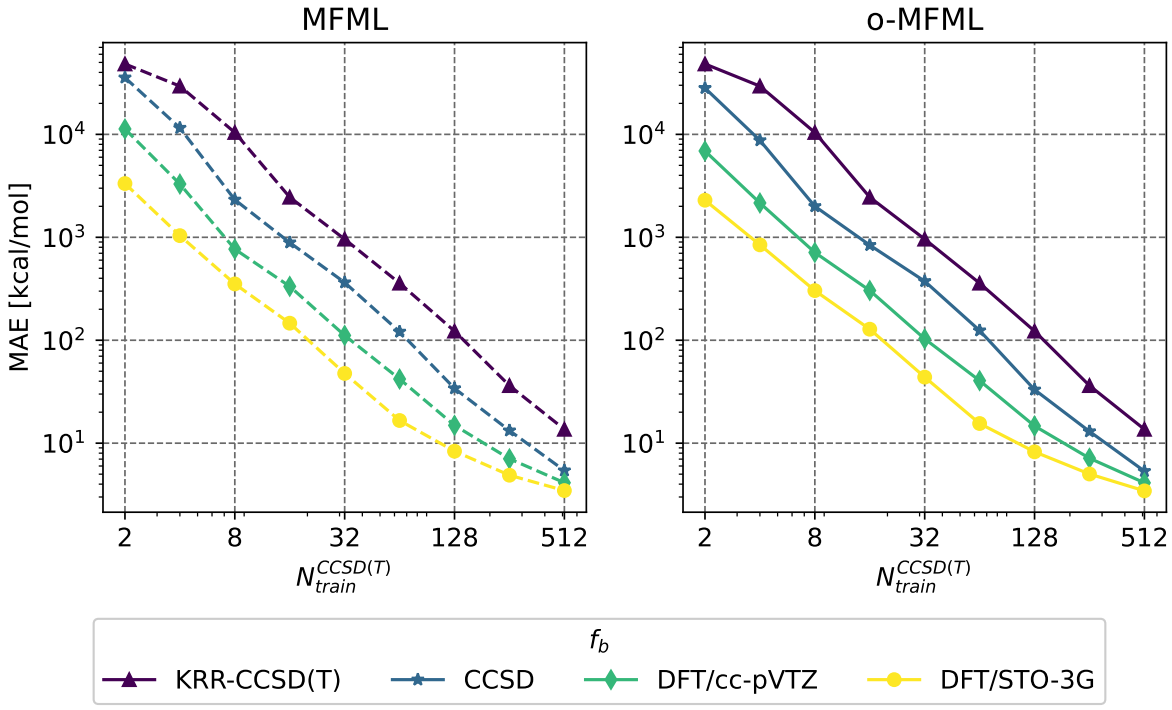


Figure 11.2: MFML and o-MFML learning curves with varying baseline fidelities. The learning curve for the single fidelity KRR model built with only DLPNO-CCSD(T) training data is also shown for reference.

MFML and o-MFML models were built with varying baseline fidelities for the prediction of energies for the monomers. The resulting learning curves are presented in Figure 11.2 for both these models. The single fidelity KRR built with only DLPNO-CCSD(T) training samples is shown for reference. With the addition of cheaper fidelities, the learning curves of the models show a constant lowered offset. That is, for the same number of training samples as used for the single fidelity KRR model, the MFML models result in a lower MAE. While the o-MFML model is a methodological improvement over the MFML method, in this case the difference is not very pronounced and the model MAEs for MFML and o-MFML are rather similar.

In this work, the Δ-ML and MFΔML methods are also evaluated. The reader is referred to Figure A.18 and Figure A.19 in Appendix A for results of Δ-ML with different values of QC_b. The overall trend is as expected based on the study from refs. [29, 144] and as dis-

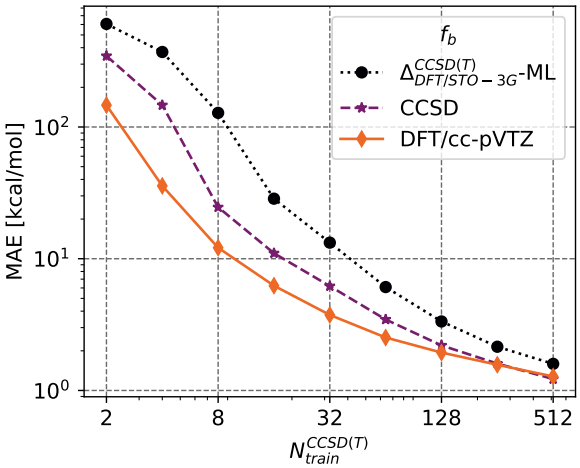


Figure 11.3: Learning curves for MFΔML. The QC baseline is DFT/STO-3G. The different baseline fidelities of the MFΔML model are shown in the legend. The learning curve of Δ-ML model built with DFT/STO-3G as QC-baseline and DLPNO-CCSD(T) target fidelity is also plotted.

cussed in Chapter 9. That is, with a QC_b that is closer to the target fidelity, the Δ-ML model shows a higher accuracy in prediction. However, as Figure A.19 indicates, the time-cost incurred in using higher QC_b far outweighs this benefit. As described in section 4.3, the MFΔML method builds a multifidelity model consisting of various Δ-ML models. The resulting learning curves are shown in Figure 11.3. In addition to the learning curves for MFΔML, the learning curve for the standard Δ-ML model built with the DFT/STO-3G as QC-baseline is shown as well. Once again, as for the case of MFML, the addition of a cheaper fidelity to the basic Δ-ML model results in a lower offset of the learning curve. However, for large enough training set sizes, $N^{\mathrm{CCSD(T)}}_{\mathrm{train}} = 512$, this offset is not very pronounced vis-á-vis the Δ-ML model. Furthermore, the learning curve for MFΔML with f_b CCSD and f_b DFT/cc-PVTZ converge at this point. This convergence could be an indication of the saturation of the model due to the very similar structures of the monomers.

Figure 11.4 depicts the difference between ML model prediction and reference DLPNO-CCSD(T) energies for the holdout test set used for the study of learning curves. The results are shown for both the MFML and MFΔML models with varying baseline fidelities. The error distribution of the single fidelity KRR with only DLPNO-CCSD(T) energies and the standard Δ-ML model with DFT/STO-3G as the QC-baseline are also shown for reference. Consider the left-hand side plot of Figure 11.4 which is the case for the single fidelity KRR and MFML models. It is seen that all the ML models predict with a difference centered around 0 kcal/mol. However, the single fidelity KRR model has a wide spread of the differ-

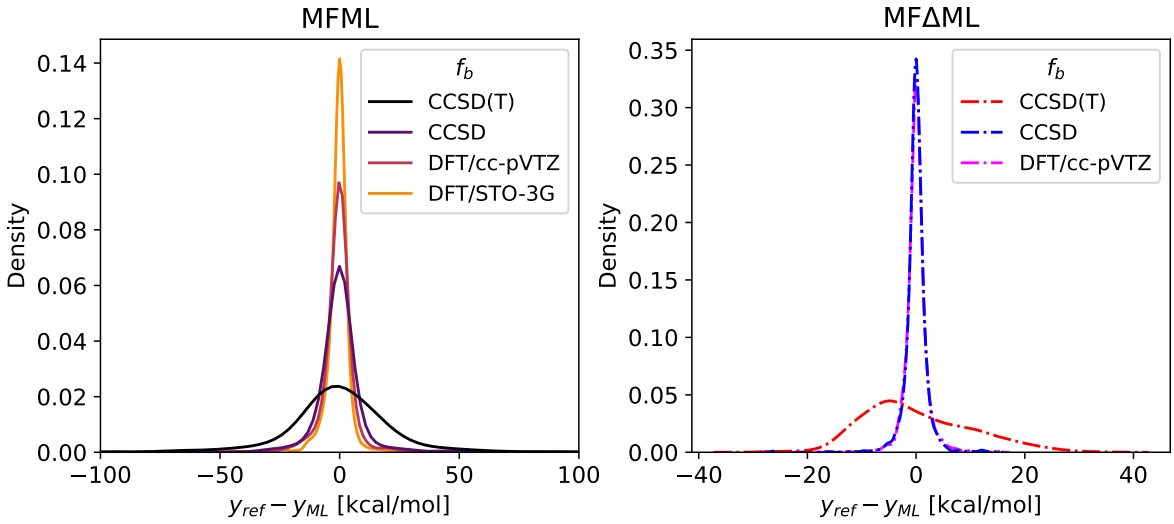


Figure 11.4: Distribution of difference in model prediction and computed reference DLPNO-CCSD(T) energies over the holdout test set of 1,500 samples for MFML and MFΔML models with varying values of f_b.

ence between reference and prediction. With each additional cheaper fidelity that is added to create the MFML model, the peak of the differences gets tighter around 0 kcal/mol meaning, the MFML models predict the DLPNO-CCSD(T) energies with increasing accuracy as one decreases the baseline fidelity. This agrees with the study of learning curves that was presented in Figure 11.2.

The right-hand side plot of Figure 11.4 depicts the distribution of the difference between reference DLPNO-CCSD(T) energies and the energies predicted by the different Δ-ML models that were studied in this work. These are built with the DFT/STO-3G fidelity as the QC baseline as explained in Section 4.3. Note that the x-axis, marking the differences, is different from that for the MFML models on the left-hand side plot, almost by an order of magnitude. On comparing the distribution of differences for the different Δ-ML models, the standard Δ-ML model (denoted in the legend by the DLPNO-CCSD(T)) has the widest distribution range with a peak that is shifted towards the left of 0 kcal/mol. With the addition of cheaper baselines to create the MFΔML models, the peak becomes narrower and centered around 0 kcal/mol. This is once again in agreement with the analysis of the learning curves for MFΔML models from Figure 11.3 performed above.

The outliers in the plots of Figure 11.4 warrant some discussion of possible reasons. The large difference in predictions could arise due to lack of diversity in the training data. Homogeneity in the training data results in the ML models ending up being overfitted to the simplistic training data and struggling to make predictions for out of sample data. Alter-

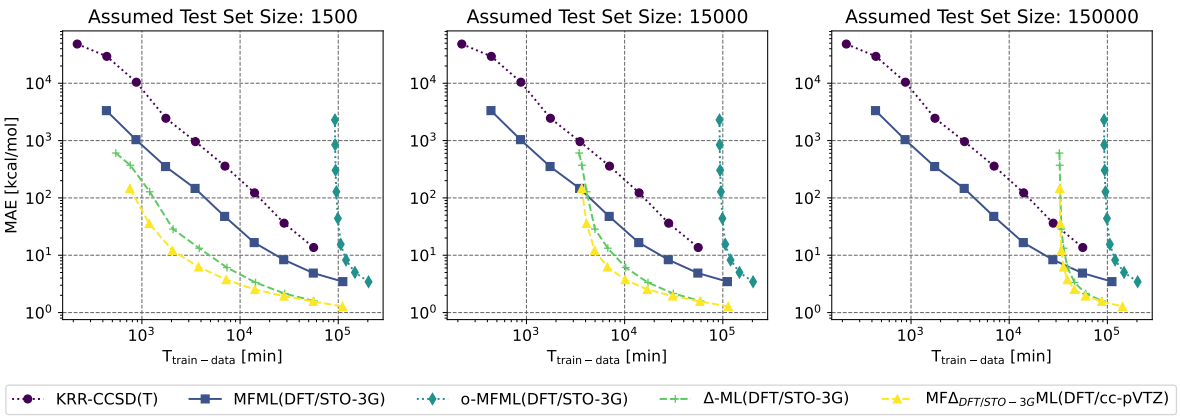


Figure 11.5: Model MAE versus the time to generate the training data. Three test set sizes are compared.

natively, outliers in prediction could be due to the complexity of certain molecules being under-represented in the training dataset, *e.g.*, cyclobuta-1,3-diene. Even so, as expressed above, the majority of the predictions are close to the reference values as seen by the peaks being centered around 0 kcal/mol.

While these are interesting results about the capabilities of both MFML and MFΔML methods, it becomes pertinent to also account for the time cost associated with these different models when predicting DLPNO-CCSD(T) energies. Figure 11.5 depicts the model MAE as a function of generating the training data for the collection of ML models that are compared in this work. This comparison is made for the single fidelity KRR, MFML and o-MFML models built with DFT/STO-3G baseline fidelity, the Δ-ML model with the QC-baseline fidelity, and the MFΔML model with the DFT/cc-PVTZ fidelity. For the MFML model, the training data cost accounts for the complete multifidelity training structure, similar to what is discussed in ref. [142]. That is, the cost of training data at all the fidelities used in the MFML model. For o-MFML model, the time-cost also includes the cost of generating a validation set over which the optimization procedure is carried out. For Δ-ML and MFΔML models the cost includes the time to make the QC-baseline calculations.

Figure 11.5 compares the time cost versus MAE for three hypothetical test set sizes, i.e., 1.5k, 15k, and 150k samples. The actual MAE values are calculated over the fixed test set of 1.5k samples. However, since the MAE values reported are for an average over ten runs, it is expected that the model MAE would be similar for a larger test set. The interesting thing to note is the time cost of generating the training data. In cases where one needs to predict energies for a few geometries, 1.5k in this case, the MFΔML model performs the best. As one increases the test set size, the time cost of making the QC-baseline calculations for the

Eval Size	**DLPNO-CCSD(T)**	**KRR**	**Δ-ML**	**MFML**	**o-MFML**	**MFΔML**
1500	1.64×10^5	5.61×10^4	5.66×10^4 (1.59)	1.11×10^5	2.04×10^5	1.11×10^5 (1.28)
15000	1.64×10^6		5.95×10^4 (1.59)			1.14×10^5 (1.28)
150000	1.64×10^7	(13.64)	8.85×10^4 (1.59)	(3.46)	(3.44)	1.43×10^5 (1.28)
1500000	1.64×10^8		3.79×10^5 (1.59)			4.33×10^5 (1.28)

Table 11.1: Time-costs (in minutes) for different sizes of the test set. The reference cost on using DLPNO-CCSD(T) conventional computation is contrasted alongside. For the ML models, the time cost is computed for $N_{\text{train}}^{\text{CCSD(T)}} = 2^9$ with remaining multifidelity data structure being accounted for as expressed in the main text. The values in the parenthesis denote the MAE of the ML models. It is to be noted that the Δ-ML and MFΔML models also have the cost of the QC-baseline fidelity.

Δ-ML and MFΔML models outweighs the potential benefit of the method. In contrast, the MFML model is unaffected by the size of the test set. This is due to the fact that the MFML approach also predicts the baseline fidelity rather than using QC computed values. In large test set size regimes, this sets the MFML to be the more efficient method. The o-MFML method, across the different test set sizes, is the most expensive model to build. This is expected since the cost of the validation set is affected by the target fidelity, which in this case is the DLPNO-CCSD(T), an expensive QC method. Table 11.1 reports the time-costs in minutes for the different ML models in contrast to using conventional QC computations for the DLPNO-CCSD(T) fidelity. The ML models are built with 2^9 training samples at the target fidelity of DLPNO-CCSD(T). It is evident that the use of any ML method is better than the use of conventional QC computational methods. Notice that the time-costs for KRR, MFML, and o-MFML are fixed regardless of the size of the test set. The Δ-ML and MFΔML, although lower in model MAE are sensitive to the size of the test set. To make this clearer, in the table a test set size of 1.5 million samples is also presented. In contrast, the MFML model is unaffected by the size of test set since even the f_b fidelity is predicted with an ML model.

While this is a glimpse into the ML related aspects presented in ref. [1], the fully trained models were thereby used to test for transferability over three additional datasets and showed great promise. The results indicate that MFML and MFΔML are efficient high-accuracy ML methods in the field of QC and provides a compelling narrative for the use of MFML in predicting DLPNO-CCSD(T) energies.

11.2 Excitation Energy Transfer in Porphyrins

This section is adapted from ref. [2] titled “Excitation energy transfer between porphyrin dyes on a clay surface: a study employing multifidelity machine learning". Only those sections which were contributed by the author Vivin Vinod are included here. Any further additions such as description of data is appropriately cited. The application is based on the work developed in Chapter 10, in particular, the Γ-curve approach.

The keen reader will recall that one motivation to developing MFML methods in Chapter 5 was to assess the method for excitation energies to better understand its scope for large scale systems. After favorable results presented in this dissertation, the MFML and Γ-curve MFML approach (see Chapter 10) were implemented in such an application. Ref. [2] presents a study of exciton transfer in several porphyrin molecules on a montmorillonite clay surface, that is $(K,Na)_x[Si_4O_8][Al_{(2-x)}Mg_xO_2(OH)_2]$. Porphyrin was chosen based on interesting results arising from experimental synthesis demonstrating high-efficiency energy transfer.

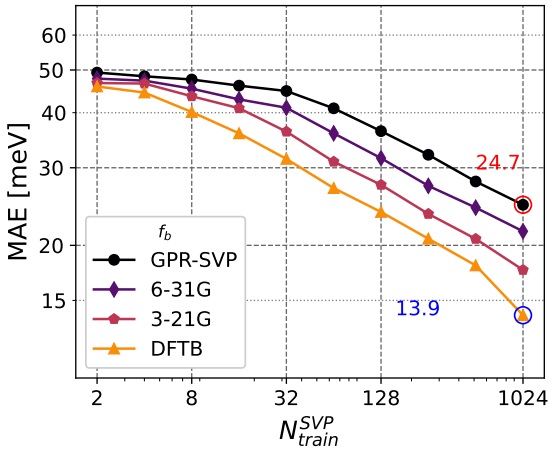
Two porphyrin molecule types, namely, m-TMPyP and p-TMPyP were studied on the clay surface for a total of 16 molecules. The QM/MM and MD simulations were run as presented in ref. [2] with a total of 40,000 snapshots per molecule being generated with QM/MM as the training geometries for ML models. The surrounding point charge electrostatic potential environment was considered. For each porphyrin molecule, time dependent LC-DFTB fidelity [173] was used to calculate the excitation energies. In order to generate multifidelity training data, the TD-DFT formalism with the CAM-B3LYP functional was used. The fidelities were distinguished on the basis set size, namely, STO-3G, 3-21G, 6-31G and def2-SVP with the excitation energies computed with a time stride of 8, 16, 32 and 64 fs, respectively. Further details on the clay surface and related assumptions can be found in ref. [2].

Expensive calculations of the excitation energy for the first excited state at the def2-SVP fidelity are replaced by evaluations of MFML models, one model for all 12 p-type molecules and one model for all m-type molecules. The final models for the prediction of the excitation energies of the p-TMPyP and m-TMPyP porphyrin pigments are multifidelity machine learning models (MFML). Such a MFML model involves the use of quantum chemistry training data at different *fidelities*, which refer to the accuracy of the data with respect to the actual value [32]. The MFML models are built with respect to some *target fidelity* and a *baseline fidelity*. The former, denoted hereon as F, refers to the most accurate (and thereby costliest) fidelity that one is interested in predicting with the ML model. The base-

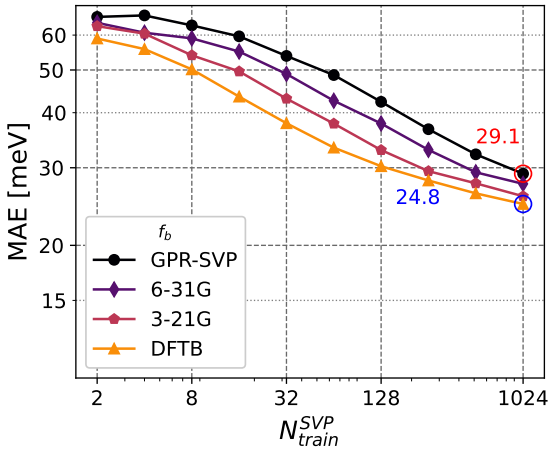
line fidelity, f_b, is the cheapest (and thereby the least accurate) fidelity data that is used in the MFML model. This MFML model is denoted by $P_{\text{MFML}}^{(F,\eta_F;f_b)}$ where $2^{\eta_F} = N_{\text{train}}^{(F)}$ is the number of training samples used at the target fidelity. The number of training samples at the subsequently cheaper fidelities are determined by the use of a *scaling factor*, γ which is conventionally set to 2 based on ref. [32]. That is, $N_{\text{train}}^{f} = \gamma \cdot N_{\text{train}}^{f-1}$ for all $f_b < f \leq F$. In this specific application, the target fidelity is TD-DFT/CAM-B3LYP with the def2-SVP basis set while the cheapest baseline fidelity is the LC-DFTB approach. These will be reported using a shorthand notation, that is SVP and DFTB, respectively. The different ML models are assessed based on the MAE using learning curves which depict the MAE as a function of the number of training samples used at the target fidelity. In addition, the MAE is studied as a function of the time-cost of generating training data for a specific ML model, be it single fidelity GPR or MFML models with different values of f_b (see section 2.7). These are reported alongside the Γ-curve, introduced in Chapter 10, which fixes the number of training samples at fidelity F and varies only the value of γ (see Chapter 10). This is shown to be superior to the conventional approach of MFML in providing a low-cost high-accuracy ML model for the prediction of excitation energies of porphyrin.

Single fidelity GPR models and MFML models were built and compared on the single pigment p-TMPyP-9 and on the union over all p-TMPyP pigments for transferability reasons. In addition tests are conducted on the union over all m-TMPyP pigments. In the all-pigment models an even sampling of the training data is used. The models are tested on a separated holdout test set for which the excitation energies are calculated at the target fidelity, that is SVP. For p-TMPyP pigments, 2,000 test samples are used, while for m-TMPyP pigments 500 test samples are considered, to account for the lower total amount of data. The resulting MFML learning curves for the different cases are shown in Figure 11.6. The shown learning curves are an average over ten learning curves created from shuffling the training data set. The different learning curves in a single study are given for a growing number of utilized fidelity levels starting from the baseline fidelity f_b, as indicated in the legend.

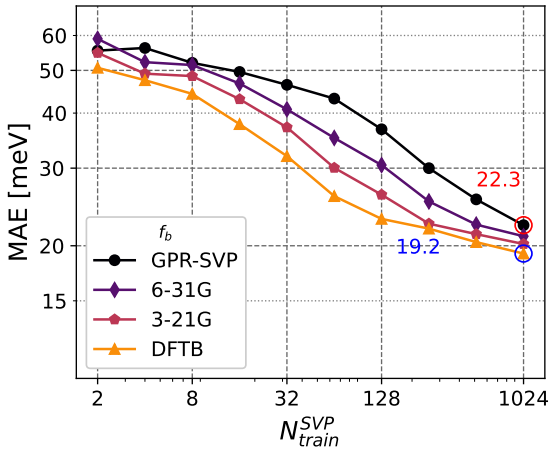
The case of training and testing on the same trajectory of p-TMPyP porphyrin molecules is shown in Figure 11.6(a) for different f_b. With the addition of cheaper baseline fidelities, one observes that the MFML model predicts with a lower MAE in comparison to the single fidelity GPR model built with training samples only from the target fidelity. For instance, with $N_{\text{train}}^{\text{SVP}} = 1024$, the GPR model results in an MAE of 24.7 meV while the MFML model with the baseline fidelity f_b set to DFTB results in an MAE of 13.9 meV. While this is a promising result, the transferability study from above indicated that a joint model for



((a)) Based only on a trajectory of porphyrin pigment p-TMPyP-9.



((b)) Concatenated trajectories of the p-TMPyP pigments.

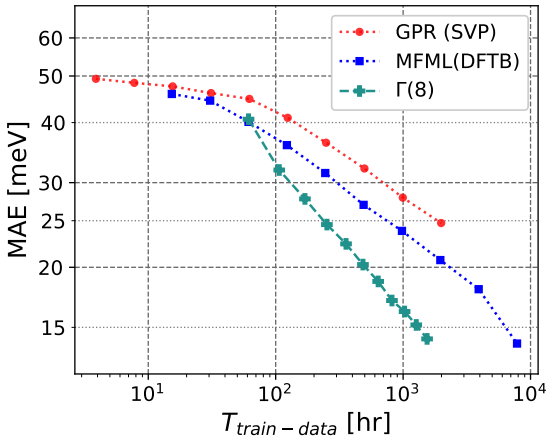


((c)) Concatenated trajectories of the m-TMPyP pigments.

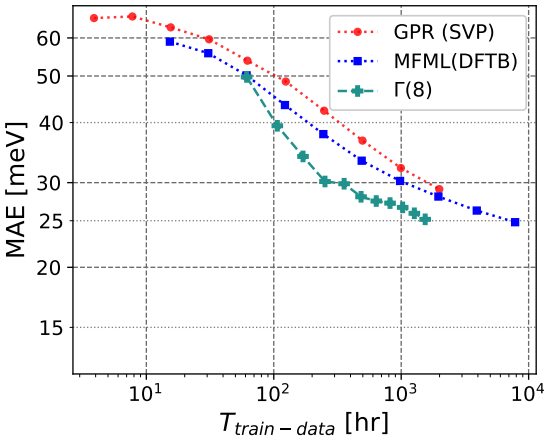
Figure 11.6: MFML learning curves for three cases of predicting excitation energies for porphyrin molecules. The prediction error (as MAE) of the single fidelity GPR and of the MFML model with the DFTB baseline fidelity are explicitly stated for $N^{\mathrm{SVP}}_{\mathrm{train}} = 1024$.

all pigments of one type should be constructed. For this reason, the final MFML models that are used to predict the excitation energies for the porphyrin molecules are built with training data taken from a pool of trajectories for each type of porphyrin molecule. The MFML learning curves for the p-TMPyP porphyrin molecules are delineated in Figure 11.6(b) for different baseline fidelities. While the addition of cheaper baselines does decrease the model MAE, this drop is not as significant as seen in the case for the single trajectory. The drop in error between single fidelity GPR and MFML with the DFTB baseline fidelity is about 6 meV for $N_{\text{train}}^{\text{SVP}} = 1024$. However, this is anticipated since both the single fidelity GPR and the MFML models have to cover a wider region of the conformational phase space (see discussion on the UMAPs in ref. [2]) as opposed to a smaller region that is to be covered in the case for the single trajectory models. A similar observation is made for the MFML learning curves for m-TMPyP porphyrin molecules as seen in Figure 11.6(c) with the single fidelity GPR model reporting an MAE of 22 meV and the MFML model with DFTB baseline reaching an MAE of 19 meV with 1024 training samples at the SVP fidelity. The slightly overall lower MAE for the m-TMPyP porphyrin dyes can be explained once again by the fact that the concatenated trajectories of this porphyrin type result in a lower number of geometries which in turn could span a smaller region of the conformational phase-space as opposed to the case for the p-TMPyP porphyrin molecules. That is, the m-TMPyP set has a smaller number of total geometries when concatenated in comparison to the total geometries of the p-TMPyP set. The larger number of total geometries for the p-TMPyP set implies that the MFML model with $N_{\text{train}}^{\text{SVP}}$ would contain a smaller amount of information about the conformation space of the molecule, in contrast to the MFML model built for m-TMPyP set. This fact is reflected in the learning curves.

