Master thesis
Zur Erlangung des akademischen Grades
Master of Science (M.Sc.)

Oxygen vacancy ordering in $YBa_2Cu_3O_{6+x}$: An *ab initio* study using cluster expansion

eingereicht von: Noah Alexy Dasch (582551)

Gutachter_innen: Prof. Dr. Dr. h. c. Claudia Draxl
Prof. Dr. Igor Sokolov

Eingereicht am Institut für Physik der Humboldt-Universität zu Berlin am:
01.09.2025

Contents

Chapter 1

Introduction

Superconductivity is characterized by the loss of electrical resistance when a material is cooled below its critical temperature T_c. This phenomenon is clearly desirable for numerous technical applications. However, the earliest identified superconductors needed to be cooled with liquid helium to temperatures near absolute zero, which limited their practical applications. This sparked interest in the search for materials with higher transition temperatures, so called high-T_c superconductors. Shortly after the discovery of high-T_c superconductivity in a La-Ba-Cu-O compound [1], the superconductor $YBa_2Cu_3O_{6+x}$ was discovered in 1987, exhibiting remarkable transition temperatures of up to around 92 K, for $x = 1$ [2–6]. It was the first observation of superconductivity above the boiling point of liquid nitrogen (77 K) [2] and expanded the practical applications of superconductors significantly, as liquid nitrogen is much cheaper than liquid helium [2, 7]. Since then, many other high-T_c superconductors were discovered [8], but up to today there is no generally accepted theory of high-T_c superconductivity [3, 9]. Yet, it is known that in the cuprates, their CuO_2 planes and the ordering of oxygen atoms and vacancies play a key role [3, 10, 11]. This work is dedicated to studying the effects of oxygen vacancy ordering on the energies and lattice constants of $YBa_2Cu_3O_{6+x}$ and to search for stable ground states in the range of $0 \leq x \leq 1$.

One approach to studying $YBa_2Cu_3O_{6+x}$, is the Hubbard model, which incorporates electron-electron interaction in the parameters of a model Hamiltonian. We choose another approach, employing *ab initio* calculations with Density Functional Theory (DFT), which is a powerful tool to investigate material properties [12–14]. The two methods can be combined within the framework of DFT+U. The introduction of a Hubbard U parameter can correct the band structure in DFT calculations of $YBa_2Cu_3O_6$, which fails to exhibit a band gap when local or semi-local xc-functionals are used, although more complex functionals, like SCAN [15], accurately reproduce it [16, 17]. We are mainly interested in the system's energetics and lattice constants. A DFT+U study revealed that choosing $U > 0$ did not improve the lattice constants compared to experimental values [18]. Also, the choice of U is not obvious [19], and its introduction limits the predictive power of the calculations [16]. Consequently, we decided against incorporating a Hubbard parameter and instead prioritize a true *ab initio* approach, ensuring that no experimental values enter our DFT calculations. The calculations are performed with the all-electron code FHI-aims [20] and the xc-functional PBEsol [21,22]. The results are combined with cluster expansion (CE) [23] and machine learning methods, using the CE code `CELL` [24] and the `scikit-learn` python library [25]. This allows us to explore a vast configurational space. We optimize CE models for energy and lattice

constants and predict the energies of more than 48,800 configurations across 29 compositions. At least six ground states are identified at a temperature of $T = 0$ K. We perform statistical thermodynamics simulations, using Metropolis Monte Carlo sampling [26, 27], to examine the properties at finite temperatures, particularly the behavior of lattice constants at varying oxygen concentrations.

Chapter 2

Physical properties of $YBa_2Cu_3O_{6+x}$

This chapter reviews physical properties of $YBa_2Cu_3O_{6+x}$ with $0 \leq x \leq 1$. In particular, we briefly discuss its high-T_c superconductivity and examine the physical properties studied in this work, such as its crystal structure and the tetragonal-orthorhombic phase transition and summarize corresponding experimental results from the literature [28, 29] for later comparison to our results.

2.1 High-T_c superconductivity

As mentioned in the Introduction, the Y-Ba-Cu-O compound, synthesized for the first time in 1987 [2], was the first superconductor to be discovered with a transition temperature above the boiling point of liquid nitrogen [2]. This was a breakthrough in the field of superconductivity [2,7], enabled by the discovery of high-T_c superconductivity in a Ba-La-Cu-0 compound by Bednorz and Müller [1] one year prior. While conventional superconductivity can be described in terms of electron-phonon coupling in the Bardeen–Cooper–Schrieffer (BCS) theory [30], this is not the case for cuprate high-T_c superconductors, such as YBCO compounds [9]. Those materials are structurally highly anisotropic, and also the resistivity in their CuO_2 planes is much smaller than the out of plane resistivity [3, 31]. Additionally, they are characterized by strong electron correlation, i.e. electrons are far from being considered as independent particles, and the requirements of the standard BCS theory [30] are not fulfilled. The high transition temperatures cannot be explained solely by electron-phonon coupling [32] and currently, no generally accepted universal theory for high-T_c superconductivity exists [3, 9].

However, it is known that superconductivity in $YBa_2Cu_3O_{6+x}$ (and other cuprate superconductors) is related to the concentration and ordering of the oxygen atoms [3, 10, 11, 33–36]. Experimental results show an increase in transition temperature with increasing oxygen content [29, 37] as well as a time delayed increase in transition temperature that is explained by the ordering of oxygen atoms and vacancies in the material [29, 33, 35]. For $YBa_2Cu_3O_{6+x}$, stable structures could be synthesized for x in the range $0 \leq x \leq 1$ [38]. Starting from an antiferromagnetic insulating parent phase at $YBa_2Cu_3O_6$ ($x = 0$), the oxygen concentration is increased and superconductivity becomes possible from about $YBa_2Cu_3O_{6.35}$ ($x = 0.35$) [34] with increasing transition temperature reaching $T_c \approx 92$ K for $YBa_2Cu_3O_7$ [3–6]. Our work is limited to this range of oxygen concentrations.

Table 2.1: The Wyckoff positions of the tetragonal phase $YBa_2Cu_3O_6$ (top) belonging to the P4/mmm space group (123) and the orthorhombic phase $YBa_2Cu_3O_7$ (bottom) belonging to the Pmmm space group (47) based on the description of Rayaprol and Kuberkar [39].

Atomic species	**Wyckoff site**	**Position**
$YBa_2Cu_3O_6$		
Y	1d	(1/2, 1/2, 1/2)
Ba	2h	(1/2, 1/2, 0.1914)
Cu1	1a	(0, 0, 0)
Cu2	2g	(0, 0, 0.3590)
O1	4i	(0, 1/2, 0.3790)
O2	2g	(0, 0, 0.1508)
$YBa_2Cu_3O_7$		
Y	1h	(1/2, 1/2, 1/2)
Ba	2t	(1/2, 1/2, 0.1854)
Cu1	1a	(0, 0, 0)
Cu2	2q	(0, 0, 0.3555)
O1	1e	(0, 1/2, 0)
O2	2q	(0, 0, 0.1568)
O3	2r	(0, 1/2, 0.3790)
O4	2s	(1/2, 0, 0.3781)

Superconductivity is not explicitly investigated in this work, but by studying the effects of oxygen vacancy ordering on the materials properties, we aim at contributing to a better understanding of cuprate high-T_c superconductors.

2.2 Crystal structure

We now take a closer look at the $YBa_2Cu_3O_6$ and $YBa_2Cu_3O_7$ which define the limits of the considered oxygen concentration range, and which will be referred to as "reference structures" from now on. $YBa_2Cu_3O_6$ has tetragonal crystal symmetry, such that lattice constants $a = b \neq c$, and belongs to space group P4/mmm (123). Its Wyckoff positions are shown in the upper part of table 2.1 [28, 39]. $YBa_2Cu_3O_7$ has orthorhombic crystal symmetry, such that all lattice constants are unequal $a \neq b \neq c$. It belongs to space group Pmmm (47) and its Wyckoff positions are shown in the lower part of table 2.1 [28, 39].

$YBa_2Cu_3O_{6+x}$ is characterized by a layered perovskite structure, where copper oxide planes alternate with planes containing yttrium and barium cations [3]. The respective unit cells are illustrated in Fig. 2.1. Yttrium atoms are depicted in cyan, barium atoms in green, copper atoms in bronze and oxygen atoms in red, while vacancies are represented as empty circles. The CuO_2 planes are indicated with a label next to them. A single yttrium atom is located at the center of the conventional unit cells, with no oxygen atoms present in the plane along the a and b directions sur-

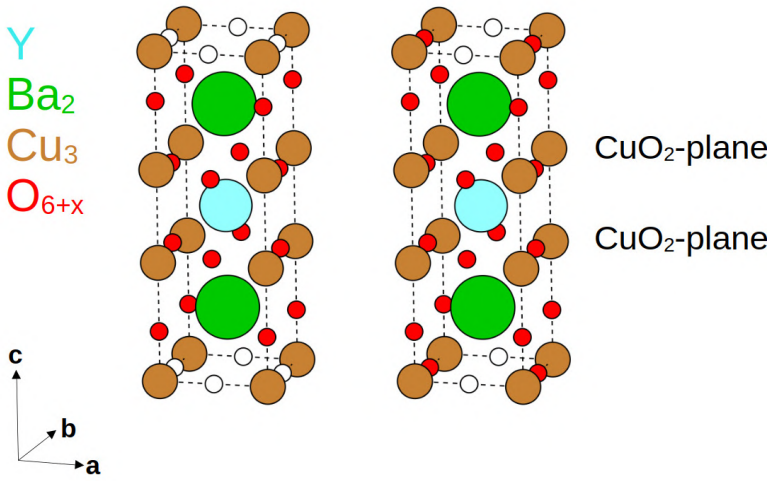

Figure 2.1: The conventional unit cells of $YBa_2Cu_3O_6$ (left) and $YBa_2Cu_3O_7$ (right) visualized by using ASE [40] and matplotlib [41] are shown. Yttrium atoms are colored in cyan, barium atoms in green, copper atoms in bronze and oxygen atoms in red. Vacancies are shown as empty circles. CuO_2 planes are indicated with a label next to them. The differences in occupation between both structures can be seen in the bottom/top cell of the unit cells, where for $YBa_2Cu_3O_7$ two oxygen atoms (one per primitive unit cell) are inserted.

rounding it. In contrast, four oxygen atoms are found in the corresponding planes around the two barium atoms. As demonstrated in Fig. 2.1, $YBa_2Cu_3O_6$ (illustrated on the the left) lacks oxygen atoms within the planes between the barium atoms (the bottom and top plane of the illustrated unit cells). The introduction of oxygen atoms into these planes results in the formation of Cu-O chains along the b direction, as is visible in the structure of $YBa_2Cu_3O_7$ (see right-hand side).

In Tab. 2.1, Cu1 corresponds to copper atoms in these planes, which in $YBa_2Cu_3O_7$ are part of the Cu-O chains, while Cu2 atoms belong to the CuO_2 planes, present in both structures. The oxygen atoms, added by doping $YBa_2Cu_3O_6$, introduce hole charge carriers that distribute non-uniformly between the CuO_2 planes and Cu-O chains [31]. $YBa_2Cu_3O_7$ is considered to be slightly overdoped, such that the maximum transition temperature of $T_c \approx 94$ K is not reached at $x = 1$ exactly, but shortly before [3, 42]. $YBa_2Cu_3O_7$ is referred to as ortho-I phase, whereas the ortho-II phase is an orthorhombic phase corresponding to the composition $YBa_2Cu_3O_{6.5}$ [3]. In the ortho-II phase, Cu-O chains alternate with Cu chains, meaning that one of the oxygen sites occupied $YBa_2Cu_3O_7$ remains empty.

2.3 Tetragonal-orthorhombic phase transition

As mentioned before, the doping of oxygen atoms in the insulator parent phase $YBa_2Cu_3O_6$ enables superconductivity starting approximately at values of $x = 0.35$ [3, 34]. All phases for which superconductivity appears are orthorhombic and the tetragonal to orthorhombic phase transition takes place close to oxygen concentrations for which superconductivity becomes possible [31, 43].

Figure 2.2 depicts experimental results for the lattice constants a (red), b (blue) and c (gray) with

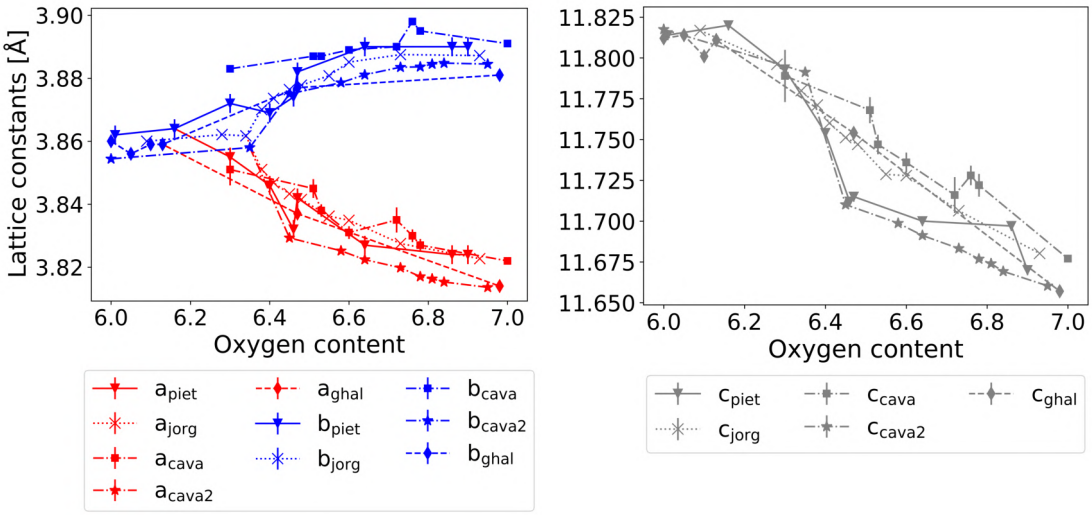

Figure 2.2: Experimental results for the lattice constants of $YBa_2Cu_3O_{6+x}$. Results by Pietraszko et al. [28] are shown as triangles and solid lines, results by Jorgensen et al. [34] as crosses and dotted lines, results by Cava et al. [29] as squares and dashdotted lines. Later results by Cava et al. [43] are shown as stars and dashdotted lines. Results by Gallagher et al. [38] are shown as diamonds and dashed lines. On the left hand side, lattice constants a (red) and b (blue) are shown and on the right hand side lattice constant c (gray). Tetragonal symmetry is broken with increasing oxygen content.

respect to oxygen content, revealing the phase transition. There is a notable variability between the results of different experimental works; therefore, we show the results from five studies that reflect this variability. Pietraszko et al. [28] fired samples at temperatures from 450 °C to 950 °C [28] and measured at room temperature. Their results are shown as triangles and solid lines. Jorgensen et al. performed neutron powder diffraction on samples quenched from diverse oxygen partial pressures at 520°C into liquid nitrogen [34]. Their results are shown as crosses and dotted lines. Cava et al. reported measurements in two papers in 1987 [29] and 1990 [43]. In the first paper, measurements are performed using neutron diffraction at room temperature on samples annealed between 360 °C and 520 °C. These results are shown as dash-dotted lines and squares. In the second work, measurements are performed at $T = 5$ K on samples annealed at 440 °C. These results are shown as stars and dash-dotted lines. Results by Ghallager et al. [38] are taken from Fig. 7 of their paper and digitized. The data is shown as diamonds and dashed lines. The corresponding standard deviations for each experiment are provided as error bars. In some cases, the error bars are smaller than the plot symbols. The figure shows that small values of x in $YBa_2Cu_3O_{6+x}$ correspond to phases with tetragonal symmetry, such that lattice constants $a = b$. The oxygen concentration, for which a phase transition is observed, varies between the experiments: The results of Pietraszko et al. show a break of tetragonal symmetry between $0.16 < x < 0.3$ [28], whereas Cava et al. [43] still observe tetragonal symmetry around $x \approx 0.35$ and find orthorhombic symmetry at $x \geq 0.45$. However, in their previous work [29] they found orthorhombic symmetry around $x = 0.3$. Jorgensen et al. find tetragonal symmetry at $x \approx 0.34$ that is broken for their next data point at $x \approx 0.38$ [34]. The results by Ghallager et al. show that the transition occurs between $0.13 < x < 0.47$ [38]. Concerning the lattice parameter c, Cava et al. in their later work and Pietraszko find a sharp decrease in slope around $x \approx 0.4$, becoming less steep around $x \approx 0.45$ [28,43], whereas a more linear behavior is found in the other experiments [29,34,38]. The

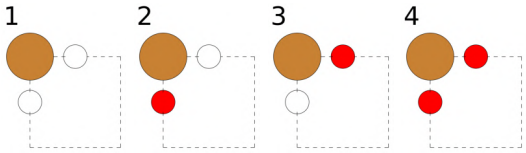

Figure 2.3: Bottom plane of the $YBa_2Cu_3O_{6+x}$ parent lattice. Copper atoms are shown in bronze; their occupation is fixed. The substitutional sites can be either vacant (empty circles) or occupied by an oxygen atom (red filled circles). There are two substitutional sites, resulting in four possible configurations. Cases two and three are equivalent due to the symmetry of the parent lattice. Case four is not part of the considered concentration range.

following lattice constants trends with increasing oxygen content are clearly visible in all results: Lattice constant b increases with increasing oxygen concentration, whereas lattice constants a and c decrease with increasing oxygen concentration. The overall cell volume decreases [28,29,34,38,43].

2.4 Combinatorial explosion

The methods used to study $YBa_2Cu_3O_{6+x}$ in this work will be outlined in the next chapter. First, we introduce the *combinatorial explosion* in the feature space of $YBa_2Cu_3O_{6+x}$, to motivate the choice of our methods. DFT calculations, which are used to determine material properties and will also be introduced in the next chapter, are computationally demanding for large unit cells. This is particularly relevant in this thesis, as the explored configuration space is extensive, making it impossible to be studied solely with DFT calculations. To illustrate the necessity of applying cluster expansion combined with machine learning methods, we will now demonstrate the *combinatorial explosion*.

With regard to $YBa_2Cu_3O_{6+x}$, there are only two lattice sites per smallest primitive unit cell, whose occupancy is not fixed. Depending on the configuration, these sites (called substitutional sites) are either occupied by an oxygen atom or vacant. The smallest primitive unit cell of configurations of $YBa_2Cu_3O_{6+x}$, which corresponds to the unit cell of $YBa_2Cu_3O_6$, is called parent lattice. Since both substitutional sites are in the same plane of the crystal, the one between the barium atoms, we only focus at this plane in the following. It contains copper atoms, whose occupancy is fixed, as well as the substitutional sites. In Fig. 2.3, as before, copper atoms are shown in bronze and substitutional sites are depicted by an empty circle for a vacancy or a red filled circle for an oxygen atom. Considering the parent lattice, there are $2^2 = 4$ possible configurations: Both substitutional sites are vacant ($x = 0$, panel 1); one substitutional site is vacant and the other one is occupied by an oxygen atom ($x = 1$, panels 2 and 3); or both substitutional sites are occupied by oxygen atoms ($x = 2$, panel 4). Due to the symmetry of the parent lattice, cases two and three are equivalent. As discussed previously, the relevant concentration range corresponds to values of x between 0 and 1, so there are only two unique possibilities in this range, which relate to the structures $YBa_2Cu_3O_6$ and $YBa_2Cu_3O_7$. These two structures will be used as references to define the energy of mixing, which is introduced in a later chapter. We can calculate their properties using DFT. To consider structures with intermediate oxygen concentrations, we need to construct larger super cells. Let us consider a super cell four times as large as the parent lattice, for which there are $2^{2\cdot4} = 2^8 = 256$ possible configurations. Considering further the crystal symmetry and the relevant concentration range, around 100 unique configurations exist. Regarding super cells up to nine times as large

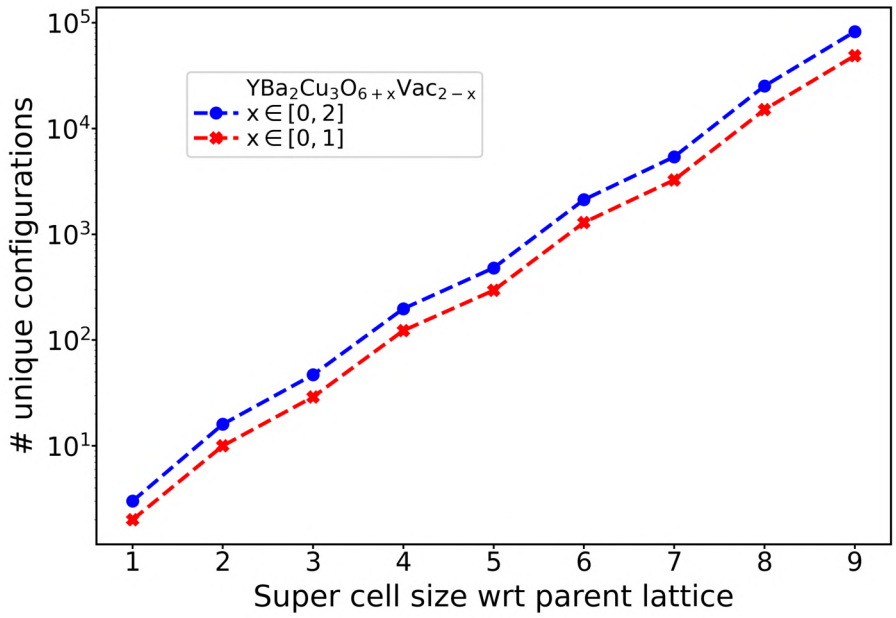

Figure 2.4: Number of unique configurations with logarithmic scaling in relation to different sized super cells, demonstrating the combinatorial explosion for $YBa_2Cu_3O_{6+x}$. The blue line shows the number of unique configurations for the whole concentration range, meaning that we go from the case, where all substitutional sites are empty to the case where all substitutional sites are occupied by oxygen, *i.e.*, $0 \leq x \leq 2$. The red line depicts the number of unique configurations for the reduced concentration range $0 \leq x \leq 1$ that is studied in this thesis.

as the parent lattice, there are $2^{2 \cdot 9} = 2^{18} = 262,144$ possible configurations, from which more than 80,000 are unique and more than 46,000 unique configurations are within the relevant oxygen concentration range. This sharp increase in the number of configurations with increasing super cell size is typical of substitutional systems and is called combinatorial explosion. It shown in detail in Fig. 2.4 for $YBa_2Cu_3O_{6+x}$. The blue line represents the number of unique configurations for the whole oxygen concentration range ($0 \leq x \leq 2$), while the red line corresponds to the relevant concentration range ($0 \leq x \leq 1$). A logarithmic y-axis scale is employed.

For even larger super cell sizes, as needed for considering finite temperature properties, it becomes practically impossible to perform this many DFT calculations. Instead, we make use of the CE method: Rather than carrying out one calculation for each possible structure, we perform calculations fo a limited amount of structures: the training set. We build a model to explore the relationship between a property and the configuration, and to predict the property for materials beyond the training set. The latter is accomplished using machine learning methods. This way, one can achieve a numerical precision similar to that of DFT calculations, while being able to predict the properties of a large number of structures in a numerically efficient way. These methods are introduced in more detail in the next chapter.

Chapter 3

Theoretical background

In this chapter, we outline the theories and methods employed in this work. We introduce the mathematical formalism of CE and give an overview of machine learning methods that are applied in this work. We explain for which use cases the nonlinear cluster expansion method [44] is recommended and introduce approaches for the non trivial task of structure selection within cluster expansion [45, 46]. Finally, we explain statistical thermodynamics simulations based on Metropolis Monte Carlo sampling [26, 27]. The variety of methods introduced in this chapter lays the foundation for the results discussed in Chap. 4.

3.1 Density functional theory

A solid is made up of atoms that are arranged in a crystalline periodic structure. To describe the physical properties of a material, in particular its electronic structure, we need to solve the many-body Schrödinger equation. Even when splitting the problem into an electronic and a nuclear contribution, we still need to deal with many-electron wavefunctions. However, in the case of solids, we cannot solve this equation using many-body wavefunction methods. A clear demonstration of this is known from the Nobel lecture of Walter Kohn, where it was shown that the wavefunction of a material with one thousand or more electrons is too large to even be stored in the whole universe [47].

DFT is based on the idea that instead of dealing with many body-wavefunctions that depend on three spatial coordinates for each electron, we can work with the electron density that only depends on three spatial coordinates in total. The theorems of Hohenberg and Kohn [12] laid the groundwork for DFT. They showed that, for a system of interacting electrons, the total energy can be expressed as a unique functional of the electron density, $n(\boldsymbol{r})$, that is minimized by the ground state density $n_0(\boldsymbol{r})$ [12]. This functional enables us to describe all kinds of many-electron systems in terms of electron densities, but unfortunately its exact form remains unknown. An approach of how to approximate this functional was formulated later by Kohn and Sham [13]. They suggested to use an auxiliary system of noninteracting particles whose single particle functions, u_i, produce

the same electron density $n(\boldsymbol{r})$ as the interacting system, such that:

$$n(\boldsymbol{r}) = \sum_{i=1}^{N} |u_i(\boldsymbol{r})|^2 = \sum_{i=1}^{N} u_i^* u_i. \tag{3.1}$$

We then start from the energy functional:

$$\begin{aligned} E[n(\boldsymbol{r})] = &- \frac{1}{2} \int d\boldsymbol{r} \sum_{i=1}^{N} u_i^*(\boldsymbol{r}) \nabla^2 u_i(\boldsymbol{r}) + \int d\boldsymbol{r}\ n(\boldsymbol{r})\ V_{ext}(\boldsymbol{r}) \\ &+ \frac{1}{2} \int d\boldsymbol{r}\ n(\boldsymbol{r}) \int d\boldsymbol{r}' \frac{n(\boldsymbol{r}')}{|\boldsymbol{r} - \boldsymbol{r}'|} + \int d\boldsymbol{r}\ n(\boldsymbol{r})\ \epsilon_{xc}(\boldsymbol{r}). \end{aligned} \tag{3.2}$$

The first term is the kinetic energy term for the single particle functions, u_i, given in atomic units. The second part comes from an external potential that summarizes the effects by the nuclei and external fields on the electrons. The third part results from the classical Coulomb interaction between the electrons, and the last part summarizes quantum mechanical exchange and correlation effects among the electrons and a correction for the kinetic energy term. The exact form of ϵ_{xc} is unknown and it has to be approximated. Varying Eq. 3.2 with respect to the single particle functions and the constraint that the density must preserve the number of electrons ($\int d\boldsymbol{r}\ n(\boldsymbol{r}) = N$) , leads to the Kohn-Sham equation:

$$\hat{H}_{KS}\ u_i(\boldsymbol{r}) = \epsilon_i\ u_i(\boldsymbol{r}), \tag{3.3}$$

with the Kohn-Sham Hamiltonian

$$\hat{H}_{KS} = -\frac{1}{2}\nabla^2 + V_{ext}(\boldsymbol{r}) + V_{Hartree}(\boldsymbol{r}) + V_{xc}(\boldsymbol{r}). \tag{3.4}$$

The Hartree potential describes the potential due to the classical Coulomb interaction, and the exchange-correlation potential V_{xc} contains exchange and correlation effects, as well as the kinetic energy correction. The Kohn-Sham equation is solved in a self-consistent manner, meaning that we start with an initial density for which we calculate the potentials and solve the Kohn-Sham equation to get the eigenvalues ϵ_i and the single particle functions u_i. From these we calculate the density again and check if it agrees with our starting point density up to a certain precision. If not, we have to iterate the process again, until the density is converged. The exchange-correlation potential is calculated as:

$$V_{xc}(\boldsymbol{r}) = \frac{\delta E_{xc}}{\delta n(\boldsymbol{r})} = \epsilon_{xc}(\boldsymbol{r}, [n]) + \int d\boldsymbol{r}' n(\boldsymbol{r}') \frac{\delta \epsilon_{xc}(\boldsymbol{r}', [n])}{\delta n(\boldsymbol{r})}. \tag{3.5}$$

Since we do not know the exact form of ϵ_{xc}, the exchange-correlation potential needs to be approximated. Common approaches include the local density approximation (LDA) [48], based on exchange correlation effects from the homogeneous electron gas, which are incorporated in ϵ_{xc}^{homo}:

$$E_{xc}^{LDA} = \int d\boldsymbol{r}\ n(\boldsymbol{r})\ \epsilon_{xc}^{homo}[n(\boldsymbol{r})] \tag{3.6}$$

and generalized gradient approximations. The latter take into account not only the local density, but also its gradient:

$$E_{xc}^{GGA} = \int d\boldsymbol{r}\ f[n(\boldsymbol{r}), \nabla n(\boldsymbol{r})]. \tag{3.7}$$

In this work, the PBEsol xc-functional [21, 22] is used which belongs to the generalized gradient approximations.

3.2 Cluster expansion and machine learning

As outlined in Sec. 2.4, due to the combinatorial explosion, it is not possible to study the vast configuration space of $YBa_2Cu_3O_{6+x}$ solely with DFT. Instead, we make use of cluster expansion combined with machine learning methods, which are introduced in the following sections.

3.2.1 Mathematical formalism of cluster expansion

To describe the mathematical formalism of the CE method, we follow the description in a paper by Sanchez et al. [23], who formulated a generalized method applicable to multi-component systems, as well as the paper by Rigamonti et al. [24], which explains the implementation of the method in the code `CELL`, that is used in this work. A crystal with N lattice sites is considered, each of which is assigned an index i. For a general system, where M_i possible atomic species could occupy site i, one defines the configuration vector $\boldsymbol{\sigma}^T = (\sigma_1, \sigma_2, ..., \sigma_N)$, where σ_i indicates which species occupies the corresponding site. There are several options for the values of σ_i, depending on the chosen basis set. Sanchez et al. [23] originally proposed to use integer values ranging from $-m_i$ to $+m_i$ with $m_i = \lfloor \frac{M_i}{2} \rfloor$. For an odd number of possible atomic species, zero is included, for an even number, it is excluded. Another option is to choose σ_i to be any integer number from 0 to $M_i - 1$ [24]. For $YBa_2Cu_3O_{6+x}$, we are interested in structures that vary in their concentration of oxygen atoms and vacancies, whereas the other atomic species are kept fixed. This corresponds to the binary case $M_i = 2$, such that $\sigma_i \in \{0, 1\}$ or $\sigma_i \in \{-1, 1\}$. Provided a suitable basis, a configuration-dependent property P of a structure can be expressed as a function of its configuration vector $P(\boldsymbol{\sigma})$ by expanding it in terms of basis functions $\Gamma_{\boldsymbol{\alpha}}(\boldsymbol{\sigma})$ with coefficients $J_{\boldsymbol{\alpha}}$ [23, 24, 49]:

$$P(\boldsymbol{\sigma}) = \sum_{\boldsymbol{\alpha}} J_{\boldsymbol{\alpha}} \Gamma_{\boldsymbol{\alpha}}(\boldsymbol{\sigma}). \tag{3.8}$$

To construct a complete and orthonormal set of basis functions $\Gamma_{\boldsymbol{\alpha}}(\boldsymbol{\sigma})$, a scalar product has to be defined. For two functions $f(\boldsymbol{\sigma})$ and $g(\boldsymbol{\sigma})$, the scalar product can be expressed as [23, 24]:

$$\langle f,\ g \rangle\ = \frac{1}{\prod_{i=1}^{N} M_i} \sum_{\sigma_1} \sum_{\sigma_2} \cdots \sum_{\sigma_N} f(\boldsymbol{\sigma}) \cdot g(\boldsymbol{\sigma}). \tag{3.9}$$

The first term on the right hand side is a normalization constant. The sums are defined such that they run over all possible configurations, so for the binary case they either run over $\{0, 1\}$ or over $\{-1, 1\}$. There are several options for constructing a basis set. One of them uses the first M_i discrete Chebyshev polynomials [23, 24]. In this work, as $M_i = 2$, using two polynomials is sufficient:

$$\gamma_0(\sigma_i) = 1 \tag{3.10}$$

and

$$\gamma_1(\sigma_i) = \sigma_i \text{ with } \sigma_i \in \{-1, 1\}. \tag{3.11}$$

γ_{α_i} are called site basis functions. The chosen polynomials are orthonormal with respect to the scalar product defined in Eq. 3.9, such that $\langle \gamma_{\alpha_i}, \gamma_{\beta_{i'}} \rangle = \delta_{\alpha_i, \beta_{i'}} \delta_{i,i'}$ [23]. The basis functions $\Gamma_{\boldsymbol{\alpha}}$ are constructed as products of these site basis functions for all combinations of the site basis function indices α_i and lattice sites i [23, 24, 49]:

$$\Gamma_{\boldsymbol{\alpha}}(\boldsymbol{\sigma}) = \prod_{i=1}^{N} \gamma_{M_i, \alpha_i}(\sigma_i). \tag{3.12}$$

Here $\boldsymbol{\alpha}$ is a vector that can take values $\alpha_i \in \{0, 1, ..., M_i - 1\}$, so in our case $\alpha_i \in \{0, 1\}$. From Eq. 3.10, we observe that $\gamma_{\alpha_i}(\sigma_i) = 1 \; \forall \; i$, with $\alpha_i = 0$. We can construct other site basis functions, that are not created by discrete Chebyshev polynomials, such that they fulfill this condition too. Thereby we can reduce the products in Eq. 3.12 to [24]

$$\Gamma_{\boldsymbol{\alpha}}(\boldsymbol{\sigma}) = \prod_{i \in \boldsymbol{\alpha}} \gamma_{M_i, \alpha_i}(\sigma_i), \tag{3.13}$$

where $\boldsymbol{\alpha}$ now is a set of tuples of indices $\boldsymbol{\alpha} \equiv \{(i, \alpha_i) |\; \alpha_i \neq 0\}$. For the binary case this simplifies to:

$$\Gamma_{\boldsymbol{\alpha}}(\boldsymbol{\sigma}) = \prod_{i \in \boldsymbol{\alpha}} \gamma_1(\sigma_i). \tag{3.14}$$

We call $\boldsymbol{\alpha}$ a cluster and the $\Gamma_{\boldsymbol{\alpha}}(\boldsymbol{\sigma})$ cluster functions [24]. Let us consider as example a 2-point cluster $\boldsymbol{\beta}$ with lattice points k and j, such that $\alpha_i = 0 \; \forall \; i \neq k$ or j. Then the cluster function would be $\Gamma_{\boldsymbol{\beta}}(\boldsymbol{\sigma}) = \gamma_1(\sigma_k)\gamma_1(\sigma_j)$ and the corresponding vector $\boldsymbol{\beta}^T = (0, \ldots, \underset{k}{1}, 0, \ldots, \underset{j}{1}, 0, \ldots, 0)$. In this case $\Gamma_{\boldsymbol{\beta}}$ only depends on the components σ_k and σ_j.

Since the site basis functions are orthonormal, the cluster functions fulfill: $\langle \Gamma_{\boldsymbol{\alpha}}(\boldsymbol{\sigma}), \Gamma_{\boldsymbol{\beta}}(\boldsymbol{\sigma}) \rangle = \delta_{\boldsymbol{\alpha}, \boldsymbol{\beta}} = \prod_{i=1}^{N} \delta_{\alpha_i} \delta_{\beta_i}$ [24]. It should be mentioned that, if $\sigma_i \in \{0, 1\}$ in Eq. 3.11 (instead of $\sigma_i \in \{-1, 1\}$), the resulting basis is not orthogonal. This basis is called the indicator-binary basis. Nevertheless, it is highly interpretable, because $\Gamma_{\boldsymbol{\alpha}}(\boldsymbol{\sigma}) = 1$ only if all sites of the cluster $\boldsymbol{\alpha} \equiv \{(i, \alpha_i) \mid \alpha_i \neq 0\}$ are substituted (occupied by oxygen atoms) and zero else. The choice of basis sets will be discussed in more detail later on, when the construction of the CE model for this work is discussed.

With the cluster functions at hand, we are able to perform a cluster expansion of the configuration-dependent property $P(\boldsymbol{\sigma})$ as shown in Eq. 3.8. Since the sum goes over all clusters $\boldsymbol{\alpha}$, it is infinite in the thermodynamic limit, where the number of atoms goes to infinity $N \to \infty$ [24]. The expansion coefficients $J_{\boldsymbol{\alpha}}$ are given by the scalar product:

[23]

$$J_{\boldsymbol{\alpha}} = \langle \Gamma_{\boldsymbol{\alpha}}(\boldsymbol{\sigma}), P(\boldsymbol{\sigma}) \rangle = \sum_{\boldsymbol{\sigma}} \Gamma_{\boldsymbol{\alpha}}(\boldsymbol{\sigma}) \; P(\boldsymbol{\sigma}). \tag{3.15}$$

We call these expansion coefficients effective cluster interactions (ECIs).

