Developments in the Theory of X-ray Absorption and Compton Scattering.

Egor Klevak

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2016

Reading Committee:

John Rehr, Chair

Gerald D. Seidler

Anton Andreev

Program Authorized to Offer Degree:
Physics

University of Washington

Abstract

Developments in the Theory of X-ray Absorption and Compton Scattering.

Egor Klevak

Chair of the Supervisory Committee:
Professor John Rehr
Physics Department

X-ray spectroscopy is a widely used technique in experimental and theoretical physics for studying electronic and chemical properties of materials. Theoretical approaches for studying x-ray absorption (XAS) and x-ray photoemision (XPS) properties are described by single particle Green's function theory. In strongly correlated systems, such as transition metal oxides, it is necessary to go beyond single particle treatment in order to reasonably describe peaks in XAS spectra. In this thesis I present a simplified, semi-empirical model which describes charge-transfer excitations in XAS, thus extending the formulations of Lee, Gunarson, and Hedin[61]. I model the XAS by a localized three-state system coupled to a photoelectron, and implemented using `FEFF` real space Green's function *ab initio* code to include solid state effects and intrinsic losses.

In the second part, I study valence electron ground state properties as a function of temperature and density for Be, Li and Si. I study thermal disorder and thermal expansion effects on high momentum components of the Compton profile (CP). To calculate the CP, I use the real space Green's function (RSGF) approach coupled with molecular dynamic simulations to create a representation of a disordered system. Thermal disorder has the effect of smearing the high momentum umklapp peak in the case of Be and Li. I find good agreement with experimental results and, to a good degree of precision I predict changes in the CP of polycrystalline Si, which undergoes a solid to liquid phase transition.

In the last part of this work, I study valence electron ground state properties of Be as a function of temperature and density in the warm dense matter (WDM) regime. I compare my thermally disordered calculations of electron momentum density (EMD) with perfect crystal calculations, using effective temperature to quantitatively describe the differences. I conclude that solid state effects are important in the WDM regime and that simplified models of degenerate Fermi gas might overestimate the temperature of the system.

TABLE OF CONTENTS

LIST OF FIGURES

GLOSSARY

TMO: Transition Metal Oxide

XAS: X-ray Absorption Spectroscopy

XPS: X-ray Photoemission Spectroscopy

LDA: Local Density Approximation

CP: Compton Profile

NIXS: Non inelastic x-ray scattering

MD: Molecular Dynamics

WDM: Warm Dense Matter

CT: Charge Transfer

EMD: Electron momentum density

BO: Born-Oppenheimer

MD: Molecular Dynamics

NRIXS: Non-resonant inelastic x-ray scattering

DFT: Density-functional theory

ACKNOWLEDGMENTS

Firstly, I would like to thank the Department of Physics at the University of Washington, which has been like home for me for the last 6 years. I enjoy and value the friendships that I made with my fellow students here: Spencer Williams, Shauna Story, Boris Dzyubenko, Grant Aivazian and others.

I would also like to thank the University of Washington Yacht Club, Corinthian Yacht Club, and the Seattle Canoe & Kayak Club. It would have been miserable and impossible to work so hard without my hobbies of sailing and kayaking. I am appreciative to WYC for welcoming me as a member and teaching me to sail, and then for allowing me teach sailing and, most importantly, for teaching me how to be friendly, social and open. I would also like to thank CYC. It was a great joy to work for them on evenings and to enjoy the most beautiful sunsets around Seattle at Leshi and Shilshole. My life is happier and more balanced since becoming a part of Seattle's sailing community.

I would like to thank my mentor, Jerry Seidler, for fruitful discussions, inspiration and support. I would like to thank my adviser, John Rehr, for support and guidance in the 5 years since I joined his group. I am also grateful to Joshua Kas and Fernando Villa, who worked closely with me, mentoring and always having time to discuss ideas and questions.

Lastly I would like thank my future wife Rebecca, for support, cheerfulness and encouragement.

DEDICATION

To my father Stephan Terentyevich Klevak

Chapter 1

INTRODUCTION

1.1 Background

The wavelength of x-rays (0.01 to 10 nm) makes them an excellent tool in the study of the properties of materials such as crystal structure, chemical composition and physical and electronic properties. The size of the wavelength is in the order of characteristic length scale distance between the atoms. It carries enough energy to ionize atoms, and tuning the incident energy of the x-ray source makes it possible to study the energy levels of the system, as well as the crystal and chemical structure. In order to understand properties probed by x-ray techniques, one needs to understand the measured x-ray spectra. Single particle theories and beyond, as well as computational tools, were developed to quantitatively describe x-ray spectra. The widely used contemporary approaches for calculations of properties are *ab initio* calculations, meaning that the calculations use assumptions for certain initial parameters: atomic numbers and crystal structure of the materials are assumed, as the starting point.