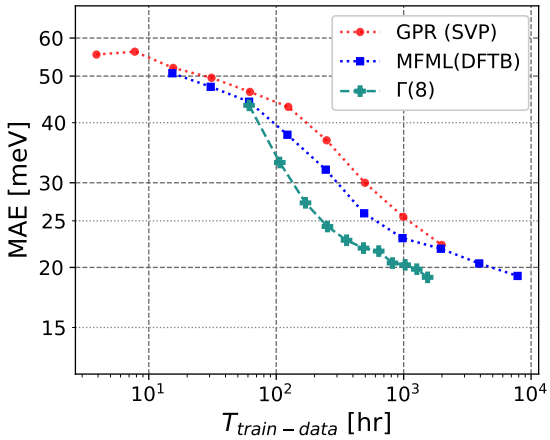
In order to better assess the computational impact of single fidelity and MFML models for porphyrin molecules, the model error is studied as a function of the cost of generating the training data used in the ML model. In this work, the QC calculation times as returned by the ORCA computing software [7] and DFTB+ software [172, 253] are used. For the GPR, this cost is directly related to the number of training samples. For the MFML model, this cost includes not only the training samples used at the top fidelity, but also the cost of the training samples used at the subsequent lower fidelities. These curves are shown in Figure 11.7. The time required for training the models and predictions over the holdout test set of the MFML model for the largest training set size used (that is, $N_{\text{train}}^{\text{SVP}} = 1024$) was 12.97 seconds and 12.45 seconds for the p-TMPyP and m-TMPyP porphyrin molecules, respectively. Since this is such a small contribution, only the total time for generating the training data is considered in the MAE versus time-cost curves.



((a)) Based only on a trajectory of porphyrin pigment p-TMPyP-9.



((b)) Concatenated trajectories of the p-TMPyP pigments.



((c)) Concatenated trajectories of the m-TMPyP pigments.

Figure 11.7: Time-cost of generating training data versus MAE in meV for a single fidelity KRR contrasted with that for the MFML model built with baseline fidelity f_b DFTB. The Γ(8)-curve is also depicted for increasing values of γ as explained in 4.1.

In addition to the single fidelity GPR and MFML models, a recently introduced MFML approach, referred to as the Γ-curve [143], is analyzed as well. In conventional MFML theory, the training samples at the various fidelities are decided by a *scaling factor*, γ, that is, $N^f_{\text{train}} = \gamma \cdot N^{f-1}_{\text{train}}$ for $f \in \{2, \ldots, F\}$. Conventionally, a MFML model is built with $\gamma = 2$. The use of different values of γ was shown to result in a more efficient approach titled the Γ-curve in Chapter 10. The Γ-curve is a plot of MAE versus time-cost of the MFML model with increasing values of γ. The Γ-curve is built with a fixed number of training samples at the target fidelity, SVP. Figure 11.7 reports the Γ(8) curve, that is, with $NM^{\text{SVP}}_{\text{train}} = 8$ with varying values of γ. Different values of $N^{\text{SVP}}_{\text{train}}$ were considered and are shown in Figure A.27.

Figure 11.7(a) depicts the MAE versus time-cost of the ML models for the case of the single trajectory of p-TMPyP porphyrin molecules. One observes that for a given time-cost on the horizontal axis, the curve for the MFML model is always below that for the single fidelity GPR curve. This implies that for a given time-cost, the MFML model results in a lower error than the single fidelity GPR model. Furthermore, the Γ(8) curve lies lower than the conventional MFML curve. Once again, this implies that for a given time-cost, the multifidelity model built along the Γ(8)-curve results in a lower MAE. A similar observation is made for the case of concatenated trajectories of the p-TMPyP and m-TMPyP molecules in Figures 11.7(b) and 11.7(c), respectively. Although for the m-TMPyP porphyrin molecules, the GPR curve does reach close to the conventional MFML curve, the Γ(8)-curve always lies beneath it. The final multifidelity models that are used in this work for the prediction of excitation energies correspond to the final data point of the Γ(8) curve, which corresponds to $\gamma = 12$. The multifidelity training structure for this model is $\{8, 12 \cdot 8 = 216, 12^2 \cdot 8 = 1152, 12^3 \cdot 8 = 13824\}$ with decreasing fidelity. For the p-molecules, this model results in an MAE of ∼ 25 meV with a time cost of about 1500 hours, while the conventional MFML model reports a similar error with a time cost of roughly 8000 hours. The single fidelity GPR model only reaches an MAE of 29 meV with a time-cost of 2000 hours. The use of the multifidelity model along the Γ(8) curve results in a time-cost benefit of roughly 5 over the conventional MFML model with $\gamma = 2$ and $N^{\text{SVP}}_{\text{train}} = 1024$. For the p-TMPyP porphyrin molecules, the corresponding time-benefit of using the multifidelity model along the Γ(8) curve over the conventional MFML model is about the same with the former reporting an MAE of about 17 meV for a time-cost of roughly 1500 hours, while the latter costs as much as 8000 hours for an MAE of about 19 meV. In terms of an overall comparison, if one were to employ conventional QC computation for the excitation energy computations at the SVP fidelity, it would cost $640,000 \times 2$ hrs $\approx$ 150 years of CPU time. This was achieved at a cost of some 1500 hrs which is a time-cost saving of ∼ 850x! Such speed

up factors are indeed a promising output from the methods developed in this dissertation.

These promising results for multifidelity methods such as MFML and Γ-curve are strong statements to their efficiency and possibility for future applications. Their implementation to large-scale systems such as 16 porphyrins on clay surfaces are is a good indicator that these can be further employed to accelerate research on complex light harvesting systems.

CHAPTER 12

CONCLUSION

And on the seventh day God ended His work which He had done, and He rested on the seventh day from all His work which He had done.

— Genesis 2:2, *The Holy Bible [NKJV]*

The use of ML in QC has been progressing at a strong pace with trained ML models being used in tandem with conventional computation methods. In some sense this has resulted in the shift of the cost of discovery from expensive single point calculations to a smaller amount of costly calculations needed to train ML models. The work collated in this dissertation provides a detailed methodological anchor point for the reduction of time-costs incurred to train ML models for the prediction of QC properties. Specifically the cost of generating training data is addressed in this document in detail with MF methods of ML being offered as a solution to reduce this overhead. MFML and o-MFML models were numerically shown to reduce the amount of costly training data needed for a specific model accuracy.

12.1 Summary

In the introductory chapter of this thesis, after motivating the need for MF methods in the field of ML-QC, 4 key objectives were laid out to be fulfilled. At this juncture, it is clear that these objectives lie completed. This dissertation started by examining existing the state of art for MF methods and how certain aspects had been extended in implementation to the ML-QC pipeline. The MFML, o-MFML, MFΔML, o-MFΔML, and Γ-curve methods were developed for the prediction of diverse scalar QC properties. MFML was shown to be a high-

accuracy low-cost method in the prediction of excitation energies for molecular trajectories of arenes. The o-MFML method was shown to be a robust and accurate successor to MFML.

After the benchmark MF dataset of QC properties, QeMFi was introduced in Chapter 7, it was used to assess the effect of non-nested configurations of training data for MFML and o-MFML. While the nested configuration of training data is recommended, o-MFML was shown to be effective even in cases of non-nested training data. Data efficiency assessments were made for MFML, o-MFML, and Δ-ML on the basis of time-costs of generating training data for each method. The MFΔML method was shown to be preferred over simple Δ-ML in cases of small number of evaluations.

A comprehensive study of the data hierarchy of MF training data was performed to understand its effect on model accuracy. A novel error metric, error contours of MFML, was introduced to better understand the contribution of each fidelity to the overall model accuracy. Inferences were drawn from this error metric which led to the development of the Γ-curve which demonstrated that high accuracy MF models can be built with low computational cost with as little as 2 training samples at the costliest fidelity. Finally, in the previous chapter practical applications of the MF methods developed in this dissertation were discussed showing how effective these methods really are in application to large systems.

12.2 Outlook

Several areas of future research can be branched off of this dissertation. Future work in MFML should address the challenges of spanning the typically wide conformation space by employing active learning techniques [25, 26, 27]. Spanning the conformation space with viable cost is a challenge that is not specific to MFML. If successful, integration of active learning strategies [25, 26] with MFML can greatly improve the capabilities over the method established in this work. This will allow the MFML models to also predict rare events, which might not be part of the usual simulated trajectory. Overall, an enhanced sampling of the conformation space would provide a great improvement for large scale systems of interest such as light-harvesting systems. Similarly, active learning can certainly be of use for a good transferability of MFML models across different molecules.

The o-MFML method introduced in Chapter 6 opens up further research avenues for MF methods in QC. For instance, the coefficients can be used as an indicator of the number of training samples to choose at each fidelity. Active learning strategies used in tandem with o-MFML could lead to large benefits in reducing the numerical costs for generating training data. For o-MFML, the optimization procedure and the choice of a validation set

implicitly decide the accuracy of the final model. A refined choice of the validation set could improve the results and provide a better optimized model even for the non-nested case. One example with QeMFi for instance is as follows: choosing validation set geometries corresponding to the largest molecule of the QeMFi dataset, o-HDBI while the sub-models are trained on the other molecules. There is indeed a wide range of modifications that can be made to fully and comprehensively assess each caveat of o-MFML. Another area of investigation with o-MFML could be the use of alternative optimization functions to compute the optimal coefficients. A potential candidate in this case is to carry out the optimization procedure in the RKHS as opposed to using a validation set as is done in this dissertation.

There is increasing interest in the applications of the ML-QC pipeline to vector QC properties. The application of o-MFML in its nested and non-nested configurations to vector properties such as molecular dipole moments and molecular forces is another interesting area of future work. As discussed in Chapter 5, the MFML approach can in principle be applied to any ML method that shows a systematic prediction error behavior. Of great interest will be its application to more recent NN approaches [22, 191, 204]. These would further enable the use of MFML for vector QC properties.

Since the QeMFi dataset from Chapter 7 contains the molecular geometries of the diverse molecules, yet another potential future outlook of this work is the use of MF data generated by other QC method such as HF, MP2, or even CASPT2. While this computation lies beyond the scope of the current work, the study on the effectiveness of MFML and o-MFML for fidelities based on different QC theory methods as opposed to different basis sets could provide interesting results for the QC community. A primer to this was already presented in Chapter 6 and Chapter 11 with several QC theory methods being combined.

Overall the work presented in this dissertation is a strong step in the direction of reducing the cost of generating training data for the use of ML in QC. Systematic reduction of error with the addition of cheaper fidelity was observed for diverse QC properties. With the application of Γ-curve MFML, it was seen that one could potentially further reduce the overall cost of generating training data for a multifidelity ML model. This dissertation has rigorously developed and tested several multifidelity methods, each with its own benefits to the realm of QC. This work has shown without a doubt that the use of such multifidelity methods in ML-QC does indeed reduce to the cost of research by mitigating the amount of compute resources spent on generating training data for ML models.

Part IV

Appendices and References

APPENDIX A

APPENDIX A - SUPPLEMENTARY RESULTS

This appendix contains supplementary information (SI) files of the works presented in this dissertation. These are adapted from the supplementary information of respective publications as indicated at the beginning of each section. These results are intended to go along with the Chapters of the main sections of this dissertation.

A.1 Additional Details and Analysis for Arenes

Adapted from the SI file of [142].

A.1.1 Generating training and evaluation data

As discussed in section 5.1, trajectories with 15000 conformations and a time step of 1 fs were generated for benzene, naphthalene, and anthracene using the MD and DFTB approaches. Starting from these conformations, training sets and evaluation sets were generated. While the first $N_{\text{train}} = 1.5 \cdot 2^{13} = 12288$ frames of the trajectory were used for training data generation, the last $N_{\text{eval}} = 2712$ conformations at a time step of 1 fs were used as the evaluation set $\mathcal{V}^F := \{(\boldsymbol{X}_q^{\text{ref}}, E_q^{\text{ref}})\}_{q=1}^{N_{\text{eval}}}$. Thereby, it is assured that the calculated prediction errors indeed reflect the generalization properties of the models on a later, unseen part of the trajectory.

A pool of training data for each fidelity was generated. This data pool is structured as follows: on the target fidelity, that is the TZVP level of theory, $1.5 \cdot 2^9 = 768$ excitation ener-

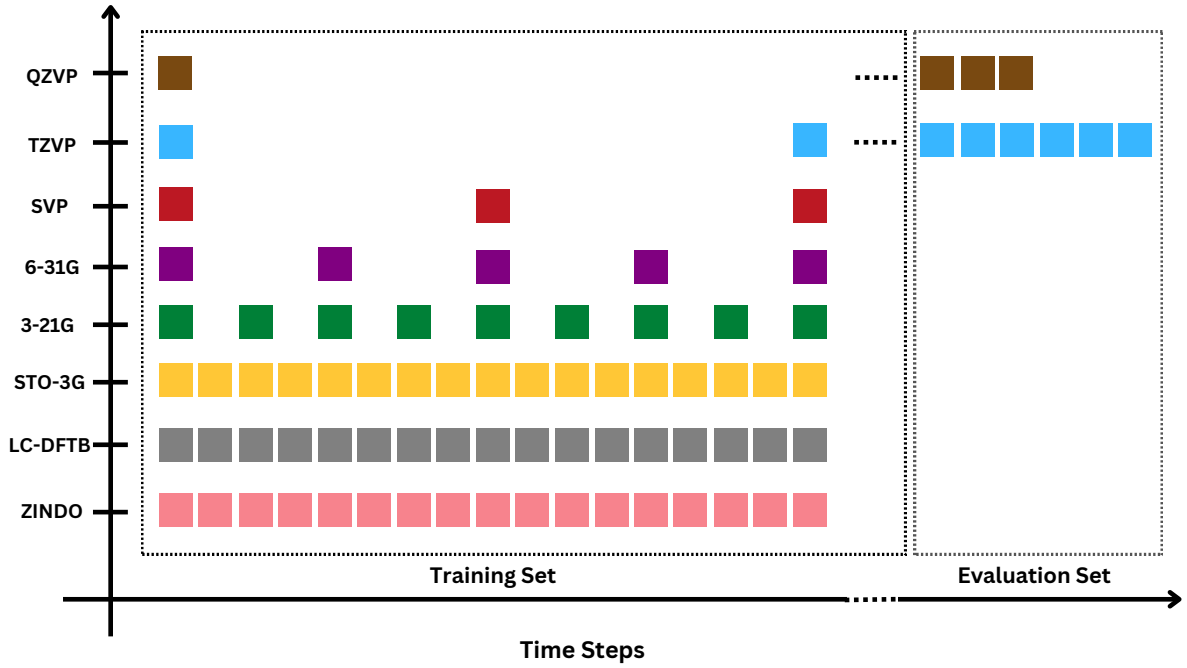


Figure A.1: The structure of the training and evaluation data set for the benzene molecule of both MD and DFTB-based trajectories. The STO-3G level of theory was sampled using a 1fs time step. The 3-21G level of theory was sampled using a 2fs time step. Each subsequently costlier level of quantum chemistry was sampled using a time step increased by a factor of 2. Thus, the TZVP level was sampled using a time step of 16fs.

gies were determined. The factor of 1.5 ensures that the training data is sufficiently different for each random shuffling needed in the model evaluation. For each subsequent lower fidelity, this number is scaled by a factor of 2 thus resulting in $1.5 \cdot 2^{13} = 12288$ excitation energy calculations at the lowest fidelity, that is STO-3G. The time step between the chosen conformations for the training data pool is increased by a factor of two for growing fidelity. That is, the STO-3G fidelity was calculated for the training trajectory using a time step of 1 fs. This resulted in 12288 excitation energies at STO-3G fidelity[1]. The next fidelity, that is 3-21G, was calculated along the training trajectory using a 2 fs time step resulting in 6144 calculations. At each subsequent fidelity, the time step was scaled upward by a factor of two. This results in 768 calculations at the TZVP fidelity with a time difference of 16 fs. Addition-

[1]This same time step was employed for the LC-DFTB and ZINDO fidelities in the case of Benzene.

ally, for benzene, the QZVP level was calculated at a time difference of 32 fs resulting in 384 calculations. Figure A.1 sketches the chosen approach. This process of generating the training data across the fidelities guarantees that the data pool contains data that is evenly distributed over the trajectory. The reader is reminded that the cheaper fidelities such as STO-3G or ZINDO are only used to enrich the final MFML model. These cheaper fidelities are not the target of prediction. Rather, the most expensive fidelities, TZVP or QZVP are the target fidelities.

Table A.1 lists the averaged single point calculation times for each fidelity for the different molecules involved in this study. These are calculated as an average of the time taken to calculate the excitation energies to the first excited state at a fidelity for the MD and DFTB-based trajectories. All results in this study were obtained on single cores (serial execution) on Intel Xeon E5-2640 2.50 GHz CPUs with 32 GB RAM running openSUSE Linux.

Molecule/Fidelity	**Benzene**	**Naphthalene**	**Anthracene**
QZVP	20.00	-	-
TZVP	3.53	12.46	31.10
SVP	1.02	3.10	8.00
6-31G	0.65	2.06	5.04
3-21G	0.43	1.25	2.85
STO-3G	0.44	1.16	2.35
LC-DFTB	0.26	-	-
ZINDO	0.02	-	-

Table A.1: Average computational times of point calculations in minutes for the excitation energies at different fidelities. The time is calculated as the average over the two trajectory types and as an average over the total number of frames.

A.1.2 ML details for the prediction of excitation energies

Beyond the Coulomb Matrices (CM) [29, 44, 191, 49, 125, 166] used in this study, various molecular descriptors or representations have previously been used in machine learning (ML) for quantum chemistry calculations. These include inverted distance matrices, Bag of Bonds (BoB) [193, 194, 192], neural networks [191, 199, 205], Smooth Overlap of Atomic Positions (SOAP) [195, 192], various *ad hoc* descriptors [201, 202], and the Faber-Christensen-Huang-Lilienfeld (FCHL) representation [196, 176, 49, 44]. While FCHL is a superior representation, it is observed that the time to generate the representation and the kernel corresponding to this representation for Kernel Ridge Regression increases strongly with the number of atoms in the molecule. Thus, it may happen that the time for predictions us-

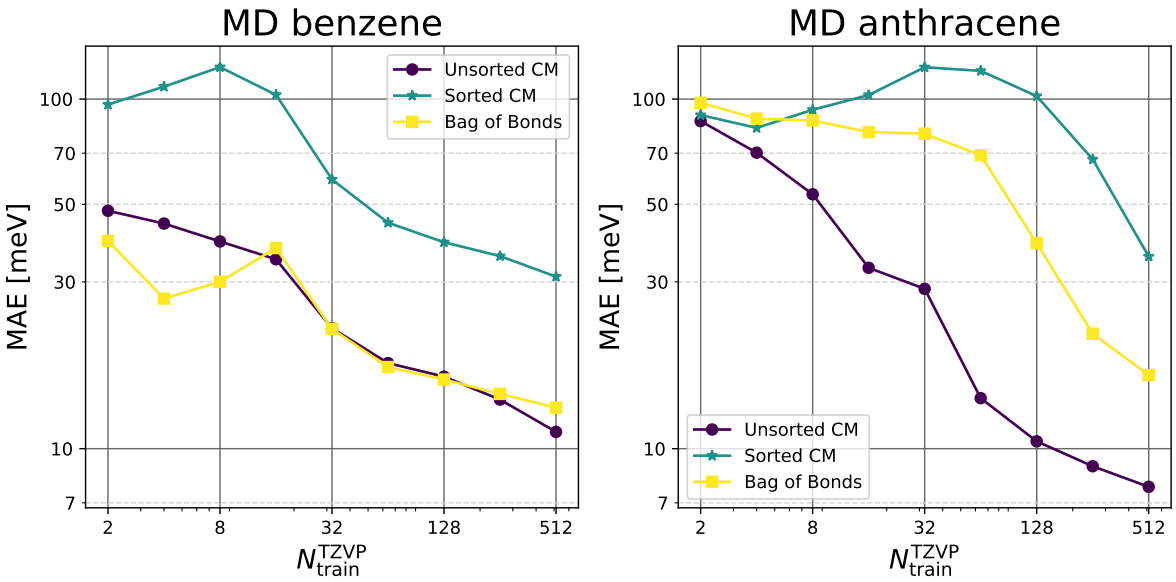


Figure A.2: Comparison of sorted and unsorted CM representations and the related BoB representation for MD based benzene and anthracene. The learning curves are built for a single fidelity KRR model with the TZVP fidelity used as both training and target. The results are shown here for a 10-run average. For both the cases, the unsorted CM representation outperforms the sorted CM and BoB.

ing ML using the FCHL representation far exceeds the cost of the conventional methods in quantum chemistry [196, 65, 50]. Therefore, this superior representation is not used, here. Instead, unsorted Coulomb matrices are used, as argued in Chapter 5. Figure A.2 shows that unsorted Coulomb matrices indeed outperform the sorted CM representation and also the BoB representation.

For a given training set $\mathcal{T} := \{(\boldsymbol{X}_i, E_i)\}_{i=1}^{N_{\mathrm{train}}}$, the coefficients of KRR, $\boldsymbol{\alpha}$, are calculated by solving the minimization problem given by

$$\boldsymbol{\alpha} = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{N_{\mathrm{train}}} \left(P_{\mathrm{KRR}}\left(\boldsymbol{X}_i\right) - E_i\right)^2 + \lambda \boldsymbol{\theta}^T \boldsymbol{K} \boldsymbol{\theta}\,, \tag{A.1}$$

where $\boldsymbol{K} = \left(k(\boldsymbol{X}_i, \boldsymbol{X}_j)\right)_{i,j=1}^{N_{\mathrm{train}}}$ is called the kernel matrix and λ is a Lavrentiev regularization parameter that is usually chosen to be small. As shown elsewhere [53, 167, 44, 63], the minimizer is found by solving the linear system of equations

$$\left(\boldsymbol{K} + \lambda \boldsymbol{I}\right) \boldsymbol{\alpha} = \boldsymbol{y}\,. \tag{A.2}$$

In the present study, the entire process of KRR including kernel generation was performed using the `qml` package [66] which employs a Cholesky decomposition to solve Eq. (A.2).

The kernels, kernel widths, and the regularization parameter are hyperparameters that can be optimized using a grid search or cross-validation techniques [160, 167, 254, 125]. In this research, however, the kernel width for the Matérn kernel has been converged manually, see Table A.2, and the regularization parameter is set to $\lambda = 10^{-9}$.

Molecule	Benzene	Naphthalene	Anthracene
MD	715.0	1300.0	2455.0
DFTB	940.0	1200.0	2200.0

Table A.2: Manually converged kernel widths of the utilized Matérn kernel for the various data sets employed in this study.

A.1.3 Supplementary results for MFML

In this section, supplementary figures and results for Chapter 5 are presented and discussed. Figures A.3 and A.5 give the preliminary data analysis and multifidelity results for the DFTB trajectories. Both have been discussed in Chapter 5. Adding up to this discussion, a particular focus is here on the multifidelity results for anthracene based on a DFTB trajectory, additional scatter plots for prediction errors across fidelities and a study that shows the impact of using several fidelities instead of two fidelities in MFML. For comparison to the centered energies used in this work, the distribution of the uncentered energies of the MD and DFTB trajectories are shown in Figure A.4.

A.1.4 Further discussion of DFTB-based anthracene

For DFTB-based anthracene, the learning curve corresponding to $P_{\text{MFML}}^{(\text{TZVP;STO-3G})}$ shows poor improvement in comparison to that of the previous model. This is especially true for smaller training sizes. This behavior of the $P_{\text{MFML}}^{(\text{TZVP;STO-3G})}$ model follows from strong variance in the STO-3G results when compared to the target fidelity TZVP, as seen in row B of Figure A.3. To make clearer the reason for such behavior in the model, the error in prediction from the ground up by building models for the MD trajectory and the DFTB trajectory is investigated.