We can rewrite the sum in Eq. 3.8 by taking into account the symmetry of the system. Let us consider S to be a symmetry operation of the system, then $P(S\boldsymbol{\sigma}) = P(\boldsymbol{\sigma})$. The clusters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ shall be symmetrically equivalent such that $S\boldsymbol{\alpha} = \boldsymbol{\beta}$, then also $J_{\boldsymbol{\alpha}} = J_{S\boldsymbol{\alpha}} = J_{\boldsymbol{\beta}}$. We can therefore rewrite the sum:

$$P(\boldsymbol{\sigma}) = \sum_{\boldsymbol{\alpha}}^{\text{s.i.}} \mathcal{M}_{\boldsymbol{\alpha}} J_{\boldsymbol{\alpha}} X_{\boldsymbol{\alpha}}(\boldsymbol{\sigma}) = \sum_{\boldsymbol{\alpha}}^{\text{s.i.}} \mathcal{M}_{\boldsymbol{\alpha}} J_{\boldsymbol{\alpha}} \frac{1}{\mathcal{M}_{\boldsymbol{\alpha}}} \sum_{\boldsymbol{\beta} \in \mathcal{O}(\boldsymbol{\alpha})} \Gamma_{\boldsymbol{\beta}}(\boldsymbol{\sigma}). \tag{3.16}$$

The first sum runs over all symmetrically inequivalent (s.i.) clusters. All clusters symmetrically equivalent to cluster $\boldsymbol{\alpha}$ are gathered in a set $\mathcal{O}(\boldsymbol{\alpha})$, the cluster orbit. The number of symmetrically equivalent clusters $\mathcal{M}_{\boldsymbol{\alpha}}$ is called cluster multiplicity. For an intensive property, denoted $\tilde{P}(\boldsymbol{\sigma})$, we can rewrite the property as: $\tilde{P}(\boldsymbol{\sigma}) = \sum_{\boldsymbol{\alpha}}^{s.i.} m_{\boldsymbol{\alpha}} J_{\boldsymbol{\alpha}} X_{\boldsymbol{\alpha}}$, with the intensive cluster multiplicity $m_{\boldsymbol{\alpha}} = \mathcal{M}_{\boldsymbol{\alpha}} \frac{V_{pc}}{V_{sc}}$, where V_{pc} is the volume of the parent cell and V_{sc} is the volume of the super cell.

The cluster correlation function

$$X_{\boldsymbol{\alpha}}(\boldsymbol{\sigma}) = \frac{1}{\mathcal{M}_{\boldsymbol{\alpha}}} \sum_{\boldsymbol{\beta} \in \mathcal{O}(\boldsymbol{\alpha})} \Gamma_{\boldsymbol{\beta}}(\boldsymbol{\sigma}) \tag{3.17}$$

is the average of cluster functions of all clusters symmetrically equivalent to $\boldsymbol{\alpha}$ [24]. As before, the sum is infinite. However, for practical applications one needs to truncate the sum to a finite number of relevant clusters, N_c. To find this set of clusters and to calculate the expansion coefficients $J_{\boldsymbol{\alpha}}$, we make use of artificial intelligence by employing machine learning methods.

3.2.2 Machine learning methods

To build a CE model, we need a set of N_t structures, for which DFT calculations are performed to obtain their properties $\boldsymbol{P}^T = (P_1, P_2, \ldots, P_{N_t})$, and a set of N_c symmetrically inequivalent clusters $\mathcal{C} = \{\boldsymbol{\alpha}, \boldsymbol{\beta}, ..., \boldsymbol{\omega}\}$ [24]. We need to find a subset of the most important clusters to reliably calculate the expansion coefficients, the ECIs.

Linear Regression

We can show that the task to calculate these coefficients can be viewed as a linear regression. Since now the number of clusters is finite, we can rearrange Eq. 3.16 by defining the cluster correlation matrix $\boldsymbol{X}$:

$$\boldsymbol{X} = \begin{pmatrix} X_{11} & X_{12} & \ldots & X_{1N_c} \\ X_{21} & X_{22} & \ldots & X_{2N_c} \\ \ldots & \ldots & \ldots & \ldots \\ X_{N_t1} & X_{N_t2} & \ldots & X_{N_tN_c}. \end{pmatrix}. \tag{3.18}$$

The rows represent the N_t structures, and the columns represent the N_c clusters. X_{11} refers to the cluster correlation for the structure with configuration vector $\boldsymbol{\sigma}_1$ for the first cluster in $\mathcal{C}$, so in this case $X_{11} = X_{\boldsymbol{\alpha}}(\boldsymbol{\sigma}_1)$, $X_{12} = X_{\boldsymbol{\beta}}(\boldsymbol{\sigma}_1)$, ..., $X_{N_tN_c} = X_{\boldsymbol{\omega}}(\boldsymbol{\sigma}_{N_t})$. The properties predicted by the machine learning model can then be written as:

$$\hat{\boldsymbol{P}}(\boldsymbol{\sigma}) = \sum_{\boldsymbol{\alpha} \in \mathcal{C}} m_{\boldsymbol{\alpha}} X_{\boldsymbol{\alpha}}(\boldsymbol{\sigma}) \boldsymbol{J}_{\boldsymbol{\alpha}} = \begin{pmatrix} X_{11} & X_{12} & \ldots & X_{1N_c} \\ X_{21} & X_{22} & \ldots & X_{2N_c} \\ \ldots & \ldots & \ldots & \ldots \\ X_{N_t1} & X_{N_t2} & \ldots & X_{N_tN_c} \end{pmatrix} \begin{pmatrix} m_1 \, J_1 \\ m_2 \, J_2 \\ \ldots \\ m_{N_c} \, J_{N_c} \end{pmatrix}. \tag{3.19}$$

Here, m_1, m_2, ... are the multiplicities of the first, second, ... cluster in $\mathcal{C}$. We consider the vector $\boldsymbol{\mathcal{J}}$ with components $m_i J_i$ to simplify the equation to:

$$\hat{\boldsymbol{P}}(\boldsymbol{\sigma}) = \boldsymbol{X}\boldsymbol{\mathcal{J}}. \tag{3.20}$$

Equation 3.20 corresponds to a linear model with input $\boldsymbol{X}$ and coefficients $\boldsymbol{\mathcal{J}}$. The latter can be found by minimizing the residual sum of squares (RSS):

$$\boldsymbol{\mathcal{J}} = \underset{\boldsymbol{\mathcal{J}}^*}{argmin} \left(\|\boldsymbol{P} - \boldsymbol{X}\boldsymbol{\mathcal{J}}^*\|_2^2 \right) = \underset{\boldsymbol{\mathcal{J}}^*}{argmin} \left(\sum_{s=1}^{N_t} \left(P_s - \hat{P}_s \right)^2 \right), \tag{3.21}$$

such that the predictions $\hat{\boldsymbol{P}} = \boldsymbol{X}\boldsymbol{\mathcal{J}}$ are close to the target values $\boldsymbol{P}$, calculated previously using DFT. Here, the ℓ_2 norm is used. More generally, the ℓ_p norm is defined as $\|x\|_p = \left(\sum_{i=1}^{N} |x_i|^p\right)^{1/p}$.

Given that the columns in matrix $\boldsymbol{X}$ are linearly independent and that the number of clusters is smaller than the number of structures in the training set ($N_c \leq N_t$), the solution is found to be [50]:

$$\boldsymbol{\mathcal{J}} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{P}. \tag{3.22}$$

But often these requirements are not fulfilled: To find the relevant ECIs, we usually start with a large pool of clusters and few calculated structures, so it is probable that $N_c > N_t$. This results in the Gram matrix $\boldsymbol{X}^T\boldsymbol{X}$ in Eq. 3.22 not being invertible anymore. To circumvent this obstacle, we make use of regularization. This is done by adding a regularization term $\Phi(\boldsymbol{\mathcal{J}}^*)$ [24]:

$$\boldsymbol{\mathcal{J}} = \underset{\boldsymbol{\mathcal{J}}^*}{argmin}\left(\|\boldsymbol{P} - \boldsymbol{X}\boldsymbol{\mathcal{J}}^*\|_2^2 + \Phi(\boldsymbol{\mathcal{J}}^*)\right) \tag{3.23}$$

There are different choices for the regularization term. A common regularization technique is ridge regression [51].

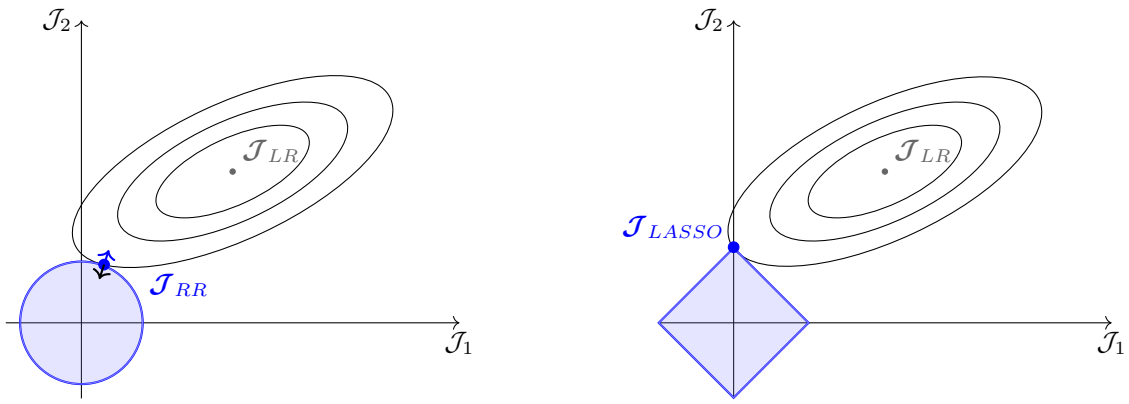

Figure 3.1: Graphical representation of regularization in terms of ridge regression [51] (left) and LASSO [52] (right). The concentric ellipses represent residual sums of squares. Their minimum lies at the center of the ellipses and corresponds to the solution $\boldsymbol{\mathcal{J}}_{LR}$ of the linear regression. Applying regularization means that we search for the $\boldsymbol{\mathcal{J}}$ that minimizes the RSS while also being inside the constraint region, shown in blue. For ridge regression the solution is therefore at the intercept of the ellipse and circle denoted as $\boldsymbol{\mathcal{J}}_{RR}$ and for LASSO at the intercept between the square and the ellipse $\boldsymbol{\mathcal{J}}_{LASSO}$.

Ridge Regression

For ridge regression [51], the regularization term in Eq. 3.23 is defined as:

$$\Phi(\boldsymbol{\mathcal{J}}^*) = \lambda\|\boldsymbol{\mathcal{J}}^*\|_2^2. \tag{3.24}$$

Inserting this in Eq. 3.23, leads to an equation that is solvable, even if the Gram matrix is not invertible, and we obtain:

$$\boldsymbol{\mathcal{J}} = (\boldsymbol{X}^T\boldsymbol{X} + \lambda\mathbb{I})^{-1}\mathbf{X^T}\mathbf{P}. \tag{3.25}$$

$\mathbb{I}$ is the identity matrix and $\lambda \in \mathbb{R}_0^+$ a positive hyperparameter used to modify the regularization strength [24, 50]. Optimal values of λ can be found by using cross validation (CV), which will be introduced in this chapter further below.

Ridge regression penalizes large values of $\boldsymbol{\mathcal{J}}$, thus producing a shrinkage of the coefficients. To understand this better, we view it as the constrained optimization problem [53]:

$$\begin{aligned} &\min_{\boldsymbol{\mathcal{J}}^*} RSS(\boldsymbol{\mathcal{J}}^*) \\ &\text{subject to } \|\boldsymbol{\mathcal{J}}^*\|_2^2 \leq t^2. \end{aligned} \tag{3.26}$$

A graphical interpretation of this can be seen on the left hand side of Fig. 3.1. The contour lines of the RSS are visualized as concentric ellipses. The minimum of the RSS and, therefore, the solution of the linear regression task $\boldsymbol{\mathcal{J}}_{LR}$ is at the center of these ellipses. Applying ridge regression means searching for the vector, $\boldsymbol{\mathcal{J}}$, that minimizes the RSS, while also fulfilling the constraint. For the two dimensional case shown in Fig. 3.1, the constraint corresponds to $\mathcal{J}_1^2 + \mathcal{J}_2^2 \leq t^2$, so points fulfilling this are inside the blue circle of radius t. At the solution, the gradients of the RSS and of the constraint with respect to $\boldsymbol{\mathcal{J}}$ are collinear, which is shown by the black and blue arrows in Fig. 3.1.

According to Eq.s 3.25 and 3.26, the vector of expansion coefficients $\boldsymbol{\mathcal{J}}$ is proportional to t and inverse proportional to λ. This means that for increasing regularization strength, i.e. λ, t and the components of $\boldsymbol{\mathcal{J}}$ decrease, leading to shrinkage.

LASSO: least absolute shrinkage and selection operator

Another regularization option corresponds to the least-absolute-shrinkage-and-selection-operator (LASSO) approach [52], which favors sparse solutions where many of the components of $\boldsymbol{\mathcal{J}}$ are exactly zero.

While in ridge regression the ℓ_2 norm is used in the regularization term, LASSO employs the ℓ_1 norm:

$$\Phi(\boldsymbol{\mathcal{J}}^*) = \lambda \|\boldsymbol{\mathcal{J}}^*\|_1. \tag{3.27}$$

Similarly to ridge regression, we can recast the problem as a constrained optimization, where we minimize the RSS while fulfilling that the ℓ_1 norm of $\boldsymbol{\mathcal{J}}$ is smaller or equal to $|t|$ [53]:

$$\begin{aligned} &\min_{\boldsymbol{\mathcal{J}}^*} RSS(\boldsymbol{\mathcal{J}}^*) \\ &\text{subject to } \|\boldsymbol{\mathcal{J}}^*\|_1 \leq |t|. \end{aligned} \tag{3.28}$$

As before, λ is a positive hyperparameter that tunes the regularization strength and needs to be found using cross validation [53].

For a graphical representation of the LASSO approach, we consider again Fig. 3.1, and focus on the graphic on the right-hand side. Now, the constraint is that $\|\boldsymbol{\mathcal{J}}^*\|_1 \leq |t|$, so, for the two dimensional case: $|\mathcal{J}_1| + |\mathcal{J}_2| \leq |t|$. Corresponding points are inside the blue square and solutions for the constrained optimization are at the intersection of the square and the ellipse. Using LASSO, the solution is more likely to be located at the corners of the square, where some of the coefficients are zero. Therefore applying LASSO means that the coefficients of $\boldsymbol{\mathcal{J}}$ are shrunk, and sparse solutions are promoted. Thus, in the context of a CE model, the it can be used to simultaneously calculate the expansion coefficients and to find a reduced set of relevant clusters.

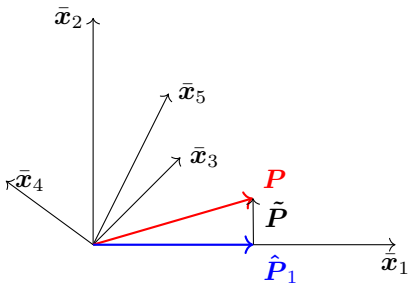

Figure 3.2: A graphical representation of OMP is shown. The vector of target values $\boldsymbol{P}$ is shown in the N_t dimensional column space, meaning that $\bar{\boldsymbol{x}}_i$ represent columns of the input matrix $\boldsymbol{X}$. Using OMP we try to find a sparse solution that best describes the true vales. In this case, the correlation between $\boldsymbol{P}$ and $\bar{\boldsymbol{x}}_1$ is highest, such that the solution $\hat{\boldsymbol{P}}_{OMP}$ has one nonzero coefficient in $\bar{\boldsymbol{x}}_1$ direction. $\hat{\boldsymbol{P}}_{OMP}$ is then subtracted from $\boldsymbol{P}$ to find the second best alignment, which would, in this case, be along $\bar{\boldsymbol{x}}_2$.

Orthogonal Matching Pursuit

Another common algorithm to promote sparsity is the orthogonal matching pursuit (OMP) [54]. To find an s-sparse solution, we iteratively select clusters (*i.e.* columns of $\boldsymbol{X}$) that highly correlate with the target $\boldsymbol{P}$ or to the current residual. A graphical representation of this is shown in Fig. 3.2 [55]: The column $\bar{\boldsymbol{x}}_1$ of the input matrix has the highest correlation with $\boldsymbol{P}$ and is selected first. We then search for the column with highest correlation to the residual $\tilde{\boldsymbol{P}}$, which excludes the part already captured by coefficient $\mathcal{J}_1$ and is therefore orthogonal to $\bar{\boldsymbol{x}}_1$. In the figure, the residual $\tilde{\boldsymbol{P}} = \boldsymbol{P} - \hat{\boldsymbol{P}}_1$ is parallel to $\bar{\boldsymbol{x}}_2$, such that this column would be selected next. At every iteration, the values of the nonzero coefficients are calculated using linear regression and a modified input matrix, containing the selected columns. This iterative process is repeated until s nonzero components are found [55].

Cross validation and error scores

Cross validation (CV) is used to test how well a model performs at predicting the properties of structures outside the training set. Consider a structure S belonging to the training set, with property P_s. A model trained with this training set yields the prediction $\hat{P}_s$. Now, we leave structure S out of the training set and train a new model. The prediction for structure S, denoted by $\hat{P}_s^{(s)}$, will in general be different from $\hat{P}_s$. While the residual $|\hat{P}_s - P_s|$ represents the error of the fit, the residual $|\hat{P}_s^{(s)} - P_s|$ is an estimate of the test error or generalization error. We compute $\hat{P}_s^{(s)}$ for each of the N_t structures until we get a vector of predictions $\hat{\mathbf{P}}^{LOO}$. This is called Leave-One-Out Cross Validation $\mathrm{CV_{LOO}}$, since always one structure or data point is left out

of the training. Its score is defined as the ℓ_2 norm of the residual $\hat{\mathbf{P}}^{loo} - \mathbf{P}$:

$$\begin{aligned} CV_{loo}^2 &= \frac{1}{N_t}\|\mathbf{P} - \hat{\mathbf{P}}^{LOO}\|_2^2 \\ &= \frac{1}{N_t}\sum_{s=1}^{N_t}(P_s - \hat{P}_s^{(s)})^2. \end{aligned} \tag{3.29}$$

Apart from that, we can also use k-fold cross validation. The data is separated into k equal parts, so-called folds. A model is trained on $k-1$ of the folds and predicted for structures in the fold left out of the training to obtain a CV score. This is done k times, each time excluding a different fold from the training. The final score is an average of the k CV-scores [56]. Leave-One-Out CV corresponds to a k-fold CV, where k is equal to the number of training structures N_t, such that each fold contains one structure. We optimize the hyperparameters of the models, *e.g.* the number of non-zero coefficients in OMP, or the λ parameter in LASSO, with respect to the CV scores. In this way, we avoid overfitting and improve the predictive power of our CE models [53].

In this work, we consider the errors of the fit and the CV scores for the following measures:

1. Mean Squared Error: MSE $= \frac{1}{N_t}\sum_{i=1}^{N_t}(P_i - \hat{P}_i)^2$
2. Root Mean Squared Error: RMSE $= \sqrt{MSE}$
3. Mean Absolute Error: MAE $= \frac{1}{N_t}\sum_{i=1}^{N_t}|P_i - \hat{P}_i|$
4. Maximal Absolute Error: MaxAE $= Max_i|P_i - \hat{P}_i|$

3.2.3 Nonlinear cluster expansion

The truncation of the sum in Eq. 3.16 impacts the quality of the model. If the cluster expansion converges slowly, a large number of clusters is needed to describe the system accurately, but this usually leads to overfitting. A truncation at a smaller subset of clusters could result in a model that fails to represent the material's properties appropriately [44]. Importantly, if the described property has a nonlinear dependence on the concentration of substituents, the standard CE may not converge at all [44]. An example for this is the c^2 problem of CE, for which we consider a property with quadratic dependence on the concentration c of substituents, $P(\boldsymbol{\sigma}) = c(\boldsymbol{\sigma})^2$ [44]. It is known that in the realms of standard CE, an infinite expansion is needed to represent this property [57, 58]. This is explained further in App. A. The c^2 problem can be resolved within nonlinear CE by introducing nonlinearities to the feature space [44].

In nonlinear CE, the feature space is extended such that the cluster correlation matrix $\boldsymbol{X}$ also entails nonlinear features. This is done by adding columns containing nonlinear combinations of the original features. There are several options for the feature space expansion. In this work we make use of a polynomial expansion, where the added columns are powers of the initial columns. For simplicity, let us assume that our input matrix has only one column (cluster); then the polynomial expansion reads:

$$\boldsymbol{X}_{nl} = \begin{pmatrix} X_{11} & X_{11}^2 & \dots & X_{11}^p \\ X_{21} & X_{21}^2 & \dots & X_{21}^p \\ \dots & \dots & \dots & \dots \\ X_{N_t 1} & X_{N_t 1}^2 & \dots & X_{N_t 1}^p \end{pmatrix}. \tag{3.30}$$

The p-dimensional vector of coefficients $\boldsymbol{\mathcal{J}}$ is then computed by using linear regression. In practice, the number of clusters is higher than one, resulting in larger matrices $\boldsymbol{X}_{nl}$, where the columns are not only multiplied with themselves, but also with each other, up to the chosen polynomial degree p.

Returning to the $P(\boldsymbol{\sigma}) = c^2$ problem, we demonstrate in App. A that, using the indicator-binary basis, the concentration of substituents is equal to the cluster correlation of the 1-point cluster $x = X_1(\boldsymbol{\sigma})$, and that the standard CE would take the following form [44]:

$$P(\boldsymbol{\sigma}) = \frac{1}{N} X_1(\boldsymbol{\sigma}) + \frac{1}{2N} \sum_{i=1}^{\infty} m_{2,i}\ X_{2,i}(\boldsymbol{\sigma}). \tag{3.31}$$

This expansion includes an infinite number of two-point cluster correlations $X_{2,i}(\boldsymbol{\sigma})$ of increasing range. In the thermodynamic limit ($N \to \infty$), the expansion coefficients are $J_1 = \frac{1}{N}$ for the 1-point cluster and $J_2 = \frac{1}{2N}$ for all 2-point clusters. Thus, we have an infinite sum with equally small expansion coefficients, such that we cannot represent it as a finite expansion that could be considered converged. However, when applying nonlinear CE, we get a finite cluster expansion with just one coefficient $J = 1$, since [44]:

$$P(\boldsymbol{\sigma}) = c^2 = X_1(\boldsymbol{\sigma})^2. \tag{3.32}$$

3.2.4 Structure selection

A non trivial task in cluster expansion is the selection of structures for the training set. In the case of CE models for the energy of formation of alloys, a common selection process is to choose the structures with lowest energies [59] and to randomly select higher-energy structures from the candidate set. A random selection has been shown to work successfully, in the case of compressive sensing and sparse models [60]. Another approach, which is more general, results from the aim to reduce the error of the predictions on unseen structures. This reduction is limited by the irreducible error stemming from the variance in the data itself, σ^2. In the case of *ab initio* data, the latter arises from, *e.g.*, the limited convergence with respect to DFT simulation parameters like the discretization of the Brillouin zone or the number of electronic orbitals in the representation of the Hamiltonian matrix. Such parameters are not accounted for by CE and together build up the irreducible error in the data. Interestingly, given training input data characterized by the Gram matrix $\boldsymbol{X}^T\boldsymbol{X}$, and assuming that the output for this data has an intrinsic error with variance σ^2, then the variance of the predicted property values for an arbitrary sample, characterized by the vector of correlations $\boldsymbol{x}_i$, is given by:

$$Var[\hat{E}_i] = \boldsymbol{x}_i^T (\boldsymbol{X}^T\boldsymbol{X})^{-1} \boldsymbol{x}_i\ \sigma^2. \tag{3.33}$$

Inspired by this result, a structure selection approach can be based on selecting a training data set that reduces this variance for a given population of structures outside the training data [46]. In order to examine the variance for a set of structures or a population of structures, we need to take the average of this expression over all structures in a population of samples (*pop*) [46]:

$$\langle Var[\hat{E}_i] \rangle_{pop} = \langle \boldsymbol{x}_i^T (\boldsymbol{X}^T\boldsymbol{X})^{-1} \boldsymbol{x}_i \rangle_{pop}\ \sigma^2 = (\boldsymbol{X}^T\boldsymbol{X})^{-1} : \langle \boldsymbol{x}_i \boldsymbol{x}_i^T \rangle_{pop}\ \sigma^2. \tag{3.34}$$

Here, the Frobenius product, defined as $\boldsymbol{A} : \boldsymbol{B} := \sum_i \sum_j A_{ij} B_{ij}$, has been used. The covariance matrix of the predicted ECIs, $(\boldsymbol{X}^T\boldsymbol{X})^{-1}\ \sigma^2$, has dimensions $N_c \times N_c$. The population-averaged

matrix $\langle \boldsymbol{x}_i \boldsymbol{x}_i^T \rangle_{pop}$, called domain matrix $\boldsymbol{D}$ [46], also has dimension $N_c \times N_c$. For the purpose of optimizing the variance of our training set, we can divide by σ^2 and consider the quantity [46]:

$$\tau = \frac{\langle Var[\hat{E}_i] \rangle_{pop}}{\sigma^2} = (\boldsymbol{X}^T \boldsymbol{X})^{-1} : \boldsymbol{D}, \tag{3.35}$$

which is the property that we want to minimize. We can compute it for different populations and training sets and choose the set producing the smallest τ.

Mueller and Ceder [46] provide various expressions for calculating the domain matrix; the following three are used in this work:

1. The identity matrix, which corresponds to an approximate result for the domain matrix derived by van de Walle and Ceder (vdWC) [45, 46].

$$\boldsymbol{D}^{vdWC} = \mathbb{I}; \tag{3.36}$$

2. A concentration dependent population average (CDPA), given by

$$D_{\alpha\beta}^{CDPA}(c) = (2c-1)^{n_\alpha + n_\beta}, \tag{3.37}$$

   where n_α and n_β are the number of sites for clusters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ [46];

3. A weighted population average (WPA) over the considered concentration range:

$$D_{\alpha\beta}^{WPA} = \int_c (2c-1)^{n_\alpha + n_\beta} \, dc. \tag{3.38}$$

   In this approximation, the entries of the domain matrix are weighted by the inverse of the number of structures per concentration. In this way, concentrations with many configurations are given less weight [46].

3.3 Monte Carlo simulations

In this work, we also study thermodynamic properties of $YBa_2Cu_3O_{6+x}$, such as the orthorhombic-to-tetragonal transition as well as the signatures of this transition on the specific heat C_p and the lattice constants at finite temperatures at varying oxygen concentration. For this, we make use of statistical thermodynamic simulations using Metropolis Monte Carlo (MC) sampling [26, 27].

We choose the canonical ensemble, such that the composition of the structures and the temperature remain constant during the sampling process. Although the volume changes, the pressure is exactly zero (fully relaxed structures), so the canonical ensemble still applies. For the MC simulations, we start with a random structure with a fixed composition. We use a CE model to predict energies of structures based on their configuration. Then, for each sampling step, the occupancies of substitutional sites are swapped for two or more randomly selected sites at once, to keep the composition constant. The energy of the new structure resulting after the swap, is predicted by the CE model. The algorithm accepts the new structure with a probability of [61]

$$P(E_0 \to E_1) = min\left(exp\left(-\frac{E_1 - E_0}{k_B T} \right), 1 \right). \tag{3.39}$$

Here k_B is the Boltzmann constant, T is the temperature and E_0, E_1 are the energies of the initial and next structure, respectively. This process is repeated for a large number of steps N_{steps}. The resulting trajectory enables us to calculate temperature-dependent properties by taking an average:

$$\langle P \rangle = \frac{1}{N_{avg.steps}} \sum_{i=1}^{N_{avg.steps}} P(\boldsymbol{\sigma}_i), \tag{3.40}$$

where $N_{avg.steps}$ is the number of steps used for averaging. It corresponds to the total number of steps minus the equilibration steps. Equilibration steps are needed before the system reaches equilibrium and they are disregarded when calculating the average property. The size of the simulation cell, the total number of steps, N_{steps}, and the number of equilibration steps need to be found in a convergence analysis. Additionally, we calculate the specific heat as

$$C_p(T) = \frac{\langle E^2 \rangle_T - \langle E \rangle_T^2}{k_B T^2}, \tag{3.41}$$

where $\langle \cdot \rangle$ indicates the averaged property according to Eq. 3.40. The numerator represents the variance of the energy, indicating fluctuations. Phase transitions, close to which the energy can fluctuate strongly, can be observed as peaks in the specific heat. Some properties, such as lattice constants at finite temperature, cannot be determined solely by extracting predictions from the CE energy model for configurations in a Monte Carlo trajectory. Instead, we build additional CE models specifically to predict the lattice constants, which offers valuable insights into how they behave with increasing oxygen concentration at a finite temperature.

Chapter 4

Results

In this chapter, we present the workflows applied in our study and present our results. We discuss the optimization of CE models for the total energy and compare to previous models for $YBa_2Cu_3O_{6+x}$ [62, 63]. We present CE models for the lattice constants and analyze their trends with increasing oxygen content at finite temperature. Additionally, we present two software codes that are developed as part of this work. First, an interface between NOMAD databases [64] and the cluster expansion code `CELL` [24], ensuring an easy combination of both tools and a FAIR (findable, accessible, interoperable and re-usable) treatment of the data [64]. It is presented in Sec. 4.4. Second, an active learning workflow that allows for an iterative structure selection in `CELL`, which is presented in Sec. 4.8. Both developments will be published in a future release of the code.

4.1 Description of $YBa_2Cu_3O_{6+x}$ in terms of cluster expansion

To describe $YBa_2Cu_3O_{6+x}$ in terms of cluster expansion, we define a parent lattice (see Chap. 3), based on the highly symmetrical $YBa_2Cu_3O_6$ structure, that belongs to space group P4/mmm (123). The corresponding Wyckoff sites are shown in Tab. 4.1. Regarding the site positions, we apply the description by Rayaprol et al. [39] introduced in Chap. 2 and verify it by comparing to the entries of $YBa_2Cu_3O_6$ [65] and $YBa_2Cu_3O_7$ [66] in the materials project. Compared to $YBa_2Cu_3O_6$, the parent lattice has an additional Wyckoff site 2f. It represents the substitutional sites in the planes between the barium atoms. The sites have positions (0, 1/2, 0) or (1/2, 0, 0). They can be either vacant (X) or occupied by an oxygen atom (O). The parent lattice has 14 sites, two of them are substitutional. Substitutional sites are assigned the value 0 (1) if they are vacant (occupied by an oxygen atom).

Cluster expansion models need to be trained with *ab initio* data. This training set is created by generating derivative structures from the parent lattice, having super cells of different shapes, and various decorations or configurations of the oxygen substituents. The details, on how such structures are selected, are explained in Sec.s 4.5 and 4.8. Due to the substitutional (dis)order, most derivative structures in the considered concentration range, such as $YBa_2Cu_3O_7$, have lower symmetry than the parent lattice. This is properly accounted for in the computed properties, such as lattice constants, which are obtained by DFT calculations, for which full lattice and atoms

Table 4.1: The Wyckoff sites of the parent lattice of $YBa_2Cu_3O_{6+x}$ are shown. Like $YBa_2Cu_3O_6$ it belongs to the P4/mmm space group, but it contains one more lattice site (Wyckoff site 2f), that can be either occupied by a vacancy (X) or an oxygen atom (O).

Atomic species	**Wyckoff site**	**Position**
Y	1d	(1/2, 1/2, 1/2)
Ba	2h	(1/2, 1/2, 0.1914)
Cu1	1a	(0, 0, 0)
Cu2	2g	(0, 0, 0.3590)
[X, O]	2f	(0, 1/2, 0)
O1	4i	(0, 1/2, 0.3790)
O2	2g	(0, 0, 0.1508)

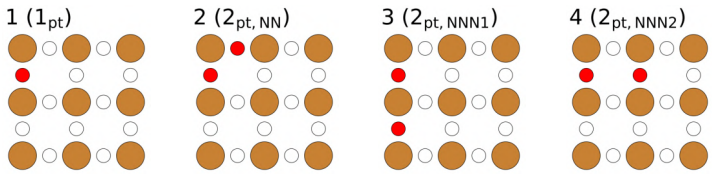

Figure 4.1: Relevant clusters for the description of $YBa_2Cu_3O_{6+x}$ are, first, the 1-point cluster (labeled 1), which contains a single oxygen atom (red). Second, the cluster containing oxygen atoms occupying nearest neighbor sites. Third and fourth, clusters of next-nearest neighbors. One of them (nr. 3) has an intermediate copper atom shown in bronze, whereas the other does not.

relaxation are performed. Being trained with this data, the resulting CE models account for the effect of the symmetry breaking on the predicted property values.

Another important element of the construction of CE models, is the selection of clusters. To build models with a high predictive power, we must select relevant clusters. Previous studies of $YBa_2Cu_3O_{6+x}$ have identified a number of relevant clusters [63,67]. The four most commonly used ones are depicted in Fig. 4.1. Clusters contain oxygen atoms (red circles). To better capture which interactions are described, we also visualize the copper atoms (bronze circles). As previously, vacant substitutional sites are depicted by a white circle. The illustrated clusters include the 1-point cluster and three 2-body clusters: the one of nearest neighbors (cluster 2), one of next-nearest neighbors with an intermediate copper atom (cluster 3) and the one of next-nearest neighbors without an intermediate copper atom (cluster 4). The ECI corresponding to cluster 3 is positive, indicating an attractive interaction that is facilitated by the intermediate copper atom. The cluster of oxygen atoms without an intermediate copper atom (cluster 4) represents a repulsive interaction. This asymmetry differs from the description of a typical Ising model. Models based on the mentioned four clusters are therefore named asymmetric next-nearest neighbor Ising (ASYNNNI) models [62,67]. As it will be shown later, an accurate description requires many more clusters than these. Nonetheless, they are able to capture some important structural characteristics like the formation of Cu-O chains, and are included in our optimized models.

Energy	**Structure**									
[meV per parent lattice]	**A**	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**J**
ΔE_{conv}	-	129.3	95.9	2.8	88.4	111.7	98.2	90.5	142.5	61.3
ΔE_{ref} [63]	-	275	244	33	190	262	192	164	331	135
ΔE_{ref} - ΔE_{conv}	-	145.7	148.1	30.2	101.6	150.3	93.8	73.5	188.5	73.2

Table 4.2: The total energy differences for the structures of $YBa_2Cu_3O_{6.5}$ are shown. Structure A, corresponding to the ortho-II phase, is lowest in energy. It is set to zero and the energies of the remaining structures are shown in relation to it. We compare our results ΔE_{conv} to the results by Draxl et al. [63] ΔE_{ref}. The total energies are given in meV per parent lattice. The total differences between the results are shown in the bottom row.

4.2 Comparison to a previous CE study of $YBa_2Cu_3O_{6.5}$

Before embarking on the construction of complex and accurate CE models of the energy of $YBa_2Cu_3O_{6+x}$, it is useful to reproduce a simple CE model by Draxl et al. [63]. This model considers, as training set, ten structures of $YBa_2Cu_3O_{6.5}$, as depicted in Fig. 4.2, where the unit cells are outlined with a black line and shaded in gray. Accordingly, we perform *ab initio* calculations on these structures and build a CE model using the clusters from Ref. [63], which, besides those in Fig. 4.1, include also those in Fig. 4.3

For all structures, *ab initio* calculations using the all-electron, full-potential code FHI-aims [20] are performed. FHI-aims utilizes numeric atom-centered basis functions, for which four default settings of increasing size (and therefore accuracy) are provided: light, tight, intermediate and really-tight basis sets. The lattice, as well as atoms positions, are relaxed using a really-tight basis set and a k-point density of approximately 7.4 Å. We compare the performance of several xc-functionals, considering also the lattice constants, as those shall be studied later in this work. We perform three comparative calculations of $YBa_2Cu_3O_6$, utilizing the PBEsol [21, 22], PBE [68] and r2SCAN [69] xc-functionals. We evaluate the obtained ratio of lattice constants $\frac{c}{a}$ and compare them to the experimental results by Pietrasko et al. [28], in which a ratio of 3.06 is observed. With PBEsol we obtain a ratio of 3.09, with r2SCAN 3.11 and with PBE 3.13. Consequently, the PBEsol xc-functional is used in all subsequent calculations. In Ref. [63], the DFT calculations are performed without doing a structure relaxation, but using fixed values $a = b = \sqrt{a'b'}$, where $a' = 3.8293$ Å, $b' = 3.8722$ Å and $c = 11.7520$ Å are the experimental lattice constants of the ortho-II phase by Grybos et al. [70].

Structure A in Fig. 4.2 corresponds to the ortho-II phase, mentioned in Chap. 2. We find that structure A is lowest in energy, and that structure D, which differs from A in neighboring order of the Cu-O chains, is next lowest in energy with a relative difference of only 1.7 meV per parent lattice. In order to distinguish ground states, it is important to energetically differentiate between these structures. Therefore, we consider 1.7 meV per parent lattice as our target accuracy. To achieve converged results, we develop a workflow of calculations using light, tight and really-tight basis sets and various k-point densities. The full convergence analysis is discussed in App. B. Finally, we are able to converge to a computational uncertainty of $u_{comp} \approx \pm 1.2$ meV per parent lattice. We perform calculations for all considered $YBa_2Cu_3O_{6.5}$ structures, following the workflow discussed in App. B.