In this dissertation I am interested in the study of properties of materials which can be probed using x-ray techniques. To solve quantum mechanical equations, Green's function techniques are widely used and therefore Green's functions can be reciprocal lattice space based (momentum space) or real space based. The real space Green's function approach is the technique of choice in *ab initio* package `FEFF`[83]. It has built-in self consistency potentials and quasiparticle effects and it is widely used in the x-ray community. There are many properties which have been successfully studied using `FEFF`, including x-ray absorption (XAS), x-ray absorption fine structure (XAFS), extended XAFS (EXAFS), near edge XAFS, x-ray absorption near edge structure (XANES), absorption fine structure (EXAFS), spin polarized XAS and XAFS and Compton Profile (CP), among others.

The challenge associated with this approach, however, is that properties calculated in `FEFF` are within single particle approximation, and therefore sometime fail in the cases of strongly correlated systems where many body effects play an important role.

The second part of the thesis is a study of thermal effects on valence electron ground state, which can be probed using non-resonant inelastic x-ray scattering (NRIXS). NRIXS is directly related to dynamic structure factor and Compton profile (CP). Double differential cross section is measured in experiment and it is is directly related to dynamic structure factor. These properties can be probed for a wide range of states of the matter and can give direct information about the density and temperature of the material, if the equation of state is available. A wide range of parameters, from room temperature to stellar conditions, with partially ionized and degenerate matter, constitute the regime called warm dense matter. The WDM regime is achievable in laboratory settings and is interesting for the study of stellar conditions. Because the matter is only partially ionized, solid state effects play a major role in the ground state properties of the valence electrons, which in turn contain information about the state parameters. Plasma models of free electrons in WDM regime have been used but they are too rough an approximation.

1.2 Goals of Dissertation

1.2.1 Beyond single particle approximation, calculation of XAS of strongly correlated materials

X-ray absorption (XAS) and x-ray photoemission spectroscopies (XPS) are used in the study of electronic and geometric properties of materials. There is strong evidence that transition metal oxides (TMOs) have a second channel for electrons to scatter. Specifically, in the XPS of NiO, CuO, MnO and FeO a second peak within $\sim$ 5 eV of the main peak is observed. XAS and XPS describe the same scattering process and are directly related. Within the single particle picture, XAS of TMOs are missing a key feature which is clearly observed in the XPS experiment. The goal of the first part of this dissertation is to go beyond single

particle approximation in XAS calculations of strongly correlated materials. In chapter 4, I present a method in which a tight binding model is used, originally discussed by Lee, Gunnarsson, Hedin (LGH)[61] to improve single particle approximation obtained using the RSGF approach to x-ray absorption spectra of TMOs.

1.2.2 Study of thermal effects on CP

In the second part of the thesis, I study temperature and density effects on the valence electron ground state of Be, Li and Si for a wide range of temperatures and densities. For Be, Li and Si I study thermal effects on valence CP at temperatures near the materials' melting points. In the WDM regime, I study the effects of temperature disorder on Be valence electron CP and valence electron momentum density in WDM regime. To study valence electron properties, non resonant inelastic x-ray scattering (NRIXS) is commonly used. NRIXS is directly related to dynamic structure factor and CP. Direct measurement in the experiment is double differential cross section. In the previous work by Mattern et al.[64–66], an *ab initio* method for calculating CP usring RSGF approach was developed. Mattern et al. calculated the CP of Be to a great degree of precision and compared with recent experiment. Using the implemented approach, electronic temperature (Fermi function) effects on the CP in the WDM regime was studied, keeping the lattice of the system in perfect crystal configuration. Mattern et al. showed that there is a strong correlation between temperature dependence and density dependence of CP. By including thermal expansion and ionic disorder in my calculations of CP, I go beyond the previous work of Mattern et al..

At room temperature and close to the melting point, I compared results of *ab initio* calculations (MD+RSGF) to the experiments. I followed the following goals:

- Study thermal effects (thermal disorder and electronic temperature) on high components of CP, and

- effect of thermal disorder on directional anisotropy of CP.

In the WDM regime, I sought to:

- Establish th role of ionic and electronic disorder on valence electron ground state: In laser shocked experiments, the temperature of electrons and ions might not be in equilibrium, and so one needs to understand the interplay between two effects on the valence electron ground state, and

- develop a method for extracting state parameters from CP experimental data.

These questions are of special importance because the regime of WDM can be achieved in ignition confinement fusion experiments[62] and is present in astrophysical studies of stellar conditions.

In chapter 2, I present key theoretical results used in RSGF approach implemented in *ab initio* code `FEFF`. Chapter 2 includes a description of self consistency loop and real space Green's function approaches for calculations of density of states, XAS, and temperature dependent CP.

In chapter 3, I present an overview of the theory which is used in molecular dynamics (MD) simulations, along with details of MD simulations using *ab initio* MD code `VASP`. I also present results of MD calculations: energy , free energy and temperature as a function of simulated time and pair distribution functions.