Figure A.6 shows the absolute differences $|\epsilon|$ of the standard KRR model $P_{\text{KRR}}^{(3-21\text{G})}$ and the model $P_{\text{KRR}}^{(\text{TZVP;STO-3G})} + P_{\text{KRR}}^{(\text{STO-3G,3-21G})}$ compared to the reference TZVP values. These errors have been evaluated on the first 200 conformations of the respective evaluation set. The reason these two very specific models are picked, is justified as follows: the learning curve indicates that it is the addition of the STO-3G fidelity, which negatively affects the

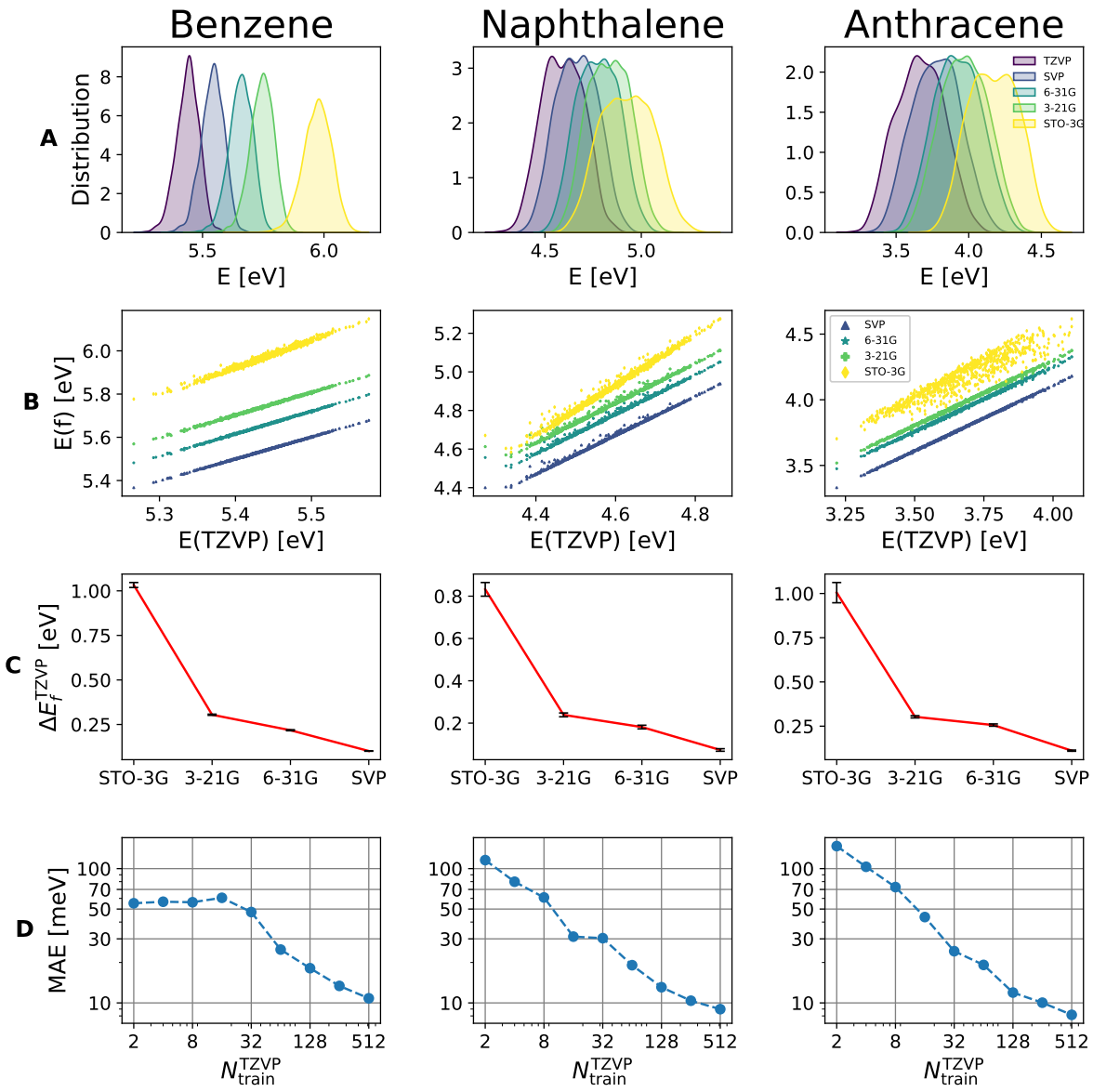


Figure A.3: A) Energy distributions of the different fidelities (basis sets) in the training sets for DFTB trajectory of benzene, naphthalene, and anthracene. The complete training data is represented here in terms of the kernel density plot. B) Comparing TZVP to the other fidelities of the training data for benzene, naphthalene, and anthracene. For this scatter plot, only those molecular conformations have been considered, which have been evaluated at the TZVP fidelity. C) Energy difference between the fidelities and target fidelity in the training set. D) Single-Fidelity model learning curves for the molecules presented with a double-logarithmic scale.

model. The concept of multifidelity machine learning (MFML) implies that the error from both these models should be similar. Failure to do so indicates an unsuitable nature of the

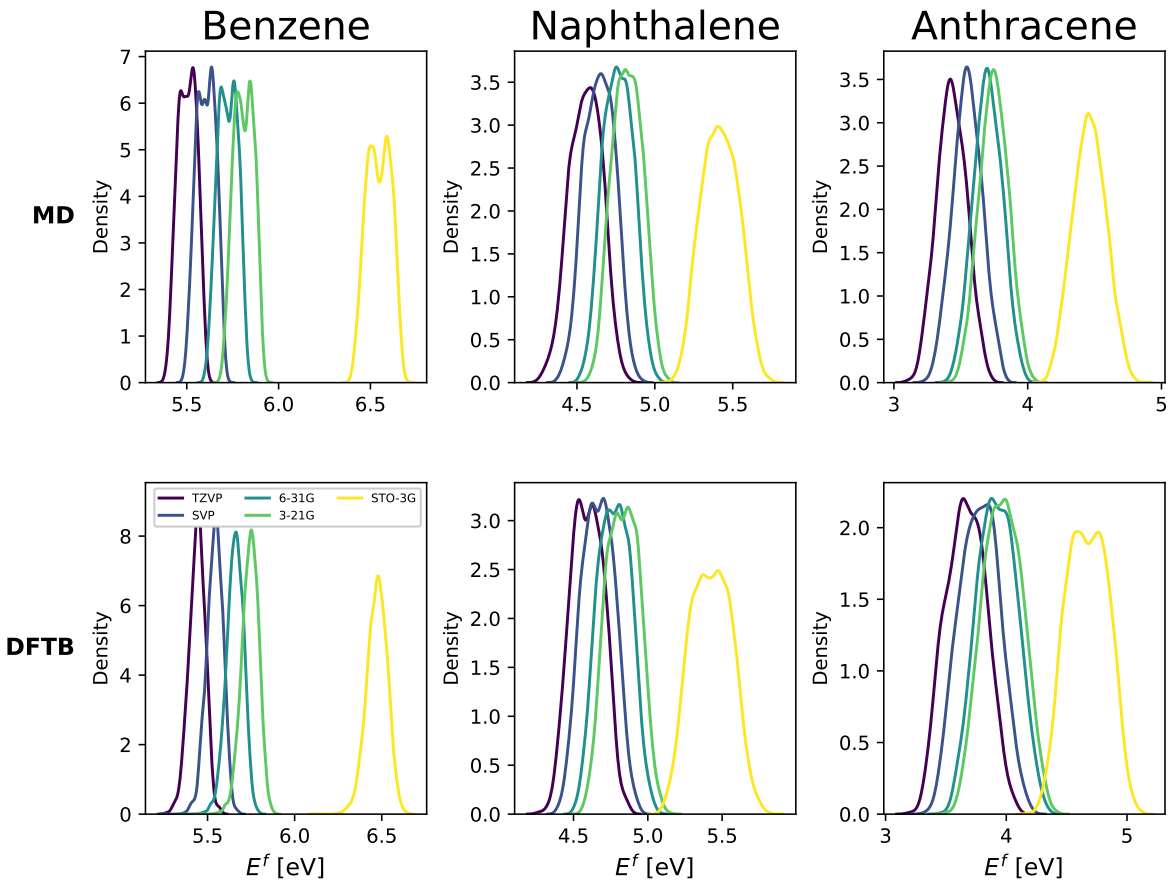


Figure A.4: Uncentered energy distributions of the different fidelities (basis sets) of the training set for the MD and DFTB trajectories of benzene, naphthalene, and anthracene.

data used at the STO-3G fidelity. These models are picked for $N_{\text{train}}^{\text{TZVP}} = 2^4$, since the learning curves for DFTB-based anthracene in row A of Figure A.5 shows sufficient difference between the different models and thereby any adverse effect arising due to the addition of the STO-3G fidelity will be more pronounced. For DFTB-based anthracene, one observes that the errors of the two models are far more spread out than for MD-based anthracene. Overall, the addition of data from STO-3G produces a large fluctuation of error in this molecule. For MD-based anthracene, on the other hand, in agreement with the observations from row B of Figure 5.2, we see that the two models produce errors with lower fluctuation between them. This indicates that the training data for DFTB-based anthracene is anomalous at the STO-3G fidelity if targeting the TZVP fidelity.

It should be noted that the learning curves for the multifidelity models built for DFTB anthracene excluding the STO-3G fidelity, in fact, follow the expectation of the approach as is evident in the left-hand side pane of row A in Figure A.5. The averaged MAE for $P_{\text{KRR}}^{(\text{TZVP})}$

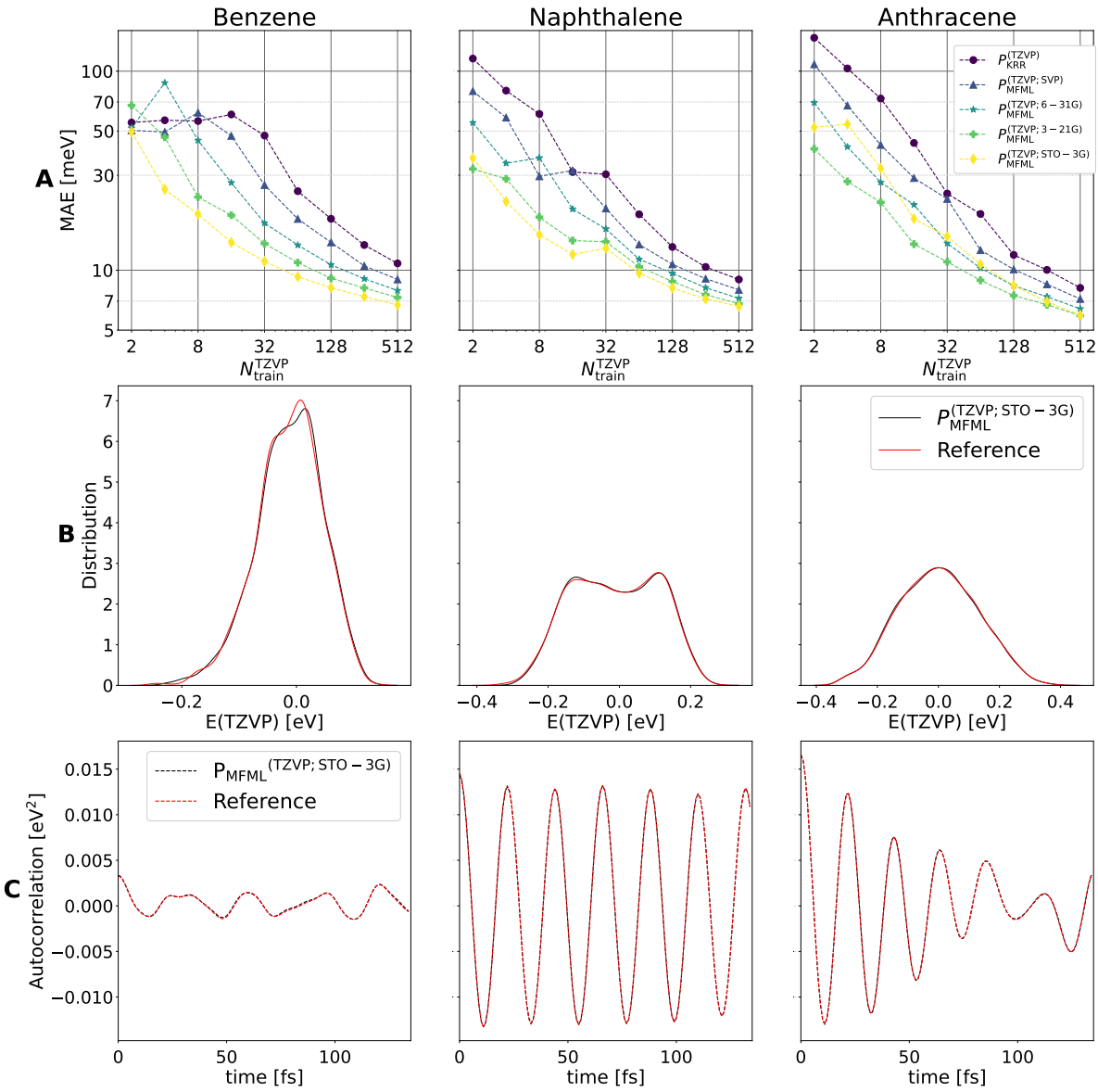


Figure A.5: Results of the MFML method for the DFTB-based trajectory of benzene, naphthalene, and anthracene. A) The learning curves associated with the MFML method are depicted with the standard KRR model represented in blue and the MFML model built with the lowest fidelity depicted in yellow. B) Energy distributions based on the holdout sets using the TZVP reference calculations (red) and the predictions from $P_{\text{MFML}}^{(\text{TZVP;STO-3G})}$ (black) for $N_{\text{train}}^{\text{TZVP}} = 512$ are shown. For all molecules, it can be observed that the predictions from the MFML model matches the reference energy distributions more accurately. C) Time autocorrelation functions (ACF) of the excitation energies. The red lines correspond to the ACF of the TZVP reference calculations from the holdout evaluation set. The black lines report the ACF of the excitation energies predicted from the MFML model for the conformations belonging to this evaluation set.

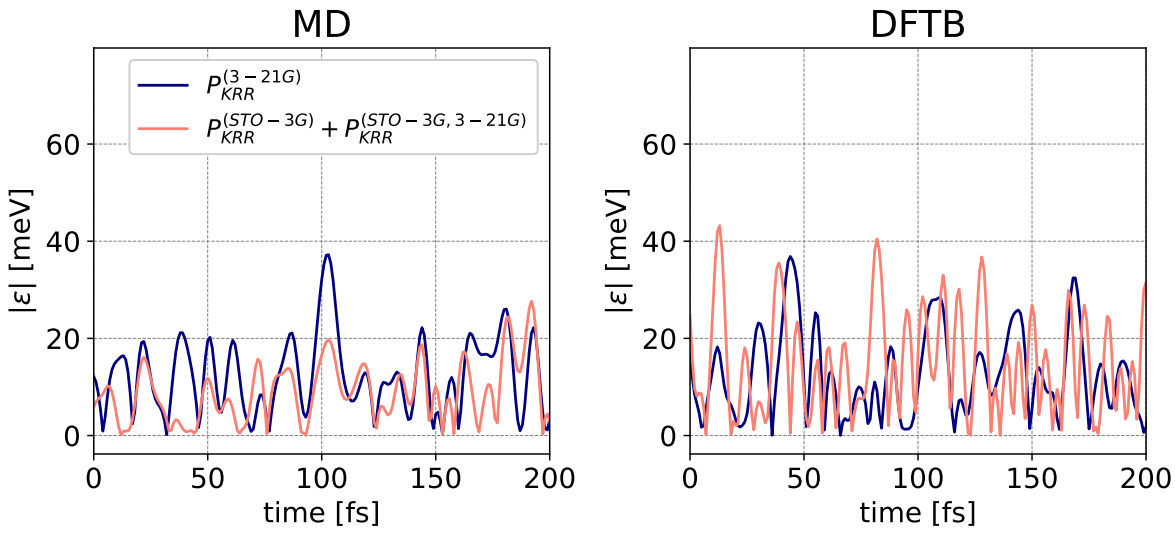


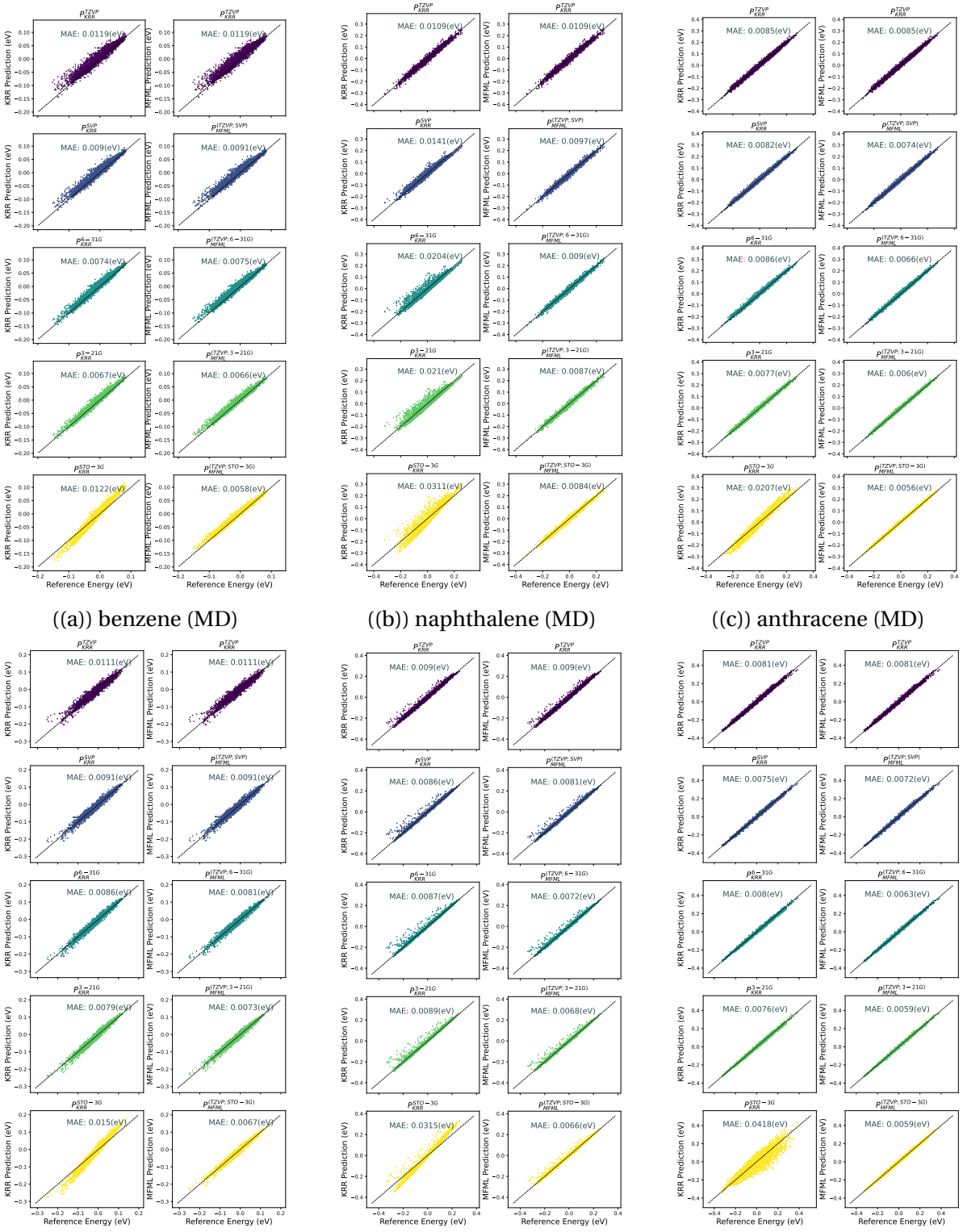
Figure A.6: Comparison of error of models for anthracene based on MD and DFTB trajectories to understand the effect of adding STO-3G. In particular, errors with respect to the target fidelity TZVP for the conventional KRR model built on the 3-21G fidelity (navy) and the model that consists of the conventional KRR model built with the STO-3G fidelity and the KRR model built on the difference between STO-3G and 3-21G fidelities (salmon) are compared. One observes that the two models are in reasonable agreement for MD-based anthracene, while there is a large difference for DFTB-based anthracene.

is 8.1 meV and is 5.9 meV for $P_{\text{MFML}}^{(\text{TZVP};3-21\text{G})}$. The multifidelity approach works as long as the required data structure is satisfied. The reader is reminded again that the robustness of the method succeeds to lower the offset of the $P_{\text{MFML}}^{(\text{TZVP};\text{STO}-3\text{G})}$ model for larger training set sizes.

A.1.5 Scatter plots of the excitation energies

To extend on the discussion of the prediction quality of the multifidelity model, Figure A.7 give scatter plots for each of the sub-models and the joint multifidelity models for the different molecules and trajectories. The plots correspond to $N_{\text{train}}^{\text{TZVP}} = 2^9$ scaled across the fidelities as explained in Section 4. Each model is built with the same randomly shuffled nested structure of the training data. For each plot, the first column corresponds to the conventional KRR models $P_{\text{KRR}}^{(\text{f})}$ built at some fidelity f. The second column corresponds to the different multifidelity models $P_{\text{MFML}}^{(\text{TZVP};\text{f})}$ built with same the baseline fidelity f_b.

One way to read these results is to begin at the bottom left-hand side of each individual plot, for example, MD-based anthracene in Figure A.7(c). This scatter plot comes from the predictions of the conventional model built on the STO-3G fidelity, when compared with the reference TZVP energies on the evaluation set. Next, moving up one cell to the scat-



((a)) benzene (MD) ((b)) naphthalene (MD) ((c)) anthracene (MD)

((d)) benzene (DFTB) ((e)) naphthalene (DFTB) ((f)) anthracene (DFTB)

Figure A.7: Scatter plots between the target fidelity, TZVP, and predictions by the single-fidelity models $P^{(f_b)}_{\mathrm{KRR}}$ (left column) and predictions by the MFML models $P^{(\mathrm{TZVP;f_b})}_{\mathrm{MFML}}$ (right column) are reported for each molecule.

ter plot corresponding to the predictions of the 3-21G conventional KRR model, predictions come closer to the reference TZVP data, i.e., one observes a numerically lower MAE. This trend continues while further moving up the left column. The top cell indicates the deviation between the TZVP reference energies and those predicted by the conventional KRR model built on the TZVP fidelity, that is $P_{\mathrm{KRR}}^{\mathrm{TZVP}}$. Obviously, this model approximates the TZVP data in the best way for the various KRR models shown. In the second column, as discussed before, the predictions by the various MFML models are studied for benzene based on an MD trajectory. At the top, the same scatter plot as for $P_{\mathrm{KRR}}^{(\mathrm{TZVP})}$ is provided, since the MFML model for the TZVP data and baseline fidelity TZVP is just the single-fidelity model. One plot further down gives the MFML model that additionally uses training data at the SVP fidelity. While the individual use of the SVP data, see left column, was still suboptimal in order to predict the TZVP data, the addition of this cheaper to compute data to the MFML model depicted on the right-hand side improves the accuracy of the MFML model. This trend continues when further moving down the results. Hence, when adding more and more lower levels of fidelity to the multifidelity model, the results are consistently improved.

The observations of the learning curves and the preliminary data analysis appear evidently in these scatter plots. Specifically, the scatter plots for naphthalene with the MD trajectory and anthracene with the DFTB trajectory show how the model $P_{\mathrm{KRR}}^{(\mathrm{STO-3G})}$ results in a wider spread due to the anomalies in the training data available at that fidelity. Considering the scatter plots for DFTB-based anthracene from Figure A.7(f), the results corresponding to $P_{\mathrm{KRR}}^{(\mathrm{STO-3G})}$ shows a large spread of the predictions, especially for the higher values. This is in strong contrast to the scatter plots resulting from the other single fidelity KRR models that one observes as one goes up the column. It is a clear indicator that the spread of the data observed in Figure A.3B results in a poorly trained model with the addition of the STO-3G fidelity. It is important to note again that the MFML models built without the STO-3G provide a very distinct reduction in the error, as seen in the second column. The robustness of the multifidelity becomes all the clearer, when one observes the results for the $P_{\mathrm{MFML}}^{(\mathrm{TZVP;STO-3G})}$, where the reported MAE is in fact similar to that for $P_{\mathrm{MFML}}^{(\mathrm{TZVP;3-21G})}$. This could indicate that the multifidelity method still improves when trained on the difference between these fidelities.

A.1.6 Impact of the use of several fidelities in MFML

To underline the positive impact of the use of more than two fidelities in the MFML method, three models are studied - a reference KRR model $P_{\mathrm{KRR}}^{\mathrm{TZVP}}$, an MFML model built with the

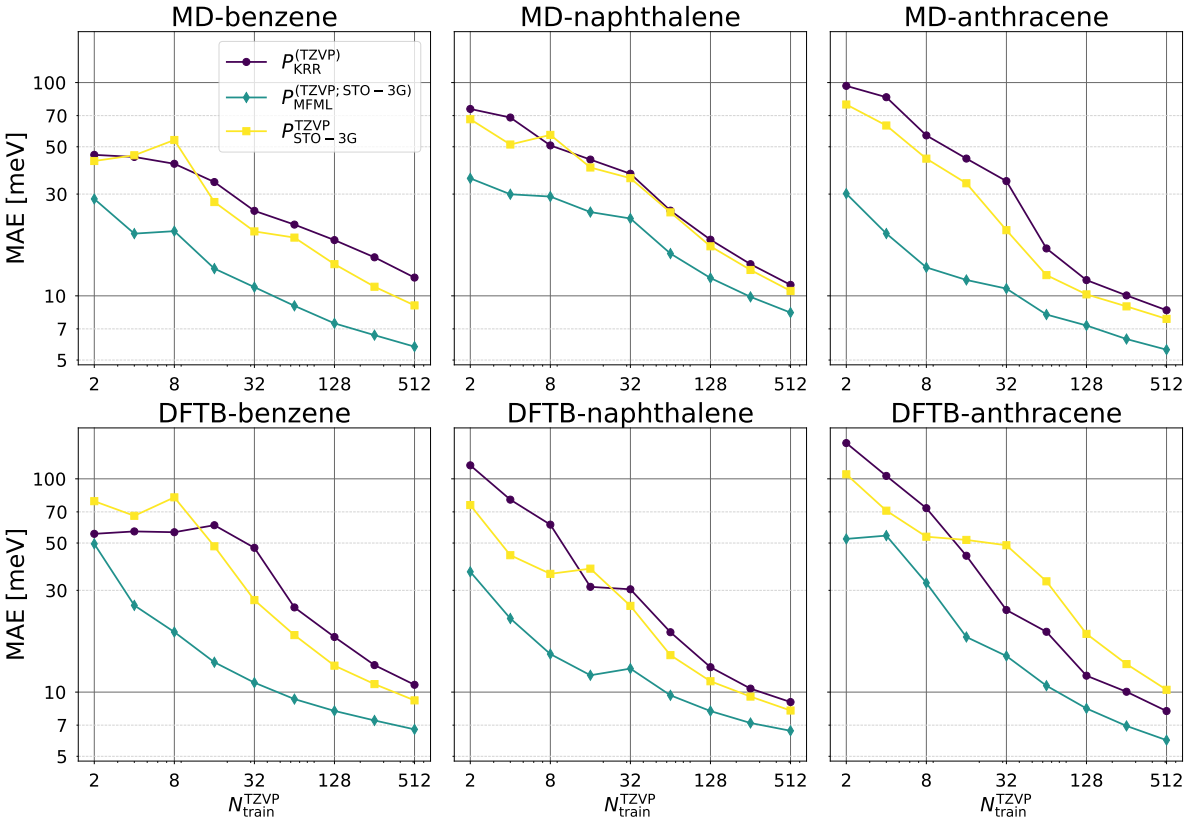


Figure A.8: Comparing the MFML and two-level ML models to motivate the need for a multifidelity training structure. The reference KRR model is also provided.

baseline fidelity of STO-3G and several intermediate fidelities, that is $P^{(\mathrm{TZVP;STO-3G})}_{\mathrm{MFML}}$, and finally, a model $P^{\mathrm{TZVP}}_{\mathrm{STO-3G}}$ which is built with the same principle as the MFML method but which assumes that the fidelity structure consists only of TZVP and STO-3G fidelity. The prediction of this model for a query representation, $\boldsymbol{X}_q$, is given as:

$$P^{\mathrm{TZVP}}_{\mathrm{STO-3G}}\left(\boldsymbol{X}_q\right) := P^{\mathrm{STO-3G}}_{\mathrm{KRR}}(\boldsymbol{X}_q) + P^{(\mathrm{STO-3G,TZVP})}_{\mathrm{KRR}}\left(\boldsymbol{X}_q\right) \ ,$$

with

$$P^{(\mathrm{STO-3G,TZVP})}_{\mathrm{KRR}}\left(\boldsymbol{X}_q\right) := \sum_{i=1}^{N^{TZVP}_{\mathrm{train}}} \alpha^{(\mathrm{STO-3G,TZVP})}_i k\left(\boldsymbol{X}_i, \boldsymbol{X}_q\right) \ .$$

Here, the $\boldsymbol{\alpha}^{(\mathrm{STO-3G,TZVP})}$ are KRR coefficients which are derived by solving the usual KRR problem by using the difference of the energies at TZVP and STO-3G levels of theory.