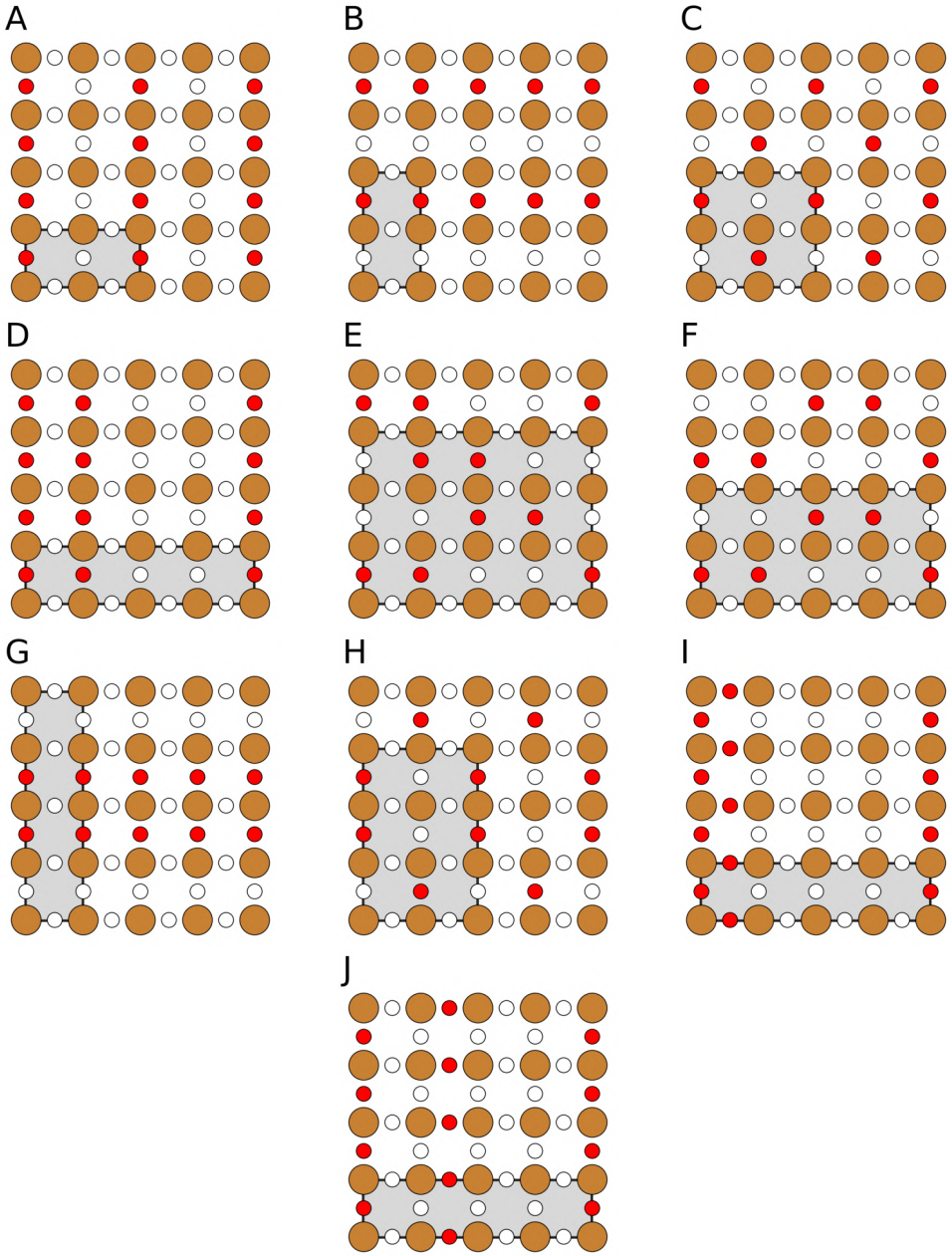

Figure 4.2: The bottom planes of the ten $YBa_2Cu_3O_{6.5}$ structures, visualized with ASE [40] and matplotlib [41]. Copper atoms are depicted as bronze, oxygen atoms as red and vacancies as white circles. The unit cells are outlined with a black line and shaded in gray.

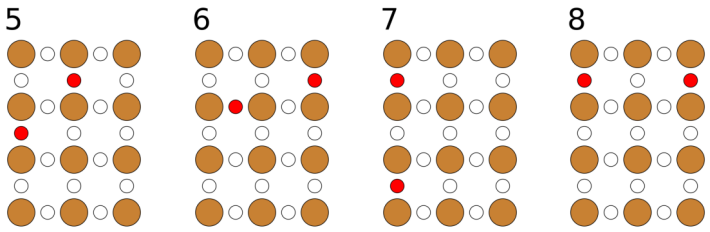

Figure 4.3: Clusters that are included in the model of $YBa_2Cu_3O_{6.5}$ by Draxl et al. [63] in addition to the four clusters from Fig. 4.1. Two-body interactions up to fifth nearest neighbors are considered.

	ECIs [meV per parent lattice]			
	J_{ref} [63]	J	$\Delta J = J_{ref}$ - J	$Sgn(J_{ref}) = Sgn(J)$?
$J_{\frac{1}{2}\frac{1}{2}}$	98	40.6	57.4	✓
J_{01}	-119	−54.1	−64.9	✓
J_{10}	14	9.6	4.4	✓
J_{11}	-1	−2.7	1.7	✓
$J_{\frac{3}{2}\frac{1}{2}}$	-17	−8.1	−8.9	✓
J_{02}	-21	−14.5	−6.5	✓
J_{20}	-6	2.3	−8.3	×

Table 4.3: Comparison of the ECIs J of our CE model to the ECIs by Draxl et al. [63]. Only 2-point interactions are considered. The subscript text indicates the translation between both contained signs. For example, J_{01} corresponds to a lattice site and its neighbor in $(0, a, 0)$ direction. The ECIs of our model differ by at least 0.4 meV and by at most 8.4 meV compared to the reference paper. All ECIs have equivalent signs in both models, except of J_{20}.

The comparison of results for the total energies of all structures is shown in Tab. 4.2. In the paper by Draxl et al. [63], energies are given in eV per formula unit of $YBa_2Cu_3O_{6.5}$. We use as unit meV/parent lattice, so we divide the total energy values by the dimension of the super cell with respect to the parent lattice. We find that structure A (ortho-II) is lowest in energy, in agreement with Ref. [63]. We compare the energetic differences ΔE of the remaining structures with respect to structure A. We obtain results, differing by at least 30.2 meV (structure D) and at most by 188.5 meV (structure I) from the result of Ref. [63][1]. The energetic ordering, from lowest to highest, agrees for most structures, except for structures H and E, as well as G and C, which have swapped sorting. The differences in the results are likely stemming from the structure relaxation that we perform, but is omitted in Ref. [63]. Considering this, our results for the energy differences are mostly consistent with Ref. [63].

We use the converged DFT results to train a CE model. As in the work by Draxl et al., we train the model on seven 2-point clusters [63]. The comparison of results for the ECIs is shown in Tab. 4.3. The notation of ECIs is as follows: $J_{\frac{1}{2}\frac{1}{2}}$ corresponds to the 2-point cluster of a site and

[1] If one interprets the formula unit in Ref. [63] as $(YBa_2Cu_3O_{6.5})_2$, as needed to represent the structures without having fractional oxygen occupation, then we observe smaller differences with our results, the maximal being 26.1 meV (structure C) and the minimal 2.2 meV (structure G).

its nearest neighbor in $(a/2, a/2, 0)$ direction, which is represented by cluster 2 in Fig. 4.1. J_{01} is the ECI for the 2-point cluster of sites with a relative translation of $(0, a, 0)$, where there is an intermediate copper atom between the sites, corresponding to cluster 3 in Fig. 4.1. J_{10} corresponds to the interaction without intermediate copper atom, visualized as cluster 4. As can be seen in Eq. 3.16, the coefficients we obtain with `CELL` [24] are a product of the multiplicity $\mathcal{M}_{\boldsymbol{\alpha}}$ and the ECIs $J_{\boldsymbol{\alpha}}$. We divide by the multiplicity of the cluster for each of the coefficients to obtain the ECIs. Our results differ by at least 1.7 meV (J_{11}) to at most 64.9 meV (J_{01})[2]. The signs of the ECIs provide information about the type of interaction: A negative ECI lowers the total energy, whereas a positive ECI increases the energy. Therefore, we can interpret a negative sign as an attractive interaction and a positive sign as a repulsion. The signs of J_{01} and J_{10} explain why structure A has the lowest energy: The negative sign of J_{01}, corresponding to the cluster with intermediate copper atoms, shows that the formation of Cu-O chains is energetically favored. Due to the repulsion of neighboring sites without an intermediate copper atom (J_{10}), an alternation of oxygen atoms and vacancies along this direction is preferred. The occupancy of sites in structure A follows exactly this pattern as can be seen in Fig. 4.2. We find that all ECIs agree in sign, except of J_{20}. However, the difference between both values is in the same order of magnitude as for other ECIs.

In the next sections, we proceed to create more structures to build CE models for the full concentration range $0 \leq x \leq 1$ of $YBa_2Cu_3O_{6+x}$.

4.3 Workflow for ground state search

Here, we provide an overview of our workflow for the search for ground states of $YBa_2Cu_3O_{6+x}$. It is visualized in Fig. 4.4. At first, an initial set of structures is created. For this study, a set of 12 structures, consisting of the ten structures of $YBa_2Cu_3O_{6.5}$ described in the previous section, together with $YBa_2Cu_3O_6$ and the ortho-I phase of $YBa_2Cu_3O_7$, served as initial set. For all structures a DFT relaxation is performed by following the workflow presented in App. B. We use the resulting energy values as target values for the construction of a CE model. We need to ensure that we use intensive properties when building the models, such that they do not scale with system size. Therefore, as discussed in App. B, we do not consider total energies, but energies of mixing, defined as:

$$E_{mix}(\boldsymbol{\sigma}, c) = \frac{E_t(\boldsymbol{\sigma})}{N_{p.l.}} - \left(E_t(\boldsymbol{\sigma}_0) + 2\,c \cdot [E_t(\boldsymbol{\sigma}_1) - E_t(\boldsymbol{\sigma}_0)]\right). \tag{4.1}$$

$N_{p.l.}$ describes the number of parent lattices needed to create the super cell of the configuration. $E_t(\boldsymbol{\sigma})$ refers to its total energy, while $E_t(\boldsymbol{\sigma}_0)$ and $E_t(\boldsymbol{\sigma}_1)$ are the total energies of the reference structures $YBa_2Cu_3O_6$ and ortho-I $YBa_2Cu_3O_7$. The configuration has a fractional oxygen concentration c, representing the concentration of oxygen atoms with respect to the substitutional sites. It relates to x, the excess number of oxygen atoms in the formula unit $YBa_2Cu_3O_{6+x}$, as $x = 2c$. Therefore, for the relevant concentration range $0 \leq x \leq 1$, it can take values $0 \leq c \leq 0.5$.

After building a CE model, we generate a full enumeration of derivative structures: We create configurations up to a certain size of super cells and predict their energies of mixing by using the model. We then check for convergence. This means that, first, we evaluate if the model's CV score is below a considered threshold, and second, whether all predicted ground state structures from

[2] In case the formula unit in the reference paper corresponds to $(YBa_2Cu_3O_{6.5})_2$, our results for the ECIs are in close agreement, differing by at most 8.4 meV ($J_{\frac{1}{2}\frac{1}{2}}$, J_{01}) and at least 0.4 meV ($J_{\frac{3}{2}\frac{1}{2}}$).

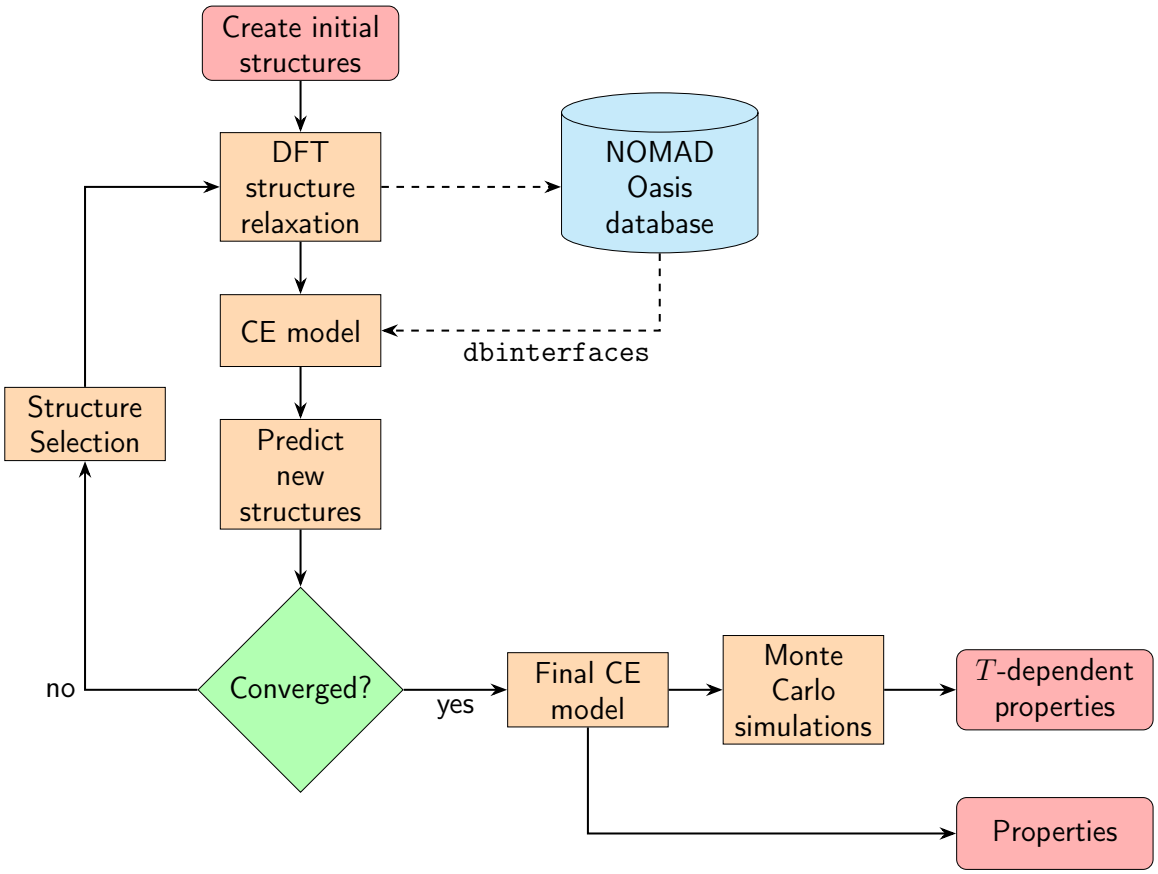

Figure 4.4: Workflow for the search for ground states. We combine DFT calculations with cluster expansion and machine learning methods. Temperature dependent properties are obtained by statistical thermodynamics simulations with Metropolis Monte Carlo sampling [26, 27].

the full enumeration have been already found. If we find a newly predicted ground state, we add it to the training set. Additionally, we select other structures to increase the diversity and size of the training set. For this, we use different approaches that are discussed in more detail in following sections. We perform DFT structure relaxations for the new configurations. With the extended training set, we build a new CE model and again generate a full enumeration. We repeat this until the model is converged.

With the converged model, we obtain predictions for the energy of mixing with a high precision. Additionally, we use the model to perform statistical thermodynamics simulations, using Metropolis Monte Carlo sampling [26, 27]. This allows us to predict temperature dependent properties, such as the specific heat C_p, for different temperatures and oxygen concentrations. In this work, we also study the lattice constants of $YBa_2Cu_3O_{6+x}$, which requires the construction of three additional CE models. These models are then used to calculate averaged lattice constants at different concentrations and finite temperature, using the statistical thermodynamics simulations.

4.4 Building an interface to NOMAD databases

In projects involving machine learning tasks, such as in this work, a lot of data is generated that can be useful for other projects, too. Consequently, all DFT calculations from this study are stored in a NOMAD [64] database. NOMAD is one of the largest data infrastructures for computational materials science and commits to the FAIR principles: to provide Findable, Accessible, Interoperable and Re-usable [71] data. We share these values and want to promote them. Currently, our data is published in the private NOMAD Oasis database, shared by the solid state theory group of the HU Berlin. From there we plan to make it publicly available in NOMAD [64] itself, to ensure that our data is FAIR.

To enhance interoperability, we develop a python module that serves as an interface between NOMAD [64] databases and `CELL` [24]. The module, called `dbinterfaces`, enables users to extract material properties from NOMAD [64] and return them in a format compatible with `CELL` [24]. This simplifies and promotes the combination of database storage and cluster expansion methods and hopefully contributes to data storage and usage in alignment with the FAIR principles.

The module consists of two python classes: `SingleEntry`, for handling single entries in the database, and `Dataset`, for processing several entries stored together in one dataset in NOMAD [64]. To build CE models, it is convenient to group the converged calculations for all structures of a training set in one dataset. Then, a list of property values for all training set structures can be extracted by a single call. Currently, the module supports extracting the following properties:

1. `get_entry_ids`:
   Returns a list of entry IDs for the entries of the dataset. This is convenient, if one wants to know which dataset entry matches which single NOMAD entry.

2. `get_total_energies`:
   Returns a list of the calculated total energy values in eV for all entries of the dataset.

3. `get_atoms_objects`:
   Returns a list of ASE Atoms objects for the structures in the dataset.

4. `get_structure_objects`:
   Returns a list of `CELL` structure objects for the whole dataset. This requires the `CELL` structure objects to be uploaded as JSON files to NOMAD together with the DFT calculations. The structure objects contain information about the super cell size and shape and the configuration vectors, which are not parsed by NOMAD. The objects do not contain the relaxed symmetry of the DFT calculations, which is stored in the atoms objects.

5. `get_lattice_constants`:
   Returns three lists: One for each lattice constant for the whole dataset. An option to normalize the lattice constants to the parent lattice is also provided. This enables users to compare the lattice constants for structures with different super cell sizes or shapes.

The methods are also available for single entries. The module, initially tailored to the needs of this project, is extensible, so users can easily add new methods for extracting additional information. Users with experience in the use of NOMADs API may also extract the full data stored in NOMAD by calling `get_archive` for SingleEntry objects or `get_data` for Datasets. From there it is possible

```
1 from clusterx.dbinterfaces.get_nomad_oasis_access_token import
      get_nomad_oasis_access_token
2 from clusterx.dbinterfaces.get_nomad_data import Dataset, SingleEntry
3 from clusterx.structures_set import StructuresSet
4
5 BASE_URL_SOL = 'https://sol-oasis.physik.hu-berlin.de/nomad-oasis/api/v1/'
6 token = get_nomad_oasis_access_token('/home/.env')
7
8 dataset = Dataset(dataset_id='-Wszp4KzTX6joTWP1DnL3A', token=token,
      pagination_page_size=12, base_url=BASE_URL_SOL)
9
10 lattice_constants = dataset.get_lattice_constants(normalize=True)
11 structures = dataset.get_structure_objects()
12
13 structures_set = StructuresSet(parent_lattice=structures[0].get_parent_lattice())
14 structures_set.add_structures(structures)
15 structures_set.set_property_values(property_name='a', property_vals=
      lattice_constants[0])
```

Figure 4.5: An example for the usage of the `dbinterfaces` module is shown. A Dataset object is created. The ID of the dataset created in NOMAD [64] is used to extract the data. We can extract a list of lattice constants, as well as the structure objects for all entries in the dataset. Next a structures set is created using the structure objects. The list of lattice constants is set as the property values. The structures set is now ready to be used to create a CE model with `CELL` [24].

to access all information parsed by NOMAD. As mentioned above, extracting `CELL` [24] structure objects from NOMAD requires to save the objects as JSON files, using `CELL`, [24] and to upload them. The data contained in the structure objects, such as parent lattice, super cell size and shape, and the configuration vector, is not accessible from the input and output files of the DFT calculations and not parsed by NOMAD. Hence, one needs to upload a file that provides this information to NOMAD [64]. The `get_structure_objects` method reads in the JSON files' content and creates a structure object from it. To ensure that the converged DFT data is correctly matched to the associated structure object file, both must be uploaded within the same directory. Only then, they are accessible with the same NOMAD entry ID. We therefore recommend saving the JSON file in the same directory as the converged calculations, from which the property values shall be extracted.

The module is both applicable to NOMAD [64] and to NOMAD Oasis databases. The latter are usually more restricted and require access via a token that is created with the login information to the Oasis. The `dbinterfaces` module allows to specify the database by providing a `base_url` argument. Additionally, it provides a method to create the access token. With `get_nomad_oasis_access_token` the login information can be read in from a file as environment variables (recommended) or it can be provided directly as arguments of the method (not recommended due to safety/privacy considerations). In Fig. 4.5 we present a short example to illustrate the usage of the module. For the example, we use a dataset stored in the NOMAD Oasis of the HU Berlin SOL group. It contains the initial structures-set described in Sec. 4.3, containing 12 structures. In the first three lines, the required modules are imported from `CELL`. In line 5, we provide the URL of the NOMAD Oasis, shared by the HU Berlin SOL group, and create an access token with the `get_nomad_oasis_access_token` method and a file containing the login information. In line 8, we create a dataset object using the `Dataset` class. We assign the URL and

token and provide the `dataset_id`. The argument `pagination_page_size` is required to specify how many entries of the dataset shall be considered. We recommend to match this number to the number of entries in the dataset. In case of timeout issues, or if only parts of the dataset shall be considered, it is possible to reduce this number. In line 10, we use `get_lattice_constants` to store the lattice constants, obtained from the DFT calculations, and normalize them to the parent lattice. We create a list of structures with the `get_structure_objects` method. This enables us to create a `CELL` [24] `StructuresSet` object by extracting the parent lattice of one of the structures. Then, we add the list of structures from the dataset to the structures set. Finally, we assign the property values for the smallest lattice constant to a property of the structures set, that we name a. This structures set could now be used to build a CE model for lattice constant a. The example shows that the creation of a `CELL` [24] `StructuresSet` object using the `dbinterfaces` module is remarkably simple and accessible, also to users with no prior experience in using APIs.

The `dbinterfaces` module will be published in a future release of `CELL`. Regarding the workflow of this project, we integrate the storage in the NOMAD [64] SOL Oasis database and the usage of the module to extract the data, by following the dashed line in Fig. 4.4.

4.5 Construction of accurate CE models for E_{mix}

In this section, we present the construction of accurate CE models for the energy of mixing of $YBa_2Cu_3O_{6+x}$. In the following, we denote our energy units as meV, while this always refers to meV per parent lattice, unless otherwise indicated. We discuss the iterative approach to model optimization, which leads us to the final models. Previous models of $YBa_2Cu_3O_{6+x}$, like the ASYNNNI model [62, 67] or extensions to it [63, 72, 73], use a limited number of cluster interactions, selected based on domain knowledge. Our approach, which combines cluster expansion with machine learning techniques - even extending to nonlinearities - allows us to explore a vast cluster / feature space. Using techniques, such as LASSO or OMP, that promote sparsity, we can identify the most important interactions from this large feature space. Finally, we obtain models that even match the accuracy of the DFT calculations for low-lying energy structures.

The first step in building a CE model is deciding what site basis functions to use. In Sec. 3.2.1, two bases were introduced: Chebyshev polynomials, and an indicator basis. Both bases allow to construct complete cluster bases, so they are in principle equivalent, because one can transform one into the other. However, since in practice one has to cut off the basis, this equivalence is broken, and the choice of basis becomes relevant. Notably, we observe that also the definition of the parent-lattice plays a non-negligible role in basis selection when finite clusters pools are used: Changing the definition of the substitutional site from [X,O] to [O,X], thereby swapping which species is occupied a 0 or a 1 in the configuration vector, has been found to have an effect on the quality of the CE models. This puzzling finding is discussed in App. C. All in all, we found that a basis of Chebyshev site basis functions, as defined in Eq.s 3.10 and 3.11, yield best results and, except said otherwise, this basis is employed.

From the *ab initio* calculations, performed on the initial set of 12 structures, we obtain their total energies and calculate their energies of mixing according to Eq. 4.1. These energies, together with the corresponding structures, form the training set that is used to build the first CE model. Concerning the initial pool of clusters, in which we search for the most relevant clusters, we include the 1-point, all 2-point clusters up to a radius of 10.89 Å, as well as 3-point clusters up to a radius of 6.09 Å, summing up to 23 clusters.

For the cluster selection we employ LASSO and a combinatorial approach (see option `subsets_cv: size + combinations` of `CELL`'s ClustersSelector class [24]). For the latter, we define a fixed subset of the initial clusters pool, containing the 1-point and 2-point clusters up to next-nearest neighbors (see Fig. 4.1). To these four clusters, all possible combinations of up to five clusters out of the initial pool are added. For each combination, a model is built, and the fit and CV_{LOO} scores are calculated. The combination with the smallest CV score is chosen. We find an optimal pool of nine clusters. Ridge regression with a regularization strength $\lambda = 10^{-8}$ is used as estimator for the fitting of the ECIs. For this first model, the indicator basis was employed. The resulting error scores for the first model are shown in the first panel of Fig. 4.6, labeled "Iteration 1". We obtain a RMSE-$CV_{LOO,\ it1}$=21.4 meV. We also perform a clusters pool optimization using LASSO with hyperparameters ranging from 10^{-9} to 1, but it does not improve the CV scores and thus, is disregarded.

Following the workflow, proposed in Fig. 4.4, as next step we generate an enumeration of derivative structures: We build all possible configurations of $YBa_2Cu_3O_{6+x}$ with super cell sizes up to four times as large as the parent lattice. We predict their energies of mixing with the previously optimized CE model. The results are visualized in the top left panel of Fig. 4.7. The training *ab initio* data are shown as black circles. The corresponding CE model predictions are depicted as black dots. The predictions of structures from the first enumeration are shown as gray dots. From these, we select structures to add to the training set in order to build the next, improved, CE model. As discussed previously, we choose the newly predicted lowest energy structures. We find four new lowest energy configurations. They are candidates for ground states. Additionally, we choose two structures, following the approach of choosing random configurations, discussed in Sec. 3.2.4 [60]. The model predictions of the selected structures are shown as red dots. For these configurations we perform *ab initio* calculations and obtain new target values (red circles). We notice that our model produces better predictions for the lowest energy states, than for the two new states with higher energies. The prediction errors for the new target values, not included in the fit, are shown in the column "Test", which is part of the table in the first panel of Fig. 4.6. The CV scores in column "CV_{LOO}" provide an estimate of these test errors. However, we obtain test errors that are smaller than the CV-scores, indicating an overestimation of the former from cross validation. This is to be expected, considering the small sizes of the training and the test sets.

With the training set, extended by the newly calculated *ab initio* data, we build new CE models, starting the second iteration of model optimization. As previously, we try different methods for the estimator (linear regression, ridge regression, LASSO) and for cluster selection (LASSO and a combinatorial search). Similar to Iteration 1, in Iteration 2, we obtain the best model using the combinatorial search, followed by a fit with ridge regression with $\lambda = 10^{-8}$. This time, we use a larger initial clusters pool that also contains 3-point clusters up to radii of 8.61 Å. In total, the initial clusters pool contains 47 clusters, compared to 23 clusters in the first iteration. Since we consider a larger initial pool, we only select up to four additional clusters in the combinatorial approach, as this quickly becomes very resource-intensive for pools of increasing size. The optimization yields a model with eight clusters. Next, we generate a second enumeration of derivative structures with super cell shapes 2x2x1 and 3x3x1. Since we consider larger super cell sizes, than in the previous iteration, we represent 2474 configurations with eleven intermediate oxygen concentrations between $YBa_2Cu_3O_6$ and $YBa_2Cu_3O_7$. We select all new configurations with shapes 2x2x1, which results in eleven structures. For the configurations with super cell shapes 3x3x1, we use the same approach as previously, and select all new lowest energy structures, plus one random structure for each composition. For the smallest concentration, only one configuration is found, such that no random configuration is selected for this composition.

Model (Iteration 1)

Standard CE
Estimator:
Ridge regression, $\lambda = 10^{-8}$
Cluster selector:
combinatorial search (comb.)
Optimized clusters pool:
9 clusters

Errors [meV]	Fit	CV_{LOO}	Test
RMSE	3.0	21.4	5.9
MAE	2.1	11.3	4.1
MaxAE	6.8	70.2	13.2

Model (Iteration 2)

Standard CE
Estimator:
Ridge regression, $\lambda = 10^{-8}$
Cluster selector:
comb. search
Optimized clusters pool:
8 clusters

Errors [meV]	Fit	CV_{LOO}
RMSE	3.5	10.3
MAE	2.6	6.9
MaxAE	8.1	34.2

Test Errors [meV]	Total	$E \leq 50$
	79.2	15.0
	57.3	12.9
	205.0	28.6

Models (Iteration 3)

Standard CE
Estimator:
Ridge regression, $\lambda = 10^{-8}$
Cluster selector:
comb. search
Optimized clusters pool:
5 clusters

Errors [meV]	Fit	CV_{LOO}
RMSE	12.8	16.3
MAE	10.3	12.5
MaxAE	31.2	47.7

Test Errors [meV]	Total	$E \leq 50$	$E \leq 0$
	25.0	4.0	3.9
	10.2	3.3	3.4
	132.1	10.0	6.4

Non linear CE
Estimator:
OMP, n=13, degree=2
Clusters pool size:
12 clusters

Errors [meV]	Fit	CV_{LOO}
RMSE	9.0	27.0
MAE	6.6	19.7
MaxAE	26.4	94.1

Test Errors [meV]	Total	$E \leq 50$	$E \leq 0$
	14.4	5.1	3.6
	8.0	4.0	2.8
	65.3	13.3	7.3

Model (Iteration 4)

Standard CE
Estimator:
OMP, n=16
Cluster selector:
LASSO, comb. search, OMP
Optimized clusters pool:
16 clusters

Errors: full dataset [meV]	Fit	CV_{LOO}
RMSE	3.3	5.8
MAE	2.4	3.5
MaxAE	11.3	33.3

Errors for $E \leq 0$ [meV]	Fit	CV_{LOO}	CV_{10f}
RMSE	1.1	1.2	1.6 ± 0.6
MAE	1.0	1.1	1.2 ± 0.2
MaxAE	2.3	3.0	3.5 ± 0.9

Figure 4.6: For each iteration, a selection of models and their parameters are displayed. Error scores are presented in tables, including fit and CV_{LOO} scores for the complete dataset of each iteration. Test errors for newly selected data are shown for the total set and subsets containing configurations with energies ≤ 50 and 0 meV. In the final iteration, no test errors are provided due to the absence of new data. Instead, fit and CV scores for configurations below 0 meV are included. Both LOO and 10-fold CV are considered, with the latter performed 100 times for different random splits to provide mean values and standard deviations.

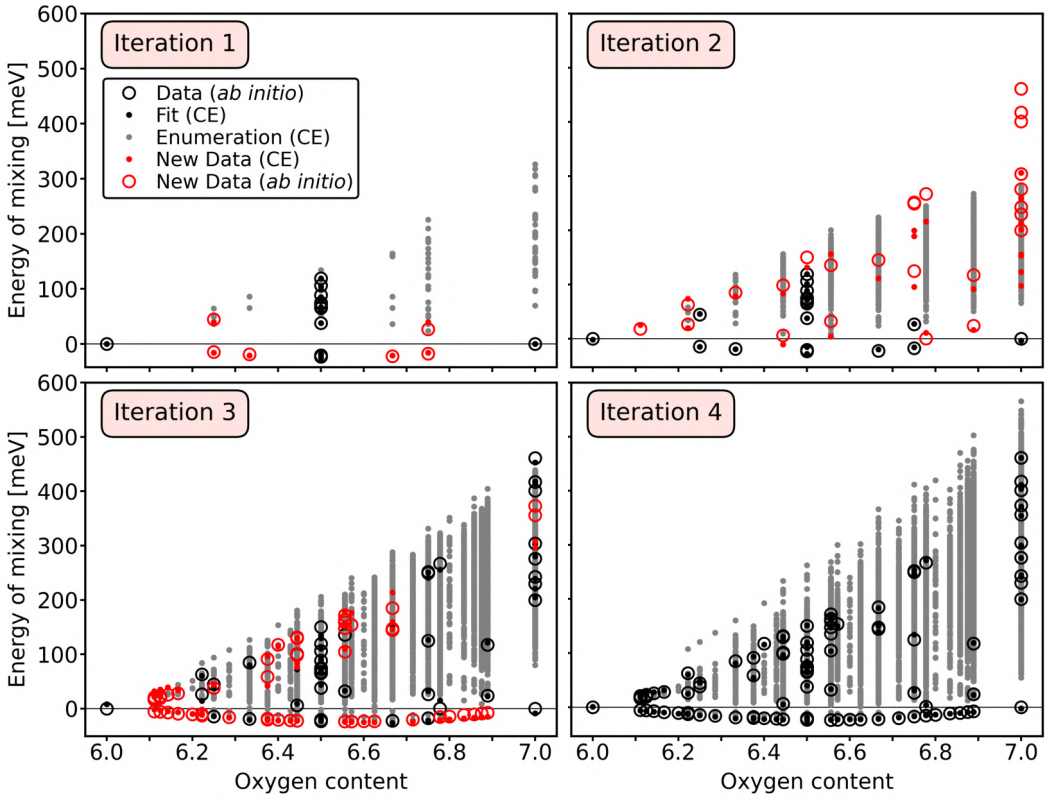

Figure 4.7: CE model optimization and intermediate ground state search for $YBa_2Cu_3O_{6+x}$: Black circles show the target values obtained by *ab initio* calculations. They are used to build a CE model. The corresponding fit predictions are shown as black dots. Next, the model is used to predict new configurations (gray dots), from which some are selected (red dots). *Ab initio* calculations are performed for the selected configurations (red circles). They are added to the training set for the next iteration of model optimization.

To efficiently select the lowest energy and random structures, we implement a new feature in `CELL` to automatically mask data sets. This is used for filtering data in the `generate_derivative_structures` method of `CELL` [24]. It allows the user to choose how many of the lowest energy structures (`n_lowest`) and how many random structures (`n_random`) to select for each composition, such that they can be filtered out and included in a set of structures. This makes the structure selection method easily accessible within the framework of `CELL` [24]. In total, 25 new structures are selected.

The corresponding data are shown on the top right of Fig. 4.7 (second panel). The fit, CV_{LOO}, and test errors of the model are displayed in the second panel of Fig. 4.6, labeled "Iteration 2". We obtain smaller CV scores than in Iteration 1: RMSE-$CV_{LOO,\,it2}$ = 10.3 meV, compared to RMSE-$CV_{LOO,\,it1}$ = 21.4 meV. However, the test errors are large. For the complete set of 25 new configurations, the RMSE-$test_{it2}$ = 79.2 meV. With the new configurations, we increase the size of the training set from 18 to 43 structures, many of them with new compositions and higher energies. It is therefore not surprising that our model, trained on a significantly smaller and less diverse training set, produces large test errors. These large test errors for configurations with high

energies are observable in the predictions of the model from the second iteration, which are depicted as gray/red dots in Fig. 4.7. We aim for a model whose predictions exhibit a high precision for low energies, where it is expected to find the ground states. Therefore, we also evaluate the test error for configurations with energies smaller than 50 meV, resulting in test errors that are more similar to the $\mathrm{CV_{LOO}}$ scores of the model. For instance: RMSE-$\mathrm{CV_{LOO,\ it2}}$ = 10.3 meV compared to RMSE-$\mathrm{test_{it2,\ E \leq 50}}$ = 15.0 meV. We expect the model to perform better in this energy region, as it was trained on configurations with energies below 50 meV.

With the training set, extended by the *ab initio* results for the 25 new configurations, we start the third iteration of model optimization. We try the same methods for the estimator and structure selector as in the previous iterations. Again, we find that a model using ridge regression with a small regularization strength of $\lambda = 10^{-8}$ and a cluster selector using the combinatorial approach, leads to the smallest CV scores. We start from the same initial clusters pool and fixed subset as in the first iteration and combine with up to five of the remaining clusters. The optimized clusters pool only contains five clusters: The fixed subset and one additional 3-point cluster. The corresponding error scores are shown in the top tables of panel "Iteration 3" in Fig. 4.6. The RMSE-$\mathrm{CV_{LOO,\ it3,\ st}}$ = 16.3 meV is larger than for the previous model. Although we aim to reduce the CV scores with each iteration, this is not unexpected, considering that the training set is a lot more diverse than it was in previous iterations.

In Iteration 3, we also explore CE model building, using the nonlinear CE method of Stroth et al. [44], discussed in Sec. 3.2.3. This allows us to compare the predictions of both models, and to choose new structures correspondingly. We perform polynomial expansions of degrees two, three, four and five and consider two initial clusters pools. One of them contains the 1-point and all 2-point clusters up to radius 10.89 Å, amounting to 12 clusters. The second, contains all clusters of the first one plus all 3-point clusters up to a radius of 6.09 Å (23 clusters). As estimator we use OMP, as described in Sec. 3.2.2. We optimize the CV score with respect to the number of nonzero coefficients. Besides $\mathrm{CV_{LOO}}$, we perform 10-fold $\mathrm{CV_{10f}}$. To estimate the error bar of the $\mathrm{CV_{10f}}$ estimator, we repeat calculations 100 times, each time generating different random splits, such that we can calculate the mean $\mathrm{CV_{10f}}$ scores and their standard deviations. The best nonlinear model is found for a polynomial expansion of degree two, using the first initial clusters pool. The model optimization is shown in Fig. 4.8. It reveals that a number of nonzero coefficients n=12 or n=13 minimizes the CV scores. The mean 10-fold CV scores are plotted as solid lines in red for the RMSE-CV, in blue for the MAE-CV and in black for the RMSE-Train error. As expected, the training error keeps decreasing with model complexity, while both CV scores increase after n=13. The standard deviations of all errors are shown as shaded areas around the mean values. We select $n = 13$ non-zero coefficients. This value also represents a minimum for a higher polynomial degree of five, indicating that it is a robust minimum. The error scores of the corresponding model are shown in the bottom tables of panel "Iteration 3" in Fig. 4.6. The model produces higher CV scores than the previously considered standard CE model, for instance: RMSE-$\mathrm{CV_{LOO,\ it3,\ nl}}$ = 27.0 meV, compared to RMSE-$\mathrm{CV_{LOO,\ it3,\ st}}$ = 16.3 meV for the standard model. Nevertheless, we decide to continue with both models for the next enumeration, as larger standard deviations of the error scores of the nonlinear model make it hard to estimate the quality of its performance on new data.

Next, we proceed to create full enumerations of derivative structures for both models, considering all possible super cell shapes and sizes up to nine times the parent lattice. We obtain 48,867 configurations. For these, we perform predictions of E_{mix} using both models, and find, for each of them, 27 lowest energy configurations. Five of these are already part of the training set. For the remaining 22 ground states found with each model, 20 are predicted as lowest energy by both, the

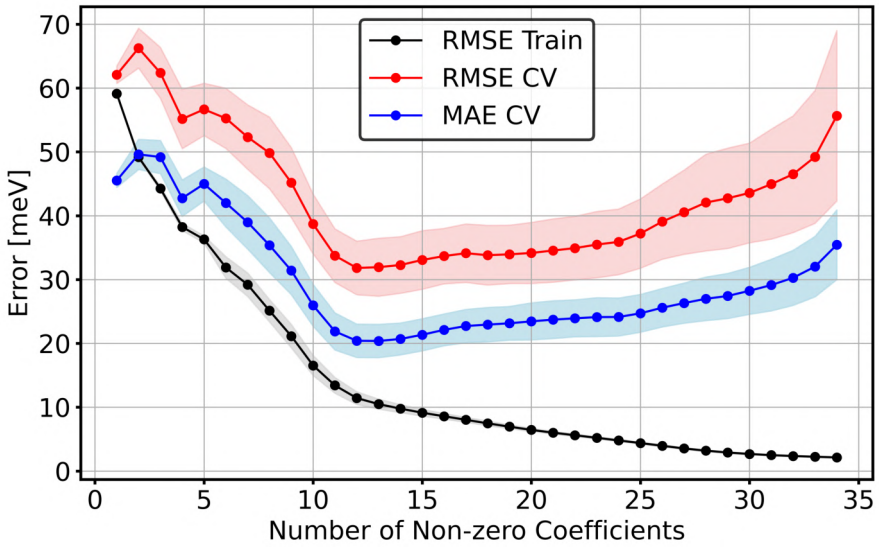

Figure 4.8: The optimization of the hyperparameter for the nonlinear CE model of degree two, using OMP, is shown. The RMSE Train error is shown in black, the RMSE-CV is shown in red and the MAE-CV score is shown in blue. We use 10-fold CV and perform 100 calculations using different random splits to obtain mean values (solid lines) and standard deviations (shaded areas). A number of non-zero coefficients n=12 or n=13 reduces the CV scores. We choose n=13 and the corresponding model is built for the third iteration of model optimization.

standard and nonlinear one, while two structures from each differ. We add all 24 configurations to the new training set. Additionally, we add 33 non-ground state structures, using the method by Mueller and Ceder [46], introduced in Sec. 3.2.4. We extend it by creating an active learning workflow, that we implement in `CELL` [24] and introduce it in detail in Sec. 4.8. In total, we add 57 new configurations, resulting in a training set of 100 configurations for the last iteration of model optimization.

With this new data, we compute test errors of the standard and nonlinear CE models obtained in Iteration 3. These errors are shown in the tables on the right hand side of panel "Iteration 3" in Fig. 4.6. We observe that, for the standard (nonlinear) CE, the CV_{LOO} scores under(over) estimate the test error. The nonlinear CE yields smaller test errors as the standard one, and both are generally smaller than test errors in the previous Iteration 2. The differences between the CV and test errors for the standard model are less severe than in the previous iteration. A plausible explanation for this is that the model was trained on a larger and more diverse training set. This highlights that the CV scores should be interpreted with caution, in the case of small training sets. It must be noted that, for both models, the large MaxAE test error scores are mainly due to a single configuration with a high energy of mixing of 373.0 meV. Although neither model's predictions meet the target error considering the complete training set, the performance is much better in the low-energy region relevant for finding ground states. This is reflected in the test errors for configurations with energies of mixing below 50 meV and below 0 meV in Fig. 4.6. Considering energies below 50 meV, the RMSE and MAE scores are ~ 2 to ~ 6 times smaller than the global test error (depending on the model), while for energies below 0 meV these are ~ 3 to ~ 6 times smaller.

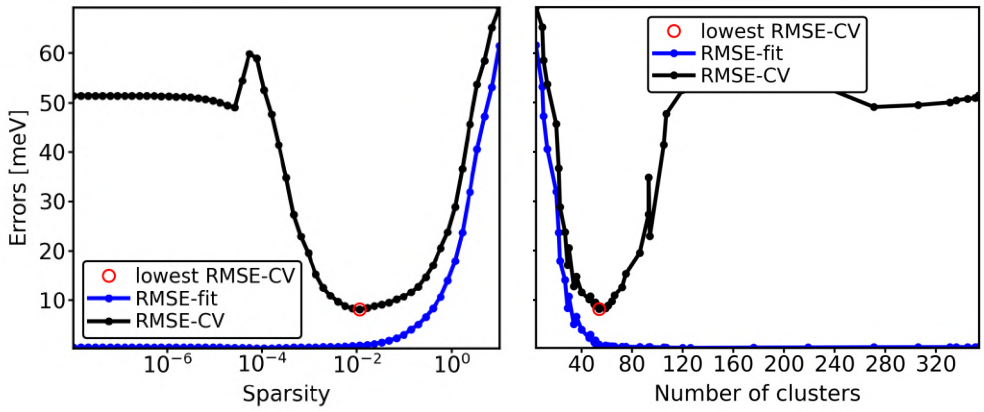

Figure 4.9: The optimization of the clusters pool in Iteration four using LASSO-CV [52] is shown. The hyperparameter (left hand side) is tuned to obtain the model with the lowest RMSE-CV score (red circle). The corresponding clusters pool sizes are shown on the right hand side.

The obtained errors, of around 3 to 5 meV for the RMSE and MAE, are much closer to the target accuracy. The *ab initio* calculations on the new structures reveal that all structures predicted as lowest energy configurations by the nonlinear model, are indeed lowest energy configurations of the new training set. In contrast, the standard CE model predicted two configurations differently, which have higher energies of mixing than the configurations predicted by the nonlinear model. The energy differences between the selected configurations are 2.6 meV (for oxygen content 6.22) and 1.7 meV (for oxygen content 6.38), which means that the first configurations are distinct, but the latter ones could be degenerate considering the computational errors of ±1.2 meV for both configurations. Despite its larger $\mathrm{CV_{LOO}}$ scores, we assess the performance of the nonlinear model to be better than that of the standard CE model from the third iteration, in the context of ground state search. The bottom left picture of Fig. 4.7 shows the predictions of the nonlinear model (gray dots) on the training data (black circles) and the selected new configurations (red).

With the extended training set, encompassing 100 configurations, we start the fourth (and last) iteration of model optimization. For this, we perform an extensive search for the best clusters pools and estimators by combining all previously mentioned methods. We find the best model for an initial clusters pool of 354 clusters, containing the 1-point, all 2-point up to a radius of 13.61 Å and 3- and 4-point clusters up to radius of 10.89 Å, on which LASSO-CV is used to select the optimal set of clusters. In Fig. 4.9, the RMSE errors versus the sparsity parameter (left panel) and number of clusters (non-zero coefficients in LASSO, right panel) are shown. An optimal sparsity parameter of $\lambda = 9.6 \cdot 10^{-3}$ is found, which corresponds to 54 clusters (see red circles). Next, we apply LASSO as estimator with an optimized $\lambda = 2.94 \cdot 10^{-3}$, such that the number of non-zero ECIs is reduced to 51. This two-stage strategy, consisting of applying LASSO twice, first for screening for relevant features and then for refining the coefficients, is a particular case of a method called relaxed LASSO [74]. It is also possible to combine LASSO with ordinary least squares fits (LASSO+OLS) or ridge regression (LASSO+RR). Applying LASSO two times, we find a model with remarkable fit errors of RMSE-Fit = 0.8 meV, MAE-Fit = 0.5 meV and MaxAE-Fit = 3.1 meV, as well as small $\mathrm{CV_{LOO}}$ scores of RMSE-$\mathrm{CV_{LOO}}$ = 3.3 meV, MAE-$\mathrm{CV_{LOO}}$ = 2.0 meV and MaxAE-$\mathrm{CV_{LOO}}$ = 16.9 meV. However, the optimized clusters pool contains only one of the two 2-point clusters of next-nearest neighbors. These two clusters embody important interactions capturing the anisotropy of the system, and are included in previous models [62, 63].

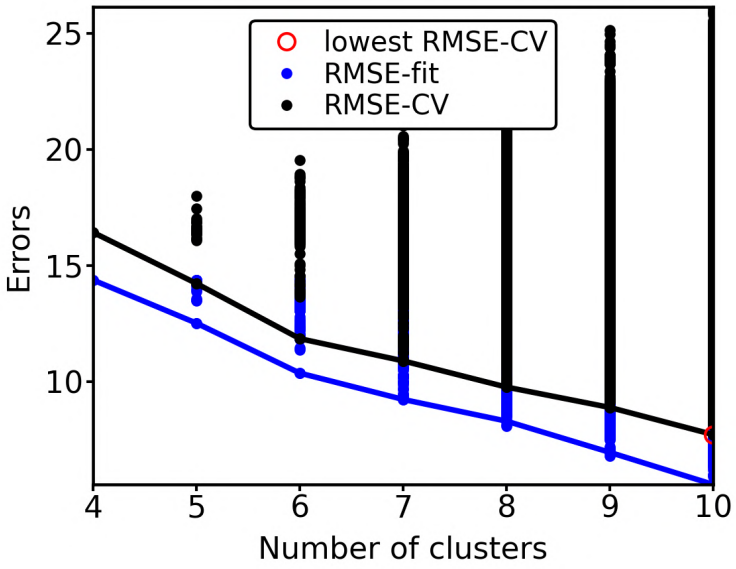

Figure 4.10: The optimization of the clusters pool in iteration four using `subsets_cv: size + combinations` is shown. The clusters pool was first optimized with LASSO from 43 to 25 clusters and is now further reduced to ten clusters.

The absence of one of them in the selected model, points to a common phenomenon in feature selection in LASSO: If many features are correlated with a "true" feature, LASSO could just pick one or a few of them as substitute, and drop the true one. One possible fix to this problem, is to restrict the feature space to reduce correlations between features. Then, LASSO is more likely to select the "true" feature.

For this purpose, we apply LASSO as cluster selector on smaller initial clusters pools. One of them contains all clusters with a radius of up to 8 Å for the 2- and 3-point clusters and a radius up to 6.5 Å for 4-point clusters, corresponding to 43 clusters. Applying LASSO+RR results in an optimized pool of 25 clusters ($\lambda = 2.82 \cdot 10^{-2}$). The corresponding model produces small error scores, such as RMSE-Fit = 2.8 meV and RMSE-CV_{LOO} = 6.2 meV. However, we aim for a model with an even smaller optimized clusters pool, as this allows for faster predictions, which is convenient for performing Monte Carlo simulations requiring several millions of property evaluations. Therefore, we reduce the optimized clusters pool further by applying a combinatorial subsets selection in the search of the optimal model. As before, we define a fixed subset up to 2-point next-nearest neighbor clusters, which we combine with up to six clusters, resulting in a pool of ten clusters. Figure 4.10 shows the result of this combinatorial subset optimization. The negative slope of the RMSE errors at the maximum number of clusters, suggests that the errors may be reduced further when considering more clusters, which is impractical in the combinatorial search. Therefore, we explore a less expensive version of the algorithm, where we manually add 2-, 3- and 4-point clusters, evaluate the reduction in the error scores and add the cluster permanently if the reduction is significant. This way, we cannot consider all combinations, but are still able to reduce the error scores. We obtain a clusters pool of 17 clusters, containing the 1-point, seven 2-point clusters, six 3-point and three 4-point clusters. Finally, we use this optimized clusters pool to build models, using OMP as estimator.

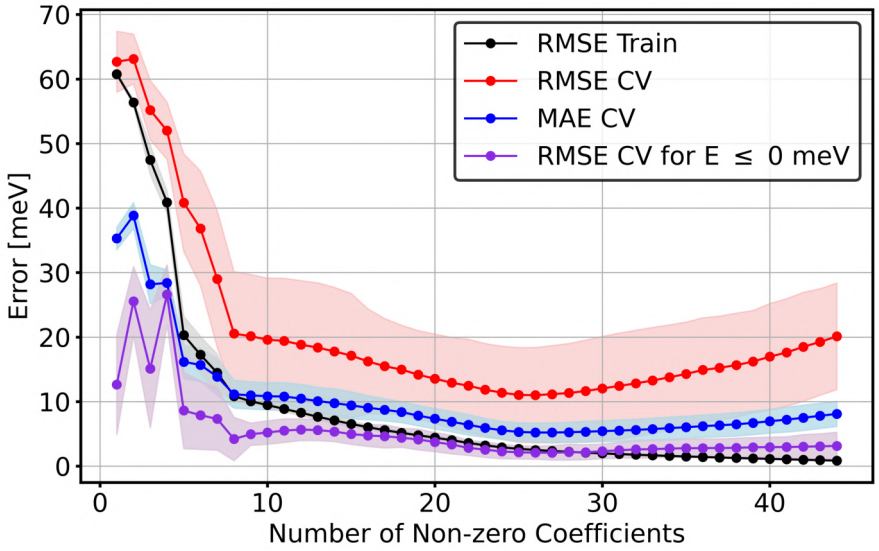

Figure 4.11: The optimization of the number of nonzero coefficients for a nonlinear CE model of polynomial degree three is shown. The RMSE-Train error (black), as well as 10-fold CV scores for the RMSE (red), MAE (blue) and the RMSE-CV for configurations with energies of mixing less than or equal to 0 meV are presented. The solid lines correspond to mean values and the shaded areas to standard deviations, obtained by performing 100 calculations, using different random splits.

We consider models with polynomial expansions of degrees one up to five. Focusing on a high accuracy for the prediction of low-energy configurations, the best model is found for degree three. The optimization of the OMP estimator is shown in Fig. 4.11. As before, mean errors (solid lines) and standard deviations (shaded areas) are presented, based on 10-fold CV. Interestingly, while the RMSE-CV scores for the whole dataset (red) are considerably larger than for low-lying structures (purple), both minimize at a similar number of non-zero parameters n. We choose n=27, which minimizes the latter.

Concerning the previously mentioned pairs of low-energy configurations, that are close in energy, we observe that 5th nearest neighbor 2-point clusters are needed to lift their degeneracy. Interestingly, there exist several studies of extended ASYNNNI models, that observed improved predictions, when including the 5th nearest neighbor 2-point cluster without intermediate copper atoms, shown as cluster 8 in Fig. 4.3 [72, 75]. This cluster is also contained in the optimized pool of the nonlinear model discussed above. It is able to lift the degeneracy for all pairs of close lying low-energy configurations, except for the one at oxygen content 6.38. As discussed previously, due to the computational error bar, it is possible that these configurations are actually degenerate. However, for the statistical thermodynamics simulations that we perform later on, it is easier to use a standard CE model instead of a nonlinear one. To obtain a good standard CE model, we need to modify the pool of the nonlinear CE slightly, replacing one of the clusters with the second 5th nearest neighbor 2-point cluster (cluster 7 in Fig. 4.3). We employ OMP as estimator and find an optimized standard CE model with 16 nonzero coefficients. Its error scores are shown in the last panel, labeled "Iteration 4", of Fig. 4.6. Its scores for the low-energy configurations

($E_{mix} \leq 0$ meV) are RMSE-Fit$_{\text{it4}}$ = 1.1 meV and RMSE-CV$_{\text{LOO, it4}}$ = 1.2 meV. These are smaller than or equal to our target error of 1.2 meV. We additionally provide the results of the 10-fold CV, to get an estimation of the standard deviations of the CV scores. Considering these, we find RMSE-CV$_{\text{10f, it4}}$ scores between 1.0 meV and 2.2 meV, for the low-energy configurations.

For this final model, we create again an enumeration up to super cell sizes nine times as large as the parent lattice. The corresponding plot is shown in the bottom right panel of Fig. 4.7. The model predicts one new lowest energy structure, but its energetic difference to the previously calculated structure is only 0.12 meV, such that we can consider them degenerate. Finally, our model is converged. Comparing to previous iterations, we were able to reduce the error scores significantly. This is likely due to the large increase of our training set size (from 12 to 18 to 43 to 100 structures) and the effort we put into optimizing the structure and cluster selection.

4.6 Comparative analysis with reported models

Next, we compare our optimized model to models of previous CE studies of $YBa_2Cu_3O_{6+x}$. Specifically, we compare to the ASYNNNI model [62] and a modeling based on the chosen clusters of Draxl et al. [63], discussed in Sec. 4.2.

As indicated by the name, in the ASYNNNI model interactions up to next-nearest neighbors are taken into account. The parameters in early versions of the model are chosen to match experimental observations, such as the prediction of the ortho-II phase [76], while later models use parameters based on *ab initio* studies [73,77]. In the simplest model, three interactions are considered: V_1, V_2 and V_3, which correspond to clusters 2, 3 and 4 in Fig. 4.1. In order to stabilize the ortho-II phase, the parameters need to fulfill $V_2 < 0 < V_3 < V_1$. As explained previously, the negative sign of V_2 indicates an attraction, such that the formation of Cu-O chains is preferred, which is also a clear result from the ground states we find, based on our *ab initio* calculations. These will be discussed in the next section. In order to compare to the ASYNNNI model, we build a CE model based on our training set of 100 configurations and the corresponding four clusters (see Fig. 4.1). We employ ridge regression with $\lambda = 10^{-8}$ to calculate the ECIs. The resulting error scores of the model are shown in panel one, labeled "ASYNNNI model", in Fig. 4.12. The model performs better for configurations with low than high energies, producing an RMSE-Fit$_{\text{ASYNNNI, } E \leq 0}$ = 4.1 meV. For the CV scores, both CV$_{\text{LOO}}$ and 10-fold CV are considered. As before, we perform 100 calculations for 10-fold CV to obtain mean errors and standard deviations. The model's predictions and the target values calculated by DFT are presented in the top left panel of Fig. 4.13. The model shows a linear behavior that is not in alignment with the actual distribution of the energies of mixing. While the model is able to distinguish between the two states at oxygen content 6.5, it fails to distinguish between the two configurations at oxygen content 6.22, as well as 6.38. Additionally, the prediction of the ortho-I phase at oxygen content 7 is off by 8.4 meV.

Next, we build a model on the complete training set, considering the 1-point and all 2-point clusters used in the study by Draxl et al. [63]. Since they consider 2-point clusters up to 5th nearest neighbors, we call the corresponding model the asymmetric 5th nearest neighbor Ising (ASY5NNI) model. The error scores are shown in the second panel of Fig. 4.12. Overall, it produces smaller fit and CV errors for the full dataset and the subset of configurations with low energies $E_{mix} \leq 0$ meV. This is to be expected, since the model contains all clusters considered in the 2D ASYNNNI model, as well as additional 2-point interactions of 3rd, 4th and 5th nearest neighbors. The obtained error scores are RMSE-Fit$_{\text{ASY5NNI}}$ = 3.5 meV and RMSE-CV$_{\text{LOO, ASY5NNI}}$ = 3.8 meV, compared to

ASYNNNI model [62]

Estimator:
Ridge regression, $\lambda = 10^{-8}$
Clusters pool:
4 clusters (up to NNN 2pt clusters)

Errors: full dataset [meV]			Errors for E$\leq$ 0 [meV]		
	Fit	**CV$_{LOO}$**	**Fit**	**CV$_{LOO}$**	**CV$_{10f}$**
RMSE	14.4	16.4	4.1	4.4	5.0 ± 1.4
MAE	9.2	10.1	3.7	3.9	4.0 ± 0.5
MaxAE	53.4	64.1	8.4	9.6	10.1 ± 1.8

ASY5NNI model [63]

Estimator:
Ridge regression, $\lambda = 10^{-8}$
Clusters pool:
1pt cluster and 2pt clusters from reference [63]

Errors: full dataset [meV]			Errors for E$\leq$ 0 [meV]		
	Fit	**CV$_{LOO}$**	**Fit**	**CV$_{LOO}$**	**CV$_{10f}$**
RMSE	11.2	13.4	3.5	3.8	4.3 ± 1.1
MAE	8.1	9.4	3.0	3.2	3.3 ± 0.5
MaxAE	31.8	41.6	8.3	9.1	9.2 ± 1.1

Optimized models

Standard CE

Estimator:
OMP, n=16
Cluster selector:
LASSO, comb. search, OMP
Optimized clusters pool:
16 clusters

Errors: full dataset [meV]			Errors for E$\leq$ 0 [meV]		
	Fit	**CV$_{LOO}$**	**Fit**	**CV$_{LOO}$**	**CV$_{10f}$**
RMSE	3.3	5.8	1.1	1.2	1.6 ± 0.6
MAE	2.4	3.5	1.0	1.1	1.2 ± 0.2
MaxAE	11.3	33.3	2.3	3.0	3.5 ± 0.9

Non linear CE

Estimator:
OMP, n=27, degree=3
Clusters selector:
LASSO, comb. search
Optimized clusters pool:
17 clusters

Errors: full dataset [meV]			Errors for E$\leq$ 0 [meV]		
	Fit	**CV$_{LOO}$**	**Fit**	**CV$_{LOO}$**	**CV$_{10f}$**
RMSE	2.5	6.6	0.6	1.4	2.1 ± 1.2
MAE	1.6	3.6	0.5	0.7	1.1 ± 0.4
MaxAE	9.3	33.6	1.7	7.2	7.1 ± 4.3

Figure 4.12: The parameters and error scores of two reported models and two of our optimized models are compared. The ASYNNNI model [62] is based on interactions up to next-nearest neighbors. The ASY5NNI model corresponds to the clusters used by Draxl et al. [63], where 2-point clusters up to 5th nearest neighbors are considered. We compare to our optimized standard CE and nonlinear CE model based on clusters pool with 16 and 17 clusters. Errors for the full set (left tables), as well as for configurations with energies below 0 meV (right tables) are provided.

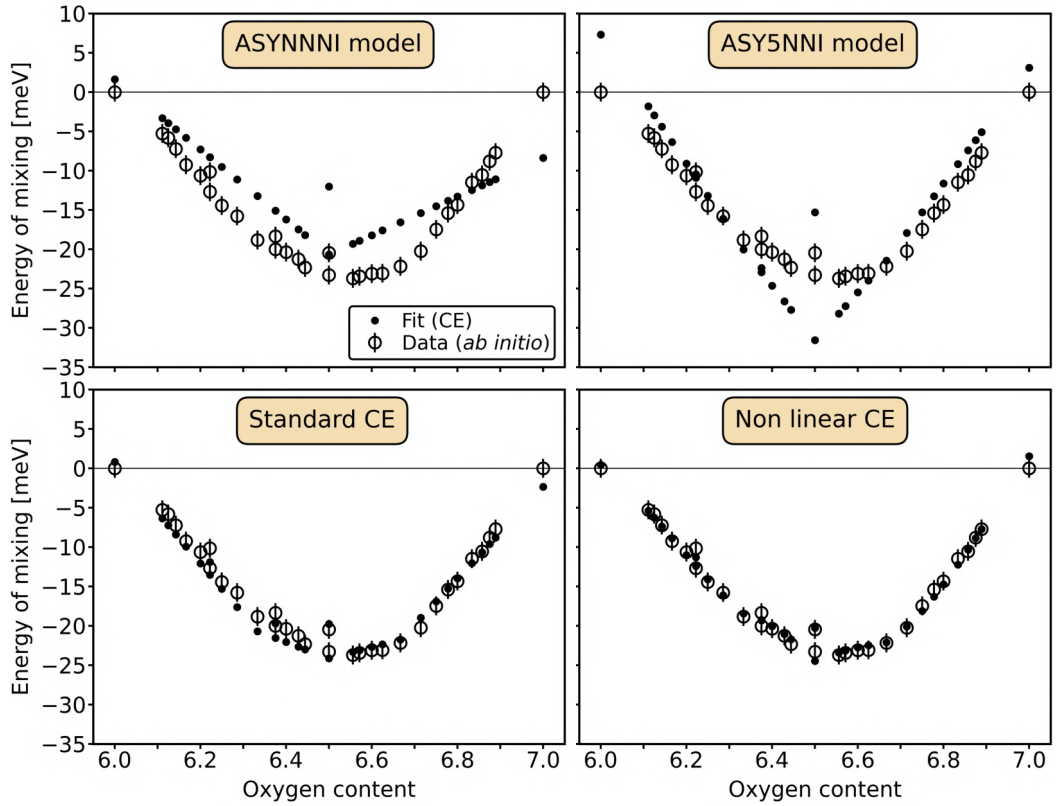

Figure 4.13: The *ab initio* target values (circles), including their error bars, and the predictions of the considered models (dots) for low-energy configurations are shown. The top left image shows the prediction of the ASYNNNI model [62], considering 2-point interactions up to next nearest neighbors. The top right image shows the ASY5NNI model, considering 2-point interactions up to 5th nearest neighbors [63]. The standard CE and nonlinear CE model correspond to our optimized models. They are each trained on clusters pools of seventeen clusters, differing by one cluster.

previously RMSE-Fit$_{\text{ASYNNNI}}$ = 4.1 meV and RMSE-CV$_{\text{LOO, ASYNNNI}}$ = 4.4 meV. The ASY5NNI model's predictions compared to the DFT target values are shown in the top right panel of Fig. 4.13. Similarly to the ASYNNNI model, it predicts a linear behavior that does not accurately capture the actual shape. Yet, in contrast to the previous model, it is able to lift the degeneracy of energetically close configurations at oxygen contents 6.22, 6.38 and 6.5. The predictions for the two smaller oxygen contents, especially for 6.22, are so close that they are hardly distinguishable in the figure.

The lower panels of Fig. 4.13 show the predictions of our two optimized models. The lower left panel corresponds to the standard CE model that we presented in Sec. 4.5, which is built using OMP and n=16. The nonlinear model on the right panel corresponds to the model with a polynomial expansion of degree three and 27 nonzero coefficients (see also Sec. 4.5 and Fig. 4.11). The predictions agree remarkably well with the *ab initio* data. The errors of both models are shown in the last panel of Fig. 4.12. The nonlinear model produces smaller fit errors, which can be simply explained by the fact that it includes more features than the standard CE model. Its CV$_{\text{LOO}}$ scores are higher (except for the MAE) than for the standard CE model. However, when

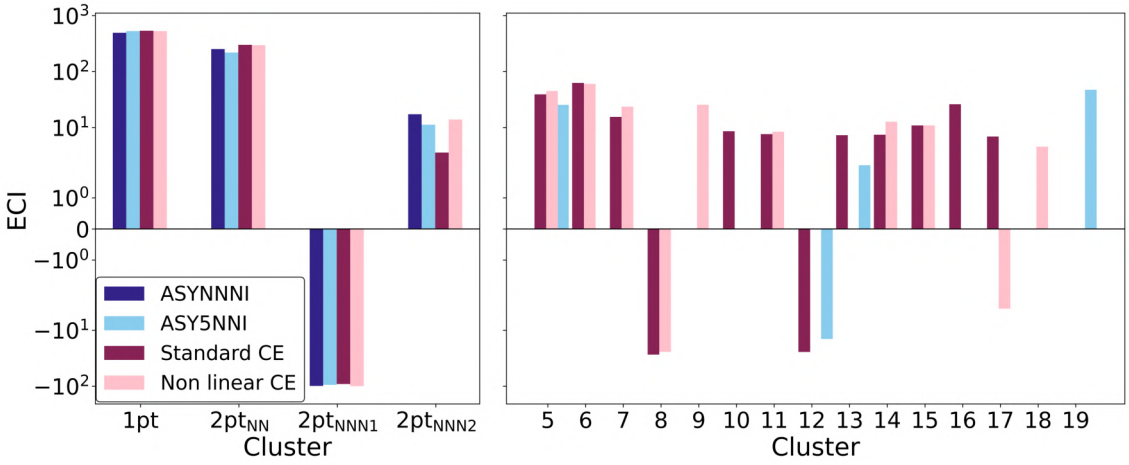

Figure 4.14: The ECIs for the considered models are shown for the subset of clusters shown in Fig. 4.1 (left-hand side). The 2pt_{NNN1} cluster is the next-nearest neighbor cluster with and the 2pt_{NNN2} without an intermediate copper atom. The ASYNNNI [62] ECIs are shown in darkblue, the ASY5NNI model's [63] in lightblue and results from our standard and nonlinear CE are shown in violet and pink. On the right hand side additional considered clusters are shown for the ASY5NNI, the standard and nonlinear CE. The scaling of the y axis corresponds to the symmetric logarithmic scale of matplotlib [78].

considering the standard deviations of the 10-fold CV, we observe that they are larger than for the standard CE model. Thus, the errors could be equivalent or even smaller for the nonlinear CE model taking into account the uncertainties of the errors. However, there is no way for us to know this, such that we cannot determine which of the models has better CV scores and therefore a higher predictive power. We observe that introducing nonlinearities to the feature space generally results in larger standard deviations for the error scores. We assume that this is due to the increase of the feature space to choose from, which could make it more difficult for the algorithm to find the best model.

A comparison of the ECIs of all four models is shown in Fig. 4.14, using the symmetric logarithmic scaling from matplotlib [78] for the y-axis. The ECIs of the different models are shown in darkblue for the ASYNNNI, lightblue for the ASY5NNI, violet for our standard and pink for our nonlinear CE model. In the left panel, the ECIs for the subset of clusters depicted in Fig. 4.1 are presented. They agree in sign for all models and are of similar magnitudes. The ECIs of the standard and nonlinear CE are more similar to each other than to the reported models. For all models, the parameters fulfill $V_2 < 0 < V_3 < V_1$ or in our notation: $ECI_{2pt_{NNN1}} < 0 < ECI_{2pt_{NNN2}} < ECI_{2pt_{NN}}$. Our models produce smaller error scores, because they have more features than the reported models, which are depicted in the right panel of Fig. 4.14. All considered clusters and nonlinear features and their ECIs are summarized in App. D. Figure 4.14 demonstrates, that the first three coefficients are significantly larger than all other coefficients, which explains why the ASYNNNI and ASY5NNI model are able to capture relevant features.

In Fig. 4.15, we show a graphical representation of the error scores by providing box plots for all considered models, using a logarithmic scale for the y-axis. The boxes represent the distribution of CV_{LOO} absolute errors, considering the interquartile range (IQR). The IQR extends from the median of the lower half of the data (25 percentile) to the upper half of the data (75 percentile).

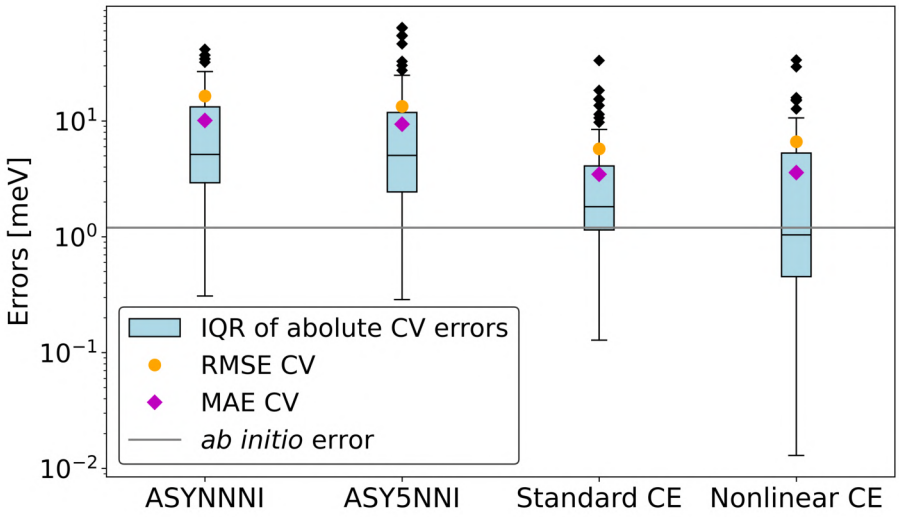

Figure 4.15: From left to right, box plots of the ASYNNNI, ASY5NNI and our optimized standard and nonlinear CE models are provided, employing a logarithmic scale for the y-axis. The interquartile range (IQR), extending from the median of 25 percentile to the median of 75 percentile of the absolute CV_{LOO} errors, are shown as blue boxes. Medians are represented by black lines. The vertical lines, called whiskers, represent the range $1.5 \cdot$ IQR. The RMSE and MAE CV_{LOO} scores are shown in orange and magenta. The computational error of the *ab initio* calculations is shown as gray horizontal line.

The vertical black lines, called whiskers, represent the range $1.5 \cdot IQR$. Outliers beyond these errors are visualized as black diamonds. The RMSE-CV and MAE-CV scores are shown in orange and magenta. The horizontal line represents the *ab initio* computational error of 1.2 meV. For the y-axis a logarithmic scaling is employed. We observe a significant improvement from the ASYNNNI and ASY5NNI models progressing to the standard and nonlinear CE model. Considering the nonlinear model, the median of the absolute errors even lies below the computational error. However, as discussed previously, the nonlinear CE model produces a wider spread of CV scores, making it harder to determine its accuracy.

We conclude that both of our optimized models perform very well, especially for the prediction of low-energy configurations. They are able to capture the actual shape of the distribution of energies of mixing, which is not achieved by the ASYNNNI or ASY5NNI model. We find that the inclusion of both 5th nearest neighbor 2-point interactions is needed to build a standard CE model that captures the actual shape, while one of them is sufficient, if nonlinearities are considered. Additionally, we find that including 3-point and 4-point clusters is relevant. Including 5th nearest neighbor interactions, but not considering interactions beyond 2-points, as in the ASY5NNI model, leads to an inaccurate linear behavior and does not deliver the desired accuracy of predictions.

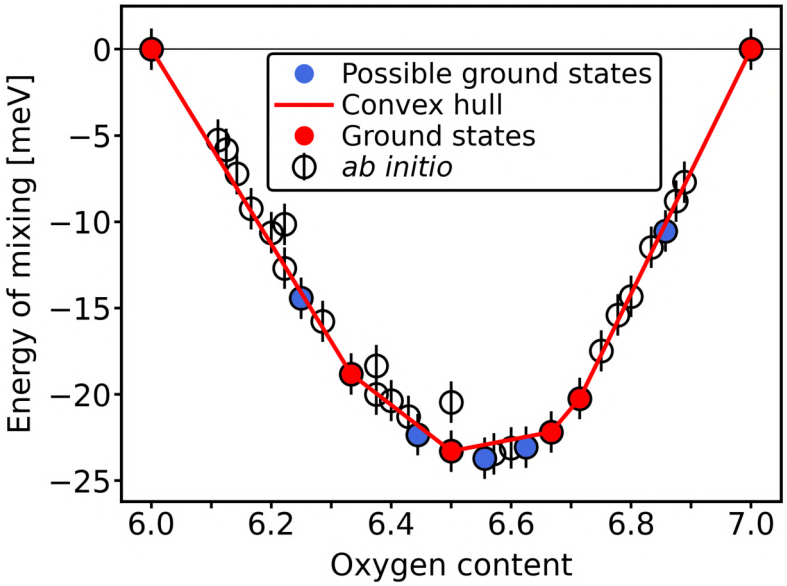

Figure 4.16: The DFT results for the low-energy states of the complete training set are shown as black circles, with error bars corresponding to the computational error $u_{comp} = \pm 1.2$. Eleven possible ground states, marked as blue/red dots, are found with a convex hull, not considering the computational error. Considering the computational error, the convex hull (red line) encompasses six ground states (red dots).