The results of this experiment are shown in Figure A.8 as learning curves for all the molecules used in this study. The results across the molecules show that the MFML model outperforms the two-level method significantly. The one case, in which the two-level method is even worse than the single-fidelity model is DFB-based anthracene, where the use of

STO-3G had a strong negative impact, anyway. Overall, these results are a strong motivation to use several levels for the training.

A.2 Ordinary Least Squares

OLS is a popular ML method to solve linear regression problems. This primer is presented here for completeness of the discussion for o-MFML. Consider a training set of size N that consists of a D-dimensional features $\boldsymbol{x}_i \in \mathbb{R}^D$ and corresponding 1-D outputs, $y_i \in \mathbb{R}$. The aim of OLS is to solve for the weights, $\boldsymbol{\omega} \in \mathbb{R}^D$, in the regression equation $y_i = \boldsymbol{\omega}^T \boldsymbol{x}_i$ for $i = \{1, 2, \ldots, N\}$. This is equivalent to minimizing the loss function given by:

$$L(\boldsymbol{\omega}) = \left(\boldsymbol{X}\boldsymbol{\omega} - \boldsymbol{y}\right)^T \left(\boldsymbol{X}\boldsymbol{\omega} - \boldsymbol{y}\right) = \left\|\boldsymbol{X}\boldsymbol{\omega} - \boldsymbol{y}\right\|_2^2 , \tag{A.3}$$

where, $\boldsymbol{y}$ is simply the vector of the outputs from the training set, and the matrix $\boldsymbol{X}$ is commonly referred to as the *design matrix* with $(\boldsymbol{X})_{i,j} = x_{i,j}$ with $i = \{1, 2, \ldots, N\}$ and $j = \{1, 2, \ldots, D\}$. The solution to this problem can be arrived at by taking the gradient of Eq. (A.3) with respect to $\boldsymbol{\omega}$ and setting it equal to 0. This readily reduces to solving the following system of equations:

$$\left(\boldsymbol{X}^T \boldsymbol{X}\right) \boldsymbol{\omega}_{OLS} = \boldsymbol{X}^T \boldsymbol{y} . \tag{A.4}$$

Here, $\boldsymbol{\omega}_{OLS}$ denotes the OLS weights of the regression problem. The Scikit-learn package [214] is used to run the OLS optimization procedure in this work.

A.3 Supplementary Results for o-MFML

Adapted from the SI file of [143].

A.3.1 Comparison of difference in data and difference in model implementation of MFML

As discussed in Chapter 6, for the MFML model with the KRR as a choice of ML method, one can show that the difference in data approach, $P_{\text{KRR}}^{(f,f+1)}$, is equivalent to the difference in model approach, $P_{\text{KRR}}^{(f+1)} - P_{\text{KRR}}^{(f)}$. Theoretically, Ref. [32] has already shown that the data-difference MFML is equivalent to the model-difference MFML for Kernel Ridge Regression (KRR). In order to numerically establish this, the following experiment was carried out for

the first excitation energy multifidelity data of arenes from Chapter 5. The entire multifidelity structure of the training set was used for each molecule. This amounts to $1.5 \cdot 2^9 = 768$ training samples at the highest fidelity with the subsequent lower fidelities being scaled with a factor of 2 resulting in $1.5 \cdot 2^{13} = 12288$ samples at the baseline fidelity. For this model setup, the data-difference MFML model was built as presented in Chapter 5. The model-difference MFML model was built as discussed section 4.1. The predictions of these two models on the same holdout evaluation set was performed and the resulting MAEs are reported in Table. A.3. The equivalence of the two models is now numerically evident with the difference in MAE in all cases being smaller than the order of 10^{-7}.

Molecule	Model MFML	Data difference MFML	Abs. difference
MD benzene	0.00556086	0.0055611	$2.465 \cdot 10^{-7}$
MD naphthalene	0.0080742	0.00807421	$7.15 \cdot 10^{-9}$
MD anthracene	0.00527777	0.00527759	$1.758 \cdot 10^{-7}$
DFTB benzene	0.00624935	0.00624899	$3.558 \cdot 10^{-7}$
DFTB naphthalene	0.00617296	0.00617296	$9.708 \cdot 10^{-10}$
DFTB anthracene	0.00554352	0.00554331	$2.114 \cdot 10^{-7}$

Table A.3: MAE (eV) of MFML for the prediction of the first excitation energy. The first column corresponds to the model built with the difference taken for different sub-models. The second column correspond to taking the difference in the data and building the MFML on these differences. The last column reports the absolute difference in the MAE for these two methods.

A.3.2 Generalization capabilities of the o-MFML for atomization energies

As an additional investigation into the effectiveness of the o-MFML model, the kernel density plots of the atomization energies at the target fidelity, that is CCSD(T)–cc-pVDZ, are presented in Figure A.9. The plots show the atomization energies from the training set used in the multifidelity model, the validation set used for the o-MFML method, and the common test set. In addition, the kernel density plots of the predicted atomization energies from the MFML and o-MFML methods for the MP2-STO3G baseline with $N_{\text{train}}^{\text{CCSD(T)}-\text{cc-pVDZ}} = 128$ are shown. The left panel of the figure shows the kernel density plots for the entire range of the atomization energies of the CCSD(T)–cc-pVDZ fidelity. The right panel zooms into the range from 50 to 140 kcal/mol and the corresponding density values. In that panel, it can be seen that both the training and validation set peak around 0 kcal/mol with a slight skew towards the negative values. The predictions and the training set density curves show

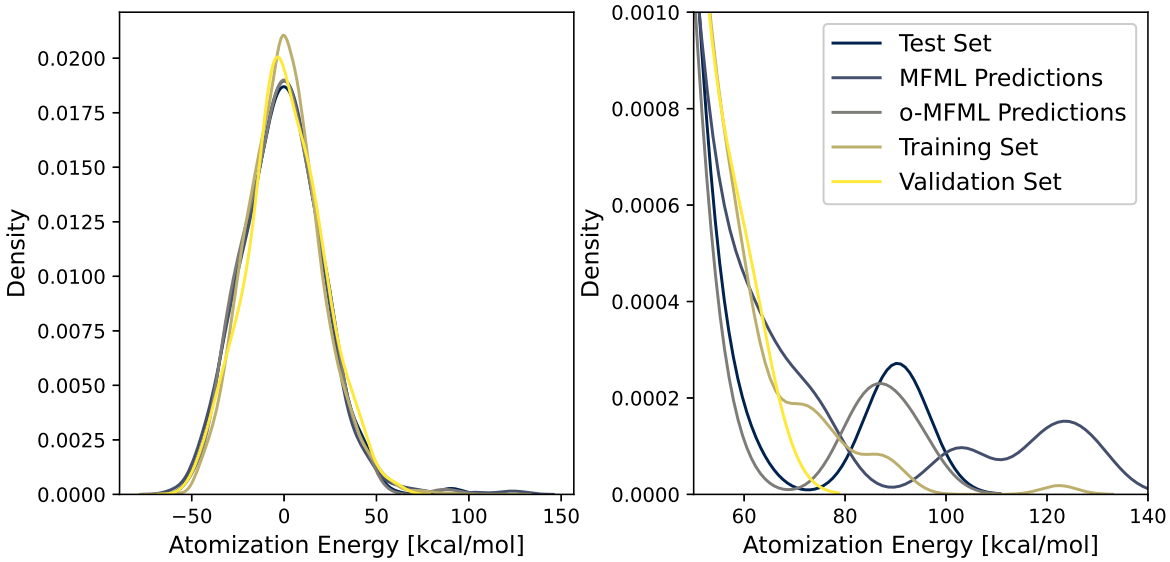


Figure A.9: Kernel density plots of the various data splits for atomization energies of the QM7b dataset. Only the CCSD(T)–cc-pVDZ fidelity is depicted. The left-hand side of the figure shows the density plots for the entire range of the atomization energies while the second plot zooms in to the range of 50-140 kcal/mol for better visualization of the outlying cases.

good agreement close to the peak at 0 kcal/mol. There appears to be an almost complete agreement in the predictions of the MFML and o-MFML methods around this region. However, there is a small deviation in the curve of the MFML method from the test set curve in the negative regime. The two methods diverge at around the 50 kcal/mol with a more pronounced difference. This finding is in agreement with the scatter plot observed in Figure 6.1.

Considering the various curves in the right panel of Figure A.9, the curve corresponding to the training set shows no specific characteristics in this region, implying the lack of specific structure which might be statistically useful for an ML model to generalize in this region. The curve of the validation set shows no presence in the region after 80 kcal/mol. Next, the test set shows a distinctive peak around 90 kcal/mol with almost complete dips in the immediate vicinity of the peak. The difficulty of learning such a niche becomes evident in the curve corresponding to the MFML method. The curve of the predicted atomization energies deviates almost entirely from that of the test set from 55 kcal/mol and overestimates the atomization energies in the range of 55-140 kcal/mol. The o-MFML, however, is able to reproduce the peak with good accuracy in addition to being able to truthfully replicate the dips of the curve.

This result is a strong indicator about the capabilities of the o-MFML method in being able to generalize over unseen data by optimally combining the information from the cheaper fidelities which allow it to reproduce the peak. Thus, the method is more capable of predicting the atomization energies at the higher ranges from 50 kcal and beyond. The optimal combination of the sub-models results in a final multifidelity model which shows a strong capability to generalize prediction across the ranges of the atomization energies. This makes a strong argument in favor of the transferable nature and generalization capacity of the multifidelity method.

A.3.3 Coefficient analysis of o-MFML

A.3.3.1 Atomization energies

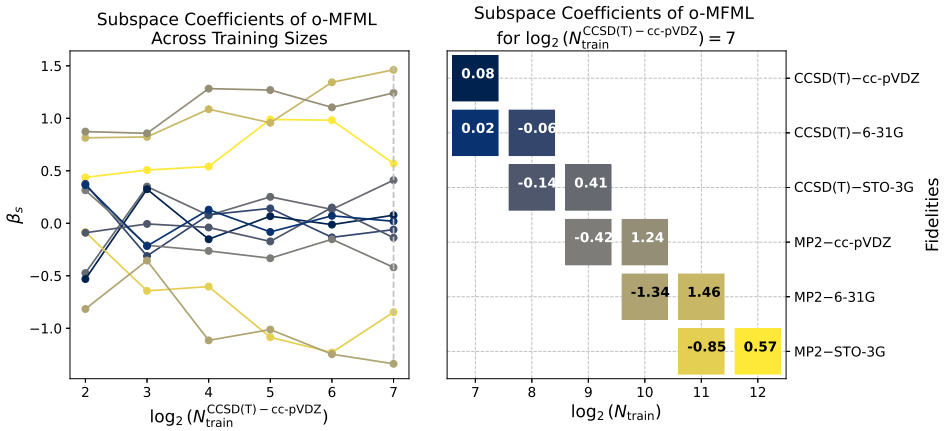


Figure A.10: Evolution of β_s^{opt} for o-MFML for the QM7b dataset. The evolution is shown across training size of CCSD(T)–cc-pVDZ and the final state of the coefficients for $N_{\text{train}}^{\text{CCSD(T)}-\text{cc-pVDZ}} = 2^8 = 128$ are shown on the right hand side which is the same as shown in Figure 6.4.

As was discussed in section 6.2.1.1, the analysis of the coefficients allowed one to completely eliminate the CCSD(T)–6-31G fidelity and still arrive at a model with similar accuracy of prediction. While this was only shown for the model built with 128 training samples at CCSD(T)–6-31G, it will be useful only if this same idea can be established for smaller training set sizes. This imposition will allow users to test with very small number of QC calculations whether the use of a certain fidelity will improve the model. This will reduce redundancy of calculation. In Figure A.10, the evolution of the coefficients β_s^{opt} is shown for varying training sizes of CCSD(T)–6-31G. On the right hand side, the final state of the coefficients is shown, which is the same as seen in Figure 6.4, provided here for easy ref-

erence. One observes that the value of the coefficients for the CCSD(T)–6-31G fidelity is almost always stable around the final values. In other words, one can simply build a small o-MFML model for some small number of training samples at fidelity F and already identify fidelities, f, which do not contribute significantly to the final multifidelity model. This fact makes the o-MFML a self-contained tool for an optimal implementation of the multifidelity method.

A.3.3.2 Excitation energies

The study of the coefficients for the o-MFML method for predicting the first excitation energies could provide useful insights into the workings of the method for such a data structure. The coefficients can be interpreted as a measure of the contribution of each sub-model into the final multifidelity model. The coefficients of the o-MFML model built with the STO-3G

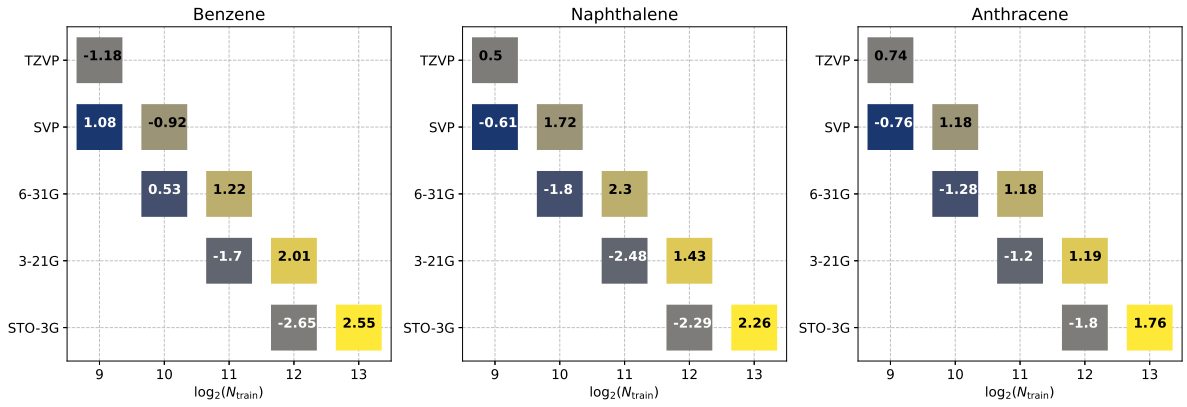


Figure A.11: o-MFML coefficient values for $N_{\text{train}}^{\text{TZVP}} = 2^9 = 512$ for MD-based trajectories of benzene, naphthalene, and anthracene.

baseline fidelity for MD-based trajectories of molecules are shown in Figure A.11. For all three molecules, one notices how different these values are from the default coefficients, which are ± 1. This optimization of the combination of the sub-models is why the corresponding learning curve shows an improvement over its counterpart in the conventional MFML. Consider the values of the various β_s^{opt} as seen in the last pane of Figure A.11 for MD-based anthracene. The coefficients are close to the ± 1 values as prescribed in the conventional MFML method. That is, the values of β_s^{opt} are close to the values of β_s^{MFML}. This indicates that the conventional MFML model was already optimal as reasoned in Chapter 6. The learning curves for the MFML and o-MFML methods were seen to be comparable for this case. This figure makes that very case with numerical evidence.

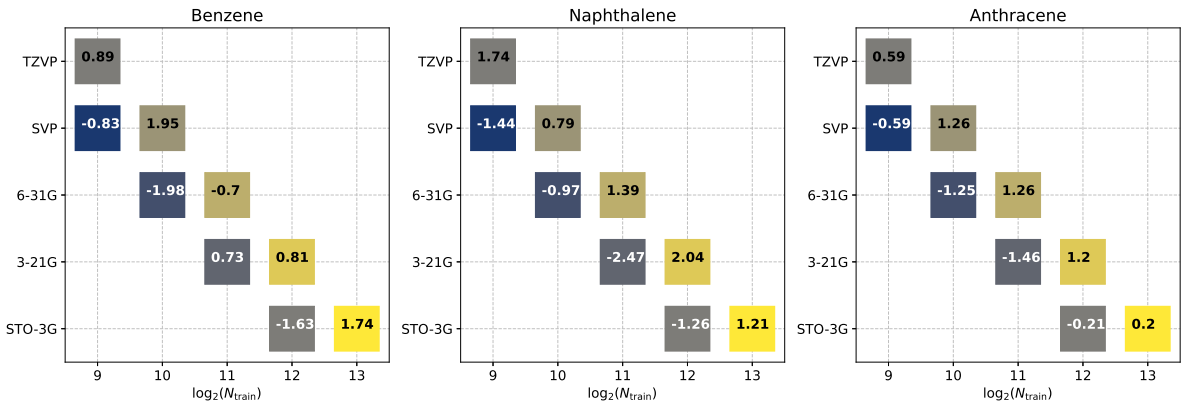


Figure A.12: The coefficient values for $N_{\text{train}}^{\text{TZVP}} = 2^9 = 512$ for DFTB-based trajectories of benzene, naphthalene, and anthracene.

Similarly, the values of β_s^{opt} of the STO-3G baseline for the DFTB-based trajectories are shown in Figure A.12. Of particular interest is the anthracene molecule. For the conventional MFML, it was previously reported in Chapter 5 that the addition of the STO-3G fidelity resulted in a breakdown of the multifidelity structure due to its distribution with respect to the target fidelity, TZVP. As a result, the learning curve of the MFML model with this baseline fidelity shown no improvement in the prediction error for first excitation energies as seen in the top row of Figure 6.6. However, the o-MFML method drastically changed this outlook for the STO-3G baseline and resulted in a learning curve (bottom row of Figure 6.6) which again displays the trend that is expected of a multifidelity method. The rightmost panel of Figure A.12 provides a window into understanding this improvement. The values of the β_s^{opt} for all fidelities other than STO-3G are close to those prescribed by the conventional MFML method. However, for the sub-models built on the STO-3G fidelity, the absolute values of the coefficients are far smaller. The o-MFML method selectively eliminates the STO-3G fidelity while retaining the required information to provide a superior MFML model for this baseline fidelity.

A.4 Additional Results for Data Efficiency Assessment

Adapted from the SI file of [144].

A.4.1 Δ-ML for atomization energies of QM7b

In order to assess whether the behavior of the Δ-ML model as seen in Chapter 9 was influenced by the fidelities only varying by basis set choices, the same test was replicated for the

QM7b dataset [49] . The QM7b dataset consists of a total of 7,211 molecules with a maximum of seven heavy atoms. The atomization energies for each of these molecules is computed with 3 levels of Qc theory, namely, Hartree Fock (denoted here as HF), Møller–Plesset perturbation theory (MP2) [211, 123, 212], and Coupled Cluster Singles and Doubles perturbative Triples (CCSD(T)) [121, 122, 15]. For each level, 3 basis sets are used, STO-3G, 6-31G, and cc-pVDZ (with increasing size). The atomization energy of a molecule is defined as the energy required to completely dissociate all the bonds of the molecule and break it into its constituent free atoms.

From the 7,211 geometries of the dataset, a random collection of 6,144 geometries were set aside as the training set, and the remaining were set aside as a test set. Since the QM7b dataset does not provide the compute times for the different fidelities, it is only used to check for the behavior of the Δ-ML models across the different levels of QC theory, as opposed to varying basis set sizes. The data efficiency benchmarks are only made using the QeMFi dataset. Regardless, the QM7b dataset serves as a key indicator in studying the effects of varying QC theory levels as opposed to basis set sizes, which is the case for QeMFi.

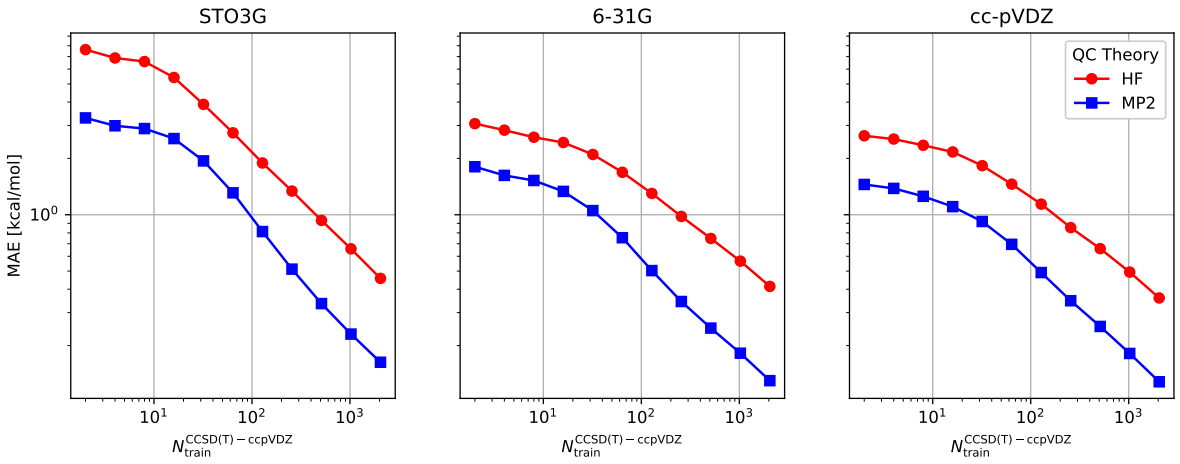


Figure A.13: Learning curves for Δ-ML with varying baseline fidelities for the atomization energies of the QM7b dataset. The different basis sets are denoted as subplot titles.

In order to assess whether the results of the Δ-ML method seen in Chapter 9 were merely due to the fidelities being described by basis set choice, with this dataset, a different approach was employed. The basis set was fixed and the fidelities were then assumed to be the different QC levels of theory. That is, for each basis set that constitutes the multifidelity dataset of QM7b, the ordered fidelity in increasing order was considered to be HF, MP2, and then CCSD(T). Thus, for each basis set choice, the Δ-ML model was built with F =CCSD(T) for HF and MP2 as f_b. Based on previous research for the QM7b dataset in ref. [32], the

Laplacian kernel was used for KRR with $\sigma = 400$. The resulting learning curves are shown in Figure A.13 for the different basis set choices. The Δ-ML model built with f_b =MP2 results in a lower model error in comparison to that built with HF as the baseline fidelity. This is observed regardless of the choice of the basis set. Thus it becomes evident that the results in Chapter 9 are not simply an artifact of the basis sets being set as fidelities. This is a general observation that the closer the baseline fidelity is to the target fidelity, the better the Δ-ML model is at prediction of the QC property.

A.4.2 Predicting QC_b for Δ-ML

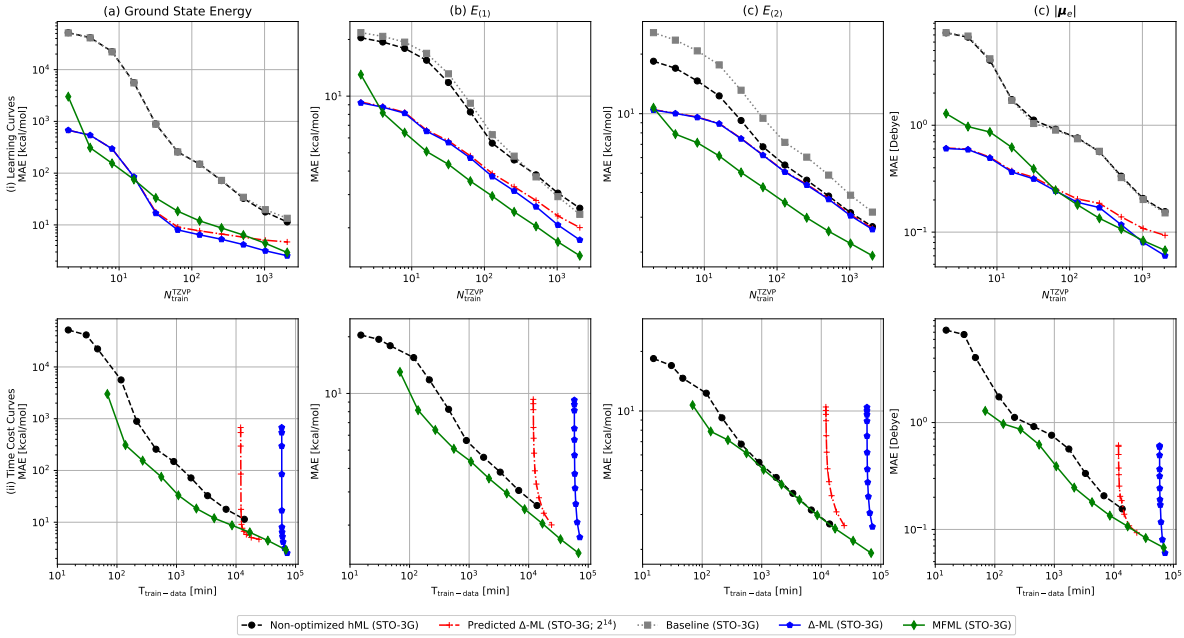


Figure A.14: Learning curves (row (i)) and time-cost assessment (row (ii)) for a non-optimized two fidelity hML model, a predicted QC_b Δ-ML variant, and MFML model for the prediction of ground state energies, first and second vertical excitation energies, and magnitude of electronic contribution to molecular dipole moments. The error in prediction of QC_b with the non-optimized hML model is also depicted (in gray). The predicted QC_b model is trained on 2^{14} training samples at the STO-3G fidelity.

Based on the results for Δ-ML variants and MFML variants from the main text, a natural line of research is to consider a case for Δ-ML, where one does not need to perform the baseline QC computations but uses an ML model to predict those QC-baseline energies instead. This would be somewhat similar to what is performed in MFML for f_b. Two such models are studied in this section. First, since the Δ-ML model uses the baseline fidelity to train the differences, these same are used to train a model for the QC-baseline prediction

over the test set. If the Δ-ML model uses 2 samples, then these same samples are used to predict the baseline over the test set. Thus there is no additional cost incurred in the this *predicted*-Δ-ML model. This model is equivalent to a 2-level hML [139] model built with the same number of training samples at both fidelities, that is, a hML model that does not run the *ad hoc* optimization procedure for the number of training samples. Therefore, this model is referred to hereon as non-optimized hML model. Second, a model is trained with 2^{14} samples for the STO-3G fidelity to predict QC_b. The predictions from this model are used to replace y^{QC_b} from Eq. (3.16). This model is referred to as 'Predicted Δ-ML (STO-3G; 2^{14})'.