4.7 Ground states

To discover which of the low-energy states of our final training set are actual ground states at $T = 0$ K we draw a convex hull. We find eleven possible ground states for $YBa_2Cu_3O_{6+x}$, which corresponds to nine intermediate ground states between the reference structures. They are marked as blue dots in Fig. 4.16. As expected, the ground states of $YBa_2Cu_3O_{6+x}$ for oxygen contents 6.0, 6.5 and 7.0 correspond to the tetragonal, ortho-II and ortho-I phase, which are experimentally observed phases [73]. The planes, containing the substitutional sites, are shown in Fig. 4.17 for all eleven states. As before, copper atoms are depicted as bronze, oxygen atoms as red and vacancies as empty circles. The unit cells are encompassed by a black line and shaded in gray. We observe a clear pattern: All ground states have continuous Cu-O chains. For a low oxygen content, the chains are far apart, but come closer, the more the oxygen content is increased.

A previous study of Andersen et al. [72], based on an extension of the ASYNNNI model, produces similar results. They also predict the ortho-VIII phase at an oxygen content of 6.625, which corresponds state 7 in Fig. 4.17. Similarly, they predict the ortho-III phase at oxygen content 6.67, which corresponds to state 8 in Fig. 4.17. The ortho-III phase is also predicted by Ceder et al. [79] for the same oxygen content at $T = 0$ K. The ortho-V phase, which Andersen et al. predict for an oxygen content of 6.6, is included in our training set of low-energy states. It is not part of the convex hull, but considering the computational error, it could be a ground state. In a X-ray diffraction study of $YBa_2Cu_3O_{6+x}$, performed by Zimmermann et al. [73], the ortho-II, ortho-III, ortho-V and ortho-VIII superstructures are observed, while only the ortho-II phase shows three dimensional ordering, indicating that it is a more stable phase. Andersen et al. also

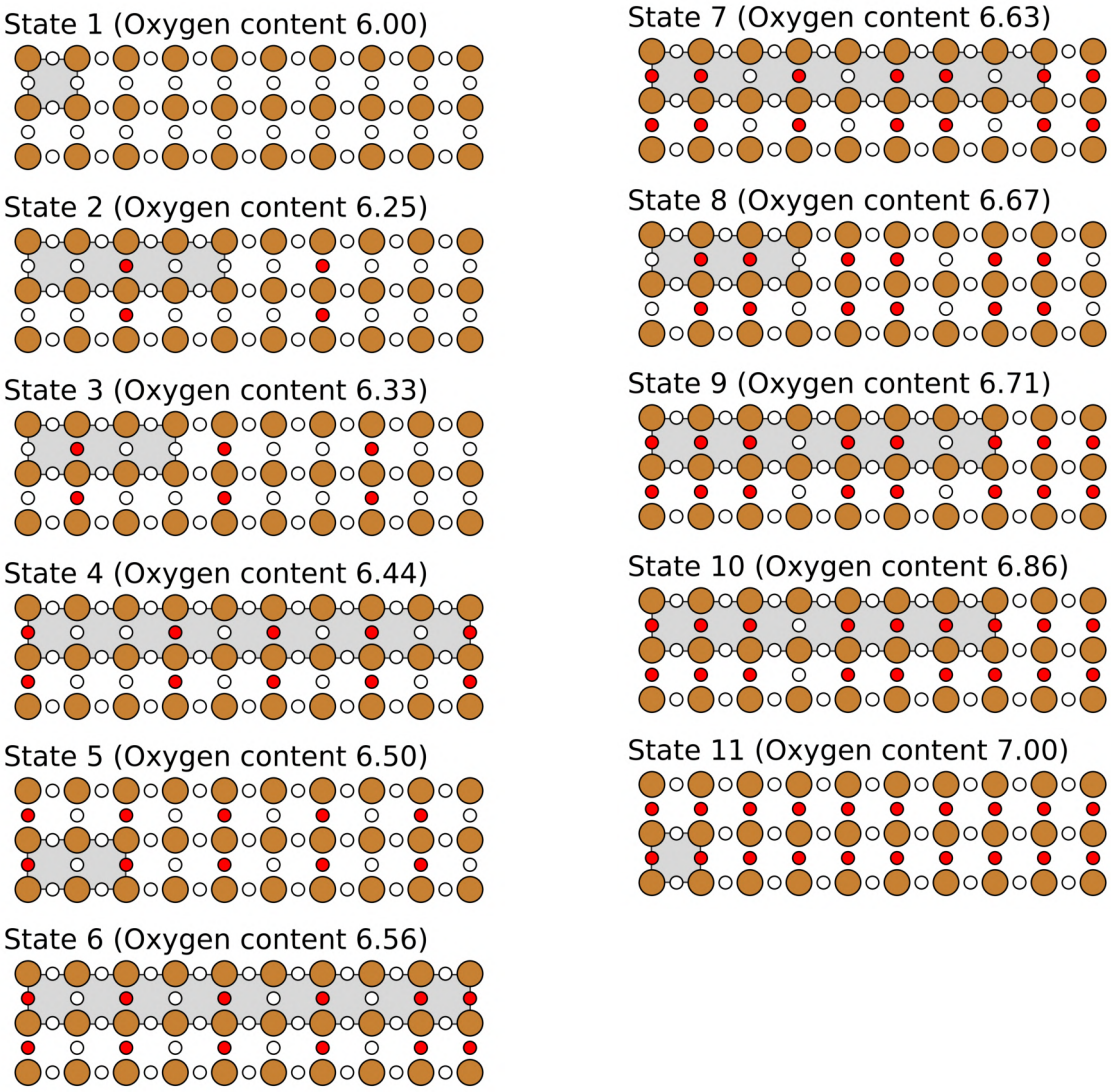

Figure 4.17: The planes, containing the substitutional sites, of the eleven possible ground states of $YBa_2Cu_3O_{6+x}$ are shown. Copper atoms are shown in bronze, oxygen atoms in red and vacancies as white circles. The unit cells are encompassed by a black line and shaded in gray. The states for oxygen contents 6.0, 6.5 and 7.0 correspond to the tetragonal, ortho-II and ortho-I phase. State 8 is the ortho-III phase and state 7 the ortho-VIII phase.

find experimental evidence of the the same orthorhombic phases [72]. Experimentally, none of the orthorhombic superstructures, apart from ortho-I, are observed to show long range ordering [73]. From our *ab initio* results, depicted in Fig. 4.16, it is difficult to determine which states are actual ground states at $T = 0$ K. Considering the error bars of our DFT calculations, we do not assume that all of the eleven states marked in blue are actual ground states. We draw a second, more conservative, convex hull, taking the error bars into account. Now, we only add data points to the convex hull, if the error bar of the corresponding point is completely below the connection line of two other points. With this approach the ortho-II phase at oxygen content 6.5 would not be considered a ground state. We decide to enforce its inclusion to the convex hull, as it is a more stable phase [73], and proceed otherwise as stated before. The resulting convex hull and ground states are shown in red in Fig. 4.16. We find six ground states, thus, four intermediate ground states, corresponding to states 3, 5, 8, and 9 in Fig. 4.17. They include, besides the ortho-II phase, the ortho-III phase. As explained, due to the error bars and because the considered configurations with oxygen contents between 6.4 and 6.7 have very similar energies of mixing, it is difficult to predict precisely which of the states are indeed ground states. Furthermore, we assume that the considered super cell sizes, up to nine as large as the parent lattice, might be too small to capture configurations that exhibit phase separation. A reasonable next step beyond this work, is to perform DFT calculations with even finer convergence criteria to allow for a clearer estimation of potential ground states of $YBa_2Cu_3O_{6+x}$.

4.8 Active learning workflow for structure selection

As discussed in Sec. 3.2.4, in order to reduce the variance of the training set, we want to reduce the value of τ from Eq. 3.35. We want to select a structure i that, when added to the training set, minimizes the variance, and therefore τ, the most. When adding a new structure to the training set, we obtain a new covariance matrix of the form:

$$\tilde{\boldsymbol{M}} = (\boldsymbol{X}^T\boldsymbol{X} + \boldsymbol{x}_i\boldsymbol{x}_i^T)^{-1}. \tag{4.2}$$

As previously, $\boldsymbol{X}$ is the matrix of cluster correlations and $\boldsymbol{x}_i$ is a column vector of the cluster correlations of a configuration i. In Sec. 3.2.4 we explained that, to calculate τ, we need to calculate its Frobenius product with the domain matrix:

$$\tilde{\tau} = (\boldsymbol{X}^T\boldsymbol{X} + \boldsymbol{x}_i\boldsymbol{x}_i^T)^{-1} : \boldsymbol{D} = \tilde{\boldsymbol{M}} : \boldsymbol{D} = tr(\tilde{\boldsymbol{M}}^T\boldsymbol{D}) = tr(\tilde{\boldsymbol{M}}\boldsymbol{D}). \tag{4.3}$$

$\tilde{\boldsymbol{M}}$ and $\boldsymbol{D}$ are square matrices of same dimension and their Frobenius product is equal to the trace $tr(\tilde{\boldsymbol{M}}^T\boldsymbol{D})$. By its definition in Eq. 4.2, $\tilde{\boldsymbol{M}}$ is the inverse of a Gram matrix and as such is a square and symmetric matrix, such that $\tilde{\boldsymbol{M}}^T = \tilde{\boldsymbol{M}}$. So, we obtain:

$$\tilde{\tau} = tr\left((\boldsymbol{X}^T\boldsymbol{X} + \boldsymbol{x}_i\boldsymbol{x}_i^T)^{-1}\boldsymbol{D}\right). \tag{4.4}$$

To simplify further, we make use of the Sherman-Morrison formula [80, 81] that states that, for a $n \times n$ invertible square matrix $\boldsymbol{A}$ and a column vector $\boldsymbol{u}$ of dimension $n \times 1$, we can rewrite: $(\boldsymbol{A} + \boldsymbol{u}\boldsymbol{u}^T)^{-1} = \boldsymbol{A}^{-1} - \frac{\boldsymbol{A}^{-1}\boldsymbol{u}\boldsymbol{u}^T\boldsymbol{A}^{-1}}{1+\boldsymbol{u}^T\boldsymbol{A}^{-1}\boldsymbol{u}}$, provided that $1 + \boldsymbol{u}^T\boldsymbol{A}^{-1}\boldsymbol{u} \neq 0$. We obtain:

$$\tilde{\tau} = tr\left(\left((\boldsymbol{X}^T\boldsymbol{X})^{-1} - \frac{(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{x}_i\boldsymbol{x}_i^T(\boldsymbol{X}^T\boldsymbol{X})^{-1}}{1 + \boldsymbol{x}_i^T(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{x}_i}\right)\ \boldsymbol{D}\right). \tag{4.5}$$

Next, we use that $tr(\boldsymbol{A} + \boldsymbol{B}) = tr(\boldsymbol{A}) + tr(\boldsymbol{B})$ to separate the equation into two terms. The first term $tr\left((\boldsymbol{X}^T\boldsymbol{X})^{-1}\ \boldsymbol{D}\right)$ does not depend on which structure we add to the training set, but only

on the original training set. The second term, that is subtracted from it, results from adding the new structure. By selecting the structure i that maximizes it, we reduce the variance as much as we can, given a set of candidate structures, we choose from. We can simplify the expression a bit further:

$$tr\left(\frac{(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{x}_i\boldsymbol{x}_i^T(\boldsymbol{X}^T\boldsymbol{X})^{-1}}{1+\boldsymbol{x}_i^T(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{x}_i}\boldsymbol{D}\right) = \frac{1}{1+\boldsymbol{x}_i^T(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{x}_i} tr\left((\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{x}_i\boldsymbol{x}_i^T(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{D}\right). \tag{4.6}$$

Above we use that the denominator is a scalar, since we multiply vectors and matrices of the dimensions: $(1 \times N_c)(N_c \times N_c)(N_c \times 1) = (1 \times 1)$. We can therefore move it outside of the trace. Next, we employ the invariance of the trace under cyclic permutations and perform permutations, such that we receive:

$$\frac{1}{1+\boldsymbol{x}_i^T(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{x}_i} tr\left(\boldsymbol{x}_i^T(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{D}(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{x}_i\right). \tag{4.7}$$

Similar to before, we can show that the term in the trace is a scalar by considering the dimensions of the single objects that are multiplied: $(1 \times N_c)(N_c \times N_c)(N_c \times N_c)(N_c \times N_c)(N_c \times 1) = (1 \times 1)$. Therefore, we can omit the trace and finally obtain:

$$\Delta\tau(\boldsymbol{X}, \boldsymbol{x}_i) = \frac{\boldsymbol{x}_i^T(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{D}(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{x}_i}{1+\boldsymbol{x}_i^T(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{x}_i}. \tag{4.8}$$

We call this value $\Delta\tau(\boldsymbol{X}, \boldsymbol{x}_i)$ as it describes how much τ from Eq. 3.35 is reduced by adding structure i to the training set. Eq. 4.8 has already been implemented in `CELL` [24] within the `StructureSelector` class in the method `select_structure` that returns the index for the structure with maximal $\Delta\tau(\boldsymbol{X}, \boldsymbol{x}_i)$. We discussed it in detail here, since, to the best of our knowledge, this approach for structure selection has not yet been discussed in a previous publication. Next, we illustrate which changes and additions are implemented in the context of this work.

Usually, following our workflow discussed in Sec. 4.3, we want to add several structures to the training set before building a new model. It is not advised to use the `select_structure` method multiple times on the same candidate set. Each time a new structure is added to the training set, the matrix of correlations $\boldsymbol{X}$ changes, since a row with the correlations for structure i is added. We need to consider this updated input matrix in the calculation of $\Delta\tau(\boldsymbol{X}, \boldsymbol{x}_i)$ in Eq. 4.8, when selecting an additional structure. If we do not, we risk to select several structures with similar correlations, that each reduce the variance of the original training set, but do not take each others correlations into account. In order to avoid selecting the same structure more than once, all chosen structures and their correlations $\boldsymbol{x}_i$ should be removed from the set of candidates. To take these considerations into account and to enable an iterative structure selection, we develop an active learning workflow.

We start with an initial training set $\mathscr{T}$ and a set of candidate structures $\mathscr{C}$. We want to apply the workflow on the previously described set of candidates from Iteration 3 in Sec. 4.5, containing 48,867 structures. During our test runs, we notice, that the set contains structures that are symmetrically equivalent. This likely stems from a problem in the symmetry packages used by `CELL` [24] that fail to recognize all symmetrically equivalent structures. To avoid selecting the same structure more than once, we need to exclude equivalent structures. This is not trivial in the case of a large set of structures. We cannot check the equivalence for all 48,867 structures by visualizing and comparing them. Also, we cannot make use of methods such as `CELL`'s `sset_equivalence_check`, which checks for equivalent structures in a structures set [24], since the candidate set is too large

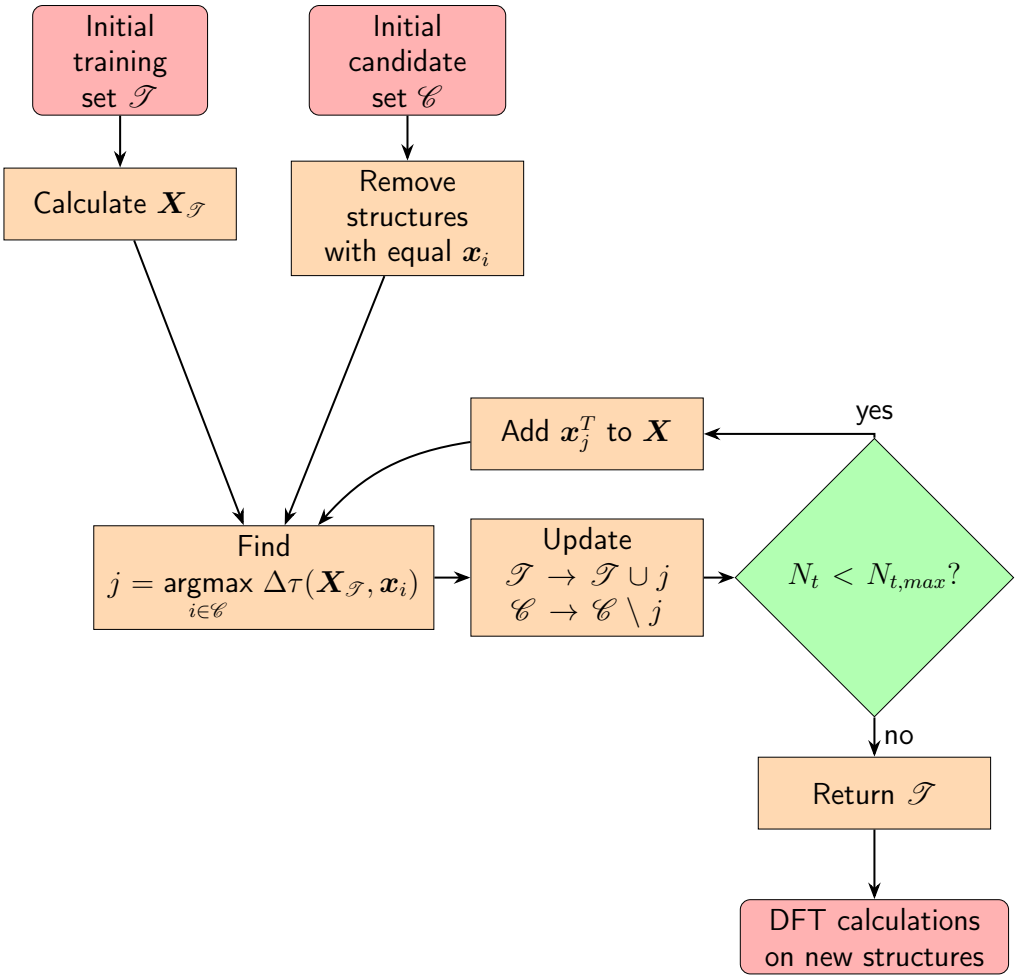

Figure 4.18: Active learning workflow for structure selection: For an initial training set $\mathscr{T}$ and candidate set $\mathscr{C}$ the correlations are calculated. Structures with equal correlations to a previous structure are removed from the candidate set. The structure that maximizes $\Delta\tau(\boldsymbol{X}, \boldsymbol{x}_i)$ is selected iteratively, while updating the training set $\mathscr{T}$ and candidate set $\mathscr{C}$ and the corresponding correlations after each iteration. Once the desired number of structures $N_{t,max}$ is found, DFT calculations are performed upon them.

for the algorithm to handle. Instead, we use a different approach: Symmetrically equivalent structures have the same correlations, so we calculate the matrix of correlations for the candidate set and remove rows with equal correlations. The corresponding structures are excluded from the training set. It is important to note, that by doing so, we ensure that no symmetrically equivalent structures are in the candidate set. However, since our basis is not complete, it is possible that some inequivalent structures have same correlations with respect to the truncated basis set and are excluded. To reduce this risk, we include the possibility to provide a larger clusters pool, employed only to calculate the correlations of the candidate set. The more expensive structure selection is based on a smaller clusters pool. By choosing an enlarged basis, it is less likely that we exclude inequivalent structures, as it is less likely that these structures will have equal correlations. For the considered candidate set, we choose a clusters pool containing the 1- and all 2-, 3- and 4-point clusters up to a radius of 10.89 Å. This results in a reduction from 48,867 to 46,694 remaining candidate structures. After excluding structures with equal correlations, we calculate $\Delta\tau(\boldsymbol{X}, \boldsymbol{x}_i)$ for every structure i in the reduced candidate set $\mathscr{C}$ and select the structure j that maximizes $\Delta\tau(\boldsymbol{X}, \boldsymbol{x}_i)$. The workflow is visualized in Fig. 4.18. In each iteration, we update the training set by adding structure j, such that $\mathscr{T} \rightarrow \mathscr{T} \cup j$. We update the candidate set by removing structure j: $\mathscr{C} \rightarrow \mathscr{C} \setminus \{j\}$. In the next iteration the correlations of structure j are added to the correlations matrix $\boldsymbol{X}$ as a row vector $\bar{\boldsymbol{x}}_j$. Then, again we calculate $\Delta\tau(\boldsymbol{X}, \boldsymbol{x}_i)$ for all structures in the updated candidate set $\mathscr{C}$ and select the structure that maximizes it. We repeat the process until we have reached the desired size of the training set $N_{t,max}$ and return the new training set, that contains all selected structures. From this, we can generate directly their input files for DFT calculations and start the next iteration of our general workflow outlined in Fig. 4.4. We call this method `select_structures_set` and it will be published in a future version of `CELL`.

The value of $\Delta\tau(\boldsymbol{X}, \boldsymbol{x}_i)$ depends on the domain matrix $\boldsymbol{D}$, for which several options are provided in `CELL` [24]. For this work, we compare three different methods, resulting from the domain matrices in Eq.s 3.36, 3.37 and 3.38. At first, we use a concentration dependent domain matrix from Eq. 3.37 to calculate τ for our initial training set with Eq. 3.35. For the considered fractional oxygen concentration range from $0 \leq c \leq 0.5$, we create thousand equally spaced steps and obtain the gray curve in Fig. 4.19. It indicates the quality of our training set for different concentrations. Our aim is to reduce the values of τ by selecting structures from the candidate set, employing our active learning workflow. We compare four training sets, optimized with different methods:

1. vdWC: Training set optimized by using the domain matrix from Eq. 3.36, as approximated by van de Walle and Ceder [45, 46].
2. CDPA with evenly spread concentrations: Calculating the domain matrix as concentration dependent population average with Eq. 3.37 [46] for a list of evenly spread concentrations between 0 and 0.5.
3. CDPA with weighted concentrations from candidate set: Calculating the domain matrix as concentration dependent average with Eq. 3.37 [46] for a list of concentrations containing all concentrations from the candidate set, while low concentrations up to $c < 0.1$ are included three times to give them more weight.
4. WPA: Calculation of the domain matrix by using Eq. 3.38. This corresponds to taking a weighted population average over the concentration range 0 to 0.5, where concentrations with more configurations are weighted less strongly [46].

For each method, we use the active learning workflow from Fig. 4.18 to iteratively select 33 structures and obtain the optimized training sets. For those we calculate τ, using Eq. 3.35, for the

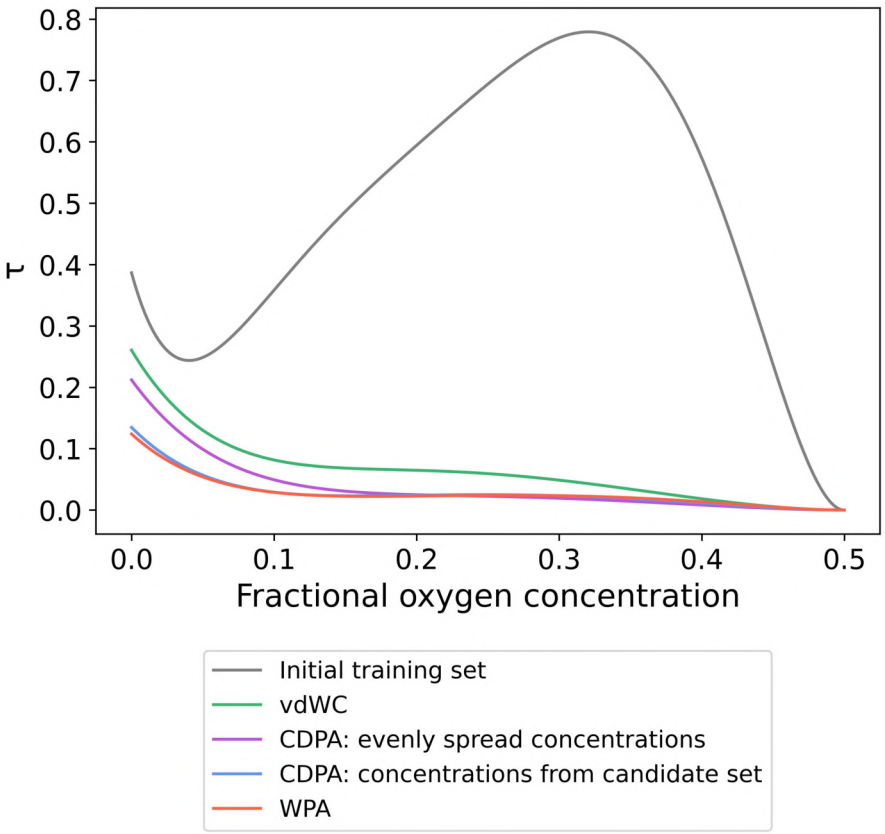

Figure 4.19: Comparison of the values of τ for the initial training set (gray) containing 67 structures and training sets optimized by the active learning workflow with different domain matrices each containing 100 structures. For the green curve (vdWC) an identity matrix was chosen as domain matrix [45,46]. For the purple and blue curve (CDPA) the concentration dependent domain matrix from Eq. 3.37 is used; once for evenly spread concentrations (purple) and once for concentrations from the candidate set while concentrations $c > 0.1$ are included two times more. For the red curve (WPA) the domain matrix from Eq. 3.38 that corresponds to a weighted average over the concentration is chosen.

considered concentration range and produce the four additional curves in Fig. 4.19. Training set 1, optimized by the vdWC method is shown in green, training set 2 (CDPA with evenly spread concentrations) in purple, training set 3 (CDPA with weighted concentrations from candidate set) in blue and training set 4 (WPA) in orange. All optimized training sets have significantly smaller values of τ, resulting in reductions up to 0.76 (97.8%) for concentration $c \approx 0.32$. It is not surprising that τ is reduced this much, since we started with an initial training set that contains 67 structures and compare to optimized sets containing 100 structures, which is an increase of almost 50% of the initial size.

We observe that the curves for training sets 4 (WPA, orange) and 2 (CDPA with weighted concentrations from candidate set, blue) are almost identical. For training set 2 we included concentrations from the candidate set with $c < 0.1$ three times, such that the domain matrix $D^{CDPA}(c)$ is calculated two times more for low oxygen concentrations $c < 0.1$. For low concentrations, there are considerably fewer structures in the candidate set, than for higher concentrations. Taking the weighted population average (WPA), where concentrations with few structures are weighted more strongly, produces very similar results to training set 2, where lower concentrations are also weighted more strongly. Comparing both methods shows that 19 of the 33 selected structures are the same. The methods vdWC and CDPA with evenly spread concentrations also reduce the values of τ significantly, but perform worse for small concentrations, especially for $c < 0.1$. From the results, presented in Fig. 4.19, we conclude that all methods behave similarly, such that $\tau(c)$ for concentrations between $0.074 \leq c \leq 0.5$ is reduced below values of 0.1, whereas for lower concentrations τ ranges up to 0.26 (vdWC), 0.21 (CDPA with evenly spread concentrations), 0.13 (CDPA with concentrations from candidate set) and 0.12 (WPA). For concentrations $c \geq 0.43$ values are reduced to $\tau < 0.01$, and $\tau < 0.001$ for $c \geq 0.48$ for all optimized training sets. We decide to select training set 4 (WPA), as it performs best for low concentrations. We can confirm the suggestion by Mueller and Ceder [46] that for a candidate set, where all structures are of equal importance, but for some concentrations many more configurations are available, a weighted population average (WPA) is the preferable method.

The active learning workflow, together with a weighted population average domain matrix, is employed to select the final 33 structures for the training set. As discussed in Sec. 4.5, we are able to reduce the RMSE and MAE values significantly and obtain a good standard CE model with only 16 features. We attribute part of this success to the active learning workflow.

4.9 CE models for lattice constants

To analyze the behavior of lattice constants with varying oxygen content, we build additional CE models. There are a few considerations that need to be taken into account. Specifically, we need to regard the disorder limit. There is no preferred $\boldsymbol{a}$ or $\boldsymbol{b}$ direction in the disordered structure, such that it has tetragonal symmetry, which needs to be considered in our modeling. In App. A we show that, when using the indicator-binary basis, the cluster correlation of the one point cluster corresponds to the fractional concentration of oxygen atoms, such that $X_1(\boldsymbol{\sigma}) = c$. In the disorder limit, the site occupancies are not correlated. The cluster correlation of a cluster $\boldsymbol{\alpha}$ with $n_{\boldsymbol{\alpha}}$ points is simply the cluster correlation of the one point cluster $X_1(\boldsymbol{\sigma})$ multiplied with itself $n_{\boldsymbol{\alpha}}$ times, such that, in the disorder limit [58]:

$$X_{\boldsymbol{\alpha}}(\boldsymbol{\sigma})^{(dis)} = X_1(\boldsymbol{\sigma})^{n_{\boldsymbol{\alpha}}} = c(\boldsymbol{\sigma})^{n_{\boldsymbol{\alpha}}}. \tag{4.9}$$

We want to capture that lattice constant $a = b$ in the disorder limit. We build a CE model for the property $P = \frac{|a-b|}{2}$. To achieve the required symmetry, we want to promote:

$$P^{(dis)}(\boldsymbol{\sigma}) = \left(\frac{|a-b|}{2}\right)^{(dis)} = \sum_{\boldsymbol{\alpha}} X_\alpha(\boldsymbol{\sigma})^{(dis)} \mathcal{J}_{\boldsymbol{\alpha}} = \sum_{\boldsymbol{\alpha}} c(\boldsymbol{\sigma})^{n_{\boldsymbol{\alpha}}} \mathcal{J}_{\boldsymbol{\alpha}} = 0. \tag{4.10}$$

In the last step, we enforce that $\frac{|a-b|}{2} = 0$ in the disorder limit. From Eq. 4.10, we can deduce that the coefficient for the 1-point cluster needs to be zero ($\mathcal{J}_{1pt} = 0$) to meet the constraint. To demonstrate this, we consider the first terms of the sum in Eq. 4.10: $c^0 \cdot \mathcal{J}_0 + c \cdot \mathcal{J}_{1pt} + c^2 \cdot (\mathcal{J}_{2pt,1} + \mathcal{J}_{2pt,2} + ...) + c^3(\mathcal{J}_{3pt,1} + \mathcal{J}_{3pt,2} + ...) + ... = 0$. There is only one symmetrically distinct 1-point cluster and to ensure that the constraint holds for different concentrations, we need to enforce that $\mathcal{J}_{1pt} = 0$. We can argue similarly for the coefficient $\mathcal{J}_0$ that corresponds to the intercept of the model. Consequently, we exclude the empty cluster and the 1-point cluster from initial clusters pools for the property $P = \frac{|a-b|}{2}$, thereby enforcing coefficients equal to zero.

To include this constraint in our model optimization, we search for the vector of coefficients that reduces Eq. 3.23, with the loss function:

$$\Phi(\boldsymbol{\mathcal{J}}^*) = \lambda \|\boldsymbol{X}^{(dis)} \boldsymbol{\mathcal{J}}^*\|_2^2 = \lambda \sum_s \left(\sum_{\boldsymbol{\alpha}} c(\boldsymbol{\sigma}_s)^{n_{\boldsymbol{\alpha}}} \boldsymbol{\mathcal{J}}^*_{\boldsymbol{\alpha}} \right)^2. \tag{4.11}$$

To build the CE model for $P = \frac{|a-b|}{2}$, we create a custom `scikit-learn` [25] estimator that minimizes the loss function, which is the sum of the residual sum of squares and $\Phi(\boldsymbol{\mathcal{J}}^*)$ as defined above. For the minimization we employ the `minimize` method from the the `scipy` python library [82] with the Sequential Least Squares Programming (SLSQP) method [83, 84]. In addition, we calculate the analytical solution by differentiating the loss function with respect to $\boldsymbol{\mathcal{J}}$ and setting to zero. We obtain:

$$\begin{aligned} \frac{\partial \mathcal{L}(\lambda, \boldsymbol{\mathcal{J}})}{\partial \boldsymbol{\mathcal{J}}} &= \frac{\partial L(\lambda, \boldsymbol{\mathcal{J}})}{\partial \boldsymbol{\mathcal{J}}} = \frac{\partial RSS(\boldsymbol{\mathcal{J}})}{\partial \boldsymbol{\mathcal{J}}} + 2\lambda \boldsymbol{X}^{(dis)T} \boldsymbol{X}^{(dis)} \boldsymbol{\mathcal{J}} \\ &= -2\boldsymbol{X}^T \boldsymbol{P} + 2\boldsymbol{X}^T \boldsymbol{X} \boldsymbol{\mathcal{J}} + 2\lambda \boldsymbol{X}^{(dis)T} \boldsymbol{X}^{(dis)} \boldsymbol{\mathcal{J}} = 0, \end{aligned} \tag{4.12}$$

and thereby:

$$\boldsymbol{\mathcal{J}} = \left(\boldsymbol{X}^T \boldsymbol{X} + \lambda \boldsymbol{X}^{(dis)T} \boldsymbol{X}^{(dis)} \right)^{-1} \boldsymbol{X}^T \boldsymbol{P}. \tag{4.13}$$

This is similar to ridge regression (see Eq. 3.25), only that the identity matrix is replaced by a quadratic form of the correlation matrix in the disorder limit $\boldsymbol{X}^{(dis)}$. We obtain equivalent results for the optimized vector of coefficients when using the customized regressor and by calculating the analytical solution for equal values of λ, apart from numerical discrepancies in the order of $3 \cdot 10^{-9}$Å to $3 \cdot 10^{-5}$Å. We choose $\lambda = 1.0$, such that the model predicts the tetragonal symmetry of $YBa_2Cu_3O_6$ correctly. The model for $P = \frac{|a-b|}{2}$ is built with the indicator-binary basis ($\sigma_i \in \{0, 1\}$ in Eq. 3.11), to match the basis for which we defined the constraint. Additionally, we build a model for $P = \frac{a+b}{2}$ using the chebyshev basis, as previously. From the predictions of these models, we can later extract the predicted property values of the lattice constants a and b. Lastly, we build a model for lattice constant c.

The errors of all models and the models' parameters are summarized in Fig. 4.20. We are able to find good models for $\frac{a+b}{2}$ and c, using relatively small clusters pools of eight and six clusters which include 1-, 2-, 3- and for the $\frac{a+b}{2}$ model also 4-point clusters. The pools are optimized by

Model for (a+b)/2

Standard CE

Estimator:
OMP, n=8
Cluster selector:
combinatorial search
Optimized clusters pool:
8 clusters

Errors [mÅ]

	Fit	**CV_{LOO}**
RMSE	0.72	0.85
MAE	0.53	0.61

Model for |a-b|/2

Standard CE

Estimator:
Custom estimator including constraint
Cluster selector:
LASSO
Optimized clusters pool:
25 clusters

Errors [mÅ]

	Fit	**CV_{LOO}**
RMSE	6.15	13.51
MAE	4.98	8.42

Model for c

Non linear CE

Estimator:
Ridge regression, $\lambda = 10^{-8}$
Cluster selector:
combinatorial search
Optimized clusters pool:
6 clusters

Errors [mÅ]

	Fit	**CV_{LOO}**
RMSE	9.84	12.34
MAE	6.07	6.91

Figure 4.20: The Fit and CV_{LOO} scores of the CE models built on properties of the lattice constants a, b and c are shown. The methods used for the calculations of ECIs and cluster selection are shown on the left-hand side in each panel. The model for the property |a-b|/2 is built with a custom regressor that incorporates a constraint for tetragonal symmetry in the disorder limit.

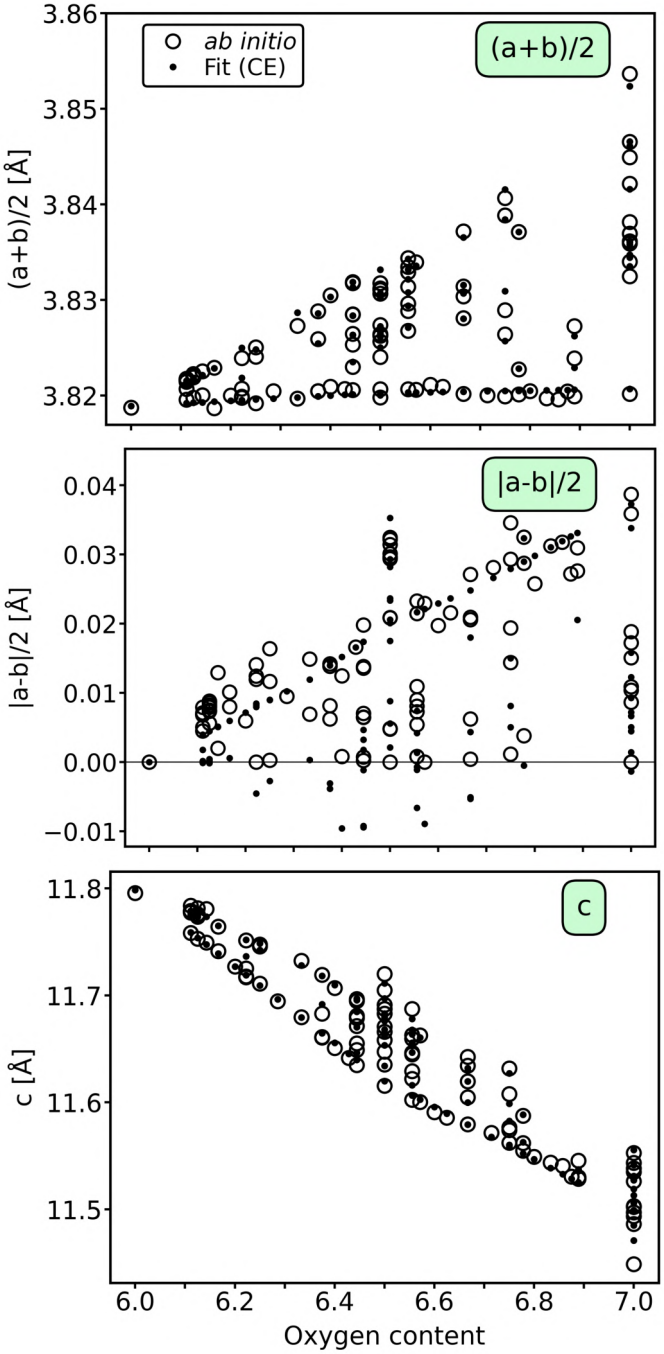

Figure 4.21: The *ab initio* target values (circles) and models' predictions (dots) for the properties of lattice constants a, b and c are shown. The property $\frac{|a-b|}{2}$, which reveals whether the configuration is tetragonal or orthorhombic, is difficult to model, such that the predictions deviate more from the target values.

a combinatorial search, starting from an initial pool of 31 clusters, including the 1-point, 2-point clusters up to a radius of 10.89 Å, 3-point clusters up to a radius of 7.71 Å and 4-point clusters up to 3.86 Å. The errors for the model trained on lattice constants c are larger than for $\frac{a+b}{2}$, for instance RMSE-Fit$_{\mathrm{a+b/2}}$ = 0.72 mÅ and RMSE-Fit$_{\mathrm{c}}$ = 4.88 mÅ. However, lattice constant c is also about three times as large as a and b and varies with increasing oxygen content approximately ten times more than $\frac{a+b}{2}$ or $\frac{|a-b|}{2}$. Therefore, we do not need to thrive for the same accuracy to make informative predictions about its behavior with increasing oxygen content. The modeling of $\frac{|a-b|}{2}$ is challenging. We try various different clusters pools, including 1-, 2-, 3-, 4- and 5-point clusters with radii ranging from 4 to 10.9 Å and optimize them with LASSO, OMP, a combinatorial search and combinations of those methods. The property values of $\frac{|a-b|}{2}$ are in the range of 0 to 40 mÅ and are difficult to model. Additionally, we need to apply the constraint, by choosing a $\lambda > 0$, which further increases the fit and CV scores. Yet, the constraint is needed to include physical reasoning into our models by promoting the tetragonal symmetry in the disorder limit. Additionally, due to limitations regarding the scalability of statistical thermodynamics simulations in `CELL`, we need to find an optimized clusters pool as small as possible. Considering this, we obtain best results for a pool of 25 clusters, optimized by applying LASSO-CV with an optimized $\lambda_{LASSO} = 7.97 \cdot 10^{-4}$ on an initial pool of 2-,3- and 4-point clusters with radii 10.89 Å, 8.0 Å and 4 Å.

The models and their predictions are shown in Fig. 4.21. The target values are depicted as circles and are normalized to the parent lattice. The predictions by the corresponding models are depicted as dots. The first panel shows the targets and predictions for $\frac{a+b}{2}$, the second panel for $\frac{|a-b|}{2}$ and the third one for lattice constant c. It is apparent, that the model for $\frac{|a-b|}{2}$ performs worse than the model for $\frac{a+b}{2}$, as the property is more difficult to model. We even get negative predictions, which can happen, but is unphysical. Negative predictions obtained by the model are interpreted as zero, but do not occur for the subsequent statistical thermodynamics simulations, discussed in the next chapter. By promoting the constraint of tetragonal symmetry in the disorder limit, the modeling of $\frac{|a-b|}{2}$ is necessary to obtain physical predictions. An extension with nonlinear features could improve the model quality, but this would require some modifications and more complex and expensive Monte Carlo simulations that are not possible to realize in the scope of this work.

4.10 Statistical thermodynamics simulations

To obtain temperature dependent properties, we perform statistical thermodynamics simulations, using MC sampling, as explained in Chap. 3. First, we calculate the specific heat according to Eq. 3.41. To predict the energies of configurations, we use the optimized standard CE model with 16 features, that was introduced in Sec. 4.5. We consider two compositions: $YBa_2Cu_3O_{6.71}$ and $YBa_2Cu_3O_{6.5}$. Close to a second-order transition, the energy varies strongly which is reflected by peaks in the specific heat spectra. Based on previous experimental studies and ASYNNNI model predictions [67, 72, 73], we expect to observe two peaks in the specific heat spectrum of $YBa_2Cu_3O_{6.71}$ and one peak for $YBa_2Cu_3O_{6.5}$. Additionally, we perform statistical thermodynamics simulations at a fixed temperature for increasing oxygen content and predict the lattice constants with the machine learning models introduced in Sec. 4.9.

As mentioned in Chap. 3, we need to converge the MC trajectories with respect to the size of the simulation cell, the total number of steps N_{steps} and the number of equilibration steps N_{eq}. Our model is complex, incorporating many more interactions than previous models, so obtaining MC

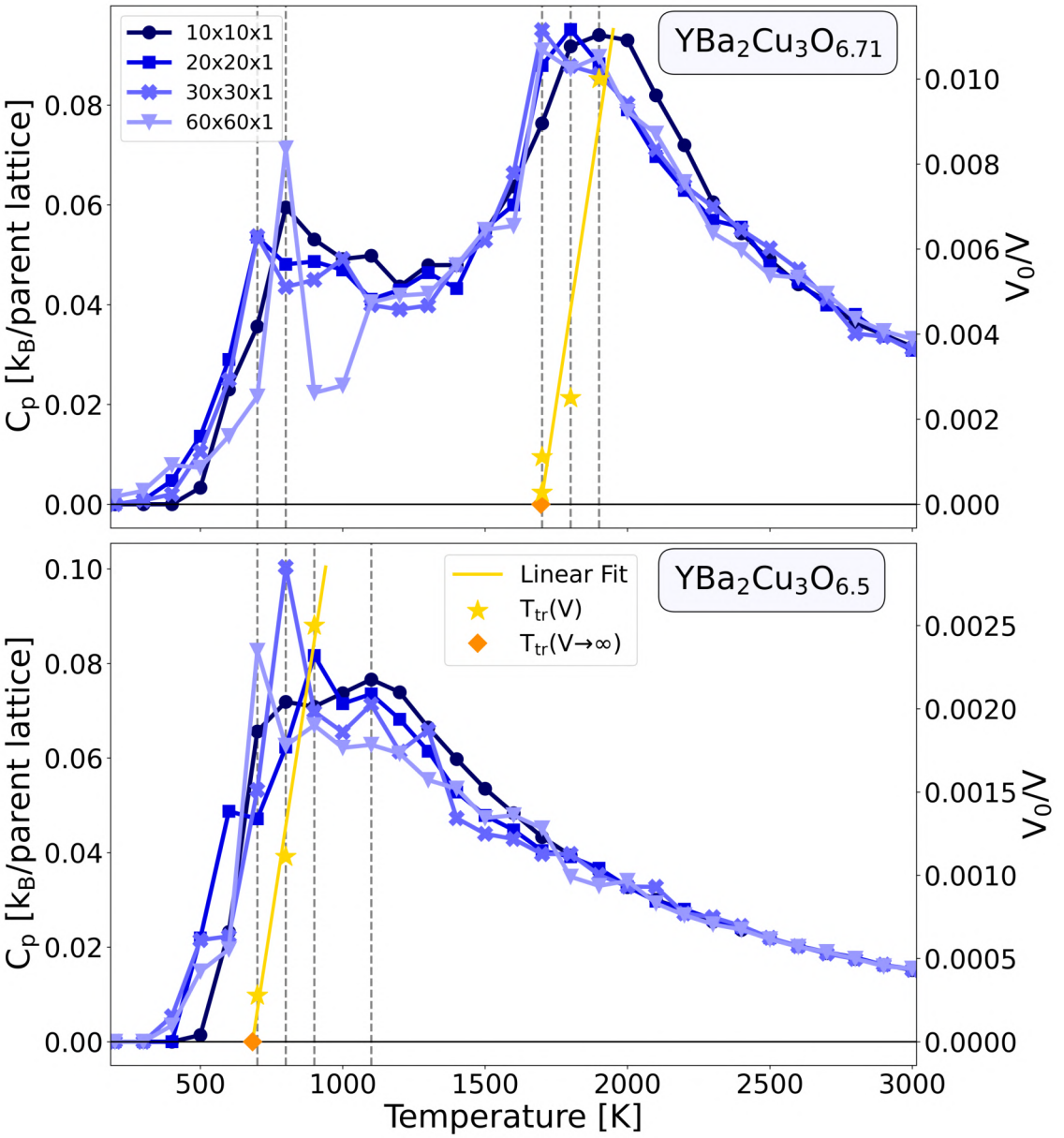

Figure 4.22: Specific heat C_p of $YBa_2Cu_3O_{6.71}$ (top) and $YBa_2Cu_3O_{6.5}$ (bottom) obtained through statistical thermodynamics simulations with Monte Carlo Metropolis sampling. C_p is shown over temperatures ranging from 200 to 3000 K, as well as for increasing simulation cell sizes, ranging from 10x10x1 (dark blue) to 60x60x1 (light blue). Simulation cell sizes are provided with respect to the size of the parent lattice. The macroscopic transition temperature $T_{tr}(V \to \infty)$ is obtained by a linear fit (yellow line) and shown as orange diamond.

trajectories is time consuming and expensive. Consequently, we apply a rather coarse temperature grid ranging from 200 to 3000 K in 100 K increments and perform simulations for four different cell sizes: 10x10x1, 20x20x1, 30x30x1 and 60x60x1 with respect to the parent lattice. The specific heat spectra for $YBa_2Cu_3O_{6.71}$ are shown in the first panel of Fig. 4.22. The results for a 10x10x1 simulation cell are shown in dark blue, with the color becoming lighter as the size of the simulation cell increases. The peak positions are accentuated by vertical, dashed gray lines. As expected, two peaks are observed. These peaks become sharper and more pronounced as the size of the simulation cell increases. Simulations with larger cells require an increasing number of equilibration steps, as there are many more possible configurations and it takes longer for the system to equilibrate. For the two smaller cells, it is sufficient to calculate approximately $N_{steps} \approx 2.7 \cdot 10^6$ steps and to set the number of equilibration steps to $N_{eq} = 5 \cdot 10^5$. This corresponds to 27,000 and 6,750 sweeps respectively, where the number of sweeps is defined as the number of steps N_{steps} divided by the number of parent lattices contained in the simulation cell volume V. Less steps/sweeps would have also been sufficient for these cells. For the 30x30x1 cell, we need to perform up to $N_{steps} = 11 \cdot 10^6$ steps, corresponding to 12,222 sweeps, while excluding $N_{eq} = 9 \cdot 10^6$ equilibration steps, when calculating the specific heat. For the 60x60x1 cell, we perform up to $N_{steps} \approx 33 \cdot 10^6$ steps, corresponding to 9,166 sweeps, of which $N_{eq} = 23 \cdot 10^6$ steps are equilibration steps. We assume that the 60x60x1 curve is not yet converged, as the position of the first peak is observed at a higher temperature than for the next smaller 30x30x1 cell. Contrary, we expect the peak positions to shift to lower temperatures with increasing size of the simulation cell, as the transition temperature scales linearly with the inverse of the simulation cell volume [61, 85]. To achieve a similar number of sweeps as for the 30x30x1 cell, we would need to perform $N_{steps} \approx 44 \cdot 10^6$ steps, which is not feasible in the scope of this work, but will hopefully be achievable in further studies due to recent improvements in the scalability of MC simulations in `CELL`.

Regarding $YBa_2Cu_3O_{6.5}$, we perform $N_{steps} \approx 10.9 \cdot 10^6$ steps for the 10x10x1 cell, corresponding to 109,000 sweeps, and exclude $N_{eq} \approx 1.0 \cdot 10^6$ equilibration steps from averaging. Similarly, for the 20x20x1 cell, we perform $N_{steps} \approx 11.7 \cdot 10^6$ steps, corresponding to 29,250 sweeps, and set $N_{eq} = 4.0 \cdot 10^6$ equilibration steps. For the 30x30x1 cell, $N_{steps} \approx 11.5 \cdot 10^6$ steps are performed, so 12,778 sweeps, with $N_{eq} = 8.2 \cdot 10^6$ equilibration steps. For the largest cell, 60x60x1, up to $N_{steps} \approx 23.5 \cdot 10^6$ steps, so 6,528 sweeps are performed and a number of $N_{eq} \approx 17 \cdot 10^6$ equilibration steps is set. As previously, for the smaller cells, less steps would have been sufficient. The specific heat spectrum of $YBa_2Cu_3O_{6.5}$ is shown in the bottom panel of Fig. 4.22. We observe one peak, that is indicative of the tetragonal-orthorhombic phase transition.

To estimate the transition temperature in the macroscopic limit, we adopt an approach of Troppenz et al. [86], who make use of the scaling of the transition temperature with the inverse of volume V, by performing a linear fit of the following form:

$$T_{tr}(V) = T_{tr}(V \to \infty) + m\frac{V_0}{V}. \tag{4.14}$$

$T_{tr}(V)$ is the transition temperature observed for a MC trajectory at a simulation cell of volume V. It is located at the position of the peak, that is observed in the specific heat spectrum. $T_{tr}(V \to \infty)$ is the transition temperature in the macroscopic limit, which we aim to determine. The slope of the linear function is denoted as m and V_0 corresponds to the volume of the parent lattice. We rearrange the equation and perform a least squares fit for the function:

$$\frac{V_0}{V} = \frac{1}{m}\left(T_{tr}(V) - T_{tr}\left(V \to \infty\right)\right). \tag{4.15}$$

The linear fits are visualized as yellow lines in Fig. 4.22. The y-axis on the right-hand side shows the volume ratio, $\frac{V_0}{V}$, that is considered in the fit. $T_{tr}(V \to \infty)$ is depicted by an orange diamond.

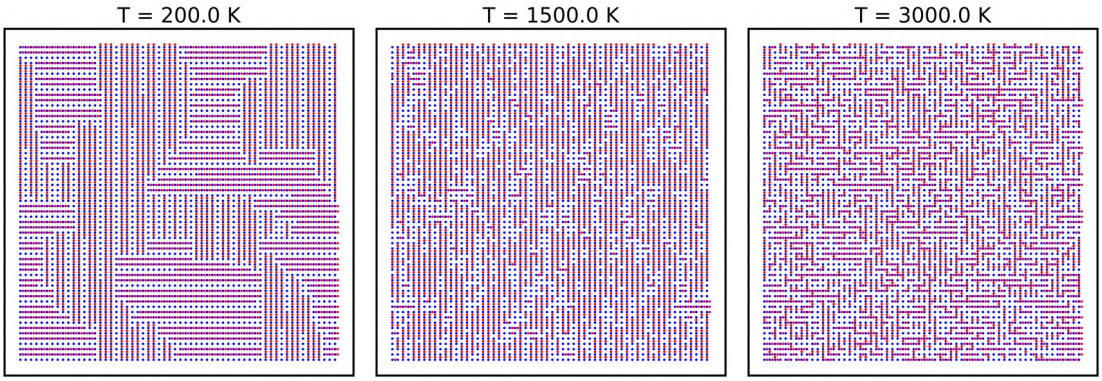

Figure 4.23: The ordering of oxygen atoms and vacancies for $YBa_2Cu_3O_{6.71}$ is depicted for temperatures of 200 K (left panel), 1500 K (middle panel) and 3000 K (right panel). To enhance contrast, copper atoms are shown in blue and oxygen atoms in red. It is apparent that at high temperatures the structure is disordered, which leads to tetragonal symmetry. At lower temperatures domains of ordered Cu-O chains form. Ortho-I and ortho-III ordering are present.

For $YBa_2Cu_3O_{6.5}$, we decide to exclude the peak position for the 10x10x1 cell from the linear fit. Due to finite size effects, this peak is broad, and the maximum position showed to be unstable, when varying the number of equilibration steps. Therefore, the peak position is not consistent enough to be included in the linear fit. We obtain the following transition temperatures in the macroscopic limit: $T_{tr,\mathrm{YBCO6.71}} = 1696 \pm 19$ K and $T_{tr,\mathrm{YBCO6.5}} = 683 \pm 21$ K. Uncertainties are obtained from the square root of the diagonal elements of the covariance matrix of the linear fit from Eq. 4.15. In order to compare these values with those of previous studies, we digitize the phase diagram in Fig. 9 of a paper by Zimmermann et al. [73]. It includes both theoretical predictions and experimental results, e.g. by Andersen et al. [87]. Experimentally, they found a transition temperature of $T_{tr,\mathrm{YBCO6.5}} \approx 670$ K for $YBa_2Cu_3O_{6.5}$. This result agrees remarkably well with our prediction of $T_{tr,\mathrm{YBCO6.5}} = 683 \pm 21$ K.

To gain a better understanding of the ordering of oxygen atoms and vacancies, we show some of the lowest energy structures obtained by the trajectories for different temperatures. First, we show three configurations of $YBa_2Cu_3O_{6.71}$ in Fig. 4.23 at temperatures of 200 K (left), 1500 K (middle) and 3000 K (right panel). Oxygen atoms are depicted as red and copper atoms as blue dots. It is apparent that, at high temperatures, the configuration is disordered (right panel). As the temperature is lowered, longer Cu-O chains begin to form. These chains are randomly distributed at high temperatures, but as the temperature decreases, they become longer and more ordered (middle panel), thereby breaking the tetragonal symmetry. At a temperature of 200 K, large domains of ordered Cu-O chains are present. We observe domains of neighboring Cu-O chains, corresponding to an ortho-I ordering, as well as ortho-III ordering in the top left of the structure. In Fig. 4.24, we present the lowest energy configurations obtained by MC sampling of $YBa_2Cu_3O_{6.5}$, considering the same temperatures as previously. As for $YBa_2Cu_3O_{6.71}$, the configurations are disordered at high temperatures, while longer Cu-O chains form as the temperature decreases. At 200 K we observe ortho-II, as well as ortho-I ordering, and some chains with a distance of $3a$.

The oxygen-vacancy ordering in $YBa_2Cu_3O_{6+x}$ is affected, not only by temperature, but also by the content of oxygen atoms. Therefore, we perform statistical thermodynamics simulations at a fixed temperature of $T = 900$ K, while increasing the oxygen content. We consider ten

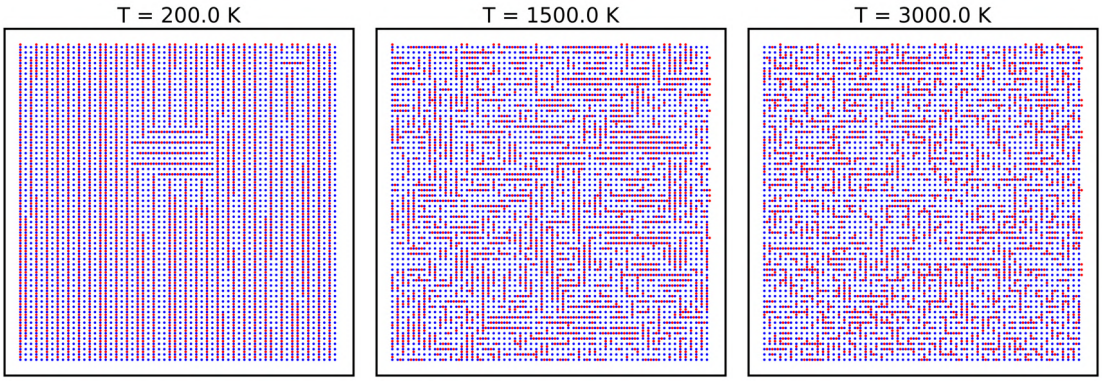

Figure 4.24: The ordering of oxygen atoms and vacancies with increasing temperatures for $YBa_2Cu_3O_{6.5}$ is depicted. Copper atoms are shown in blue and oxygen atoms in red. Disordered tetragonal configurations are observed at high temperatures, while for low temperatures domains with long Cu-O chains form. At 200 K both ortho-I, as well as ortho-II ordering is observed.

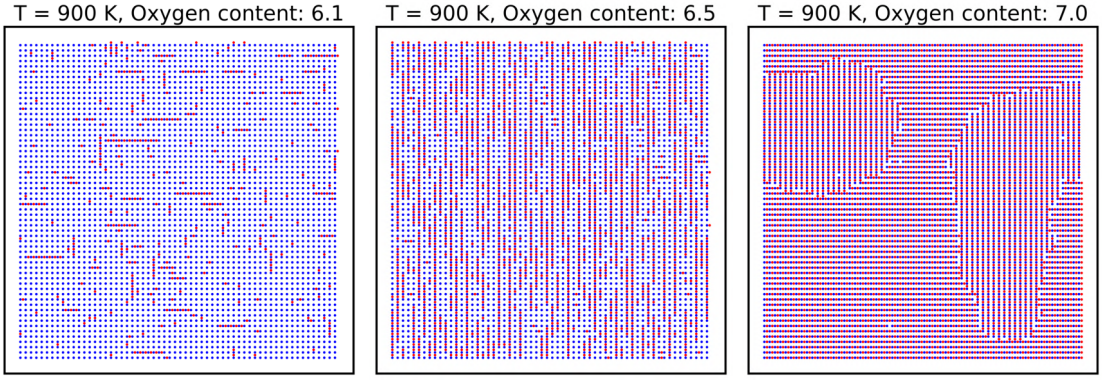

Figure 4.25: The ordering of oxygen atoms and vacancies obtained by MC sampling with a 60x60x1 simulation cell at fixed temperature $T = 900$ K is shown for increasing oxygen content.

compositions, with oxygen contents ranging from 6.0 to 7.0. Three configurations, obtained from this MC trajectory, are shown in Fig. 4.25 for oxygen contents of 6.1 (left panel), 6.5 (middle panel) and 7.0 (right panel). At low concentrations, some randomly distributed Cu-O chains are present, as well as randomly distributed oxygen atoms. As expected, increasing the oxygen concentration enables the formation of long Cu-O chains that increase the orthorhombicity of the structures. At an oxygen content of 6.5, long Cu-O chains are present, exhibiting ortho-I and ortho-II ordered sections. At an oxygen content of 7.0, three distinct domains are observed, all of which exhibit ortho-I ordering.

To the best of our knowledge, a theoretical prediction of the behavior of lattice constants with increasing oxygen content, using CE models, has not yet been performed. In the following, we employ the previously constructed CE models for lattice constant properties (see Sec. 4.9) to predict the lattice constants for the ten configurations obtained as last structure of the corresponding MC trajectory. By using the last structure, we ensure to obtain an equilibrated, yet random configuration. The larger the considered simulation cell size, the more informative the predicted lattice

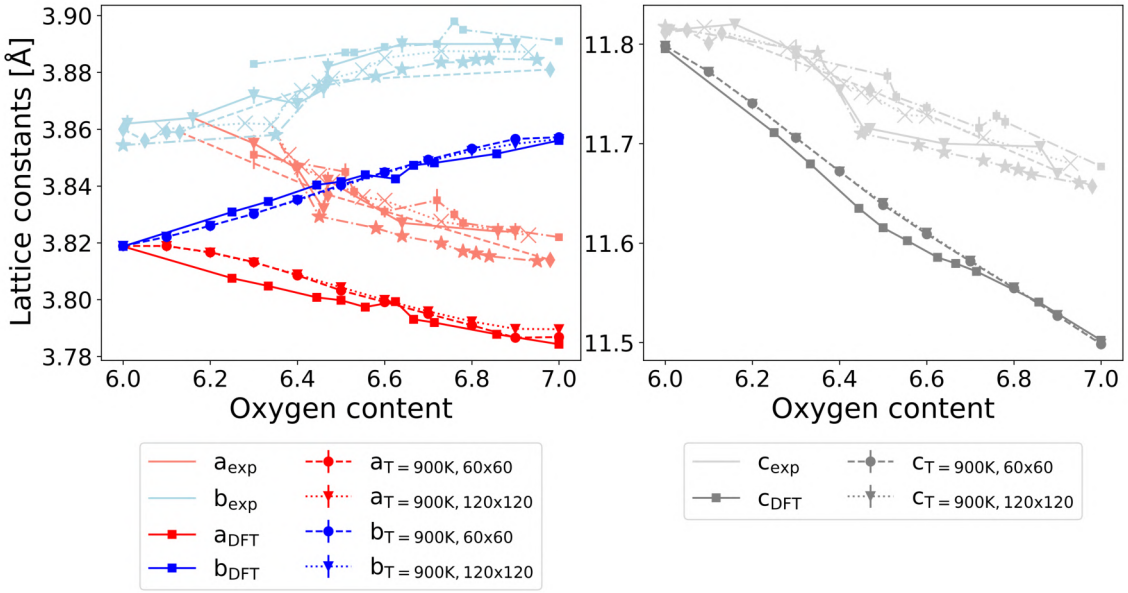

Figure 4.26: The lattice constants of $YBa_2Cu_3O_{6+x}$ at different contents of oxygen atoms are shown. Experimental results, as introduced in Chap. 2, are shown in light blue, light red and light gray. Additionally, results from *ab initio* calculations (solid lines and squares) and thermodynamic simulations at $T = 900$ K with a 60x60x1 (dashed line and circles) and 120x120x1 (dotted line and triangles) simulation cell are shown in red for a, blue for b and gray for c.

constants become, as larger super cells allow for a more comprehensive sampling of the configuration space. This improves the quality of lattice constants predictions. The lattice constants converge much faster than the specific heat, such that less steps need to be performed. We consider simulation cells of sizes 60x60x1 and 120x120x1 and perform at least $N_{steps} = 9 \cdot 10^6$ steps. To check for convergence and to estimate the statistical error, we calculate the lattice constants five times. We start from trajectories with at least 8 million steps, successively increase the number of steps and predict the lattice parameters each time. Then, we calculate average values and standard deviations. The results are shown in Fig. 4.26. The standard deviations are at most $8 \cdot 10^{-4}$Å and therefore not visible in the plot. We include the experimental results, introduced in Chap. 2, to enable comparison with our theoretical predictions. As before, the lattice constants a and b are shown in the left panel in red and blue, while c is shown in the right panel in gray. The experimental results are shown in lighter shades of the colors to distinguish them from the theoretical results. For comparison, we also present the lattice constants obtained by DFT calculations of the eleven possible ground states, introduced in Sec. 4.7. They are shown as solid lines and squares. The predictions of our models at $T = 900$ K are shown as dashed lines and circles for the 60x60x1 cell and as dotted lines and triangles for the 120x120x1 cell. Our models predict almost tetragonal symmetry at oxygen content 6.1, while predicting similar lattice constants as the *ab initio* data at an oxygen content of 7.0. This demonstrates that our model can account for the effects of disorder at finite temperatures that lead to tetragonal symmetry. Interestingly, our model shows a slight change in slope for the lattice constants a and b around 6.4, where also a change in slope is observed experimentally. Regarding the lattice constant c, the *ab initio* data show a slight kink around an oxygen content of 6.4, which is less pronounced in the predictions at finite temperature. In this exact region, experiments show greater variability in the observed

values of c. Thus, our models show qualitative agreement with the trends in the lattice constants with increasing oxygen content. Nevertheless, our model is not able to correctly predict tetragonal symmetry at low oxygen concentrations. Comparing to the phase diagram of Zimmermann et al. [73], based on the experimental data by Andersen et al. [87], we expect to observe tetragonal symmetry until an oxygen content around 6.5 at a temperature of $T = 900$ K, which is not predicted by our model. The minor differences at low oxygen contents between results obtained by 60x60x1 and 120x120x1 simulation cells, indicate that finite size effects in the MC simulations are not significantly accountable for the differences in our models' predictions to experimental observations. In Sec. 4.9, we discussed the difficulty of optimizing models for the $\frac{|a-b|}{2}$ property and how an increasing regularization strength λ worsened the predictions of our model for the training set. However, a larger regularization strength would promote tetragonal symmetry in the disorder limit more strongly and could produce more accurate results. Further studies, beyond the scope of this work, could compare models of different regularization strengths, but still it would be unclear which regularization strength is appropriate to choose. Another reasonable next step would be to consider other ways of incorporating the constraint of tetragonal symmetry into the lattice constant models, which could possibly improve their performance.

Chapter 5

Discussion and outlook

We started by discussing the importance of high-T_c superconductivity and recognizing that its physical mechanisms are not fully understood yet. $YBa_2Cu_3O_{6+x}$ was introduced as the first material to be discovered with a transition temperature T_c above the boiling point of liquid nitrogen [2]. We acknowledged that its superconductivity is related to the content of oxygen atoms, as well as to the ordering of oxygen atoms and vacancies [3, 10, 11, 33–36]. This was the starting point for our motivation to study the effects of oxygen-vacancy ordering on the energetics and lattice constants of configurations of $YBa_2Cu_3O_{6+x}$, considering the whole composition range of $0 \leq x \leq 1$.

In Chap. 2, we provided an overview of some physical properties of $YBa_2Cu_3O_{6+x}$, beginning with its high-T_c superconductivity. We discussed that oxygen doping transforms the antiferromagnetic insulator parent phase $YBa_2Cu_3O_6$ into a metal, with superconductivity emerging at oxygen concentrations above 6.35. We examined its crystal structure, focusing particularly on the plane between the barium atoms that contains copper atoms and the substitutional sites. We compared experimental results for the measurements of lattice constants of $YBa_2Cu_3O_{6+x}$ from five different studies [28, 29, 34, 38, 43]. All experiments showed a tetragonal-orthorhombic phase transition with increasing oxygen content and revealed a consistent trend in the lattice constants: lattice constants a and c decrease with increasing oxygen content, while b decreases. The *combinatorial explosion* illustrated that an approach solely using DFT is not sufficient to study a wide range of $YBa_2Cu_3O_{6+x}$ configurations. We recognized that cluster expansion [23], combined with machine learning methods, overcomes this limitation and introduced the corresponding methods in the next chapter.

In Chap. 3, we presented DFT [12–14] as a method for studying the electronic structure and properties of materials. By combining it with cluster expansion and machine learning methods, we can perform DFT calculations on a smaller subset of structures. The results are then used to build models to predict configurations beyond the training set. All, while maintaining a precision similar to that of the DFT calculations. Nonlinear cluster expansion [44] was discussed as a valuable extension of the standard CE. Various approaches for the selection of structures for the training set, such as choosing lowest energy configurations and random configurations or selecting configurations to reduce the variance of the training set, were outlined. [45, 46, 59, 60]. To obtain temperature dependent properties, we presented Metropolis Monte Carlo sampling [26, 27] as a technique to perform statistical thermodynamic simulations.

In Chap. 4, we presented the results of this work. At first, we demonstrated how CE is applied to study $YBa_2Cu_3O_{6+x}$ and built a small initial model, based on ten structures of $YBa_2Cu_3O_{6.5}$, to compare to the results of a previous CE study by Draxl et al. [63]. In agreement with the original study, we observe that the structure corresponding to the ortho-II phase exhibits the lowest energy and find a similar ordering of the energetic differences of the other structures compared to the ortho-II phase. Only the ordering for two pairs of structures was swapped. Discrepancies for the calculated energetic differences compared to Draxl et al. [63] are likely stemming from the full structure relaxation that was performed in this work, but not in the paper we compare to [63]. Other factors, like improvements in the employed DFT codes over the last decades might also contribute. Regarding the resulting CE model, based on the same clusters as those in the work by Draxl et al. [63], we find that the ECIs have equal signs for both models for all, but one ECI. Overall, our results reproduce the key results of the work by Draxl et al. [63], exhibiting an attractive interaction for next-nearest neighbor oxygen sites with an intermediate copper atom and a repulsive interaction for oxygen sites without an intermediate copper atom.

Next, the workflow for the ground state search of $YBa_2Cu_3O_{6+x}$ was summarized, starting with the *ab initio* calculations for the initial structures. We presented the `dbinterfaces` python module, which was developed as part of this work and serves as an interface to material databases provided by NOMAD [64]. Users can upload their *ab intio* data to the databases and extract relevant properties for the whole training set, in a format that is directly applicable in `CELL`. The easy usage of the module was demonstrated by an example use case. The module is incorporated in the workflow for ground state search, such that the data extracted by the module is used to build CE models, which predict properties of configurations beyond the training set. From these new configurations, some are chosen to be added to a new, extended training set. At first, configurations which were predicted to be lowest in energy and random configurations were chosen, while later in this project a more elaborate approach of structure selection was developed, that will be reviewed in a following paragraph. With the extended training set, a new model was optimized and again configurations were added to the training set, based on their predicted properties. We discussed the criteria of evaluating the convergence of a model, by assessing if it predicts new ground states and by analyzing its cross validation scores.

We presented the results of this iterative optimization process for CE models of the energies of mixing. We compared the performance of models built by utilizing various ML techniques, including ridge regression, LASSO, OMP and nonlinear CE. In each iteration, we analyzed the corresponding models' fit, cross-validation, and, if possible, test errors. We demonstrated a significant improvement in model performance throughout the iterations. We progressed from an initial training set of twelve structures to the complete training set, encompassing *ab initio* results of 100 configurations of $YBa_2Cu_3O_{6+x}$. We discussed the importance of selecting relevant clusters and outlined the various techniques used for the optimization of selected clusters, including LASSO, OMP and a combinatorial search for best subsets. Finally, we built a standard CE model with only 16 features that accurately predicts the distribution of energies of mixing. This model produces error scores that are smaller than or similar to the computational error of the *ab initio* calculations for low-energy configurations. In addition, we presented a nonlinear CE model with smaller fit errors, but noted the difficulty of evaluating the performance of nonlinear models due to the high standard deviation of their errors.

We compared the performance of the optimized standard and nonlinear CE models to reported models based on the interactions considered in the ASYNNNI model [62] and the clusters used in the work by Draxl et al. [63], which we refer to as ASY5NNI model. We note that these models do not correspond to the original ASYNNNI model [62] and the model by Draxl et al. [63], since

we train them on the full set of 100 configurations that was developed in this thesis. To the best of our knowledge, no other CE model of $YBa_2Cu_3O_{6+x}$ was trained on a set of comparable size previously. We demonstrated that both reported models predict a linear behavior, which does not align with the actual distribution of energies of mixing for low-energy states of the training set. In contrast, both of our models predict the accurate distribution. We observed that at least one of the 5th nearest neighbor 2-body clusters is necessary to distinguish close lying low-energy configurations in the training set. Comparing the ECIs of the four considered models, we found that they agree in sign and are of similar orders of magnitude across all models for 1-point and 2-point clusters up to next-nearest neighbors, corresponding to four clusters in total. Three of these ECIs (corresponding to clusters 1,2 and 3 in Fig. 4.1) were found to be significantly larger in magnitude than the ECIs of the other considered clusters (see Fig. D.2). Yet, we emphasize again that many more than four clusters are needed to obtain accurate predictions of the energies, especially in low energy regions.

By analyzing the *ab initio* results for states with the lowest energies of mixing, several possible ground states at 0 K were identified. The exact trajectory of the convex hull was difficult to ascertain due to the computational error of the DFT calculations and proximity of states with similar energies. Without considering error bars, our results reveal a convex hull indicating eleven possible ground states (see Fig. 4.17). As anticipated [63, 67, 73], the ground states for the compositions $YBa_2Cu_3O_6$ and $YBa_2Cu_3O_7$ correspond to the tetragonal and ortho-I phases, while the ground state at oxygen content 6.5 corresponds to the ortho-II phase. All eleven configurations feature continuous Cu-O chains, which are spaced further apart at low oxygen contents but become closer with increasing concentration. We identified the ortho-VIII and ortho-III phases, which are also predicted by an extended version of the ASYNNNI model by Andersen et al. [87] and observed experimentally [73]. The ortho-V phase, although observed experimentally [73], is not predicted to be a ground state by our *ab initio* results. Due to the computational error, it is not clear whether or not it corresponds to a ground state. To determine the minimal number of ground states, we constructed a more conservative convex hull, predicting only configurations as ground state if their energies of mixing, including the error bars, lie below the connection line of two other states. The ortho-II phase, unlike other ortho superstructures (except ortho-I), exhibits three-dimensional ordering in experiment [73], suggesting greater stability. Thus, we enforce that it is included in the convex hull. We identified six ground states, including the tetragonal, ortho-II, ortho-III and ortho-I phase. We acknowledge that the considered super cell sizes of up to nine times as large as the parent lattice, might be inadequate to capture states exhibiting phase separation. Future analyses could benefit from enumerating configurations with even larger super cell sizes. Furthermore, we emphasize the possibility for further *ab initio* calculations on the identified low-energy configurations to minimize the computational errors even further and to gain more insight into the stability of these states. However, given that many of the configurations are energetically very close, it is not guaranteed that more clarity can be gained.