The analysis for the predicted-Δ-ML model is shown in Figure A.14 along with the conventional Δ-ML and MFML methods for the prediction of the different QC properties studied in this work. Row (i) shows the learning curves as a function of the number of training samples used at TZVP. In addition, the learning curve for the prediction of QC_b used in the non-optimized hML model over the test set is provided for reference. The results indicate that the error of predicting the baseline QC-fidelity is a huge contributor to the overall error of the predicted-Δ-ML model, as also stated in ref. [139], where the authors report that the use of identical number of training points for all fidelities does not provide any benefit over the single fidelity model in terms of model error. The two curves are almost entirely overlaid on each other for the most part with only a small deviation observed for $N_{\text{train}}^{\text{TZVP}} = 2^{11}$ for most of the QC properties. For $E_{(2)}$, the error of the prediction of the QC-baseline is higher possibly due to the added chemical complexity of the second vertical excitation state. Although the learning curves for Δ-ML and MFML were already discussed in Figure 9.3 and Figure 9.4 respectively, here they are visible in contrast with each other. The two learning curves seem to converge to similar MAE values for large training set sizes. Further, the Predicted Δ-ML (STO-3G; 2^{14}) model initially has MAE comparable to the Δ-ML model for the case of ground state energies but saturates for $N_{\text{train}}^{\text{TZVP}} > 2^7$. For $E_{(1)}$, $E_{(2)}$, and $|\boldsymbol{\mu}_e|$ this model has MAE comparable to the MFML model for the most part with some saturation observed for larger training set sizes.

Figure A.14(ii) studies the time-cost analysis of the two predicted QC_b models in contrast with the Δ-ML and MFML models. First it is observed that the non-optimized hML model does not provide any benefit over MFML as was reasoned above. Second, the predicted Δ-ML (STO-3G; 2^{14}) model for all QC properties is shifted to the right-hand side of the plots due to the cost of training data for the prediction of QC_b : STO-3G. For all QC properties, the MFML method has a lower MAE than this predicted Δ-ML variant for a given time-cost. The Δ-ML model is shifted by a large time-cost that is incurred due to the QC-

baseline calculation as was discussed previously in light of Figure 9.8. These results from Figure A.14 strongly indicate that the use of the predicted Δ-ML variants does not provide any foreseeable benefit.

A.4.3 Validation set and o-MFML learning curves

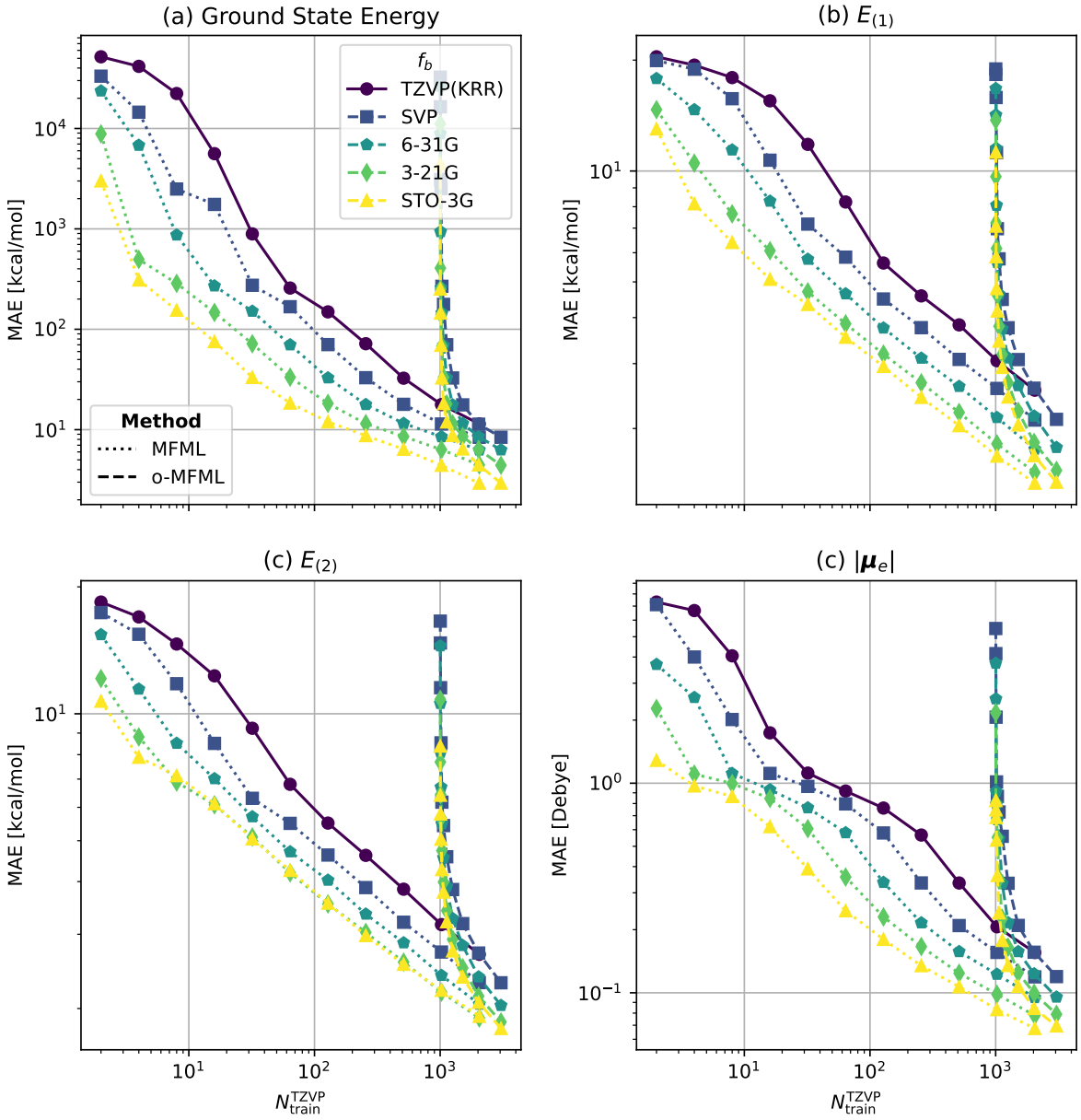


Figure A.15: MFML and o-MFML learning curves for the various QC properties with the validation set size being accounted for o-MFML.

In o-MFML, as has been discussed in Chapter 6, one uses a validation set consisting

of representations and QC properties computed at the highest fidelity, target fidelity. In all the learning curves of the main text, the learning curves for o-MFML did not include these. The validation set cost was included only in the time-cost assessments. Figure A.15 shows the reader the kind of learning curves one would get if the validation set was included in standard learning curves. The figure depicts MFML and o-MFML learning curves for the prediction of several QC properties from the QeMFi dataset in association with the work presented in Chapter 9. It becomes evident that the incorporation of this information into the standard learning curves does not help in making any additional inferences and only makes the visual less appealing overall. This is the reason the main text omits this in the standard learning curves. However, it must be reiterated that the cost associated with the validation set is very much considered in all efficiency assessments made in this dissertation.

A.4.4 Training time of the models

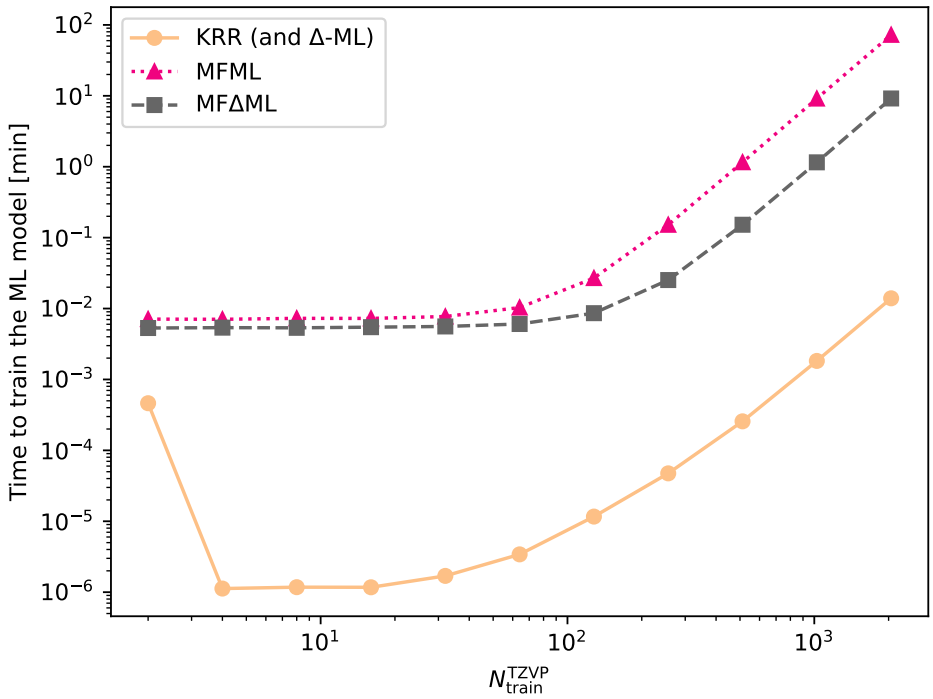


Figure A.16: Time to train different ML models as a function of number of training samples used at the TZVP fidelity. The time is reported here in minutes to provide ready comparison to the time-cost assessment curves of the main text.

Throughout this dissertation the discussion has been focused on the efficiency of MFML in terms of training data costs. It is also interesting to briefly look at the time cost of actually

training the models, that is, the time taken to solve the linear system of equations for KRR for the different sub-models of MFML. This was performed for the setup related to the data efficiency analysis in Chapter 9. The time taken to train the different single fidelity and MF models is depicted graphically in Figure A.16. The single fidelity KRR model is the cheapest, naturally, due to the fact that there is only one fidelity that needs to be trained. The MFΔML is slightly cheaper than the MFML for a similar reason, there is one fidelity less in this model than the MFML. However, as can be seen also in Table A.4, the costs for even large training set sizes is at most ~ 70 minutes which is marginal compared to the time to generate the training data which was of the order of 10^5 minutes for $N_{\text{train}}^{\text{TZVP}} = 2048$. It is for this reason that the time-cost of training the models are not included in the cost-efficiency analyses of Chapter 9.

$N_{\text{train}}^{\text{TZVP}}$	KRR	MFML	MFΔML
2	0.0277	0.4257	0.3188
4	0.0001	0.4232	0.3226
8	0.0001	0.4364	0.3209
16	0.0001	0.4353	0.3277
32	0.0001	0.4631	0.3354
64	0.0002	0.6198	0.3632
128	0.0007	1.6126	0.5163
256	0.0028	9.1021	1.5134
512	0.0154	69.3421	9.1189
1024	0.1095	548.575	69.2028
2048	0.8366	4343.7689	550.4958

Table A.4: Time taken to train different ML models based on number of training samples chosen at the TZVP fidelity. The time is reported here in seconds and rounded to the fourth decimal.

A.5 Supplementary Results for Ground State Energies of Monomers

Adapted from the SI file of [1].

These are certain additional analyses and results for ref. [1] discussed in section 11.1. A preliminary data analysis of the multifidelity training structure for monomers is shown in Figure A.17. This is to keep with the recommendation prescribed in Chapter 5 to check for discrepancy in the multifidelity data structure, which could cause issues with the training

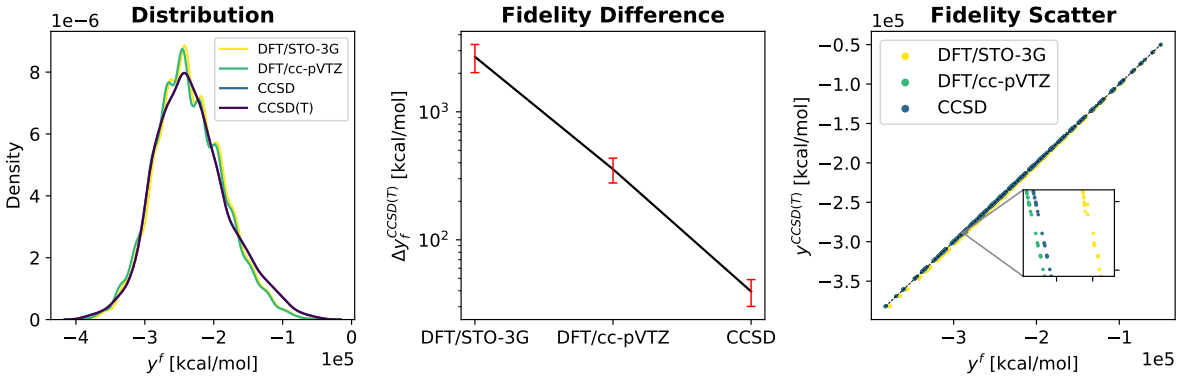


Figure A.17: Preliminary multifidelity data analysis of the monomers for the different fidelities used in this work.

of the MFML models. The left-hand side of the figure shows the distribution of the energies for different fidelities. The CCSD and CCSD(T) fidelities are nearly identical in distribution and show a normally distributed landscape of the energies. The cheaper DFT fidelities show several spikes in the data distribution. The middle pane of Figure A.17 shows the absolute difference of a given fidelity f with respect to the target fidelity CCSD(T). That is, $\Delta_f^{CCSD(T)} = |y^{CCSD(T)} - y^f|$ for varying f. The difference to the target fidelity is seen to be monotonically decreasing with increasing accuracy of the fidelities. The error bars of the plot indicate the standard deviation of the absolute differences. The right most plot of Figure A.17 shows the scatter of the energies of a fidelity with respect to the target fidelity energies at CCSD(T) for the training data. Since the energy range covered is quite large, $10^5 - 4 \cdot 10^5$ kcal/mol, the scatter is not clearly visible. To aid this, an inset is provided around $2.8 \cdot 10^5$.

Figure A.18 compares the learning curves for Δ-ML with different baselines. The results indicate that the closer the QC-baseline is to the target fidelity, the lower the error can get. Indeed, the case for the CCSD QC-baseline reports the MAE 0.2 kcal/mol with $N_{train}^{CCSD(T)} = 512$, which is similar to the case reported in Ref. [208] for a similar number of training samples where the Δ-ML approach is used to learn the perturbative difference between CCSD and CCSD(T) fidelities.

The time-cost comparisons of the different Δ-ML models are presented in Figure A.19 for differing QC-baselines. Three test set sizes are shown: 1,500, 15,000, and 150,000. With increasing proximity of the QC-baseline to the target fidelity, although the MAE decreases, the cost of implementing the Δ-ML model for new predictions becomes unreasonable. With a large test set size, the use of the costlier QC-baselines can not be justified as seen from the plot for 150,000 test set size, with the Δ-ML model with CCSD QC-baseline being

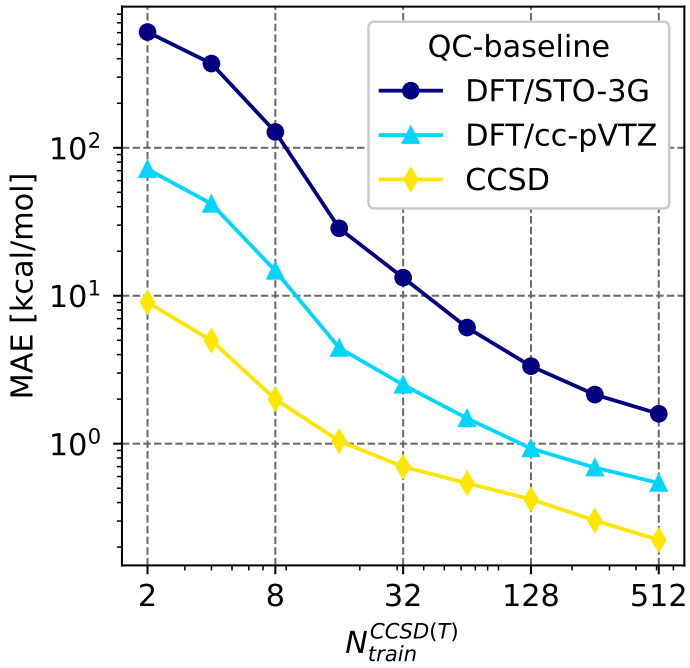


Figure A.18: Learning curves for Δ-ML approach of KRR with different QC-baseline for the target fidelity CCSD(T). It is observed that the closer the QC-baseline is to the target fidelity, the lower the model error is.

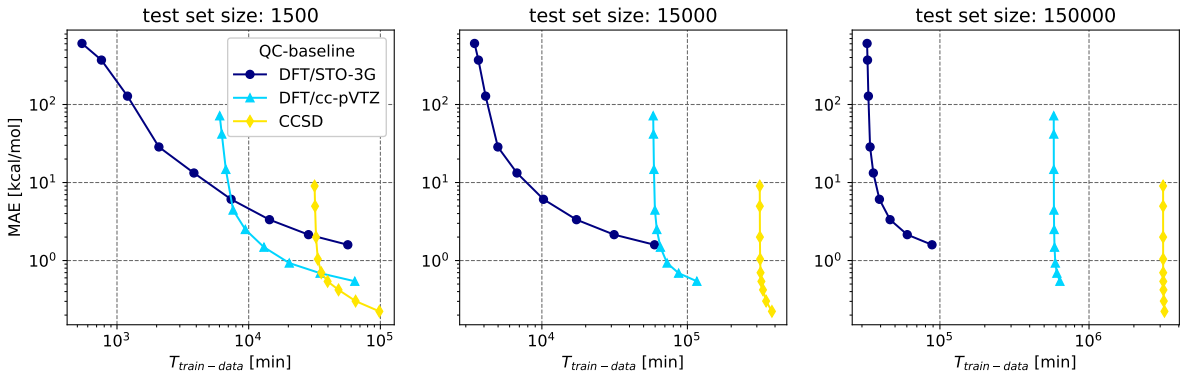


Figure A.19: Training data time-cost for Δ-ML approach of KRR with different QC-baseline for the target fidelity CCSD(T). The cost of the QC-baseline calculations is considered for different test set sizes.

two orders of magnitude costlier than the one with DFT-STO3G as the QC-baseline. This high cost is associated with the cost of making the QC-baseline calculations, which are then added to the prediction of the difference between the QC-baseline and target fidelity.

A.6 Supplementary Results for Excitation Energies of Porphyrin

Adapted from the SI file of [2].

This section reports additional results for ref. [2] discussed in section 11.2.

A.6.1 Multifidelity data analysis

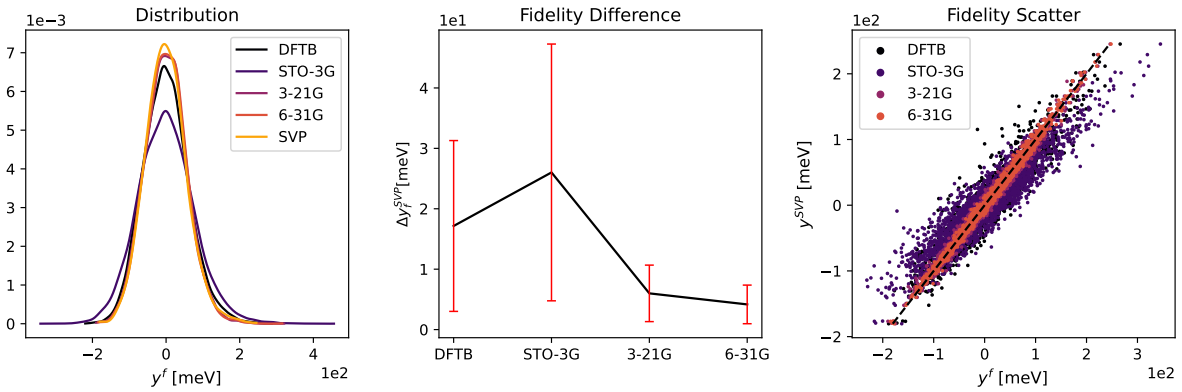


Figure A.20: Preliminary multifidelity data analysis for porphyrin molecule p-TMPyP-9. The STO-3G fidelity shows an unfavorable distribution with respect to the target fidelity of SVP.

Figure A.20 reports the preliminary multifidelity data structure analysis as recommended in Chapter 5 for one single molecule, p-TMPyP-9 of porphyrin, on the clay surface that is discussed in section 11.2. This first analysis includes all the fidelities that were generated for this system. With such a preliminary study, based on the fidelity difference plot and fidelity scatter plot of Figure A.20, it was ascertained that the STO-3G fidelity would not offer significant improvement in the MFML model. Therefore, a new fidelity hierarchy was assumed omitting STO-3G, namely, LC-DFTB, 3-21G, 6-31G, and def2-SVP, in increasing order of accuracy.

Figure A.21 shows the preliminary data analysis for the case of the single molecule, p-TMPyP-9 with the STO-3G fidelity removed from consideration. In this case, a ordered hierarchy is visible in the fidelity difference plot and the scatter of energies with respect to the target fidelity of SVP shows a tighter clustering.

A similar data analysis was performed for the concatenated trajectories of the p-TMPyP porphyrin molecules. The results are delineated in Figure A.22 which includes the STO-3G fidelity. Although the excitation energies all show uni-modal distributions, in the assumed

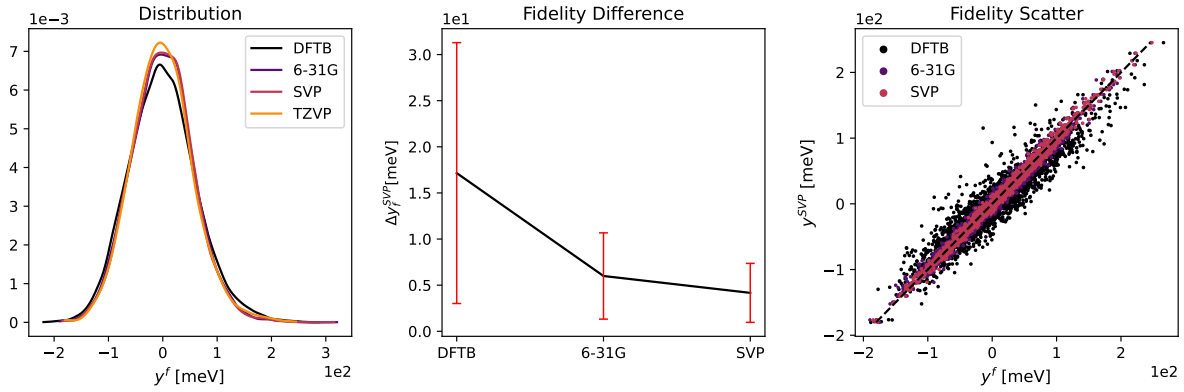


Figure A.21: Preliminary multifidelity data analysis for porphyrin molecule p-TMPyP-9 omitting STO-3G from the hierarchy.

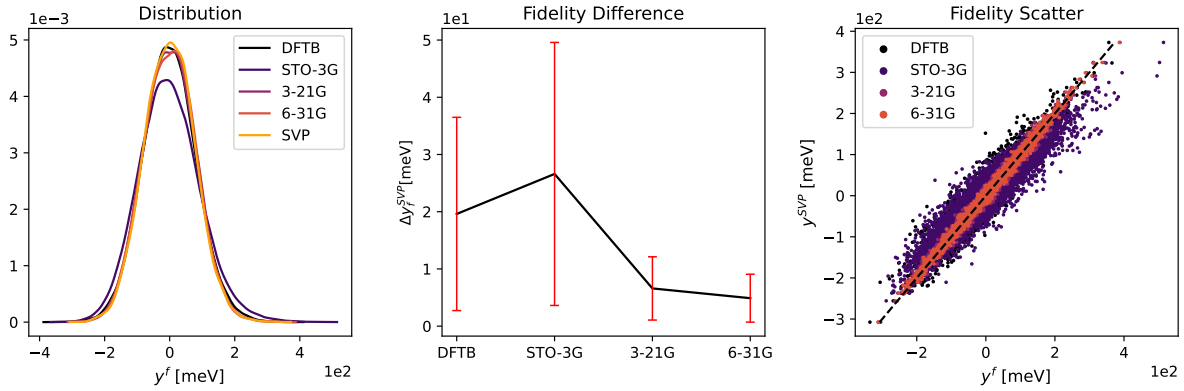


Figure A.22: Preliminary multifidelity data analysis for the concatenated trajectories of the p-TMPyP porphyrin molecules. The STO-3G fidelity shows a poor positioning in the data hierarchy and an unfavorable distribution with respect to the target fidelity of SVP.

hierarchy structure, the STO-3G fidelity once again shows a poor fidelity difference with wide range of outliers as seen in the error bars on the center plot for fidelity difference. Furthermore, in the scatter plot of the energies of different fidelities with respect to SVP, the STO-3G fidelity has a loose clustering. This once again indicates that STO-3G might not be favorable to use in the multifidelity training data structure. On removing this fidelity from the assumed hierarchy structure, one immediately notices that the difference in fidelities shows a monotonic decrease and the scatter plot shows a tight clustering as seen in Figure A.23. Therefore, the STO-3G fidelity is no longer considered in the main application for training MFML models for p-TMPyP porphyrin molecules.

For completeness, the preliminary multifidelity data analysis was repeated for the concatenated trajectories of m-TMPyP porphyrin molecules with similar inferences as stated previously. The resulting plots are shown in Figure A.24 and Figure A.25 including the STO-

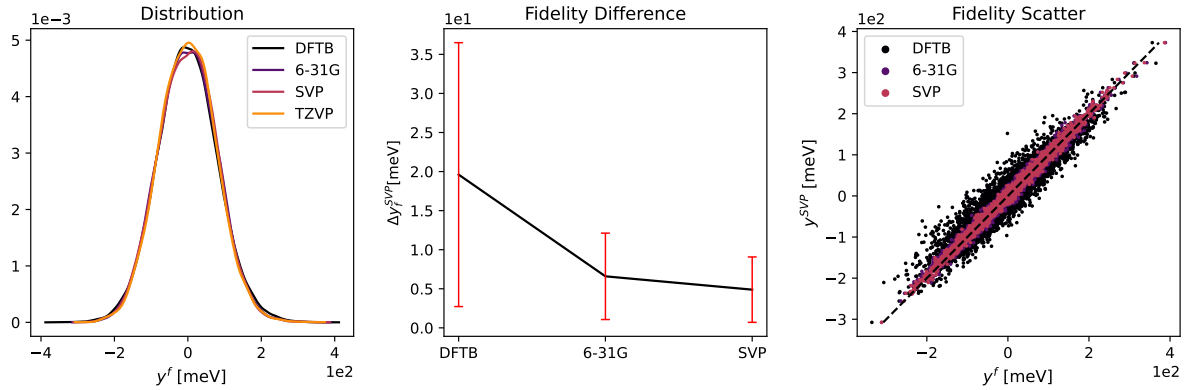


Figure A.23: Preliminary multifidelity data analysis for the concatenated trajectories of the p-TMPyP porphyrin molecules after omitting STO-3G data.

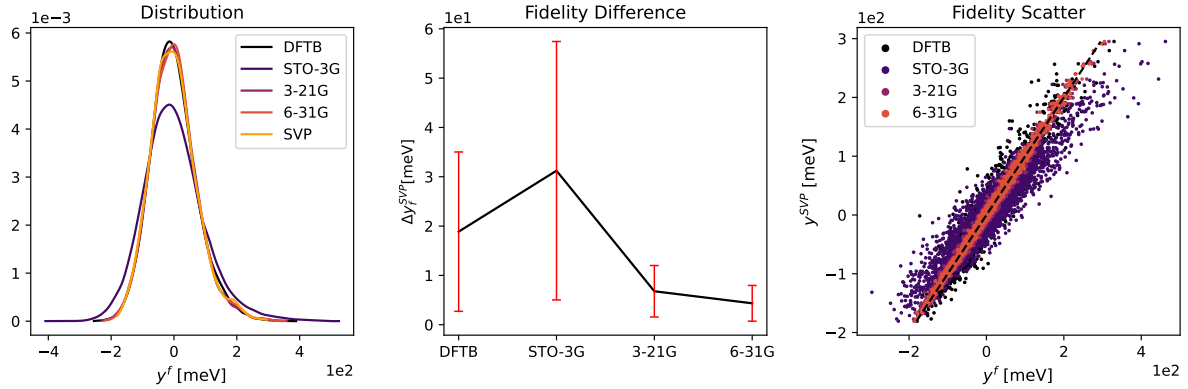


Figure A.24: Preliminary multifidelity data analysis for the concatenated trajectories of the m-TMPyP porphyrin molecules. The STO-3G fidelity shows unfavorable distribution with respect to the target fidelity of SVP.

3G fidelity and omitting it respectively.

Based on the analysis of the data presented in this section, it was decided to omit STO-3G from the multifidelity data structure for the models used to predict excitation energies of porphyrin.

A.6.2 Full learning curves

While the learning curves for porphyrins in section 11.2 are for MFML models which omit the STO-3G fidelity, in Figure A.26, learning curves for all fidelities are provided. That is, the learning curves for different f_b including the STO-3G fidelity are shown. The learning curves show the anticipated reduction in error as one adds the cheaper fidelities to the multifidelity models. For the concatenated trajectories of porphyrin, it can be seen how

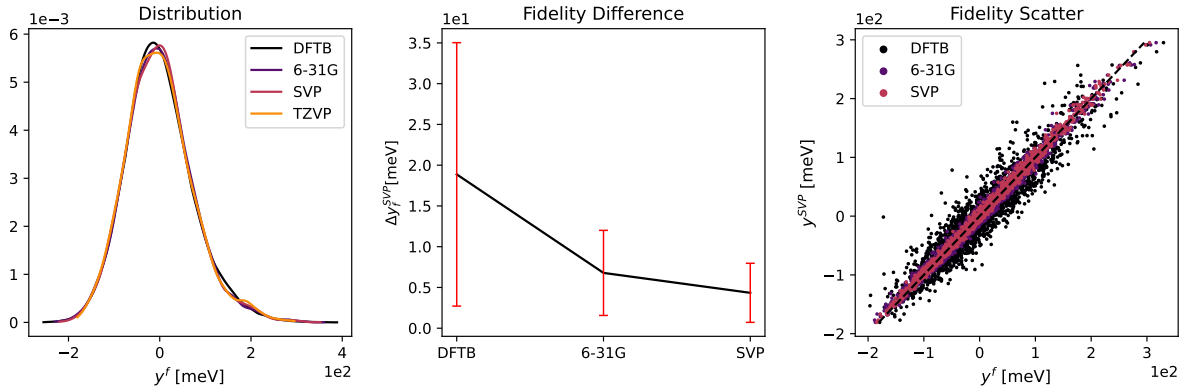


Figure A.25: Preliminary multifidelity data analysis for the concatenated trajectories of the m-TMPyP porphyrin molecules after removing STO-3G from the multifidelity training data structure.

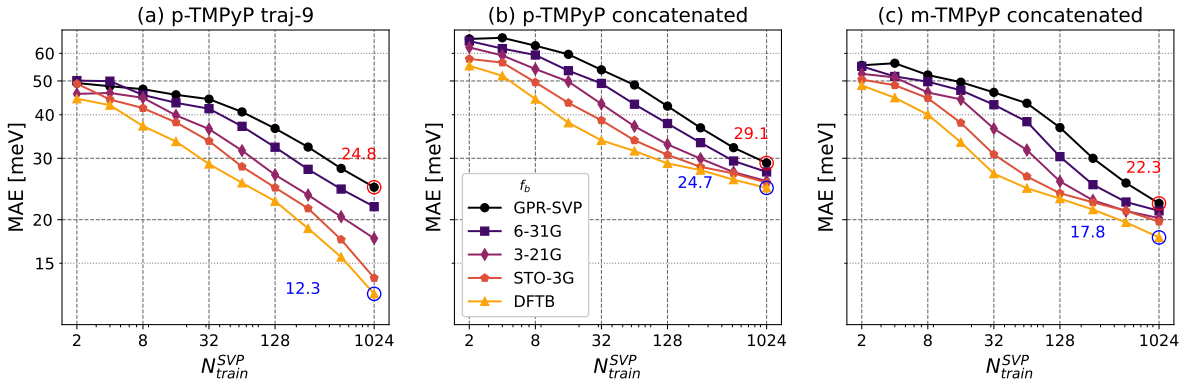


Figure A.26: MFML learning curves for porphyrins with STO-3G fidelity included.

the addition of the STO-3G fidelity provides close to no improvement for large training set sizes. With the full fidelity structure including STO-3G, the standard MFML model results in MAEs of 12.3 meV, 24.7 meV, and 17.8 meV for p-TMPyP-9, concatenated p-TMPyP, and concatenated m-TMPyP porphyrin molecules respectively.

A.6.3 Additional Γ-curves

In addition to the Γ(8)-curve MFML that is used in the main predictions for porphyrins, other variants were also tested for their accuracy versus efficiency. Theses reported in Figure A.27 for the different set-up used for porphyrins. The different Γ-curves show consistent improvement in comparison to standard MFML. it becomes evident that the Γ(8)-curve provides the most meaningful efficiency for a given error, although the others are within similar ranges.

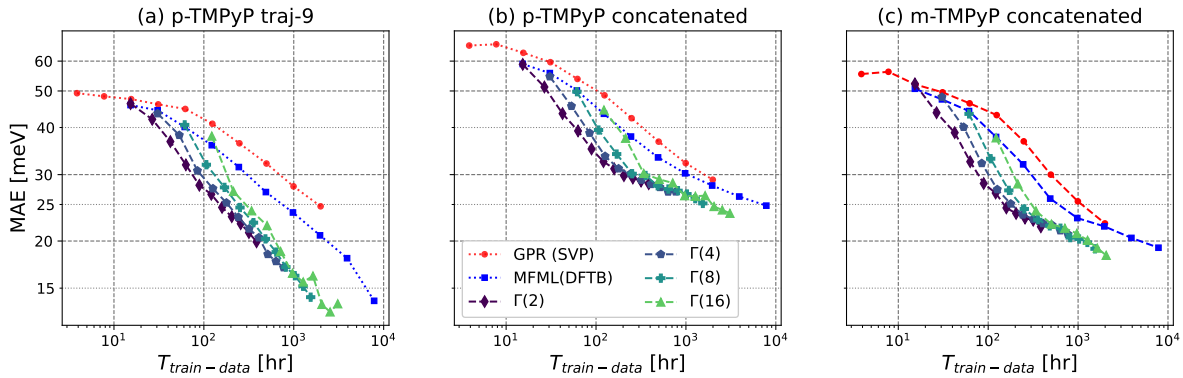


Figure A.27: Time-cost versus model error for additional Γ-curve variants.

APPENDIX B

APPENDIX B - SELECTED QUANTUM CHEMISTRY DETAILS

For the sake of completeness, certain details of the QC methods are provided here. This ensures that the dissertation can be understood in its fullness. Furthermore, this section becomes pertinent since it is the output of such QC calculations which form the training data for the ML models in this work. Only a general overview of QC methods is provided here, the interested reader is directed to refs. [13, 255, 120] and related works for an in-depth treatment of QC methods.

For this segment, we will consider the Born-Oppenheimer approximation [256] wherein there is considered to be a separation of the electronic and nuclear degrees of freedom of the molecule. Broadly speaking, there are two pathways to perform calculations of the electronic properties of molecules, namely, wave function theory (WFT), and density functional theory (DFT) [257]. WFT involves the solution of the electronic form of the Schrödinger equation from Eq. (1.3) with the electronic Hamiltonian operator $\hat{\boldsymbol{H}}_{el}$ given as

$$\hat{\boldsymbol{H}}_{el}(\boldsymbol{R},\boldsymbol{r})\,|\Psi(\boldsymbol{R},\boldsymbol{r})\rangle = E_n\,|\Psi(\boldsymbol{R},\boldsymbol{r})\rangle\ . \tag{B.1}$$

Here, $\boldsymbol{r}$ and $\boldsymbol{R}$ denote the electronic and nuclear coordinates respectively. Once eq. (B.1) is solved for the eigenvalues and eigenvectors, any QC property of the molecule/system can be arrived at. The computational solution of the Schrödinger equation is known in theory but in reality is not feasible for molecules which are complex [258]. WFT is the approximations that are used to cater to such systems making the computation of larger and more complex systems more feasible. On the other hand, DFT methods compute the energy of a system in terms of the ground state electron density. While DFT is quicker and is perhaps

one of the most commonly used method in the field of QC research, it is not exact in the sense that the equations that need to be solved are themselves unknowns [22]. The accuracy of the results of DFT, therefore, depend on the choice of the functional that is used in the calculation process [258]. Below, WFT and DFT are discussed in brief.

B.1 Wave Function Theory

The Hartree-Fock method is often the starting point to discuss *ab initio* methods[1]. In the Hartree-Fock method, the electronic wave function from Eq. (B.1) is denoted by a single Slater determinant[259], ϕ_0. This results in the problem being changed from a decoupled N_e-electron problem to N_e coupled single electron problems. The exact details of how ϕ_0 is computed is omitted here. However, naïvely speaking, it involves the electronic wave functions and some form of spin orbital. The spin orbital in most cases is the molecular orbitals which are reformulated as a linear combination of the atomic orbitals which are in turn expressed in terms of a *basis set*. In theory, the latter expansion is performed for an infinite basis set size. This is approximated in the calculations of WFT by employing a finite basis set, also called the *basis set size limit*. The approximations thus made are understood to result in an incomplete description of the molecule/system because no information of the electronic correlation, which accounts for how one electron is affected by the others, is included.

In the Hartree-Fock method, the electronic correlation is accounted for by the correlation energy since in this method, the effect of the ensemble of electrons is considered as an average when it influences another electron. This forms the basis of solving the electronic Schrödinger equation. subsequent QC theories involve the use of more Slater determinants and the corresponding wave function is then expanded in a three step approach: first as a linear combination of the determinants, each of which is expanded in terms of a molecular orbital, each of which is in turn expanded in terms of the atomic orbitals. In this process, two category of coefficients can be optimized: those for the determinants and those for molecular orbitals. The Hartree-Fock method is often used as the reference calculation point to arrive at the orbitals which are then fixed for the remainder of the calculation. The wave function is expanded as a weighted sum of the different Slater determinants for some excited state i as:

$$|\Psi_i\rangle = \sum_I c_{iI} |\psi_I\rangle \ , \tag{B.2}$$

[1] Meaning derived from first-principles without any parametrization.

with ψ_0 being calculated from the Hartree-Fock method. The coefficients c_{iI} are calculated by minimizing the total energy of the molecule with the constraints of fixed orbitals. This general approach is called the configuration interaction (CI) method.

A size-extensive and size-consistent version of the CI method is the coupled cluster (CC) method. CC is considered the gold standard of *ab initio* method for ground state energy calculation [260]. The single and double excitation are accounted for by using the excitation operator, $\hat{T}$ as

$$|\Psi_{CC}\rangle = \exp(\hat{T})\,|\phi_0\rangle = \left(1 + \hat{T} + \frac{1}{2!}\hat{T}^2 + \ldots\right)|\phi_0\rangle \ . \tag{B.3}$$

There are several other CI methods such as CASCF, MCSCF, full CI, and MR-CI. These are omitted here since they are not used through this dissertation. Interested readers are referred to ref. [260]. It is also to be noted that the use of CI methods and WFT has a high compute costs associated with them.

WFT is the primary QC method used to compute the atomization energies of the QM7b dataset used in Chapter 6 with a variation of the CC method, namely CCSD(T), being employed.

B.2 Density Functional Theory

DFT formulates the problem of solving the Schrödinger equation in terms of the electron density, $\rho(\boldsymbol{r})$, of the system. In the presence of an external potential $V(\boldsymbol{r})$, there is a direct correspondence to $\rho(\boldsymbol{r})$ as the potential affects the electron density. This energy is written down in the form of a *universal functional* of the electronic density, $F[\rho(\boldsymbol{r})]$. Thus, the ground state energy of the system is computed through the following equation:

$$E[\rho(\boldsymbol{r})] = \int V(\boldsymbol{r})\rho(\boldsymbol{r})d\boldsymbol{r} + F[\rho(\boldsymbol{r})] \ . \tag{B.4}$$

$F[\rho(\boldsymbol{r})]$ is separated into Coulombic and non-Coulombic parts. The non-Coulombic components can be split into the kinetic energy of the electrons that do not interact and the correlation term which describes the interaction of electrons, that is the exchange-correlation functional. This is the unknown component of DFT. Finding the right correlation functional is a key component of DFT and the approximation of it is critical to the usage of the method. For further details the reader is directed to the original derivations in refs. [261, 261]. The review and in depth critique presented in ref. [258] is a splendid read to better comprehend the effectiveness of DFT for the calculation of electronic properties.

DFT forms the core of most computations made in the works presented in this dissertation. Chapter 7 presents a multifidelity dataset with QC properties computed with DFT.

BIBLIOGRAPHY

[1] V. Vinod, D. Lyu, M. Ruth, P. R. Schreiner, U. Kleinekathöfer, and P. Zaspel, "Predicting molecular energies of small organic molecules with multi-fidelity methods," *J. Comp. Chem.*, vol. 46, no. 6, p. e70056, 2025.

[2] D. Lyu, M. Holzenkamp, V. Vinod, Y. M. Holtkamp, S. Maity, C. R. Salazar, U. Kleinekathöfer, and P. Zaspel, "Excitation energy transfer between porphyrin dyes on a clay surface: A study employing multifidelity machine learning," *arXiv*, 2024.

[3] Q. Rao and J. Frtunikj, "Deep learning for self-driving cars: Chances and challenges," in *Proceedings of the 1st International Workshop on Software Engineering for AI in Autonomous Systems*, pp. 35–38, 2018.

[4] M. Daily, S. Medasani, R. Behringer, and M. Trivedi, "Self-driving cars," *Computer*, vol. 50, no. 12, pp. 18–23, 2017.

[5] G. Schneider, "Virtual screening: an endless staircase?," *Nat. Rev. Drug Discov.*, vol. 9, no. 4, pp. 273–276, 2010.

[6] F. Neese, "Software update: The ORCA program system, version 4.0," *WIREs Comput. Mol. Sci.*, vol. 8, p. e1327, 09 2018.

[7] F. Neese, F. Wennmohs, U. Becker, and C. Riplinger, "The ORCA quantum chemistry program package," *J. Chem. Phys.*, vol. 152, p. 224108, 06 2020.

[8] M. J. Frisch, G. W. Trucks, H. B. Schlegel, G. E. Scuseria, M. A. Robb, J. R. Cheeseman, G. Scalmani, V. Barone, B. Mennucci, G. A. Petersson, H. Nakatsuji, M. Caricato, X. Li, H. P. Hratchian, A. F. Izmaylov, J. Bloino, G. Zheng, J. L. Sonnenberg, M. Hada, M. Ehara, K. Toyota, R. Fukuda, J. Hasegawa, M. Ishida, T. Nakajima, Y. Honda, O. Kitao, H. Nakai, T. Vreven, J. A. Montgomery, Jr., J. E. Peralta, F. Ogliaro, M. Bearpark, J. J. Heyd, E. Brothers, K. N. Kudin, V. N. Staroverov, R. Kobayashi, J. Normand, K. Raghavachari, A. Rendell, J. C. Burant, S. S. Iyengar, J. Tomasi, M. Cossi, N. Rega, J. M. Millam, M. Klene, J. E. Knox, J. B. Cross,

V. Bakken, C. Adamo, J. Jaramillo, R. Gomperts, R. E. Stratmann, O. Yazyev, A. J. Austin, R. Cammi, C. Pomelli, J. W. Ochterski, R. L. Martin, K. Morokuma, V. G. Zakrzewski, G. A. Voth, P. Salvador, J. J. Dannenberg, S. Dapprich, A. D. Daniels, O. Farkas, J. B. Foresman, J. V. Ortiz, J. Cioslowski, and D. J. Fox, "Gaussian 09 Revision E.01."

[9] D. G. A. Smith, L. A. Burns, A. C. Simmonett, R. M. Parrish, M. C. Schieber, R. Galvelis, P. Kraus, H. Kruse, R. Di Remigio, A. Alenaizan, A. M. James, S. Lehtola, J. P. Misiewicz, M. Scheurer, R. A. Shaw, J. B. Schriber, Y. Xie, Z. L. Glick, D. A. Sirianni, J. S. O'Brien, J. M. Waldrop, A. Kumar, E. G. Hohenstein, B. P. Pritchard, B. R. Brooks, I. Schaefer, Henry F., A. Y. Sokolov, K. Patkowski, I. DePrince, A. Eugene, U. Bozkaya, R. A. King, F. A. Evangelista, J. M. Turney, T. D. Crawford, and C. D. Sherrill, "PSI4 1.4: open-source software for high-throughput quantum chemistry," *J. Chem. Phys.*, vol. 152, p. 184108, 05 2020.

[10] M. Barbatti, M. Bondanza, R. Crespo-Otero, B. Demoulin, P. O. Dral, G. Granucci, F. Kossoski, H. Lischka, B. Mennucci, S. Mukherjee, M. Pederzoli, M. Persico, M. Pinheiro Jr, J. Pittner, F. Plasser, E. Sangiogo Gil, and L. Stojanovic, "Newton-X platform: New software developments for surface hopping and nuclear ensembles," *J. Chem. Theory Comput.*, vol. 18, no. 11, pp. 6851–6865, 2022.

[11] G. Landrum, P. Tosco, B. Kelley, R. Rodriguez, D. Cosgrove, R. Vianello, sriniker, P. Gedeck, G. Jones, NadineSchneider, E. Kawashima, D. Nealschneider, A. Dalke, M. Swain, B. Cole, S. Turk, A. Savelev, A. Vaucher, M. Wójcikowski, I. Take, V. F. Scalfani, R. Walker, D. Probst, K. Ujihara, tadhurst cdd, A. Pahl, guillaume godin, J. Lehtivarjo, F. Bérenger, and J. Bisson, "rdkit/rdkit: 2024_09_2 (Q3 2024) Release," 10 2024.

[12] E. Schrödinger, "An undulatory theory of the mechanics of atoms and molecules," *Phys. Rev.*, vol. 28, pp. 1049–1070, Dec 1926.

[13] D. J. Griffiths and D. F. Schroeter, *Introduction to quantum mechanics*. Cambridge university press, 2019.

[14] M. Griebel and J. Hamaekers, "Tensor product multiscale many-particle spaces with finite-order weights for the electronic schrödinger equation," *Zeitschrift für Physikalische Chemie*, vol. 224, no. 3-4, pp. 527–543, 2010.

[15] T. D. Crawford and H. F. Schaefer III, *An Introduction to Coupled Cluster Theory for Computational Chemists*, ch. 2, pp. 33–136. John Wiley & Sons, Ltd, 2000.

[16] A. Singharoy, C. Maffeo, K. H. Delgado-Magnero, D. J. K. Swainsbury, M. Şener, U. Kleinekathöfer, J. W. Vant, J. Nguyen, A. Hitchcock, B. Isralewitz, I. Teo, D. E. Chandler, J. E. Stone, J. C. Phillips, T. V. Pogorelov, M. I. Mallus, C. Chipot, Z. Luthey-Schulten, D. P. Tieleman, C. N. Hunter, E. Tajkhorshid, A. Aksimentiev, and K. Schulten, "Atoms to phenotypes: Molecular design principles of cellular energy metabolism," *Cell*, vol. 179, pp. 1098–1111.e23, 2019.

[17] J. P. Perdew and K. Schmidt, "Jacob's ladder of density functional approximations for the exchange-correlation energy," *AIP Conf. Proc.*, vol. 577, pp. 1–20, 07 2001.

[18] R. Ramakrishnan and O. A. von Lilienfeld, *Machine Learning, Quantum Chemistry, and Chemical Space*, ch. 5, pp. 225–256.
John Wiley & Sons, Ltd, 2017.

[19] L. David, A. Thakkar, R. Mercado, and O. Engkvist, "Molecular representations in AI-driven drug discovery: a review and practical guide," *J. Cheminformatics*, vol. 12, no. 1, pp. 1–22, 2020.

[20] Z. J. Baum, X. Yu, P. Y. Ayala, Y. Zhao, S. P. Watkins, and Q. Zhou, "Artificial intelligence in chemistry: Current trends and future directions," *J. Chem. Inf. Model.*, vol. 61, no. 7, pp. 3197–3212, 2021.

[21] O. A. von Lilienfeld, R. Ramakrishnan, M. Rupp, and A. Knoll, "Fourier series of atomic radial distribution functions: A molecular fingerprint for machine learning models of quantum chemical properties," *Int. J. Quantum Chem.*, vol. 115, no. 16, pp. 1084–1093, 2015.

[22] J. Westermayr and P. Marquetand, "Machine learning for electronically excited states of molecules," *Chem. Rev.*, vol. 121, pp. 9873–9926, 11 2020.

[23] P. O. Dral, ed., *Quantum Chemistry in the Age of Machine Learning*.
Elsevier, 2023.

[24] J. S. Smith, B. Nebgen, N. Lubbers, O. Isayev, and A. E. Roitberg, "Less is more: Sampling chemical space with active learning," *J. Chem. Phys.*, vol. 148, p. 241733, 05 2018.

[25] E. Uteva, R. S. Graham, R. D. Wilkinson, and R. J. Wheatley, "Active learning in Gaussian process interpolation of potential energy surfaces," *J. Chem. Phys.*, vol. 149, no. 17, 2018.

[26] N. Wilson, D. Willhelm, X. Qian, R. Arróyave, and X. Qian, "Batch active learning for accelerating the development of interatomic potentials," *Comput. Mater. Sci.*, vol. 208, p. 111330, 2022.

[27] V. Zaverkin, D. Holzmüller, I. Steinwart, and J. Kästner, "Exploring chemical and conformational spaces by batch mode deep active learning," *Digital Discovery*, vol. 1, no. 5, pp. 605–620, 2022.

[28] M. G. Fernández-Godino, "Review of multi-fidelity models," *Adv. Comput. Sci. Eng.*, vol. 1, p. 351–400, 12 2023.

[29] R. Ramakrishnan, P. O. Dral, M. Rupp, and O. A. von Lilienfeld, "Big Data Meets Quantum Chemistry Approximations: The Δ-Machine Learning Approach," *J. Chem. Theory Comput.*, vol. 11, pp. 2087–2096, 05 2015.

[30] S. Manzhos and T. J. Carrington, "Neural network potential energy surfaces for small molecules and reactions," *Chem. Rev.*, vol. 121, no. 16, pp. 10187–10217, 2021.

[31] J. Westermayr, M. Gastegger, K. T. Schütt, and R. J. Maurer, "Perspective on integrating machine learning into computational chemistry and materials science," *J. of Chem. Phys.*, vol. 154, p. 230903, 06 2021.

[32] P. Zaspel, B. Huang, H. Harbrecht, and O. A. Von Lilienfeld, "Boosting quantum machine learning models with a multilevel combination technique: Pople Diagrams revisited," *J. Chem. Theory Comput.*, vol. 15, no. 3, pp. 1546–1559, 2019.

[33] M. Hegland, J. Garcke, and V. Challis, "The combination technique and some generalisations," *Linear Algebra Appl.*, vol. 420, no. 2-3, pp. 249–275, 2007.

[34] J. Benk and D. Pflüger, "Hybrid parallel solutions of the black-scholes pde with the truncated combination technique," in *2012 International Conference on High Performance Computing & Simulation (HPCS)*, pp. 678–683, IEEE, 2012.

[35] C. Reisinger, "Analysis of linear difference schemes in the sparse grid combination technique," *IMA J. Numer. Anal.*, vol. 33, no. 2, pp. 544–581, 2013.

[36] H. Harbrecht, M. Peters, and M. Siebenmorgen, "Combination technique based k-th moment analysis of elliptic problems with random diffusion," *J. Comput. Phys.*, vol. 252, pp. 128–141, 2013.

[37] M. Hegland, B. Harding, C. Kowitz, D. Pflüger, and P. Strazdins, "Recent developments in the theory and application of the sparse grid combination technique," in *Software for Exascale Computing - SPPEXA 2013-2015* (H.-J. Bungartz, P. Neumann,

and W. E. Nagel, eds.), (Cham), pp. 143–163, Springer International Publishing, 2016.

[38] A.-L. Haji-Ali, F. Nobile, and R. Tempone, "Multi-index monte carlo: when sparsity meets sampling," *Numerische Mathematik*, vol. 132, no. 4, pp. 767–806, 2016.

[39] M. Kennedy and A. O'Hagan, "Predicting the output from a complex computer code when fast approximations are available," *Biometrika*, vol. 87, p. 1–13, 03 2000.

[40] L. L. Gratiet and J. Garnier, "Recursive co-kriging model for design of computer experiments with multiple levels of fidelity," *Int. J. Uncertainty Quantif.*, vol. 4, no. 5, 2014.