As mentioned previously, we developed an active learning workflow to select structures that reduce the variance of the training set. To select multiple structures, we implemented a workflow, that recalculates the input matrix and candidate correlations each time a new structure is selected, while excluding symmetrically equivalent structures. The selection process is based on the works by Mueller, Ceder [46] and van de Walle [45], as well as the extension to the method incorporated in `CELL` [24]. We used this active learning workflow to select the final 33 structures for our training set. We compared four different selection methods, based on three different population averages of the candidate set [46]. The quality of the training set was shown to improve significantly: We reduced the variance measure, τ, by up to 97.8 %. Notably, many selected structures featured large super cells: 19 of the 33 selected structures had super cells nine times as large as the parent

lattice, which is the maximum dimension for the considered candidate set. Ten structures had super cell dimensions of eight, two had dimensions of seven, and one each had dimensions of six and five. To save computational resources, we recommend prioritizing configurations with smaller super cell sizes if they achieve a similar reduction of τ compared to larger configurations. This has not yet been considered in this work, but is planned to be included for future applications of the workflow.

As a novel approach, we built CE models to predict the lattice constants of $YBa_2Cu_3O_{6+x}$ with increasing oxygen content. Three models are optimized, considering the following lattice constant properties: $\frac{a+b}{2}$, $\frac{|a-b|}{2}$ and c. To promote tetragonal symmetry in the disorder limit, we incorporated a constraint into the modeling of $\frac{|a-b|}{2}$, which is a measure of the orthorhombicity of the configuration. This property was difficult to predict with a high accuracy, and introducing the constraint worsened the predictions for training set structures. However, it is important to include it in order to base the model on physical reasoning. Consequently, we needed to use a large clusters pool to model the property sufficiently well. An even more extensive search for small, yet suitable, clusters pools could improve the results. For some outliers the predictions strongly deviated from the target values, which could be a good starting point for a further analysis of the *ab initio* results. Additionally, building a nonlinear CE model for $\frac{|a-b|}{2}$ could reduce the errors. Since this requires modifications in `CELL` and more expensive Monte Carlo simulations, we did not yet expand to nonlinear features. Nevertheless, this is a reasonable next step beyond the scope of this work. Furthermore, we could try to find other methods of enforcing tetragonal symmetry in the disorder limit, which could improve the models' performance. To summarize, it is planned to refine the modeling of the lattice constants further for future studies.

To predict the lattice constants at finite temperature and to obtain the specific heat for certain compositions, we performed statistical thermodynamics simulations using MC Metropolis sampling. For $YBa_2Cu_3O_{6.71}$ and $YBa_2Cu_3O_{6.5}$, we observed the expected number of peaks (two and one) in their specific heat spectra and predicted the corresponding transition temperatures. The transition temperature for $YBa_2Cu_3O_{6.5}$ agrees remarkably well with experimental observations. An analysis of the lowest energy configurations from the obtained MC trajectories, revealed disordered tetragonal structures at high temperatures. As the temperature decreases, longer and more ordered Cu-O chains form, breaking tetragonal symmetry. At low temperatures, we observed large domains of ordered Cu-O chains. For $YBa_2Cu_3O_{6.71}$, ortho-I and ortho-III ordering are observed, which is consistent with experimental and other theoretical predictions [72, 73]. In $YBa_2Cu_3O_{6.5}$, ortho-I and ortho-II ordering are observed, as expected [73]. The optimized models for lattice constant properties were used to predict the lattice constants at a finite temperature of $T = 900$ K with oxygen contents ranging from 6.0 to 7.0. We compared our results both to the *ab initio* lattice constants for the lowest energy configurations of the training set, as well as to the experimental results discussed in Chap. 2. Our model shows qualitative agreement with the trends in the lattice constants, capturing the transition from (almost) tetragonal to orthorhombic symmetry with increasing oxygen content. Experimental results show a change in slope in the lattice constant curves around an oxygen content of 6.4, which is also reflected in our models' predictions. The experimental data exhibits a greater variability for values of lattice constant c in this region. Similarly, we observe a slight kink in the *ab initio* data in this region, that is reduced in the predictions of the model, such that we also observe a greater variability, which could result from finite temperature effects. Apart from this qualitative successes of our model, it was not able to predict exact tetragonal symmetry at low oxygen contents, indicating the need for further optimization of the lattice constant models.

Beyond the scope of this work, the next planned steps involve performing further statistical ther-

modynamics simulations of $YBa_2Cu_3O_{6+x}$ to build a comprehensive phase diagram covering the entire composition range, based on our computational results. This will be possible due to the mentioned improvements in the scalability of `CELL` regarding MC simulations. Additionally, changing the utilized xc-functional from PBEsol [21, 22] to SCAN [15] or r2SCAN [69] could be considered. As mentioned in the introduction, both SCAN and r2SCAN xc-functionals have demonstrated the ability to accurately capture the antiferromagnetic ground state of $YBa_2Cu_3O_6$ [16, 17]. For the properties studied in this work, however, the PBEsol xc-functional was sufficient, as discussed in Sec. 4.2. Using more complex functionals, like SCAN or r2SCAN, which are expensive and can be numerically unstable, would have been difficult to realize due to the extensive number of calculations required for this work. Nevertheless, additional calculations using these functionals for configurations with small oxygen content could be beneficial, especially if one is interested in studying band structures and the density of states.

In conclusion, we optimized CE models that accurately predict the distribution of energies of mixing for $YBa_2Cu_3O_{6+x}$ configurations, demonstrating the necessity of considering many more than four interactions for precise predictions. We identified at least six ground states at temperatures of 0 K. We predicted the transition temperatures for two compositions of $YBa_2Cu_3O_{6+x}$, observing alignment with experimental results. Additionally, our models trained for lattice constant properties, captured their behavior with increasing oxygen content at finite temperature, showing qualitative agreement with experimental observations. Future steps will involve refining the modeling of lattice constants and creating a comprehensive phase diagram.

Appendix A

The c^2 problem in standard CE

The following discussion is based on the paper by Stroth et al. [44], introducing the nonlinear CE method. Often the discussed topic is referred to as x^2 problem of standard CE, but we use c to stay consistent with our notation. We consider the property $P(\boldsymbol{\sigma}) = c^2$, where c is the concentration of substituents with respect to the substitutional sites. In our case we substitute oxygen atoms, such that $c = \frac{N_O}{N}$ where N_O is the number of oxygen atoms and N the number of substitutional sites. We use the indicator-binary basis where $\gamma_0(\sigma_i) = 1$ and $\gamma_1(\sigma_i) = \sigma_i$ with $\sigma_i \in \{0, 1\}$. $\sigma_i = 1$ if a site is occupied by an oxygen atom and zero if it is vacant. Then c is simply the cluster correlation of the one point cluster:

$$X_1(\boldsymbol{\sigma}) = \frac{1}{\mathcal{M}_1} \sum_{\boldsymbol{\alpha}_{1pt}} \prod_{i=1}^{N_p} \gamma_{\alpha_i}(\sigma_i) = \frac{1}{N} \sum_{i=1}^{N} \sigma_i = \frac{N_O}{N} = c. \tag{A.1}$$

Here we have started with the definition of cluster correlations from Eq. 3.16, considering only the 1-point clusters. We make use of the fact that a one point cluster contains only one point, so the number of points in the cluster is $N_p = 1$. By definition this point is assigned with $\gamma_1 = \sigma_i$. So the product of site basis functions running over all points inside the cluster is only σ_i. We also know that there are as many 1-point clusters as substitutional sites N, all of which are symmetrically equivalent, thus the cluster multiplicity is $\mathcal{M}_1 = N$. Since we use the indicator-binary basis, σ_i is only different from zero if the corresponding site is occupied by an oxygen atom. Therefore, the sum over σ_i for all sites $\sum_{i=1}^{N} \sigma_i = N_O$ results in the number of oxygen atoms. Finally, we obtain the result $X_1(\boldsymbol{\sigma}) = \frac{N_O}{N}$, which is the substituent concentration with respect to the substitutional sites x.

Using this result we can rewrite the property:

$$\begin{aligned} P(\boldsymbol{\sigma}) = c^2 &= X_1(\boldsymbol{\sigma}) X_1(\boldsymbol{\sigma}) \\ &= \left(\frac{1}{N} \sum_{i=1}^{N} \sigma_i \right) \left(\frac{1}{N} \sum_{j=1}^{N} \sigma_j \right) \\ &= \frac{1}{N^2} \left(\sum_{i=1}^{N} \sigma_i^2 + \sum_{i=1}^{N} \sum_{j \neq i} \sigma_i \sigma_j \right). \end{aligned} \tag{A.2}$$

In the second row, we have simply inserted our previous result $X_1(\boldsymbol{\sigma}) = \frac{1}{N}\sum_{i=1}^{N}\sigma_i$. Then we split the sum in a part where both indices match, such that $i = j$ and a part for which $i \neq j$. The resulting term has two parts: One corresponds to the cluster correlation for 1-point clusters divided by the number of substitutional sites. The second term corresponds to the sum of cluster functions for 2-point clusters.

Still using the indicator-binary basis, we know that $\sum_{i=1}^{N}\sigma_i^2 = \sum_{i=1}^{N}\sigma_i$ since $\sigma_i \in \{0,1\}$ and insert $\sum_{i=1}^{N}\sigma_i^2 = \sum_{i=1}^{N}\sigma_i = NX_1(\boldsymbol{\sigma})$ for the first term. Considering the second term, we again split the sum, in a sum over symmetrically inequivalent 2-point clusters and a sum over symmetrically equivalent clusters. We can then use the definition of the cluster correlations again to insert for $\sum_{\boldsymbol{\beta}\in\mathcal{O}(\boldsymbol{\alpha})}\Gamma_{\boldsymbol{\beta}}(\boldsymbol{\sigma}) = \mathcal{M}_2 X_2(\boldsymbol{\sigma})$. In the next step, we recall the relationship between the cluster multiplicity $\mathcal{M}$ and the intensive cluster multiplicity m: $\mathcal{M}_{2,i} = m_{2,i}\frac{Vsc}{V_{pc}} = m_{2,i}\frac{N}{2}$. We used that in our case the ratio between the super cell and parent cell volume is the same as the number of substitutional sites divided by two, since there are two substitutional sites per parent cell and the super cell volume is a whole-number multiple of the parent cell volume. Following these steps, we obtain:

$$\begin{aligned} P(\boldsymbol{\sigma}) &= \frac{1}{N}X_1(\boldsymbol{\sigma}) + \frac{1}{N^2}\sum_{i=1}^{\infty,s.i.}\sum_{\boldsymbol{\beta}\in\mathcal{O}(\boldsymbol{\alpha})}\Gamma_{\boldsymbol{\alpha}}(\boldsymbol{\sigma}) \\ &= \frac{1}{N}X_1(\boldsymbol{\sigma}) + \frac{1}{N^2}\sum_{i=1}^{\infty}\mathcal{M}_{2,i}X_{2,i}(\boldsymbol{\sigma}) \\ &= \frac{1}{N}X_1(\boldsymbol{\sigma}) + \frac{1}{N^2}\frac{N}{2}\sum_{i=1}^{\infty}m_{2,i}X_{2,i}(\boldsymbol{\sigma}) \\ &= \frac{1}{N}X_1(\boldsymbol{\sigma}) + \frac{1}{2N}\sum_{i=1}^{\infty}m_{2,i}X_{2,i}(\boldsymbol{\sigma}). \end{aligned} \tag{A.3}$$

Equation A.3 is an infinite expansion with the expansion coefficients $J_1 = \frac{1}{N}$ for the 1-point cluster and $J_2 = \frac{1}{2N}$ for all 2-point clusters. As J_2 is equally small for every 2-point cluster, there is no finite expression of this expansion that we can consider converged and truncating the sum would result in spurious interactions due to the left out terms. To resolve this problem, we make use of nonlinear CE, as described in Sec. 3.2.3.

Appendix B

Convergence analysis with FHI-aims

For DFT calculations, one has to find a set of parameters that yields converged and therefore trustworthy results. For a set of converged parameters the results should stabilize and not improve significantly when optimizing the parameters further. What changes we classify as significant depends on the convergence target.

To estimate this target, as well as the parameter values to set for the calculations, we perform a detailed convergence analysis on the initial set of structures introduced in Sec. 4.2. We want to predict structures and estimate their stability in the range of $0 \leq x \leq 1$ for $YBa_2Cu_3O_{6+x}$. To do so, we consider the energy of mixing and want to achieve convergence with respect to it. It describes the difference between the total energy of a structure with configuration vector $\boldsymbol{\sigma}$ and a linear interpolation between the energies of the reference structures $E_t(\boldsymbol{\sigma}_0)$ (for $YBa_2Cu_3O_6$) and $E_t(\boldsymbol{\sigma}_1)$ (for $YBa_2Cu_3O_7$), evaluated at c:

$$E_{mix}(\boldsymbol{\sigma}, c) = \frac{E_t(\boldsymbol{\sigma})}{N_{p.l.}} - \left(E_t(\boldsymbol{\sigma}_0) + (c - c_0) \cdot \frac{E_t(\boldsymbol{\sigma}_1) - E_t(\boldsymbol{\sigma}_0)}{c_1 - c_0} \right). \tag{B.1}$$

The fractional concentration c describes the fraction of substitutional sites that is occupied by oxygen atoms. $N_{p.l.}$ is the number of parent lattices contained in the super cell of the considered structure. $E_t(\boldsymbol{\sigma})$ refers to its total energy. The fractional concentration of $YBa_2Cu_3O_6$, defined as c_0, is 0 as none of the substitutional sites are occupied by an oxygen atom, while $c_1 = 0.5$, as half of the substitutional sites are vacant and the other half is occupied by oxygen atoms in $YBa_2Cu_3O_7$. Therefore, for our considerations, the energy of mixing simplifies to:

$$E_{mix}(\boldsymbol{\sigma}, c) = \frac{E_t(\boldsymbol{\sigma})}{N_{p.l.}} - \left(E_t(\boldsymbol{\sigma}_0) + 2c \cdot [E_t(\boldsymbol{\sigma}_1) - E_t(\boldsymbol{\sigma}_0)] \right). \tag{B.2}$$

Our aim is to distinguish the structures energetically and to study the differences of the energies of mixing for the calculated/predicted structures. When considering structures with configuration vectors $\tilde{\boldsymbol{\sigma}}$ and $\boldsymbol{\sigma}$, that have the same fractional concentration of oxygen atoms (as is the case for the ten structures of $YBa_2Cu_3O_{6.5}$), the difference of energy of mixing between these structures simplifies to their energetic difference, normalized by $N_{p.l.}$:

$$E_{mix}(\tilde{\boldsymbol{\sigma}}) - E_{mix}(\boldsymbol{\sigma}) = \frac{E_t(\tilde{\boldsymbol{\sigma}})}{\tilde{N}_{p.l.}} - \frac{E_t(\boldsymbol{\sigma})}{N_{p.l.}}. \tag{B.3}$$

Thus, we set a convergence target with respect to (normalized) energetic differences. For simplicity, we use the following notation: Considering structure A that has a configuration vector $\boldsymbol{\sigma}_A$ we refer to the normalized total energy normalized of structure A as:

$$E_A \coloneqq \frac{E_t(\boldsymbol{\sigma}_A)}{N_{p.l.}}.$$

The same applies to all other considered structures. By definition, the energies of mixing for $YBa_2Cu_3O_6$ and $YBa_2Cu_3O_7$ are zero. In the following, we omit to state that the energy units are normalized $N_{p.l.}$, but we keep this in mind. To estimate the convergence target, we perform *ab initio* calculations with FHI-aims [20]. As explained in Sec. 4.2 we start our initial calculations by using a k-point density of approximately 7.4 Å and a really-tight basis set and obtain as smallest energetic difference 1.7 meV between structures D and A. Consequently, we set this as our convergence target. We are mainly interested in adjusting the following calculation parameters to achieve convergence: the size of the basis set, the k-point density and the threshold for the force acting upon the atoms after performing a geometry optimization.

We first consider basis set size and k-point density. We compare calculations, that all start from the same geometry and are performed without a geometry optimization, such that we can estimate the influence of the parameter values, without considering effects resulting from differing optimized geometries. To study the influence of the k-point density, we compare four different k-grids of increasing size. We adjust the number of k-points in each direction of the reciprocal lattice, according to the dimension of the real space super cell. Considering the parent lattice, lattice constant c is approximately three times as large as lattice constants a and b. Thus, we choose a k-grid with $k_1 = k_2$ k-points in $\boldsymbol{b}_1$ and $\boldsymbol{b}_2$ direction and $k_3 = \frac{k_1}{3}$ k-points in $\boldsymbol{b}_3$ direction, with $\boldsymbol{b}_i, i \in \{1, 2, 3\}$, being the reciprocal lattice vectors. The first chosen k-grid is of size 12x12x4 for the parent lattice. We adjust the values according to the super cell sizes: for example structure A has a super cell that is twice as large in x direction and equally large in y and z direction as the parent lattice. We therefore choose a k-grid of size 6x12x4. The number of k-points in $\boldsymbol{b}_3$ direction is the same for all structures and only increases, when increasing the k-point density. To compare the k-grid sizes easily among different structures, we calculate the k-point density d_k. It represents the number of k-points per reciprocal space volume:

$$d_k^3 = \frac{k_1 \cdot k_2 \cdot k_3}{|(\boldsymbol{b}_1 \times \boldsymbol{b}_2) \cdot \boldsymbol{b}_3|}. \tag{B.4}$$

The cross product of reciprocal space vectors $\boldsymbol{b}_1$ and $\boldsymbol{b}_2$ is orthogonal to both and since, for all super cells, $\boldsymbol{b}_3$ is orthogonal to $\boldsymbol{b}_1$ and $\boldsymbol{b}_2$, we can simplify further:

$$d_k^3 = \frac{k_1 \cdot k_2 \cdot k_3}{|b_1 \cdot b_2 \cdot sin(\gamma) \cdot \boldsymbol{e}_3 \cdot \boldsymbol{b}_3|} = \frac{k_1 \cdot k_2 \cdot k_3}{|b_1 \cdot b_2 \cdot b_3 \cdot sin(\gamma)|} = \frac{k_1}{|b_1|}\frac{k_2}{|b_2|}\frac{k_3}{|b_3|}\frac{1}{|sin(\gamma)|}. \tag{B.5}$$

Here $\boldsymbol{b_i} = b_i \boldsymbol{e}_i$, with $\boldsymbol{e}_i$ the unit vectors for each direction. The angle between $\boldsymbol{b}_1$ and $\boldsymbol{b}_2$ is γ. Most considered super cells are rectangular, such that $\gamma = 90°$, but other super cell shapes are also possible. We want to find k-grids for which $\frac{k_1}{|b_1|} \approx \frac{k_2}{|b_2|} \approx \frac{k_3}{|b_3|} \coloneqq \rho$, thus

$$d_k^3 \approx \frac{\rho^3}{sin(\gamma)}. \tag{B.6}$$

By this definition, the k-point density is given in units $\frac{1}{\text{Å}^{-1}} = \text{Å}$. The chosen k-grid sizes correspond to k-point densities of approximately 7.4 Å, 14.8 Å, 22.2 Å and 29.6 Å. Figure B.1 shows the energetic differences of structures D and A in the left panel and of structures C and A in the

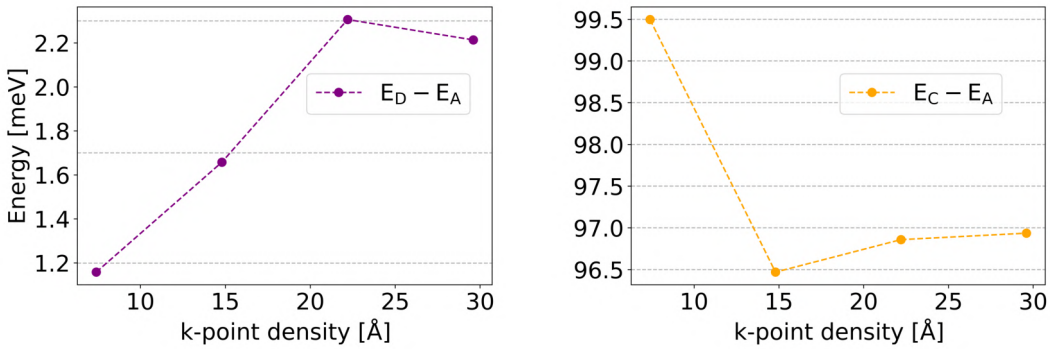

Figure B.1: The differences in energies of mixing of structures D and A (left panel) and of structures C and A (right panel) in relation to the k-point density are shown. Energies are normalized to the parent lattice. All calculations use the tight basis set in FHI-aims [20], without geometry optimization. The grid spacing represents the estimated computational error.

right panel, with respect to the k-point density. All calculations are performed with the tight basis set of FHI-aims [20]. We note that, for all considered k-point densities, the energy difference is positive, meaning that structures D and C always have a higher energy than structure A. This is expected, since structure A corresponds to the ortho-II phase and thus is the expected ground state of $YBa_2Cu_3O_{6.5}$ [63]. When increasing the k-point density over a value of 22.2 Å (corresponding to the third data point in each plot), the results in energy vary only slightly indicating a convergent behavior of the energetic difference with respect to k-point density. Thus, we choose a k-point density of approximately 22.2 Å and estimate the computational error that arises from this choice. For the energetic difference of structures D and A and the chosen k-point density, the energy deviates by 0.6 meV from the result with next lower and by 0.10 meV from the result with next higher k-point density. For the energy difference of structures C and A the corresponding values are 0.4 meV and 0.08 meV. To account for possibly larger differences for other structures, that are not considered in the convergence analysis, we estimate the error a bit larger and assess for the error due to k-point density choice:

$$u_{kgrid} = \pm 0.5 \text{ meV}. \tag{B.7}$$

The grid spacing in Fig. B.1 is chosen to match this value, to visualize that the estimated computational error is larger than differences of results with k-point densities $d_k \geq 22.2$Å.

Next, we study the effects of different basis set sizes. In FHI-aims [20], there are four predefined basis sets: light, intermediate, tight and really-tight. Light settings take into account a minimal set of basis functions, whereas really-tight take into account a large set of basis functions and therefore results in more expensive calculations. We consider again the energetic differences of structures D and A and of C and A. We perform one calculation for each of the basis set sizes and for two different k-point densities. Figure B.2 shows the results for the considered energetic differences. In both cases, it can be seen that a change in the k-point density leads to an energy shift that is approximately the same for all basis set sizes. Therefore, we have to treat the computational errors resulting from the k-grid and basis set size as independent contributions. Comparing really-tight and tight basis set sizes the differences in energy results are: $\Delta_{rt-t}E_{D-A} \approx 0.52$ meV and $\Delta_{rt-t}E_{C-A} \approx 0.93$ meV. Comparing tight and intermediate basis set sizes, the differences in

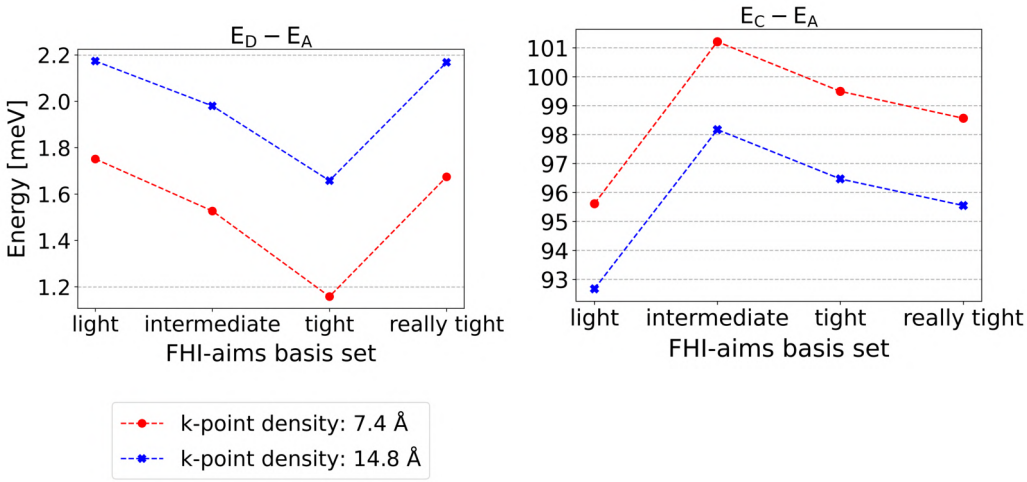

Figure B.2: The differences in energies of mixing of structures D and A (left panel) and of structures C and A (right panel) are shown with respect to the basis set size in FHI-aims [20]. The calculations are performed for two different k-point densities. The grid spacing represents the estimated computational error of ±1.0 meV resulting from calculations with a really-tight basis set.

energy results are: $\Delta_{t-i}E_{D-A} \approx 0.37$ meV and $\Delta_{t-i}E_{C-A} \approx -1.72$ meV. To reduce the overall computational error and to match the desired convergence target of 1.7 meV, we decide for really-tight basis sets. Similar to before, we estimate the error higher, to account for higher differences for structures not considered in the analysis. As computational error, stemming from the basis set size, we estimate:

$$u_{basis} = \pm 1.0 \text{ meV}. \tag{B.8}$$

Lastly, we want to converge the force threshold for the geometry optimization, that follows the BFGS algorithm [88]. The geometry optimization is stopped when the force magnitudes, that act on each atom, are below the chosen threshold. We consider a force threshold in the range of 5 meV/Å to 24 meV/Å. Figure B.3 shows that the results for the energy values, in the considered range of force thresholds, fluctuate by around 0.16 meV. Again, to account for higher differences for structures not considered in the convergence analysis, we estimate a larger error:

$$u_{force} = \pm 0.2 \text{ meV}. \tag{B.9}$$

All structures are relaxed to a force target of at most 24 meV/Å.

So far, we obtain converged results for a k-point density of 22.2 Å combined with a really-tight basis set and a force threshold up 24 meV/Å. A calculation with these parameters is very expensive, especially due to the geometry optimization on a large k-grid and with a large basis set size. To save computational resources, we use the following workflow instead of performing one geometry optimization with the mentioned parameters:

1. Geometry optimization with light basis set and k-point density 7.4 Å;
2. Geometry optimization with tight basis set and k-point density 7.4 Å;
3. Fixed geometry calculation with really-tight basis set and k-point density 22.2 Å;
4. Check that forces acting on atoms are below 24 meV/Å;

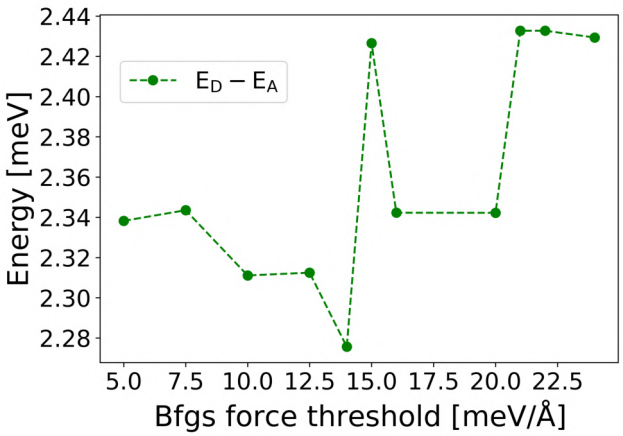

Figure B.3: The difference in energy of mixing of structures D and A with respect to the force threshold in the BFGS geometry optimization [88] FHI-aims [20] is shown. The optimizations are performed using a tight basis set and a k-point density of 22.2 Å. The computational error for using force thresholds up to 24 meV/Å is estimated as $u_{force} = \pm 0.2$ meV.

5. If force threshold not achieved: geometry optimization with really-tight basis set and k-point density 22.2 Å.

To get a better starting point for the expensive calculations, we always start with a geometry optimization using the light basis set and a small k-point density. In order to save computational resources, we then perform the main geometry optimization on the small k-point density of 7.4 Å using a tight basis set. With the largest basis set, we perform a fixed geometry calculation without optimizing the geometry further on an increased k-point density of 22.2 Å. Finally, we check if the desired force target is achieved. If so, we have obtained the converged geometry and property values. If not, we have to perform another geometry optimization, this time using the really-tight basis set and a k-point density of 22.2 Å.

Next, we want to estimate the error that comes in from using this workflow, instead of directly performing the really-tight basis set geometry relaxation in combination with a k-point density of 22.2 Å and the 24 meV/Å threshold. To estimate this, we compare the results shown in Tab. B.1. First, a geometry optimization with a light basis set is performed, followed by a geometry optimization with tight basis set and a k-point density of 7.4 Å. The result is shown in the first row of Tab. B.1. All other calculations, shown in Tab. B.1, use this optimized geometry as starting point, to save computational resources. It is apparent that increasing the k-point density, while maintaining the tight basis set, significantly increases the energetic difference of structures D and A and significantly reduces the energetic difference of structures C and A. This is consistent with our previous results. Additionally, we observe that the difference between a tight fixed geometry calculation (on top of the initial geometry optimization) and a tight geometry optimization yield very similar results for the same k-point density. The energetic difference is further increased for structures D and A and decreased for structures C and A, when switching from a tight to really-tight basis set. This too, is in agreement with the previous results. We observe that a fixed geometry calculation with really-tight basis set and a k-point density of 22.2 Å (on top of

Basis set	Geometry	k-point density [Å]	Energy differences [meV]	
			$\mathbf{E_D}$ - $\mathbf{E_A}$	$\mathbf{E_C}$ - $\mathbf{E_A}$
Tight	optimization	7.4	1.14	99.50
Tight	fixed	22.2	2.32	96.84
Tight	optimization	22.2	2.34	96.88
Really-tight	fixed	22.2	2.84	95.91
Really-tight	optimization	22.2	2.86	95.84

Table B.1: The differences in energies of mixing of structures D and A and of structures C and A calculated with different basis set and k-grid sizes with and without geometry optimization in FHI-aims [20] are shown.

the initial geometry optimization with tight settings and k-point density 7.4 Å) yields almost the same result, as a geometry optimization with really-tight basis set and k-point density 22.2 Å. The absolute differences between the results are 0.02 meV for structures D and A and 0.07 meV for structures C and A. A comparison of the computation times, reveals a significant difference between the workflow procedure and a direct relaxation with converged parameters. Applying the workflow to structure C resulted in a computation time of 1785 minutes, while a relaxation with really-tight basis set, k-point density 22.2 Å and otherwise same computational parameters (number of nodes/available memory) as before took 6904 minutes. For structure C, a direct relaxation with converged parameters therefore takes almost four times as long as the workflow procedure. We can conclude that it is reasonable to follow the proposed workflow, due to the large savings in computational resources. By doing so, we introduce only a small computational error. As before, we estimate a larger error to account for larger differences for structures, beyond the considered ones, and asses:

$$u_{workflow} = \pm 0.1 \text{ meV}. \tag{B.10}$$

We consider all the computational errors to be independent contributions to the total error. To be accurate, the computational error, that is introduced by using the proposed workflow, is a combination of the errors due to k-point density and basis set size. However, since we did not observe a significant correlation between k-point density and basis set size, we choose to treat them as independent contributions. The estimated total computational error of the DFT calculations is calculated by Pythagorean addition as $u_{comp} = \pm\sqrt{u_{basis}^2 + u_{kgrid}^2 + u_{force}^2 + u_{workflow}^2}$. Finally, we obtain:

$$u_{comp} = \pm\sqrt{0.5^2 + 1.0^2 + 0.2^2 + 0.1^2} \text{ meV} = \pm 1.2 \text{ meV}. \tag{B.11}$$

We achieved our convergence target since $u_{comp} = \pm 1.2$ meV $< \pm 1.7$ meV. The suggested workflow is applied to all DFT calculations that are used as input to build the CE models.

Finally, we note that we also performed a convergence analysis for the parameter `sc_accuracy_rho`. It is a convergence criterion for the scf cycle, based on the charge density. Its default values are between 10^{-6} and $10^{-3}\frac{e}{a_0^3}$, with e being the elementary charge and a_0 the Bohr radius. It is calculated according to the number of atoms in the unit cell n_{atoms}, via $10^{-6} \cdot \sqrt{\frac{n_{atoms}}{6}}$ [89]. A smaller value relates to a more converged result. We test, whether a reduction of this parameter significantly improves the results. To do this, we compare our results to results obtained by using

	sc_accuracy_rho		
Energy difference	**default**	**1.4e-6**	**Difference**
E_D - E_A [meV]	1.140	1.134	0.006
E_E - E_A [meV]	90.592	90.588	0.004
Computation time for structures D and A	887 min	1385 min	498 min
Computation time for structures E and A	2148 min	3311 min	1163 min

Table B.2: Comparison of energetic differences for different values of `sc_accuracy_rho` (default values and $1.4 \cdot 10^{-6}$ for all structures). The structures are relaxed with tight settings and a k-point density of 7.4 Å.

the smallest (and therefore most precise) default value of `sc_accuracy_rho`, which is found for $YBa_2Cu_3O_6$, as it contains the smallest number of atoms. The corresponding value is $1.4 \cdot 10^{-6}$ eV/Å. We evaluate the energetic differences of structures D and A and of structures E and A with the default values of `sc_accuracy_rho` and the comparative smallest default value. We consider structure E instead of structure C, as it contains the most atoms and therefore has the highest default value for `sc_accuracy_rho`. The results are shown in Tab. B.2. The calculations are performed using a tight basis set and a k-point density of only 7.4 Å, so the results differ from the previously shown well converged ones. We observe that the differences in results $6 \cdot 10^{-3}$ meV and $4 \cdot 10^{-3}$ meV are much smaller than the estimated computational error of 1.2 meV and therefore insignificant. However, the computation time increases drastically, if smaller values are chosen for `sc_accuracy_rho`. Therefore, we use the default values for all calculations, without reducing them.

Appendix C

Influence of parent lattice on model optimization

There are two equally valid ways to define the parent lattice. In both cases, we start from a primitive structure that we define according to the Wyckoff positions shown in Tab. 4.1. The substitutional sites correspond to Wyckoff position 2f (0, 1/2, 0). We provide two occupational options for this site by assigning either ['X', 'O'] or ['O', 'X'] as symbols for those positions. In the first case, a substitutional site that is vacant is assigned 0 in the configuration vector $\boldsymbol{\sigma}$ and a substitutional site occupied by an oxygen atom is assigned 1. In the second case, the assignment is swapped. The non substitutional sites are always assigned a value of 0 and their occupancy remains fixed. The configuration vectors have to be defined, such that they match with the assignment of the corresponding parent lattice. Theoretically, both parent lattice definitions should result in models with the same error scores. However, we have found that this is not necessarily the case, when using the indicator-binary basis, which we explain in the following.

To investigate the influence of the parent lattice on a model, we write two scripts that only differ in the definition of the parent lattice and the corresponding configuration vectors of the defined structures. We consider $YBa_2Cu_3O_6$, $YBa_2Cu_3O_7$ and the ten structures described in Sec. 4.2 as input. In both scripts, we perform model optimizations and compare the results. We create three cluster pools for a super cell of shape 4x4x1 with respect to the parent lattice:

1. Clusters pool one:
   Contains the 1-point and all 2-point clusters up to a radius of 10.89 Å;
2. Clusters pool two:
   Contains all 2-point clusters up to a radius of 10.89 Å;
3. Clusters pool three:
   Contains the 1-point, all 2-point and three point clusters up to a radius of 10.89 Å and 3-point clusters up to a radius of 7 Å.

At first, we use the indicator-binary basis, where $\gamma_0(\sigma_i) = 1$ and $\gamma_1(\sigma_i) = \sigma_i$ with $\sigma_i \in \{0, 1\}$. We have used this basis for the model optimization of our first model described in Sec. 4.5. We build

Model scores

	Model 1 (plat [X, O])		**Model 2** (plat [O, X])	
Errors [meV]	**Fit**	**CV**	**Fit**	**CV**
RMSE	0.0	41.7	0.0	45.7
MAE	0.0	36.0	0.0	43.0
MaxAE	0.0	73.4	0.0	72.3

Effective Cluster Interactions

Index	**Nr. of points**	**Radius**	**Multiplicity**	**ECI, Model1**	**ECI, Model2**
0	1	0.000	2	390.5	−595.7
1	2	2.723	4	489.2	489.1
2	2	3.851	2	−419.5	−419.5
3	2	3.851	2	77.8	77.8
4	2	5.446	4	−19.8	−19.8
5	2	6.088	8	164.4	164.4
6	2	7.701	2	−100.7	−100.7
7	2	7.701	2	42.2	42.2
8	2	8.169	8	−160.4	−160.4
9	2	8.610	4	36.0	36.0
10	2	8.610	4	16.1	16.1
11	2	10.891	4	−22.6	−22.6

Table C.1: Comparison of the error scores and ECIs of two models where only the assignment in the parent lattice differs: For model 1 empty sites are assigned 0 in the configuration vectors and sites occupied by oxygen are assigned one; for model two the assignment is switched. For both model an indicator-binary basis is used, together with a clusters pool containing the one point cluster and all two point clusters for a 4x4x1 super cell. The model is built using ridge regression with an intercept and hyperparameter $\alpha = 10^{-8}$.

a model by using ridge regression with a small regularization strength $\lambda = 10^{-8}$ and an intercept. We use clusters pool one without optimizing it. The results are shown in Tab. C.1. Both models have similar, but not equal error scores, where Model 1 results in a smaller CV and MAE, but larger MaxAE score. The ECIs are equal, up to the considered precision, for all but two clusters. A large difference of 986.2 meV is observed for the one point cluster. This is not surprising: When using the indicator-binary basis, the 1-point cluster of Model 1 represents a single occupation with an oxygen atom, which results in an increase in energy (positive sign). In case of Model 2, it represent a vacancy, that lowers the energy (negative sign). The ECIs for the 2-point clusters are equal for both models, except for a small deviation of 0.1 meV for the nearest neighbor two point cluster (Index 1). The transformation from the basis of Model 1 to the basis of Model 2 in case of a 2-point cluster, corresponds to:

$$\sigma_i \cdot \sigma_j \to (1 - \sigma_i) \cdot (1 - \sigma_i) = 1 - \sigma_j - \sigma_i + \sigma_i \sigma_j. \tag{C.1}$$

We notice that the prefactor of the $\sigma_i \sigma_j$ term remains unchanged after the transformation. How-

Model scores

	Model 1 (plat [X, O])		**Model 2** (plat [O, X])	
Errors [meV]	**Fit**	**CV**	**Fit**	**CV**
RMSE	16.0	373.8	4.4	135.3
MAE	9.8	267.7	3.6	86.6
MaxAE	45.0	780.9	6.6	397.1

Effective Cluster Interactions

Index	**Nr. of points**	**Radius**	**Multiplicity**	**ECI, Model1**	**ECI, Model2**
0	2	2.723	4	619.6	399.6
1	2	3.851	2	−405.8	−410.8
2	2	3.851	2	126.7	84.7
3	2	5.446	4	54.6	−42.1
4	2	6.088	8	294.9	74.8
5	2	7.701	2	−1.8	−104.0
6	2	7.701	2	105.8	40.6
7	2	8.169	8	−29.9	−250.0
8	2	8.610	4	45.8	19.9
9	2	8.610	4	61.1	−1.7
10	2	10.891	4	−49.1	−32.2

Table C.2: Comparison of the errors and ECIs of two models where only the assignment in the parent lattice differs: For model 1 empty sites are assigned zero, sites occupied by oxygen are assigned one; for model two it is the other way around. For both model an indicator-binary basis is used, together with a clusters pool containing all two point clusters for a 4x4x1 super cell. The models are built using ridge regression with an intercept and hyperparameter $\alpha = 10^{-8}$.

ever, there are additional terms of order zero and order one, that contribute to the intercept and the ECI of the 1-point cluster. This explains the large difference for the ECIs of the 1-point cluster and why we find different intercepts for both models: 0 for Model 1 and 493.1 for Model 2. This likely is the reason why the error scores of the models are affected by the choice of parent lattice. This influence becomes more significant, when we build a model using clusters pool two, that excludes the one point cluster. The results can be seen in Tab. C.2. Now, the zero and first order contributions of the basis transformation introduce spurious interactions to the ECIs of the 2-point clusters, which leads to different results for both models. The error scores, as well as the values of the ECIs, differ significantly in magnitude, as well as in sign for some of the clusters. This also affects processes of model optimization, such as cluster selection.

Both parent lattices are of equal validity and there is no physical reasoning for which to choose. Because of this, it is concerning that the choice of parent lattice influences the results of cluster selection and model optimization. To avoid an influence by the choice of parent lattice, we decide to use the chebyshev basis of Eq.s 3.10 and 3.11 for all subsequent models, unless indicated otherwise. Concerning the basis based on Chebyshev polynomials, the basis transformation between both

parent lattice choices is simply $\sigma_i \rightarrow -\sigma_i$. So the magnitude of the ECIs is the same for models trained on different parent lattices, while the sign is equal in case of clusters with an even and opposed for clusters with an odd number of points. The error scores and intercepts are not affected by the choice of parent lattice, when using the chebyshev basis. Therefore, the chebyshev basis seems to be the more robust choice. We employ it for all energy models, except for the first one which has been optimized before we investigated the influence of parent lattice and basis set choice on model optimization.

Appendix D

ECIs of optimized energy models

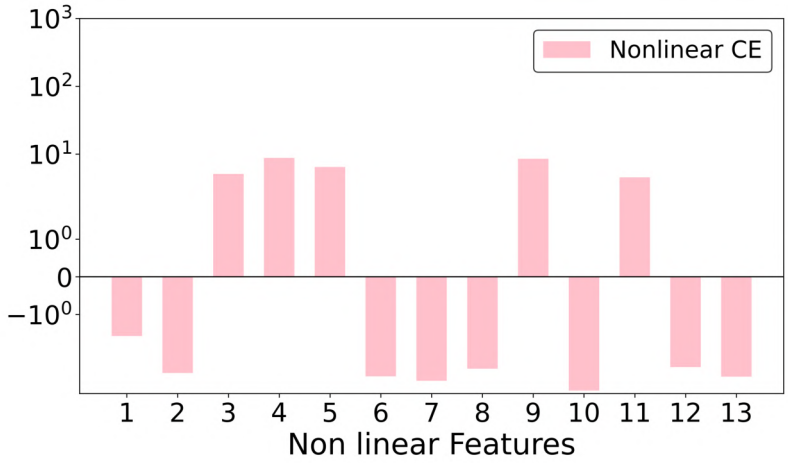

Figure D.1: The ECIs corresponding to the additional nonlinear features of the nonlinear CE model are depicted on a symmetric logarithmic y-axis scale [78]. The features are created by nonlinear combinations of the initial features (clusters).

In Fig. 4.14, we have presented a bar plot, comparing the ECIs of the ASYNNNI, ASY5NNI and of our optimized standard and nonlinear CE models. The nonlinear features of the latter were not yet presented and are summarized here in Fig. D.1. The scale of the y-axis is a symmetric logarithmic one, provided by matplotlib [78]. All clusters, for which their corresponding ECIs are presented in Fig. 4.14, are illustrated in Fig. D.2. As previously, copper atoms are depicted as bronze, oxygen atoms as red and vacancies as empty circles. Technically, the clusters only contain the oxygen atoms or the corresponding sites, but we show the copper atoms and vacancies as well, for a better understanding of the considered interactions. In Tab. D.1, we present the explicit values of the ECIs for each of the four models. Their labeling is matched to Fig. D.2. The first table contains the standard CE features/the clusters and the second table contains the nonlinear features.

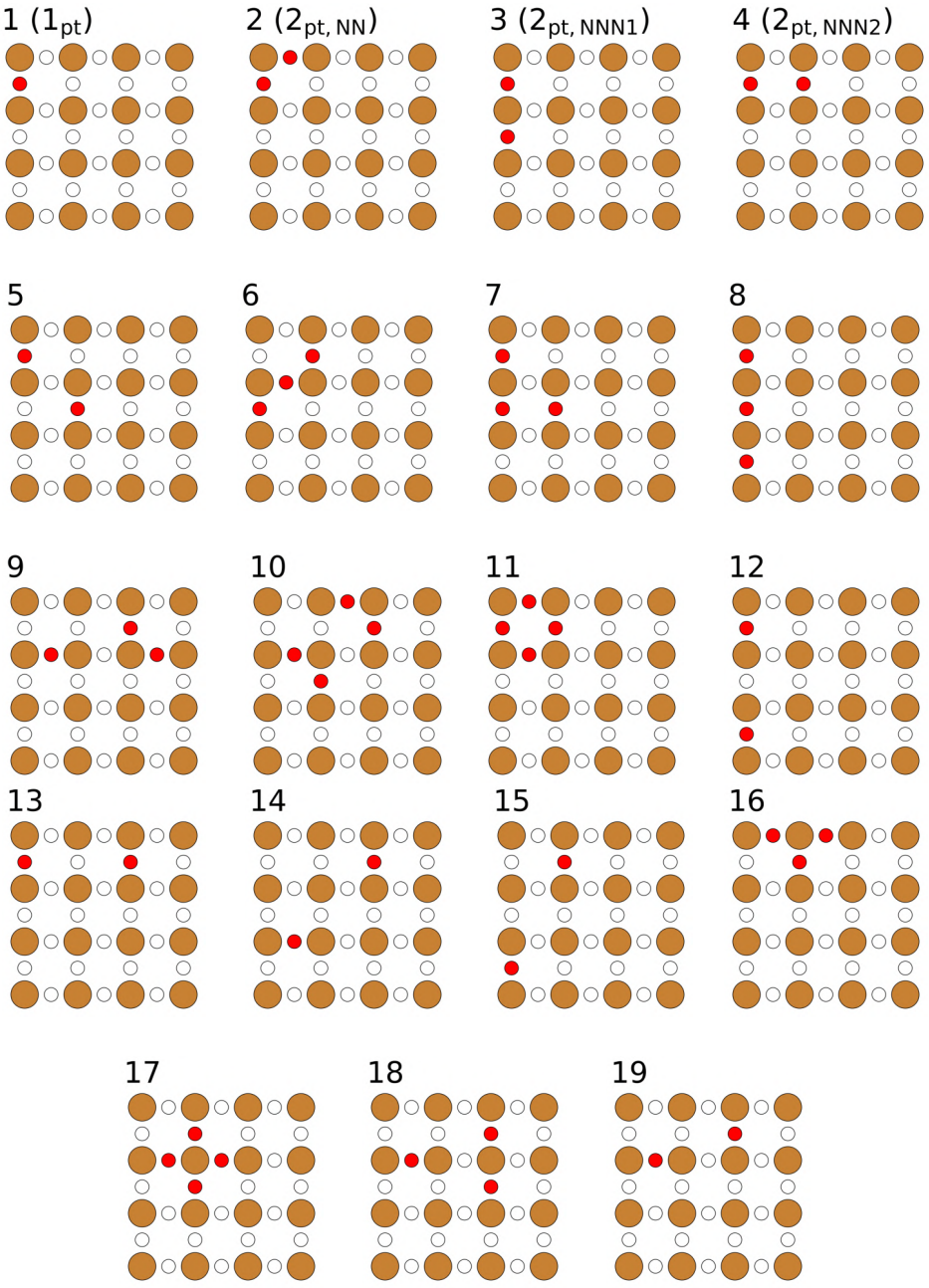

Figure D.2: All clusters used as features for the optimized energy-of-mixing models are visualized. The clusters are labeled according to the numbers assigned to them in Fig. 4.14.

Linear features

	ECI			
Cluster	**ASYNNNI [62]**	**ASY5NNI [63]**	**Standard CE**	**Nonlinear CE**
c1	495.3722	530.1739	537.5716	530.0286
c2	252.6851	219.7508	301.6315	296.4093
c3	−98.9491	−94.6825	−91.5205	−99.8848
c4	17.3115	11.2639	3.5821	13.9385
c5	X	25.4918	39.3362	45.1231
c6	X	X	63.0158	60.4763
c7	X	X	15.5480	23.6542
c8	X	X	−27.3341	−24.3841
c9	X	X	0.0000	25.6763
c10	X	X	8.6109	0.0000
c11	X	X	7.6555	8.4593
c12	X	47.4461	−24.4539	X
c13	X	−14.3388	7.3281	0.0000
c14	X	X	7.4484	12.7724
c15	X	X	10.923	10.9147
c16	X	X	26.190	0.0000
c17	X	X	6.9161	−4.1249
c18	X	X	X	4.5773
c19	X	2.1338	X	X

Nonlinear features

Index	**Feature**	**ECI**
1	$c2 \cdot c9$	−1.5708
2	$c4 \cdot c9$	−4.0521
3	$c8^2$	5.1007
4	$c10 \cdot c13$	8.7680
5	$c11^2$	6.4629
6	$c11 \cdot c14$	−4.5453
7	$c11 \cdot c15$	−5.2636
8	$c17^2$	−3.5051
9	$c3 \cdot c17 \cdot c13$	8.5484
10	$c5 \cdot c10 \cdot c11$	−7.3375
11	$c10^2 \cdot c17$	4.5444
12	$c13^3$	−3.3257
13	$c16 \cdot c17^2$	−4.5871

Table D.1: The used clusters and the corresponding ECIs are shown in the top table for the ASYNNNI and ASY5NNI an for both of our optimized energy models. The bottom tables displays the additional nonlinear features used in the nonlinear CE and the corresponding ECIs.

Abbreviations

Abbreviation	Meaning
DFT	Density functional theory
CE	Cluster expansion
ECI	Effective cluster interaction
RSS	Residual sum of squares
LR	Linear regression
RR	Ridge regression
LASSO	Least absolute shrinkage and selection operator
OMP	Orthogonal matching pursuit
CV	Cross validation
$\mathbf{CV_{LOO}}$	Leave-One-Out cross validation
$\mathbf{CV_{10f}}$	10-fold cross validation
MSE	Mean squared error
RMSE	Root mean squared error
MAE	Mean absolute error
MaxAE	Maximal absolute error
MC sampling	Monte carlo metropolis sampling
ASYNNNI	Asymmetric next-nearest neighbor Ising model
ASY5NNI	Asymmetric 5th nearest neighbor Ising model
IQR	Interquartile Range

Bibliography

[1] J. G. Bednorz and K. A. Müller, "Possible high Tc superconductivity in the Ba-La-Cu-O system," *Zeitschrift für Physik B Condensed Matter*, vol. 64, pp. 189–193, June 1986.

[2] M.-K. Wu, J. R. Ashburn, C. Torng, P.-H. Hor, R. L. Meng, L. Gao, Z. J. Huang, Y. Wang, and a. Chu, "Superconductivity at 93 k in a new mixed-phase Y-Ba-Cu-O compound system at ambient pressure," *Physical review letters*, vol. 58, p. 908, Mar. 1987.

[3] K. Khallouq, "Exploring high-temperature superconductivity in the YBCO system: From theory to experiments," Springer Nature, 2024.

[4] Y. Zhang and X. Xu, "Yttrium barium copper oxide superconducting transition temperature modeling through gaussian process regression," *Computational Materials Science*, vol. 179, p. 109583, June 2020.

[5] K. M. Elsabawy, "Raman spectra, microstructure and superconducting properties of Sb(III)–YBCO composite superconductor," *Physica C: Superconductivity*, vol. 432, pp. 263–269, Nov. 2005.

[6] K. M. Elsabawy, "Superconductivity, structure visualization, mechanical strength promotion and raman spectra of hafnium-doped-123-YBCO synthesized via urea precursor route," *Cryogenics*, vol. 51, pp. 452–459, Aug. 2011.

[7] T. P. Sheahen, *Introduction to High-Temperature Superconductivity*. Springer New York, 2006.

[8] S. P. Kruchinin, "Physics of high-T_c superconductors," *Reviews in Theoretical Science*, vol. 2, pp. 124–145, June 2014.

[9] X. Zhou, W.-S. Lee, M. Imada, N. Trivedi, P. Phillips, H.-Y. Kee, P. Törmä, and M. Eremets, "High-temperature superconductivity," *Nature Reviews Physics*, vol. 3, pp. 462–465, May 2021.

[10] H. Friis Poulsen, N. Hessel Andersen, J. Vrtting Andersen, H. Bohrt, and O. G. Mouritsen, "Relation between superconducting transition temperature and oxygen ordering in YBa2Cu3O6+x," *Nature*, vol. 349, pp. 594–596, Feb. 1991.

[11] H. H. Zhao, J. D. Shen, G. Y. Wang, X. Y. Jia, W. Mi, C. Liu, J. O. Wang, Q. Y. Xu, and Q. Li, "The relationship between orbital hybridization and superconductivity of sm-doped YBa2Cu3O7–δ studied by x-ray spectroscopy," *Low Temperature Physics*, vol. 49, pp. 187–192, Feb. 2023.

[12] P. Hohenberg and W. Kohn, "Inhomogeneous electron gas," *Physical review*, vol. 136, p. B864, Nov. 1964.

[13] W. Kohn and L. J. Sham, "Self-consistent equations including exchange and correlation effects," *Physical review*, vol. 140, no. 4A, p. A1133, 1965.

[14] R. M. Martin, *Electronic Structure: Basic Theory and Practical Methods*. Cambridge University Press, Apr. 2004.

[15] J. Sun, A. Ruzsinszky, and J. P. Perdew, "Strongly constrained and appropriately normed semilocal density functional," *Physical Review Letters*, vol. 115, p. 036402, July 2015.

[16] Y. Zhang, C. Lane, J. Furness, B. Barbiellini, J. Perdew, R. Markiewicz, A. Bansil, and J. Sun, "Competing stripe and magnetic phases in the cuprates from first principles," *Proceedings of the National Academy of Sciences*, vol. 117, pp. 68–72, Dec. 2019.

[17] J. Ning, C. Lane, B. Barbiellini, R. Markiewicz, A. Bansil, A. Ruzsinszky, J. Perdew, and J. Sun, "Comparing first-principles density functionals plus corrections for the lattice dynamics of $YBa_2Cu_3O_6$," *The Journal of Chemical Physics*, vol. 160, Feb. 2024.

[18] S. F. Saipuddin, A. Hashim, M. H. Samat, N. E. Suhaimi, and M. F. M. Taib, "Dft+u calculation in determining structural and electronic properties of YBa2Cu3O7-δ," *AIP Conference Proceedings*, vol. 2368, p. 040001, Nov. 2021.

[19] I. Ramli, S. S. Mohd Tajudin, M. R. Ramadhan, D. P. Sari, S. Shukri, M. I. Mohamed-Ibrahim, B. Kurniawan, and I. Watanabe, "Magnetic properties of $YBa_2Cu_3O_6$ studied by density functional theory calculations," *Materials Science Forum*, vol. 966, pp. 257–262, Aug. 2019.

[20] V. Blum, R. Gehrke, F. Hanke, P. Havu, V. Havu, X. Ren, K. Reuter, and M. Scheffler, "Ab initio molecular simulations with numeric atom-centered orbitals," *Computer Physics Communications*, vol. 180, pp. 2175–2196, Nov. 2009.

[21] J. P. Perdew, A. Ruzsinszky, G. I. Csonka, O. A. Vydrov, G. E. Scuseria, L. A. Constantin, X. Zhou, and K. Burke, "Restoring the density-gradient expansion for exchange in solids and surfaces," *Physical Review Letters*, vol. 100, p. 136406, Apr. 2008.

[22] J. P. Perdew, A. Ruzsinszky, G. I. Csonka, O. A. Vydrov, G. E. Scuseria, L. A. Constantin, X. Zhou, and K. Burke, "Erratum: Restoring the density-gradient expansion for exchange in solids and surfaces [phys. rev. lett.100, 136406 (2008)]," *Physical Review Letters*, vol. 102, p. 039902, Jan. 2009.

[23] J. M. Sanchez, F. Ducastelle, and D. Gratias, "Generalized cluster description of multicomponent systems," *Physica A: Statistical Mechanics and its Applications*, vol. 128, pp. 334–350, Nov. 1984.

[24] S. Rigamonti, M. Troppenz, M. Kuban, A. Hübner, and C. Draxl, "CELL: a python package for cluster expansion with a focus on complex alloys," *npj Computational Materials*, vol. 10, p. 195, Aug. 2024.

[25] F. Pedregosa, V. Michel, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, D. Cournapeau, G. Varoquaux, A. Gramfort, B. Thirion, A. Passos, M. Brucher, M. Perrot, and Duchesnay, "Scikit-learn: Machine learning in python," *Journal of machine Learning research*, vol. 12, pp. 2825–2830, Oct. 2011.

[26] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *The Journal of Chemical Physics*, vol. 21, pp. 1087–1092, June 1953.

[27] W. K. Hastings, "Monte carlo sampling methods using markov chains and their applications," Apr 1970.

[28] A. Pietraszko, M. Wolcyrz, R. Horyń, Z. Bukowski, K. Lukaszewicz, and J. Klamut, "Orthorhombic — tetragonal phase transition and oxygen index of YBa2Cu3O6+δ," *Crystal Research and Technology*, vol. 23, pp. 351–357, Mar. 1988.

[29] R. Cava, B. Batlogg, C. Chen, E. Rietman, S. Zahurak, and D. Werder, "Oxygen stoichiometry, superconductivity and normal-state properties of YBa2Cu3O7–δ," *Nature*, vol. 329, pp. 423–425, Oct. 1987.

[30] J. Bardeen, L. N. Cooper, and J. R. Schrieffer, "Theory of superconductivity," *Phys. Rev.*, vol. 108, pp. 1175–1204, 1957.

[31] C. Min Cheong and S. Kien Chen, "First principle calculation of electronic structures and hole concentration of YBCO family compounds," *Materials Today: Proceedings*, vol. 96, pp. 94–99, Nov. 2024.

[32] O. Gunnarsson and O. Rösch, "Interplay between electron–phonon and coulomb interactions in cuprates," *Journal of Physics: Condensed Matter*, vol. 20, p. 043201, Jan. 2008.

[33] B. Veal, H. You, A. Paulikas, H. Shi, Y. Fang, and J. Downey, "Time-dependent superconducting behavior of oxygen-deficient YBa2Cu3Ox: possible annealing of oxygen vacancies at 300 K," Sept. 1990.

[34] J. Jorgensen, B. Veal, A. Paulikas, L. Nowicki, G. Crabtree, H. Claus, and W. Kwok, "Structural properties of oxygen-deficient YBa2Cu307," Feb. 1990.

[35] J. Jorgensen, S. Pei, P. Lightfoor, H. Shi, A. Paulikas, and B. Veal, "Time-dependent structural phenomena at room temperature in quenched YBa2Cu3O6.41: Local oxygen ordering and superconductivity," *Physica C: Superconductivity*, vol. 167, pp. 571–578, May 1990.

[36] R. Beyers, B. T. Ahn, G. Gorman, V. Lee, S. Parkin, M. Ramirez, K. Roche, J. Vazquez, T. Gür, and R. Huggins, "Oxygen ordering, phase separation and the 60-K and 90-K plateaus in YBa2Cu3Ox," *Nature*, vol. 340, pp. 619–621, Aug. 1989.

[37] W. Farneth, R. Bordia, E. Mccarron, M. Crawford, and R. Flippen, "Influence of oxygen stoichiometry on the structure and superconducting transition temperature of YBa2Cu3Ox," vol. 66, pp. 953–959, Elsevier, June 1988.

[38] P. Gallagher, H. O'Bryan, S. Sunshine, and D. Murphy, "Oxygen stoichiometry in Ba2YCu3Ox," *Materials Research Bulletin*, vol. 22, pp. 995–1006, July 1987.

[39] S. Rayaprol and D. Kuberkar, "On the role of calcium in inducing superconductivity -the study of La-2125 superconductors," July 2005.

[40] A. H. Larsen, J. J. Mortensen, J. Blomqvist, I. E. Castelli, R. Christensen, M. Dułak, J. Friis, M. N. Groves, B. Hammer, C. Hargus, E. D. Hermes, P. C. Jennings, P. B. Jensen, J. Kermode, J. R. Kitchin, E. L. Kolsbjerg, J. Kubal, K. Kaasbjerg, S. Lysgaard, J. B. Maronsson, T. Maxson, T. Olsen, L. Pastewka, A. Peterson, C. Rostgaard, J. Schiøtz, O. Schütt, M. Strange, K. S. Thygesen, T. Vegge, L. Vilhelmsen, M. Walter, Z. Zeng, and K. W. Jacobsen, "The atomic simulation environment—a Python library for working with atoms," *Journal of Physics: Condensed Matter*, vol. 29, p. 273002, June 2017.

[41] J. D. Hunter, "Matplotlib: A 2D graphics environment," *Computing in Science & Engineering*, vol. 9, pp. 90–95, June 2007.

[42] J. Tallon and N. Flower, "Stoichiometric YBa2Cu3O7 is overdoped," *Physica C: Superconductivity*, vol. 204, pp. 237–246, Jan. 1993.

[43] R. Cava, A. Hewat, E. Hewat, B. Batlogg, M. Marezio, K. Rabe, J. Krajewski, W. Peck, and L. Rupp, "Structural anomalies, oxygen ordering and superconductivity in oxygen deficient Ba2YCu3Ox," *Physica C: Superconductivity*, vol. 165, pp. 419–433, Feb. 1990.

[44] A. Stroth, C. Draxl, and S. Rigamonti, "Cluster expansion toward nonlinear modeling and classification," *Physical Review Research*, June 2025.

[45] A. van de Walle and G. Ceder, "Automating first-principles phase diagram calculations," *Journal of Phase Equilibria*, vol. 23, p. 348, Aug. 2002.

[46] T. Mueller and G. Ceder, "Exact expressions for structure selection in cluster expansions," *Physical Review B*, vol. 82, p. 184107, Nov. 2010.

[47] W. Kohn, "Nobel lecture: Electronic structure of matter—wave functions and density functionals," vol. 71, p. 1253, APS, Oct. 1999.

[48] J. P. Perdew and Y. Wang, "Accurate and simple analytic representation of the electron-gas correlation energy," *Physical review B*, vol. 45, no. 23, p. 13244, 1992.

[49] A. Van De Walle, "Multicomponent multisublattice alloys, nonconfigurational entropy and other additions to the alloy theoretic automated toolkit," *Calphad*, vol. 33, pp. 266–278, June 2009.

[50] M. P. Deisenroth, A. A. Faisal, and C. S. Ong, *Mathematics for machine learning*. Cambridge University Press, 2020.

[51] A. E. Hoerl and R. W. Kennard, "Ridge regression: Biased estimation for nonorthogonal problems," *Technometrics*, vol. 12, no. 1, pp. 55–67, 1970.

[52] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, pp. 267–288, Dec. 2018.

[53] G. James, D. Witten, T. Hastie, R. Tibshirani, and J. Taylor, *An Introduction to Statistical Learning with Applications in Python*. Springer Nature Switzerland AG 2023, 2023.

[54] Y. Pati, R. Rezaiifar, and P. Krishnaprasad, "Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition," in *Proceedings of 27th Asilomar Conference on Signals, Systems and Computers*, pp. 40–44 vol.1, 1993.

[55] A. Ang, "Orthogonal matching pursuit algorithm a brief introduction." https://angms.science/doc/RM/OMP.pdf. Accessed: 2025-04-21.

[56] "scikit-learn.org." https://scikit-learn.org/stable/modules/cross_validation.html. Accessed: 2025-04-22.

[57] J. M. Sanchez, "Cluster expansion and the configurational theory of alloys," *Physical Review B*, vol. 81, p. 224202, June 2010.

[58] T. Mueller, "Comment on "cluster expansion and the configurational theory of alloys"," *Physical Review B*, vol. 95, p. 216201, June 2017.

[59] A. Seko, Y. Koyama, and I. Tanaka, "Cluster expansion method for multicomponent systems based on optimal selection of structures for density-functional theory calculations," *Physical Review B—Condensed Matter and Materials Physics*, vol. 80, p. 165122, Oct. 2009.

[60] L. J. Nelson, G. L. W. Hart, F. Zhou, and V. Ozoliņš, "Compressive sensing as a paradigm for building physics models," *Physical Review B*, vol. 87, p. 035125, Jan. 2013.

[61] D. P. Landau and K. Binder, *A Guide to Monte Carlo Simulations in Statistical Physics*. Cambridge University Press, 4 ed., 2014.

[62] D. de Fontaine, L. T. Wille, and S. C. Moss, "Stability analysis of special-point ordering in the basal plane in $YBa_2Cu_3O_{7-\delta}$," *Phys. Rev. B*, vol. 36, pp. 5709–5712, Oct 1987.

[63] C. Ambrosch-Draxl, P. A. Korzhavyi, and B. Johansson, "First-principles study of oxygen ordering in YBa2Cu3O7-x," *Physica C: Superconductivity*, vol. 341, pp. 1997–1998, Nov. 2000.

[64] M. Scheidgen, L. Himanen, A. N. Ladines, D. Sikter, M. Nakhaee, Fekete, T. Chang, A. Golparvar, J. Márquez, S. Brockhauser, S. Brückner, L. M. Ghiringhelli, F. Dietrich, D. Lehmberg, T. Denell, A. Albino, H. Näsström, S. Shabih, F. Dobener, M. Kühbach, R. Mozumder, J. F. Rudzinski, N. Daelman, J. M. Pizarro, M. Kuban, C. Salazar, P. Ondračka, H.-J. Bungartz, and C. Draxl, "NOMAD: A distributed web-based platform for managing materials science research data," *Journal of Open Source Software*, vol. 8, p. 5388, Oct. 2023.

[65] Materials Project, "Ba2Y(CuO2)3." https://next-gen.materialsproject.org/materials/mp-22215, 2025. Accessed: 2025-07-06.

[66] Materials Project, "Ba2YCu3O7." https://next-gen.materialsproject.org/materials/mp-20674, 2025. Accessed: 2025-07-06.

[67] G. Ceder, M. Asta, W. C. Carter, M. Kraitchman, D. de Fontaine, M. E. Mann, and M. Sluiter, "Phase diagram and low-temperature behavior of oxygen ordering in $YBa_2Cu_3O_z$ using ab initio interactions," *Phys. Rev. B*, vol. 41, pp. 8698–8701, May 1990.

[68] J. P. Perdew, K. Burke, and M. Ernzerhof, "Generalized gradient approximation made simple," *Phys. Rev. Lett.*, vol. 77, pp. 3865–3868, Oct. 1996.

[69] J. W. Furness, A. D. Kaplan, J. Ning, J. P. Perdew, and J. Sun, "Accurate and numerically efficient r2SCAN meta-generalized gradient approximation," *The Journal of Physical Chemistry Letters*, vol. 11, pp. 8208–8215, Sept. 2020.

[70] J. Grybos, D. Hohlwein, T. Zeiske, R. Sonntag, F. Kubanek, K. Eichhorn, and T. Wolf, "Atomic displacements in the ortho-II phase of YBa2Cu3O6.50 by synchrotron X-ray diffraction," *Physica C: Superconductivity*, vol. 220, pp. 138–142, Feb. 1994.

[71] FAIRmat, "About fairmat." https://www.fairmat-nfdi.eu/fairmat/about-fairmat/consortium-fairmat, 2025. Accessed: 2025-07-18.

[72] N. Andersen, M. von Zimmermann, T. Frello, M. Käll, D. Mønster, P.-A. Lindgård, J. Madsen, T. Niemöller, H. Poulsen, O. Schmidt, J. Schneider, T. Wolf, P. Dosanjh, R. Liang, and W. Hardy, "Superstructure formation and the structural phase diagram of YBa2Cu3O6+x," *Physica C: Superconductivity*, vol. 317-318, pp. 259–269, May 1999.

[73] M. v. Zimmermann, J. R. Schneider, T. Frello, N. H. Andersen, J. Madsen, M. Käll, H. F. Poulsen, R. Liang, P. Dosanjh, and W. N. Hardy, "Oxygen-ordering superstructures in underdoped YBa2Cu3O6+x studied by hard x-ray diffraction," *Physical Review B*, vol. 68, p. 104515, Sept. 2003.

[74] N. Meinshausen, "Relaxed lasso," *Computational Statistics Data Analysis*, vol. 52, pp. 374–393, Sept. 2007.

[75] D. Mønster, P.-A. Lindgård, and N. H. Andersen, "Simple solution to problems concerning oxygen ordering in $YBa_2Cu_3O_{6+x}$," *Phys. Rev. B*, vol. 60, pp. 110–113, Jul 1999.

[76] L. T. Wille, A. Berera, and D. de Fontaine, "Thermodynamics of oxygen ordering in $YBa_2Cu_3O_z$," *Phys. Rev. Lett.*, vol. 60, pp. 1065–1068, Mar 1988.

[77] P. Sterne and L. Wille, "Oxygen vacancy ordering in YBa2Cu3O7-y," *Physica C: Superconductivity and its Applications*, vol. 162-164, pp. 223–224, Dec. 1989.

[78] MatplotlibDevelopers, "Symmetric logarithmic scale demo." https://matplotlib.org/stable/gallery/scales/symlog_demo.html. Accessed: 2025-08-18.

[79] G. Ceder, M. Asta, and M. de Fontaine, "Computation of the OI-OII-OIII phase diagram and local oxygen configurations for YBa2Cu3Oz with z between 6.5 and 7," *Physica C: Superconductivity*, vol. 177, pp. 106–114, June 1991.

[80] J. Sherman and W. J. Morrison, "Adjustment of an inverse matrix corresponding to a change in one element of a given matrix," vol. 21, pp. 124–127, JSTOR, Mar. 1950.

[81] G. H. Golub and C. F. Van Loan, *Matrix computations*. Johns Hopkins University Press, 4 ed., 2013.

[82] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, A. Vijaykumar, A. P. Bardelli, A. Rothberg, A. Hilboll, A. Kloeckner, A. Scopatz, A. Lee, A. Rokem, C. N. Woods, C. Fulton, C. Masson, C. Häggström, C. Fitzgerald, D. A. Nicholson, D. R. Hagen, D. V. Pasechnik, E. Olivetti, E. Martin, E. Wieser, F. Silva, F. Lenders, F. Wilhelm, G. Young, G. A. Price, G.-L. Ingold, G. E. Allen, G. R. Lee, H. Audren, I. Probst, J. P. Dietrich, J. Silterra, J. T. Webber, J. Slavič, J. Nothman, J. Buchner, J. Kulick, J. L. Schönberger, J. V. de Miranda Cardoso, J. Reimer, J. Harrington, J. L. C. Rodríguez, J. Nunez-Iglesias, J. Kuczynski, K. Tritz, M. Thoma, M. Newville, M. Kümmerer, M. Bolingbroke, M. Tartre, M. Pak, N. J. Smith, N. Nowaczyk, N. Shebanov, O. Pavlyk, P. A. Brodtkorb, P. Lee, R. T. McGibbon, R. Feldbauer, S. Lewis, S. Tygier, S. Sievert, S. Vigna, S. Peterson, S. More, T. Pudlik, T. Oshima, T. J. Pingel, T. P. Robitaille, T. Spura, T. R. Jones, T. Cera, T. Leslie, T. Zito, T. Krauss, U. Upadhyay, Y. O. Halchenko, and Y. Vázquez-Baeza, "Scipy 1.0: fundamental algorithms for scientific computing in python," *Nature Methods*, vol. 17, pp. 261–272, Feb. 2020.

[83] C. L. Lawson and R. J. Hanson, *Solving Least Squares Problems*. Society for industrial and Applied Mathematics, 1987.

[84] J. Nocedal, *Numerical optimization (springer series in operations research)*. Springer, 2006.

[85] M. S. S. Challa, D. P. Landau, and K. Binder, "Finite-size effects at temperature-driven first-order transitions," *Phys. Rev. B*, vol. 34, pp. 1841–1852, Aug 1986.

[86] M. Troppenz, S. Rigamonti, J. O. Sofo, and C. Draxl, "Partial order-disorder transition driving closure of band gap: Example of thermoelectric clathrates," *Physical Review Letters*, vol. 130, p. 166402, Apr. 2023.

[87] N. Andersen, B. Lebech, and H. Poulsen, "The structural phase diagram and oxygen equilibrium partial pressure of YBa2Cu3O6+x studied by neutron powder diffraction and gas volumetry," *Physica C: Superconductivity*, vol. 172, no. 1, pp. 31–42, 1990.

[88] R. Fletcher, *Practical methods of optimization*. John Wiley & Sons, May 2000.

[89] "All-electron electronic structure theory with numeric atom-centered basis functions, a users' guide." https://fhi-aims.org/uploads/documents/FHI-aims.221103_1.pdf. Accessed: 2025-04-07.

Acknowledgments

I thank Prof. Dr. Claudia Draxl for introducing me to this topic and for her belief in my ability to explore it. I greatly appreciate her valuable insights of previous studies of $YBa_2Cu_3O_{6+x}$ and her support in discussing and reviewing my progress. I benefited greatly from her experience and impressive overview of the scientific landscape.

I am deeply grateful for the continuous support of Dr. Santiago Rigamonti, who has played a crucial role in my development in academic writing and the overall quality of my work. I value the numerous discussions we had and the time he dedicated to addressing the state of my work, sharing his thoughts on further improvements, and providing his deep understanding of cluster expansion and machine learning methods. I have learned a lot during our collaboration, and this thesis would not be what it is without the guidance and support of my supervisors.

Additionally, I would like to thank the entire SOL group for always creating a friendly and supportive environment that encourages both learning and scientific curiosity. I appreciate the time and effort people took to assist me, whenever I encountered challenges, may it be my office door not opening, a code not compiling correctly, engaging in discussions about current research or improving the quality and scalability of codes to make my project possible.

Lastly, I also want to thank my friends and family who supported me during stressful times, ensuring that I have the capacity to complete this project. I also want to appreciate my service dog, who may not care about written acknowledgments, but is a fantastic support in everything I encounter.

Selbstständigkeitserklärung

Ich erkläre hiermit, dass ich die vorliegende Arbeit selbstständig verfasst und noch nicht für andere Prüfungen eingereicht habe. Sämtliche Quellen einschließlich Internetquellen, die unverändert oder abgewandelt wiedergegeben werden, insbesondere Quellen für Texte, Grafiken, Tabellen, Bilder sowie die Nutzung von generativer Künstlicher Intelligenz für die Erstellung von Texten und Abbildungen, sind als solche kenntlich gemacht. Mir ist bekannt, dass bei Verstößen gegen diese Grundsätze ein Verfahren wegen Täuschungsversuchs bzw. Täuschung eingeleitet wird.

Berlin, 01.09.2025
Ort, Datum

Noah Alexy Dasch