[41] V. Vinod and P. Zaspel, "QeMFi: A multifidelity dataset of quantum chemical properties of diverse molecules," *Sci. Data*, vol. 12, p. 202, 02 2025.

[42] V. Vinod and P. Zaspel, "Assessing non-nested configurations of multifidelity machine learning for quantum-chemical properties," *Mach. Learn.: Sci. Technol.*, vol. 5, p. 045005, 10 2024.

[43] V. Vinod and P. Zaspel, "Investigating data hierarchies in multifidelity machine learning for excitation energies," *J. Chem. Theory Comput.*, vol. 21, no. 6, pp. 3077–3091, 2025.

[44] M. Rupp, A. Tkatchenko, K.-R. Müller, and O. A. von Lilienfeld, "Fast and accurate modeling of molecular atomization energies with machine learning," *Phys. Rev. Lett.*, vol. 108, pp. 05830–1 – 05830–5, Jan 2012.

[45] B. Huang and O. A. Von Lilienfeld, "Understanding molecular representations in machine learning: The role of uniqueness and target similarity," *J. Chem. Phys.*, vol. 145, no. 16, p. 161102, 2016.

[46] M. Veit, D. M. Wilkins, Y. Yang, J. DiStasio, Robert A., and M. Ceriotti, "Predicting molecular dipole moments by combining atomic partial charges and atomic dipoles," *J. Chem. Phys.*, vol. 153, p. 024113, 07 2020.

[47] K. Schütt, O. Unke, and M. Gastegger, "Equivariant message passing for the prediction of tensorial properties and molecular spectra," in *Proceedings of the 38th International Conference on Machine Learning* (M. Meila and T. Zhang, eds.), vol. 139 of *Proceedings of Machine Learning Research*, pp. 9377–9388, PMLR, 07 2021.

[48] M. Haghighatlari, J. Li, F. Heidar-Zadeh, Y. Liu, X. Guan, and T. Head-Gordon, "Learning to make chemical predictions: The interplay of feature representation, data, and machine learning methods," *Chem*, vol. 6, p. 1527–1542, 07 2020.

[49] G. Montavon, M. Rupp, V. Gobre, A. Vazquez-Mayagoitia, K. Hansen, A. Tkatchenko, K.-R. Müller, and O. A. Von Lilienfeld, "Machine learning of molecular electronic properties in chemical compound space," *New J. Phys.*, vol. 15, no. 9, p. 095003, 2013.

[50] M. Krämer, P. M. Dohmen, W. Xie, D. Holub, A. S. Christensen, and M. Elstner, "Charge and exciton transfer simulations using machine-learned hamiltonians," *J. Chem. Theory Comput.*, vol. 16, pp. 4061–4070, 2020.

[51] B. Huang and O. A. von Lilienfeld, "Quantum machine learning using atom-in-molecule-based fragments selected on the fly," *Nat. Chem.*, vol. 12, pp. 945–951, 10 2020.

[52] R. F. W. Bader, *Atoms in Molecules*, pp. 64–86.
John Wiley & Sons, Ltd, 2002.

[53] C. K. Williams and C. E. Rasmussen, *Gaussian processes for machine learning*, vol. 2.
MIT press Cambridge, MA, 2006.

[54] H. Wendland, *Scattered Data Approximation*.
Cambridge University Press, 12 2004.

[55] R. M. Dudley, *Real analysis and probability*.
Chapman and Hall/CRC, 2018.

[56] F. Larkin, "Gaussian measure in hilbert space and applications in numerical analysis," *The Rocky Mountain Journal of Mathematics*, pp. 379–421, 1972.

[57] M. Kanagawa, P. Hennig, D. Sejdinovic, and B. K. Sriperumbudur, "Gaussian Processes and kernel methods: A review on connections and equivalences," *arXiv*, 07 2018.

[58] G. Wahba, *Spline models for observational data*.
SIAM, 1990.

[59] A. Caponnetto and E. De Vito, "Optimal rates for the regularized least-squares algorithm," *Found. Comput. Math.*, vol. 7, pp. 331–368, 2007.

[60] V. Vovk, *Kernel Ridge Regression*, pp. 105–116.
Berlin, Heidelberg: Springer Berlin Heidelberg, 2013.

[61] B. Schölkopf, R. Herbrich, and A. J. Smola, "A generalized representer theorem," in *Computational Learning Theory* (D. Helmbold and B. Williamson, eds.), (Berlin, Heidelberg), pp. 416–426, Springer Berlin Heidelberg, 2001.

[62] B. Schölkopf and A. J. Smola, *Learning with kernels: support vector machines, regularization, optimization, and beyond*.
MIT press, 2018.

[63] K. Vu, J. C. Snyder, L. Li, M. Rupp, B. F. Chen, T. Khelif, K.-R. Müller, and K. Burke, "Understanding kernel ridge regression: Common behaviors from simple functions to density functionals," *Int. J. Quantum Chem.*, vol. 115, pp. 1115–1128, 2015.

[64] J. C. Snyder, M. Rupp, K. Hansen, K.-R. Müller, and K. Burke, "Finding density functionals with machine learning," *Phys. Rev. Lett.*, vol. 108, no. 25, p. 253002, 2012.

[65] F. A. Faber, A. S. Christensen, B. Huang, and O. A. von Lilienfeld, "Alchemical and structural distribution based representation for universal quantum machine learning," *J. Chem. Phys.*, vol. 148, p. 241717, June 2018.

[66] A. Christensen, F. Faber, B. Huang, L. Bratholm, A. Tkatchenko, K. Muller, and O. v. Lilienfeld, "QML: A python toolkit for quantum machine learning," 2017.

[67] R. A. Van De Geijn and E. S. Quintana-Ortí, *The science of programming matrix computations*.
University of Texas, 2008.

[68] B. Peherstorfer, K. Willcox, and M. Gunzburger, "Survey of multifidelity methods in uncertainty propagation, inference, and optimization," *SIAM Review*, vol. 60, p. 550–591, 01 2018.

[69] P. Piperni, A. DeBlois, and R. Henderson, "Development of a multilevel multidisciplinary-optimization capability for an industrial environment," *AIAA Journal*, vol. 51, no. 10, pp. 2335–2352, 2013.

[70] Iooss, Bertrand and Lemaître, Paul, *A Review on Global Sensitivity Analysis Methods*, pp. 101–122.
Springer US, 2015.

[71] E. Qian, B. Peherstorfer, D. O'Malley, V. V. Vesselinov, and K. Willcox, "Multifidelity monte carlo estimation of variance and sensitivity indices," *SIAM/ASA Journal on Uncertainty Quantification*, vol. 6, p. 683–706, 01 2018.

[72] S. van Rijn, S. Schmitt, M. Olhofer, M. van Leeuwen, and T. Bäck, "Multi-fidelity surrogate model approach to optimization," in *Proceedings of the Genetic and Evo-*

lutionary Computation Conference Companion, GECCO '18, p. 225–226, Association for Computing Machinery, 07 2018.

[73] M. Zaefferer, D. Gaida, and T. Bartz-Beielstein, "Multi-fidelity modeling and optimization of biogas plants," *Applied Soft Computing*, vol. 48, p. 13–28, 11 2016.

[74] A. Forrester, A. Sobester, and A. Keane, *Exploiting Gradient Information*, ch. 7, pp. 155–165.
John Wiley & Sons, Ltd, 2008.

[75] P. Benner, S. Gugercin, and K. Willcox, "A survey of projection-based model reduction methods for parametric dynamical systems," *SIAM Review*, vol. 57, no. 4, pp. 483–531, 2015.

[76] R. Molléro, X. Pennec, H. Delingette, A. Garny, N. Ayache, and M. Sermesant, "Multifidelity-CMA: a multifidelity approach for efficient personalisation of 3D cardiac electromechanical models," *Biomechanics and Modeling in Mechanobiology*, vol. 17, p. 285–300, 02 2018.

[77] P. Perdikaris, D. Venturi, and G. E. Karniadakis, "Multifidelity information fusion algorithms for high-dimensional systems and massive data sets," *SIAM J. Sci. Comput.*, vol. 38, p. B521–B538, 01 2016.

[78] A. Forrester, A. Sobester, and A. Keane, *Engineering design via surrogate modelling: a practical guide*.
John Wiley & Sons, 2008.

[79] M. Cutler, T. J. Walsh, and J. P. How, "Reinforcement learning with multi-fidelity simulators," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, p. 3888–3895, IEEE, 05 2014.

[80] R. Lam, D. L. Allaire, and K. E. Willcox, *Multifidelity Optimization using Statistical Surrogate Modeling for Non-Hierarchical Information Sources*, ch. 06.
AIAA SciTech Forum, American Institute of Aeronautics and Astronautics, 01 2015.

[81] P. Perdikaris, D. Venturi, J. O. Royset, and G. E. Karniadakis, "Multi-fidelity modelling via recursive co-kriging and gaussian–markov random fields," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 471, 07 2015.

[82] B. Peherstorfer, T. Cui, Y. Marzouk, and K. Willcox, "Multifidelity importance sampling," *Computer Methods in Applied Mechanics and Engineering*, vol. 300, p. 490–509, 03 2016.

[83] X. Meng and G. E. Karniadakis, "A composite neural network that learns from multi-fidelity data: Application to function approximation and inverse pde problems," *J. Comput. Phys.*, vol. 401, p. 109020, 2020.

[84] M. M. Valero, L. Jofre, and R. Torres, "Multifidelity prediction in wildfire spread simulation: Modeling, uncertainty quantification and sensitivity analysis," *Env. Model. Soft.*, vol. 141, p. 105050, 07 2021.

[85] D. H. Song and D. M. Tartakovsky, "Transfer learning on multifidelity data," *J. Mach. Learn. Model. Comput.*, vol. 3, no. 1, pp. 31–47, 2022.

[86] Y. Xu, V. Keshavarzzadeh, R. M. Kirby, and A. Narayan, "A bandit-learning approach to multifidelity approximation," *SIAM J. Sci. Comput.*, vol. 44, p. A150–A175, 02 2022.

[87] W. R. Thompson, "On the likelihood that one unknown probability exceeds another in view of the evidence of two samples," *Biometrika*, vol. 25, no. 3/4, pp. 285–294, 1933.

[88] L. Lu, R. Pestourie, S. G. Johnson, and G. Romano, "Multifidelity deep neural operators for efficient learning of partial differential equations with application to fast inverse design of nanoscale heat transport," *Physical Review Research*, vol. 4, p. 023210, 06 2022.

[89] S. Pawar, O. San, P. Vedula, A. Rasheed, and T. Kvamsdal, "Multi-fidelity information fusion with concatenated neural networks," *Scientific Reports*, vol. 12, p. 5900, 04 2022.

[90] P. Perdikaris, D. Venturi, J. Royset, and G. E. Karniadakis, "Multi-fidelity modelling voa recursive co-kriging and Gaussian-Markov random fields," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 07 2015.

[91] V. Balabanov, B. Grossman, L. Watson, W. Mason, and R. Haftka, "Multifidelity response surface model for hsct wing bending material weight," in *7th AIAA/USAF/-NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, 1998.

[92] A. I. Forrester, N. W. Bressloff, and A. J. Keane, "Optimization using surrogate models and partially converged computational fluid dynamics simulations," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 462, no. 2071, pp. 2177–2204, 2006.

[93] S. R. Arridge, J. P. Kaipio, V. Kolehmainen, M. Schweiger, E. Somersalo, T. Tarvainen, and M. Vauhkonen, "Approximation errors and model reduction with an application in optical diffusion tomography," *Inverse Problems*, vol. 22, p. 175, 01 2006.

[94] V. Kolehmainen, M. Schweiger, I. Nissilä, T. Tarvainen, S. R. Arridge, and J. P. Kaipio, "Approximation errors and model reduction in three-dimensional diffuse optical tomography," in *Biomedical Optics and 3-D Imaging*, p. BTu3A.5, Optica Publishing Group, 2012.

[95] V. Kolehmainen, M. Schweiger, I. Nissilä, T. Tarvainen, S. R. Arridge, and J. P. Kaipio, "Approximation errors and model reduction in three-dimensional diffuse optical tomography," *J. Opt. Soc. Am. A*, vol. 26, pp. 2257–2268, Oct 2009.

[96] N. M. Alexandrov, R. M. Lewis, C. R. Gumbert, L. L. Green, and P. A. Newman, "Approximation and model management in aerodynamic optimization with variable-fidelity models," *Journal of Aircraft*, vol. 38, no. 6, pp. 1093–1101, 2001.

[97] V. Balabanov, B. Grossman, L. Watson, W. Mason, and R. Haftka, *Multifidelity response surface model for HSCT wing bending material weight.*
AIAA, 1998.

[98] R. Chen, J. Xu, S. Zhang, C.-H. Chen, and L. H. Lee, "An effective learning procedure for multi-fidelity simulation optimization with ordinal transformation," in *2015 IEEE International Conference on Automation Science and Engineering (CASE)*, pp. 702–707, 2015.

[99] K. J. Chang, R. T. Haftka, G. L. Giles, and P.-J. Kao, "Sensitivity-based scaling for approximating structural response," *Journal of Aircraft*, vol. 30, no. 2, pp. 283–288, 1993.

[100] R. Vitali and B. Sankar, *Correction response surface design of stiffened composite panel with a crack.*
AIAA, 1999.

[101] R. Vitali, O. Park, R. T. Haftka, B. V. Sankar, and C. A. Rose, "Structural optimization of a hat-stiffened panel using response surfaces," *Journal of Aircraft*, vol. 39, no. 1, pp. 158–166, 2002.

[102] A. J. Keane, "Cokriging for robust design optimization," *AIAA Journal*, vol. 50, no. 11, pp. 2351–2364, 2012.

[103] Y. Zhang, N. H. Kim, C. Park, and R. T. Haftka, "Multifidelity surrogate based on single linear regression," *AIAA Journal*, vol. 56, no. 12, pp. 4944–4952, 2018.

[104] S. E. Gano, J. E. Renaud, and B. Sanders, "Hybrid variable fidelity optimization by using a kriging-based scaling function," *AIAA Journal*, vol. 43, no. 11, pp. 2422–2433, 2005.

[105] P. M.Zadeh and V. Toropov, "Multi-fidelity multidisciplinary design optimization based on collaborative optimization framework," in *9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, 2002.

[106] H. S. Kim, M. Koç, and J. Ni, "A hybrid multi-fidelity approach to the optimal design of warm forming processes using a knowledge-based artificial neural network," *International Journal of Machine Tools and Manufacture*, vol. 47, no. 2, pp. 211–222, 2007.

[107] M. Ghoreyshi, K. Badcock, and M. Woodgate, "Integration of multi-fidelity methods for generating an aerodynamic model for flight simulation," in *46th AIAA Aerospace Sciences Meeting and Exhibit*, 2008.

[108] S. Sarkar, "Multi-fidelity learning with heterogeneous domains," 2019.

[109] O. Day and T. M. Khoshgoftaar, "A survey on heterogeneous transfer learning," *J. Big Data*, vol. 4, pp. 1–42, 2017.

[110] K. E. Fisher, M. F. Herbst, and Y. M. Marzouk, "Multitask methods for predicting molecular properties from heterogeneous data," *J. Chem. Phys.*, vol. 161, p. 014114, 07 2024.

[111] R. Caruana, "Multitask learning," *Mach. Learn.*, vol. 28, no. 1, pp. 41–75, 1997.

[112] J. Baxter, "A model of inductive bias learning," *J. Artif. Intell. Res.*, vol. 12, pp. 149–198, 03 2000.

[113] E. V. Bonilla, K. Chai, and C. Williams, "Multi-task gaussian process prediction," in *Advances in Neural Information Processing Systems* (J. Platt, D. Koller, Y. Singer, and S. Roweis, eds.), vol. 20, Curran Associates, Inc., 2007.

[114] E. V. Bonilla, F. V. Agakov, and C. K. I. Williams, "Kernel multi-task learning using task-specific features," in *Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics* (M. Meila and X. Shen, eds.), vol. 2 of *Proceedings of Machine Learning Research*, (San Juan, Puerto Rico), pp. 43–50, PMLR, 21–24 Mar 2007.

[115] K. Yu, W. Chu, S. Yu, V. Tresp, and Z. Xu, "Stochastic relational models for discriminative link prediction," in *Advances in Neural Information Processing Systems 19: Proceedings of the 2006 Conference*, The MIT Press, 09 2007.

[116] R. A. Friesner, "Ab initio quantum chemistry: Methodology and applications," *Proc. Natl. Acad. Sci.*, vol. 102, no. 19, pp. 6648–6653, 2005.

[117] J. A. Pople, M. Head-Gordon, D. J. Fox, K. Raghavachari, and L. A. Curtiss, "Gaussian-1 theory: A general procedure for prediction of molecular energies," *J. Chem. Phys.*, vol. 90, pp. 5622–5629, 05 1989.

[118] L. A. Curtiss, K. Raghavachari, G. W. Trucks, and J. A. Pople, "Gaussian-2 theory for molecular energies of first- and second-row compounds," *J. Chem. Phys.*, vol. 94, pp. 7221–7230, 06 1991.

[119] E. Runge and E. K. U. Gross, "Density-Functional Theory for time-dependent systems," *Phys. Rev. Lett.*, vol. 52, pp. 997–1000, 03 1984.

[120] N. T. Maitra, "Perspective: Fundamental aspects of time-dependent density functional theory," *J. Chem. Phys.*, vol. 144, p. 220901, 06 2016.

[121] I. Purvis, George D. and R. J. Bartlett, "A full coupled-cluster singles and doubles model: The inclusion of disconnected triples," *J. Chem. Phys.*, vol. 76, pp. 1910–1918, 02 1982.

[122] R. J. Bartlett and M. Musiał, "Coupled-cluster theory in quantum chemistry," *Rev. Mod. Phys.*, vol. 79, pp. 291–352, 2 2007.

[123] S. R. Yost and M. Head-Gordon, "Efficient implementation of NOCI-MP2 using the resolution of the identity approximation with application to charged dimers and long C-C bonds in ethane derivatives," *J. Chem. Theory Comput.*, vol. 14, no. 9, pp. 4791–4805, 2018.

[124] L. Hu, X. Wang, L. Wong, and G. Chen, "Combined first-principles calculation and neural-network correction approach for heat of formation," *J. Chem. Phys.*, vol. 119, pp. 11501–11507, 12 2003.

[125] K. Hansen, G. Montavon, F. Biegler, S. Fazli, M. Rupp, M. Scheffler, O. A. Von Lilienfeld, A. Tkatchenko, and K.-R. Müller, "Assessment and validation of machine learning methods for predicting molecular atomization energies," *J. Chem. Theory Comput.*, vol. 9, no. 8, pp. 3404–3419, 2013.

[126] L. Ruddigkeit, R. van Deursen, L. C. Blum, and J.-L. Reymond, "Enumeration of 166 billion organic small molecules in the chemical universe database GDB-17," *J. Chem. Inf. Model.*, vol. 52, no. 11, pp. 2864–2875, 2012.

[127] R. Ramakrishnan, P. Dral, M. Rupp, and O. von Lilienfeld, "Quantum chemistry structures and properties of 134 kilo molecules," *Sci. Data*, vol. 1, p. 140022, 08 2014.

[128] L. A. Curtiss, P. C. Redfern, and K. Raghavachari, "Gaussian-4 theory using reduced order perturbation theory," *J. Chem. Phys.*, vol. 127, p. 124105, 09 2007.

[129] A. J. Cohen, P. Mori-Sánchez, and W. Yang, "Challenges for density functional theory," *Chem. Rev.*, vol. 112, p. 289–320, 01 2012.

[130] R. Ramakrishnan, M. Hartmann, E. Tapavicza, and O. A. von Lilienfeld, "Electronic spectra from TDDFT and machine learning in chemical space," *J. Chem. Phys.*, vol. 143, p. 084111, 08 2015.

[131] G. Pilania, J. E. Gubernatis, and T. Lookman, "Multi-fidelity machine learning models for accurate bandgap predictions of solids," *Comput. Mater. Sci.*, vol. 129, pp. 156–163, 03 2017.

[132] A. Palizhati, S. B. Torrisi, M. Aykol, S. K. Suram, J. S. Hummelshøj, and J. H. Montoya, "Agents for sequential learning using multiple-fidelity data," *Sci. Rep.*, vol. 12, no. 1, p. 4694, 2022.

[133] O. Egorova, R. Hafizi, D. C. Woods, and G. M. Day, "Multifidelity statistical machine learning for molecular crystal structure prediction," *J. Phys. Chem. A*, vol. 124, no. 39, pp. 8065–8078, 2020.

[134] A. Nandi, C. Qu, P. L. Houston, R. Conte, and J. M. Bowman, "Δ-machine learning for potential energy surfaces: a PIP approach to bring a DFT-based PES to CCSD (T) level of theory," *J. Chem. Phys.*, vol. 154, no. 5, p. 051102, 2021.

[135] C. Chen, Y. Zuo, W. Ye, X. Li, and S. P. Ong, "Learning properties of ordered and disordered materials from multi-fidelity data," *Nature Computational Science*, vol. 1, p. 46–53, 01 2021.

[136] C. Fare, P. Fenner, M. Benatan, A. Varsi, and E. O. Pyzer-Knapp, "A multi-fidelity machine learning approach to high throughput materials screening," *npj Computational Materials*, vol. 8, p. 1–9, 12 2022.

[137] Y. Yang, M. S. Eldred, J. Zádor, and H. N. Najm, "Multifidelity neural network formulations for prediction of reactive molecular potential energy surfaces," *J. Chem. Inf. Model.*, vol. 63, p. 2281–2295, 04 2023.

[138] L. C. Blum and J.-L. Reymond, "970 million druglike small molecules for virtual screening in the chemical universe database GDB-13," *J. Am. Chem. Soc.*, vol. 131, p. 8732, 2009.

[139] P. O. Dral, A. Owens, A. Dral, and G. Csányi, "Hierarchical machine learning of potential energy surfaces," *J. Chem. Phys.*, vol. 152, no. 20, p. 204110, 2020.

[140] G. Leen, J. Peltonen, and S. Kaski, "Focused multi-task learning in a Gaussian process framework," *Mach. Learn.*, vol. 89, pp. 157–182, 2012.

[141] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *J. Big Data*, vol. 3, pp. 1–40, 2016.

[142] V. Vinod, S. Maity, P. Zaspel, and U. Kleinekathöfer, "Multifidelity machine learning for molecular excitation energies," *J. Chem. Theory Comput.*, vol. 19, no. 21, pp. 7658–7670, 2023.

[143] V. Vinod, U. Kleinekathöfer, and P. Zaspel, "Optimized multifidelity machine learning for quantum chemistry," *Mach. Learn.: Sci. Technol.*, vol. 5, p. 015054, 03 2024.

[144] V. Vinod and P. Zaspel, "Benchmarking data efficiency in Δ-ML and multifidelity models for quantum chemistry," *arXiv*, no. 2410.11391, 2024.

[145] G. D. Scholes, G. R. Fleming, A. Olaya Castro, and R. van Grondelle, "Lessons from nature about solar light harvesting," *Nat. Chem.*, vol. 3, pp. 763–764, 2011.

[146] T. J. Zuehlsdorff, A. Montoya-Castillo, J. A. Napoli, T. E. Markland, and C. M. Isborn, "Optical spectra in the condensed phase: Capturing anharmonic and vibronic features using dynamic and static approaches," *J. Chem. Phys.*, vol. 151, p. 074111, 08 2019.

[147] E. Cignoni, V. Slama, L. Cupellini, and B. Mennucci, "The atomistic modeling of light-harvesting complexes from the physical models to the computational protocol," *J. Chem. Phys.*, vol. 156, p. 120901, 2022.

[148] S. Maity and U. Kleinekathöfer, "Recent progress in atomistic modeling of light-harvesting complexes: A mini review," *Photosynth. Res.*, vol. 156, pp. 147–162, 2023.

[149] F. Häse, S. Valleau, E. Pyzer-Knapp, and A. Aspuru-Guzik, "Machine learning exciton dynamics," *Chem. Sci.*, vol. 7, no. 8, pp. 5139–5147, 2016.

[150] M. S. Chen, T. J. Zuehlsdorff, T. Morawietz, C. M. Isborn, and T. E. Markland, "Exploiting machine learning to efficiently predict multidimensional optical spectra in complex environments," *J. Phys. Chem. Lett.*, vol. 11, pp. 7559–7568, 09 2020.

[151] A. Gupta, S. Chakraborty, D. Ghosh, and R. Ramakrishnan, "Data-driven modeling of $s_0 \rightarrow s_1$ excitation energy in the BODIPY chemical space: High-throughput computation, quantum machine learning, and inverse design," *J. Chem. Phys.*, vol. 155, p. 244102, 12 2021.

[152] Z. Chen, F. C. Bononi, C. A. Sievers, W.-Y. Kong, and D. Donadio, "UV-visible absorption spectra of solvated molecules by quantum chemical machine learning," *J. Chem. Theory Comput.*, vol. 18, pp. 4891–4902, 08 2022.

[153] E. Cignoni, L. Cupellini, and B. Mennucci, "Machine learning exciton Hamiltonians in light-harvesting complexes," *J. Chem. Theory Comput.*, 01 2023.

[154] P. O. Dral and M. Barbatti, "Molecular excited states through a machine learning lens," *Nat. Rev. Chem.*, vol. 5, no. 6, pp. 388–405, 2021.

[155] P. O. Dral, "Quantum chemistry in the age of machine learning," *J. Phys. Chem. Lett.*, vol. 11, no. 6, pp. 2336–2347, 2020.

[156] J. Behler, "Constructing high-dimensional neural network potentials: a tutorial review," *Int. J. Quantum Chem.*, vol. 115, no. 16, pp. 1032–1050, 2015.

[157] Q. Lin, Y. Zhang, B. Zhao, and B. Jiang, "Automatically growing global reactive neural network potential energy surfaces: A trajectory-free active learning strategy," *J. Chem. Phys.*, vol. 152, no. 15, p. 154104, 2020.

[158] N. Bernstein, G. Csányi, and V. L. Deringer, "De novo exploration and self-guided learning of potential-energy surfaces," *npj Computational Materials*, vol. 5, no. 1, p. 99, 2019.

[159] V. L. Deringer, C. J. Pickard, and G. Csányi, "Data-driven learning of total and local energies in elemental boron," *Phys. Rev. Lett.*, vol. 120, no. 15, p. 156001, 2018.

[160] P. O. Dral, F. Ge, B. X. Xue, Y.-F. Hou, M. Pinheiro, J. Huang, and M. Barbatti, *MLatom 2: An Integrative Platform for Atomistic Machine Learning*, pp. 13–53.
Springer International Publishing, 2022.

[161] L. Zhang, T. Su, M. Li, F. Jia, S. Hu, P. Zhang, and W. Ren, "Accurate band gap prediction based on an interpretable Δ-machine learning," *Mater. Today Commun.*, vol. 33, p. 104630, 2022.

[162] S. Verma, M. Rivera, D. O. Scanlon, and A. Walsh, "Machine learned calibrations to high-throughput molecular excited state calculations," *J. Chem. Phys.*, vol. 156, p. 134116, 2022.

[163] A. Patra, R. Batra, A. Chandrasekaran, C. Kim, T. D. Huan, and R. Ramprasad, "A multi-fidelity information-fusion approach to machine learn and predict polymer bandgap," *Comput. Mater. Sci.*, vol. 172, p. 109286, 2020.

[164] A. J. Therrien, M. J. Kale, L. Yuan, C. Zhang, N. J. Halas, and P. Christopher, "Impact of chemical interface damping on surface plasmon dephasing," *Faraday discuss.*, vol. 214, pp. 59–72, 2019.

[165] J. Westermayr, F. A. Faber, A. S. Christensen, O. A. von Lilienfeld, and P. Marquetand, "Neural networks and kernel ridge regression for excited states dynamics of CH2NH: From single-state to multi-state representations and multi-property machine learning models," *Mach. Learn.: Sci. Technol.*, vol. 1, no. 2, 2020.

[166] M. Rupp, M. R. Bauer, R. Wilcken, A. Lange, M. Reutlinger, F. M. Boeckler, and G. Schneider, "Machine learning estimates of natural product conformational energies," *PLoS Comput. Biol.*, vol. 10, no. 1, p. e1003400, 2014.

[167] M. Rupp, "Machine learning for quantum mechanics in a nutshell," *Int. J. Quantum Chem.*, vol. 115, no. 16, pp. 1058–1073, 2015.

[168] S. Maity, B. M. Bold, J. D. Prajapati, M. Sokolov, T. Kubař, M. Elstner, and U. Kleinekathöfer, "DFTB/MM molecular dynamics simulations of the FMO light-harvesting complex," *J. Phys. Chem. Lett.*, vol. 11, pp. 8660–8667, 2020.

[169] A. W. S. Da Silva and W. F. Vranken, "ACPYPE-Antechamber Python parser interface," *BMC Res. Notes*, vol. 5, no. 1, p. 367, 2012.

[170] M. J. Abraham, T. Murtola, R. Schulz, S. Páll, J. C. Smith, B. Hess, and E. Lindahl, "GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers," *SoftwareX*, vol. 1-2, pp. 19–25, 2015.

[171] M. Gaus, A. Goez, and M. Elstner, "Parametrization and benchmark of DFTB3 for organic molecules," *J. Chem. Theory Comput.*, vol. 9, no. 1, pp. 338–354, 2013.

[172] B. Hourahine, B. Aradi, V. Blum, F. Bonafé, A. Buccheri, C. Camacho, C. Cevallos, M. Y. Deshaye, T. Dumitrică, A. Dominguez, S. Ehlert, M. Elstner, T. Van Der Heide, J. Hermann, S. Irle, J. J. Kranz, C. Köhler, T. Kowalczyk, T. Kubař, I. S. Lee, V. Lutsker, R. J. Maurer, S. K. Min, I. Mitchell, C. Negre, T. A. Niehaus, A. M. N. Niklasson, A. J. Page, A. Pecchia, G. Penazzi, M. P. Persson, J. Řezáč, C. G. Sánchez, M. Sternberg, M. Stöhr, F. Stuckenberg, A. Tkatchenko, V. W.-Z. Yu, and T. Frauenheim, "DFTB+, a software package for efficient approximate density functional theory based atomistic simulations," *J. Chem. Phys.*, vol. 152, p. 124101, 03 2020.

[173] B. M. Bold, M. Sokolov, S. Maity, M. Wanko, P. M. Dohmen, J. J. Kranz, U. Kleinekathöfer, S. Höfener, and M. Elstner, "Benchmark and performance of long-range corrected time-dependent density functional tight binding (LC-TD-

DFTB) on rhodopsins and light-harvesting complexes," *Phys. Chem. Chem. Phys.*, vol. 22, pp. 10500–10518, 2020.

[174] A. Damjanović, I. Kosztin, U. Kleinekathöfer, and K. Schulten, "Excitons in a photosynthetic light-harvesting system: A combined molecular dynamics, quantum chemistry and polaron model study," *Phys. Rev. E*, vol. 65, p. 031919, 2002.

[175] L. González and R. Lindh, eds., *Quantum Chemistry and Dynamics of Excited States*. Wiley, 2020.

[176] B. Huang and O. A. von Lilienfeld, "Ab initio machine learning in chemical compound space," *Chem. Rev.*, vol. 121, pp. 10001–10036, aug 2021.

[177] E. O. Pyzer-Knapp, K. Li, and A. Aspuru-Guzik, "Learning from the harvard clean energy project: The use of neural networks to accelerate materials discovery," *Adv. Funct. Mater.*, vol. 25, no. 41, pp. 6495–6502, 2015.

[178] P. Raccuglia, K. C. Elbert, P. D. Adler, C. Falk, M. B. Wenny, A. Mollo, M. Zeller, S. A. Friedler, J. Schrier, and A. J. Norquist, "Machine-learning-assisted materials discovery using failed experiments," *Nature*, vol. 533, no. 7601, pp. 73–76, 2016.

[179] R. Ramprasad, R. Batra, G. Pilania, A. Mannodi-Kanakkithodi, and C. Kim, "Machine learning in materials informatics: recent applications and prospects," *npj Comput. 'Materials*, vol. 3, no. 1, p. 54, 2017.

[180] M. Rupp, O. A. von Lilienfeld, and K. Burke, "Guest Editorial: Special Topic on Data-Enabled Theoretical Chemistry," *J. Chem. Phys.*, vol. 148, p. 241401, 06 2018.

[181] K. T. Butler, D. W. Davies, H. Cartwright, O. Isayev, and A. Walsh, "Machine learning for molecular and materials science," *Nature*, vol. 559, no. 7715, pp. 2336–2347, 2018.

[182] O. A. von Lilienfeld, "Quantum machine learning in chemical compound space," *Angew. Chem. Int. Ed.*, vol. 57, no. 16, pp. 4164–4169, 2018.

[183] E. Cignoni, L. Cupellini, and B. Mennucci, "Machine Learning Exciton Hamiltonians in Light-Harvesting Complexes," *J. Chem. Theory Comput.*, 01 2023.

[184] S. Kondati Natarajan, T. Morawietz, and J. Behler, "Representing the potential-energy surface of protonated water clusters by high-dimensional neural network potentials," *Phys. Chem. Chem. Phys.*, vol. 17, pp. 8356–8371, 2015.

[185] S. Chmiela, A. Tkatchenko, H. E. Sauceda, I. Poltavsky, K. T. Schütt, and K.-R. Müller, "Machine learning of accurate energy-conserving molecular force fields," *Sci. Adv.*, vol. 3, no. 5, p. e1603015, 2017.

[186] S. Chmiela, H. E. Sauceda, K.-R. Müller, and A. Tkatchenko, "Towards exact molecular dynamics simulations with machine-learned force fields," *Nat. Commun.*, vol. 9, no. 1, p. 3887, 2018.

[187] C. Qu, P. L. Houston, R. Conte, A. Nandi, and J. M. Bowman, "Breaking the coupled cluster barrier for machine-learned potentials of large molecules: The case of 15-atom acetylacetone," *J. Phys. Chem. Lett.*, vol. 12, no. 20, pp. 4902–4909, 2021.

[188] D. T. Ahneman, J. G. Estrada, S. Lin, S. D. Dreher, and A. G. Doyle, "Predicting reaction performance in C–N cross-coupling using machine learning," *Science*, vol. 360, no. 6385, pp. 186–190, 2018.

[189] M. Gastegger, J. Behler, and P. Marquetand, "Machine learning molecular dynamics for the simulation of infrared spectra," *Chem. Sci.*, vol. 8, pp. 6924–6935, 2017.

[190] F. Jensen, *Introduction to Computational Chemistry*.
Wiley, 3rd edition ed., 2017.

[191] K. T. Schütt, H. E. Sauceda, P.-J. Kindermans, A. Tkatchenko, and K.-R. Müller, "Schnet–a deep learning architecture for molecules and materials," *J. Chem. Phys.*, vol. 148, no. 24, p. 241722, 2018.

[192] A. P. Bartók, R. Kondor, and G. Csányi, "On representing chemical environments," *Phys. Rev. B*, vol. 87, no. 18, p. 184115, 2013.

[193] K. Hansen, F. Biegler, R. Ramakrishnan, W. Pronobis, O. A. Von Lilienfeld, K.-R. Müller, and A. Tkatchenko, "Machine learning predictions of molecular properties: Accurate many-body potentials and nonlocality in chemical space," *J. Phys. Chem. Lett.*, vol. 6, no. 12, pp. 2326–2331, 2015.

[194] S. De, A. P. Bartók, G. Csányi, and M. Ceriotti, "Comparing molecules and solids across structural and alchemical space," *Phys. Chem. Chem. Phys.*, vol. 18, no. 20, pp. 13754–13769, 2016.

[195] A. P. Bartók, M. C. Payne, R. Kondor, and G. Csányi, "Gaussian approximation potentials: The accuracy of quantum mechanics, without the electrons," *Phys. Rev. Lett.*, vol. 104, no. 13, p. 136403, 2010.

[196] A. S. Christensen, L. A. Bratholm, F. A. Faber, and O. A. von Lilienfeld, "FCHL revisited: Faster and more accurate quantum machine learning," *J. Chem. Phys.*, vol. 152, p. 044107, jan 2020.

[197] D. Weininger, "SMILES, a chemical language and information system. 1. introduction to methodology and encoding rules," *J. Chem. Inf. Comput. Sci.*, vol. 28, no. 1, pp. 31–36, 1988.

[198] B. Kang, C. Seok, and J. Lee, "Prediction of molecular electronic transitions using random forests," *J. Chem. Inf. Model.*, vol. 60, no. 12, pp. 5984–5994, 2020.

[199] K. T. Schütt, F. Arbabzadah, S. Chmiela, K. R. Müller, and A. Tkatchenko, "Quantum-chemical insights from deep tensor neural networks," *Nat. Commun.*, vol. 8, no. 1, pp. 1–8, 2017.

[200] K. T. Schütt, M. Gastegger, A. Tkatchenko, K.-R. Müller, and R. J. Maurer, "Unifying machine learning and quantum chemistry with a deep neural network for molecular wavefunctions," *Nat. Comm.*, vol. 10, p. 5024, 11 2019.

[201] G. Pilania, C. Wang, X. Jiang, S. Rajasekaran, and R. Ramprasad, "Accelerating materials property predictions using machine learning," *Sci. Rep.*, vol. 3, no. 1, pp. 1–6, 2013.

[202] J. Carrete, W. Li, N. Mingo, S. Wang, and S. Curtarolo, "Finding unprecedentedly low-thermal-conductivity half-heusler semiconductors via high-throughput materials modeling," *Phys. Rev. X*, vol. 4, no. 1, p. 011019, 2014.

[203] J. S. Smith, O. Isayev, and A. E. Roitberg, "ANI-1: An extensible neural network potential with DFT accuracy at force field computational cost," *Chem. Sci.*, vol. 8, no. 4, pp. 3192–3203, 2017.

[204] X. Gao, F. Ramezanghorbani, O. Isayev, J. S. Smith, and A. E. Roitberg, "TorchANI: A free and open source pytorch-based deep learning implementation of the ani neural network potentials," *J. Chem. Inf. Modeling*, vol. 60, no. 7, pp. 3408–3415, 2020.

[205] O. T. Unke and M. Meuwly, "Physnet: A neural network for predicting energies, forces, dipole moments, and partial charges," *J. Chem. Theory Comput.*, vol. 15, no. 6, pp. 3678–3693, 2019.

[206] G. Sun and P. Sautet, "Toward fast and reliable potential energy surfaces for metallic Pt clusters by hierarchical delta neural networks," *J. Chem. Theory Comput.*, vol. 15, no. 10, pp. 5614–5627, 2019.

[207] Y. Liu and J. Li, "Permutation-Invariant-Polynomial neural-network-based δ-machine learning approach: A case for the HO2 self-reaction and its dynamics study," *J. Phys. Chem. Lett.*, vol. 13, no. 21, pp. 4729–4738, 2022.

[208] M. Ruth, D. Gerbig, and P. R. Schreiner, "Machine learning of Coupled Cluster (T)-energy corrections via delta (Δ)-learning," *J. Chem. Theory and Comp.*, vol. 18, no. 8, pp. 4846–4855, 2022.

[209] C. Reisinger, "Analysis of linear difference schemes in the sparse grid combination technique," *IMA J. Numer. Anal.*, vol. 33, pp. 544–581, 09 2012.

[210] J. Garcke, "Regression with the optimised combination technique," in *Proceedings of the 23rd international conference on Machine learning*, pp. 321–328, 2006.

[211] D. Quiñonero, C. Garau, A. Frontera, P. Ballester, A. Costa, and P. M. Deyà, "Structure and binding energy of anion-π and cation-π complexes: A comparison of mp2, ri-mp2, dft, and df-dft methods," *J. Phys. Chem. A*, vol. 109, no. 20, pp. 4632–4637, 2005.

[212] J. Pogrebetsky, A. Siklitskaya, and A. Kubas, "MP2-based correction scheme to approach the limit of a complete pair natural orbitals space in DLPNO-CCSD(T) calculations," *J. Chem. Theory Comput.*, vol. 19, no. 13, pp. 4023–4032, 2023.

[213] K. L. Bak, P. Jørgensen, J. Olsen, T. Helgaker, and W. Klopper, "Accuracy of atomization energies and reaction enthalpies in standard and extrapolated electronic wave function/basis set calculations," *J. Chem. Phys.*, vol. 112, pp. 9229–9242, 06 2000.

[214] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.

[215] M. Pinheiro Jr, S. Zhang, P. O. Dral, and M. Barbatti, "WS22 database, Wigner Sampling and geometry interpolation for configurationally diverse molecular datasets," *Sci. Data*, vol. 10, p. 95, 02 2023.

[216] M. P. Jr, S. Zhang, P. O. Dral, and M. Barbatti, "WS22 database: combining Wigner Sampling and geometry interpolation towards configurationally diverse molecular datasets," 08 2022.

[217] Y.-F. Hou, F. Ge, and P. O. Dral, "Explicit learning of derivatives with the KREG and pKREG models on the example of accurate representation of molecular potential energy surfaces," *J. Chem. Theory Comput.*, vol. 19, no. 8, pp. 2369–2379, 2023.

[218] K. Ravi, V. Fediukov, F. Dietrich, T. Neckel, F. Buse, M. Bergmann, and H.-J. Bungartz, "Multi-fidelity gaussian process surrogate modeling for regression problems in physics," *Mach. Learn.: Sci. Tech.*, vol. 5, p. 045015, 10 2024.

[219] J. S. Smith, R. Zubatyuk, B. Nebgen, N. Lubbers, K. Barros, A. E. Roitberg, O. Isayev, and S. Tretiak, "The ANI-1ccx and ANI-1x data sets, coupled-cluster and density functional theory properties for molecules," *Sci. Data*, vol. 7, p. 134, 05 2020.

[220] M. Nakata and T. Shimazaki, "PubChemQC project: A large-scale first-principles electronic structure database for data-driven chemistry," *J. Chem. Inf. Model.*, vol. 57, p. 1300–1308, 06 2017.

[221] L. Zhang, S. Zhang, A. Owens, S. N. Yurchenko, and P. O. Dral, "VIB5 database with accurate ab initio quantum chemical molecular potential energy surfaces," *Sci. Data*, vol. 9, p. 84, 03 2022.

[222] A. Fediai, P. Reiser, J. E. O. Peña, P. Friederich, and W. Wenzel, "Accurate GW frontier orbital energies of 134 kilo molecules," *Sci. Data*, vol. 10, p. 581, 09 2023.

[223] S. Nandi, T. Vegge, and A. Bhowmik, "MultiXC-QM9: Large dataset of molecular and reaction energies from multi-level quantum chemical methods," *Sci. Data*, vol. 10, p. 783, 11 2023.

[224] V. Vinod and P. Zaspel, "QeMFi: A multifidelity dataset of quantum chemical properties of diverse molecules (1.1.0) [dataset]," *Zenodo*, 10 2024.

[225] X. Zhu, K. C. Thompson, and T. J. Martínez, "Geodesic interpolation for reaction pathways," *J. Chem. Phys.*, vol. 150, p. 164103, 04 2019.

[226] L. McInnes, J. Healy, and J. Melville, "UMAP: Uniform manifold approximation and projection for dimension reduction," *arXiv*, no. 1802.03426, 2020.

[227] N. Menon and A. Basak, "Multi-fidelity surrogate with heterogeneous input spaces for modeling melt pools in laser-directed energy deposition," *Addit. Manuf.*, vol. 94, p. 104440, 2024.

[228] Y. Zhu, Y. Chen, Z. Lu, S. Pan, G.-R. Xue, Y. Yu, and Q. Yang, "Heterogeneous transfer learning for image classification," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 25, pp. 1304–1309, 2011.

[229] W. Li, L. Duan, D. Xu, and I. W. Tsang, "Learning with augmented features for supervised and semi-supervised heterogeneous domain adaptation," *IEEE Transactions on Pattern analysis and machine intelligence*, vol. 36, no. 6, pp. 1134–1148, 2013.

[230] X. Shi, Q. Liu, W. Fan, S. Y. Philip, and R. Zhu, "Transfer learning on heterogenous feature spaces via spectral transformation," in *2010 IEEE international conference on data mining*, pp. 1049–1054, IEEE, 2010.

[231] F. Häse, S. Valleau, E. Pyzer-Knapp, and A. Aspuru-Guzik, "Machine learning exciton dynamics," *Chem. Sci.*, vol. 7, pp. 5139–5147, 07 2016.

[232] D. Khatamsaz, B. Vela, and R. Arróyave, "Multi-objective Bayesian alloy design using multi-task Gaussian processes," *Mater. Lett.*, vol. 351, p. 135067, 11 2023.

[233] S. Heinen, D. Khan, G. F. von Rudorff, K. Karandashev, D. J. A. Arrieta, A. J. A. Price, S. Nandi, A. Bhowmik, K. Hermansson, and O. A. von Lilienfeld, "Reducing training data needs with minimal multilevel machine learning (M3L)," *Mach. Learn.: Sci. Technol.*, vol. 5, p. 025058, 06 2024.

[234] F. Noé, A. Tkatchenko, K.-R. Müller, and C. Clementi, "Machine learning for molecular simulation," *Annu. Rev. Phys. Chem.*, vol. 71, pp. 361–390, 2020.

[235] W. Samek, G. Montavon, S. Lapuschkin, C. J. Anders, and K.-R. Müller, "Explaining deep neural networks and beyond: A review of methods and applications," *Proceedings of the IEEE*, vol. 109, no. 3, pp. 247–278, 2021.

[236] J. Westermayr and P. Marquetand, "Deep learning for uv absorption spectra with schnarc: First steps toward transferability in chemical compound space," *J. Chem. Phys.*, vol. 153, p. 154112, 10 2020.

[237] Y. Zhang, S. Ye, J. Zhang, C. Hu, J. Jiang, and B. Jiang, "Efficient and accurate simulations of vibrational and electronic spectra with symmetry-preserving neural network models for tensorial properties," *J. Phys. Chem. B*, vol. 124, no. 33, pp. 7284–7290, 2020.

[238] M. Véril, A. Scemama, M. Caffarel, F. Lipparini, M. Boggio-Pasqua, D. Jacquemin, and P.-F. Loos, "QUESTDB: A database of highly accurate excitation energies for the electronic structure community," *WIREs Comput. Mol. Sci.*, vol. 11, no. 5, p. e1517, 2021.

[239] P.-F. Loos, A. Scemama, A. Blondel, Y. Garniron, M. Caffarel, and D. Jacquemin, "A mountaineering strategy to excited states: Highly accurate reference energies and benchmarks," *J. Chem. Theory Comput.*, vol. 14, no. 8, pp. 4360–4379, 2018.

[240] S. L. Krug, D. Khan, and O. A. von Lilienfeld, "Alchemical harmonic approximation based potential for iso-electronic diatomics: Foundational baseline for Δ-machine learning," *arXiv*, no. 2409.18007, 2024.

[241] I. Batatia, P. Benner, Y. Chiang, A. M. Elena, D. P. Kovács, J. Riebesell, X. R. Advincula, M. Asta, M. Avaylon, W. J. Baldwin, F. Berger, N. Bernstein, A. Bhowmik, S. M. Blau, V. Cărare, J. P. Darby, S. De, F. D. Pia, V. L. Deringer, R. Elijošius, Z. El-Machachi,

F. Falcioni, E. Fako, A. C. Ferrari, A. Genreith-Schriever, J. George, R. E. A. Goodall, C. P. Grey, P. Grigorev, S. Han, W. Handley, H. H. Heenen, K. Hermansson, C. Holm, J. Jaafar, S. Hofmann, K. S. Jakob, H. Jung, V. Kapil, A. D. Kaplan, N. Karimitari, J. R. Kermode, N. Kroupa, J. Kullgren, M. C. Kuner, D. Kuryla, G. Liepuoniute, J. T. Margraf, I.-B. Magdău, A. Michaelides, J. H. Moore, A. A. Naik, S. P. Niblett, S. W. Norwood, N. O'Neill, C. Ortner, K. A. Persson, K. Reuter, A. S. Rosen, L. L. Schaaf, C. Schran, B. X. Shi, E. Sivonxay, T. K. Stenczel, V. Svahn, C. Sutton, T. D. Swinburne, J. Tilly, C. van der Oord, E. Varga-Umbrich, T. Vegge, M. Vondrák, Y. Wang, W. C. Witt, F. Zills, and G. Csányi, "A foundation model for atomistic materials chemistry," *arXiv*, no. 2401.00096, 2024.

[242] R. Izsák, "Single-reference coupled cluster methods for computing excitation energies in large molecules: the efficiency and accuracy of approximations," *WIREs Comput. Mol. Sci.*, vol. 10, no. 3, p. e1445, 2020.

[243] I. Sandler, J. Chen, M. Taylor, S. Sharma, and J. Ho, "Accuracy of DLPNO-CCSD (T): Effect of basis set and system size," *J. Phys. Chem. A*, vol. 125, no. 7, pp. 1553–1563, 2021.

[244] S. Stocker, G. Csányi, K. Reuter, and J. T. Margraf, "Machine learning in chemical reaction space," *Nat. Comm.*, vol. 11, p. 5505, 10 2020.

[245] P. Schwaller and T. Laino, *Data-Driven Learning Systems for Chemical Reaction Prediction: An Analysis of Recent Approaches*, ch. 4, pp. 61–79.
American Chemical Society, 2019.

[246] M. Ruth, D. Gerbig, and P. R. Schreiner, "Machine Learning for Bridging the Gap between Density Functional Theory and Coupled Cluster Energies," *J. Chem. Theory and Comp.*, vol. 19, no. 15, pp. 4912–4920, 2023.

[247] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data. Eng.*, vol. 22, no. 10, pp. 1345–1359, 2010.

[248] F. H. Vermeire and W. H. Green, "Transfer learning for solvation free energies: From quantum chemistry to experiments," *Chem. Eng. J.*, vol. 418, p. 129307, 2021.

[249] C. A. Grambow, Y.-P. Li, and W. H. Green, "Accurate thermochemistry with small data sets: A bond additivity correction and transfer learning approach," *J. Phys. Chem. A*, vol. 123, no. 27, pp. 5826–5835, 2019.

[250] V. Gupta, K. Choudhary, F. Tavazza, C. Campbell, W.-k. Liao, A. Choudhary, and A. Agrawal, "Cross-property deep transfer learning framework for enhanced predictive analytics on small materials data," *Nat. Comm.*, vol. 12, 11 2021.

[251] P. C. St. John, Y. Guan, Y. Kim, B. D. Etz, S. Kim, and R. S. Paton, "Quantum chemical calculations for over 200,000 organic radical species and 40,000 associated closed-shell molecules," *Sci. Data*, vol. 7, no. 1, p. 244, 2020.

[252] K. T. Schütt, H. Glawe, F. Brockherde, A. Sanna, K.-R. Müller, and E. K. Gross, "How to represent crystal structures for machine learning: Towards fast prediction of electronic properties," *Phys. Rev. B*, vol. 89, no. 20, p. 205118, 2014.

[253] M. Sokolov, B. M. Bold, J. J. Kranz, S. Höfener, T. A. Niehaus, and M. Elstner, "Analytical time-dependent long-range corrected density functional tight binding (td-lc-dftb) gradients in DFTB+: Implementation and benchmark for excited-state geometries and transition energies," *J. Chem. Theory Comput.*, vol. 17, pp. 2266–2282, 2021.

[254] G. C. Cawley and N. L. Talbot, "On over-fitting in model selection and subsequent selection bias in performance evaluation," *J. Mach. Learn. Res.*, vol. 11, pp. 2079–2107, 2010.

[255] L. González and R. Lindh, *Quantum chemistry and dynamics of excited states: methods and applications*.
John Wiley & Sons, Ltd, 2020.

[256] M. Born and R. Oppenheimer, "Zur Quantentheorie der Molekeln," *Annalen der Physik*, vol. 389, no. 20, pp. 457–484, 1927.

[257] W. Kohn, "Nobel lecture: Electronic structure of matter—wave functions and density functionals," *Rev. Mod. Phys.*, vol. 71, pp. 1253–1266, Oct 1999.

[258] H. S. Yu, S. L. Li, and D. G. Truhlar, "Perspective: Kohn-Sham density functional theory descending a staircase," *J. Chem. Phys.*, vol. 145, p. 130901, 10 2016.

[259] P. Atkins and R. Friedman, *Molecular Quantum Mechanics*.
Oxford University Press, 11 2010.

[260] T. Helgaker, P. Jørgensen, and J. Olsen, *Coupled-Cluster Theory*, p. 648–723.
Wiley, 08 2000.

[261] P. Hohenberg and W. Kohn, "Inhomogeneous electron gas," *Phys. Rev.*, vol. 136, p. B864–B871, 11 1964.

NOTES

These pages may be used by the reader in order to make notes.