In chapter 4, I present a study of TMOs, using the tight binding model originally discussed by Lee, Gunnarsson, Hedin (LGH)[61] to improve single particle approximation results. Parameters for the (LGH) are obtained from analysis of experimental XPS.

In chapter 5, I present a study of ground state properties of the valence electrons of Be, Li and Si. In particular, I study temperature effects on the valence electron ground state, which can be probed using Compton profile measurement. I am interested in the effects of thermal disorder on the CP and the momentum density of valence electrons, as well as the effect of high momentum components of the CP. Thermal disorder was simulated using molecular dynamics (MD) simulation package. I compare the results of these studies to

the experimental results available. The project emerged from a much more ambitious task: study of matter in the warm dense regime.

In chapter 6, I present calculations of thermally disordered Compton profile (CP) and valence electron momentum density (EMD) of Be in Warm Dense Matter regime (WDM). I examine parameter space available to *ab initio* MD code `VASP` and RSGF code `FEFF`. I compare results of CP and EMD calculations for thermally disordered MD simulated structure with perfect crystal calculations by comparing model temperature T^* obtained by fitting Fermi gas model to EMD of the two calculations. I discuss possibility of extending Fermi gas model to the WDM regime for inferring parameters of state of the matter.

In chapter 7, I summarizing results and objectives of the thesis and conclude with discussion future directions of the work.

1.3 Conventions and Notation

Throughout this work I use Hartree atomic units $e = \hbar = m = 1$ unless otherwise specified. Energies are measured in Hartree (equal to 2 Rydberg or twice the binding energy of the 1s electron in hydrogen atom) or electron volts, where the conversion factor is 1 hartree = 27.2 eV. Distances are measured in bohr equal to average radius of 1s electron in hydrogen atom equal to 0.523 $\mathring{A}$

Chapter 2

X-RAY SPECTRA CALCULATIONS USING *AB INITIO* CODE FEFF

2.1 Absorption coefficient

Absorption coefficient $\mu(x)$ describes how much intensity of an incoming electromagnetic beam is attenuated per unit length of an absorbing substance as a function of electromagnetic energy $\hbar\omega$. It is given by:

$$\mu(E) = -\frac{d\,ln(I)}{dx}, \tag{2.1}$$

where I is the beam intensity and x is the thickness of the sample. Absorption coefficient greatly depends on the energies at which it is measured and depends on both absorption and scattering of incident radiation caused by several mechanisms:

- Rayleigh or Thomson scattering,
- Compton scattering,
- Photoelectron absorption, and
- Pair production (electron positron production in the fields of nucleus and electron).

The schematic representation of the electromagnetic absorption is presented in Fig. (2.1) for iron with the characteristic contributions with relative intensities as a function of momenta on the log-log scale. Dominant contribution for energies up to $\sim$ 500 keV is photoelectric absorption, labeled "PE", with Rayleigh having only an exponentially smaller contribution, labeled "R". In photoelectric absorption, the photon is absorbed by the atom ejecting an

electron from outer shells, ionizing the atom. Ionized atoms return to the neutral state by emitting lower energy photons, which are generally absorbed.

Compton scattering makes a major contribution to the absorption coefficient $\mu(E)$ at energies above 5000 keV and up to 12000 keV, where the positron electron production has importance labeled "PP". In Compton scattering, the incident x-ray photon is deflected from original direction by an interaction with an electron. Scattered x-ray, transfers some of the energy and momentum to the target electron, therefore the energy of the photon is less than the original one. The Compton process is called incoherent scattering, because the change in the direction and energy of the photon is not always orderly or consistent.

Thomson scattering, also known as Rayleigh scattering, is an elastic or coherent scattering process in which the energy of the photon stays constant while the direction is changed. It makes only a minor contribution to the absorption process. At energies around 1.02 MeV, pair production can occur in which an electron and positron pair is created. This process is dominant at the energies around 10 MeV. A positron is a short lived particle which annihilates with an electron, to produce two photons. It is an important attenuation mechanism in the case of high energy x-ray beam and heavy nucleus samples. In this thesis I am interested in an x-ray regime which corresponds roughly to energies up to 10000 keV.

In the Fig. (fig. 2.2) I present an schematic shape of absorption coefficient $\mu(E)$ for the region of energies which correspond to binding energy of the first (n = 1) and second shell (n = 2) (up to 10000 keV) of the nucleus, where n corresponds to the principal quantum number. The experimental data show three main features:

- absorption coefficient is decreasing with the energy which corresponds to the general behavior of photoelectron absorption coefficient;

- sharp peaks at certain energies, called edges, resemble step functions show increase in the absorption corresponding to the binding energies of electron in the atom; and

- the absorption pattern exhibit oscillation above the edge energy.

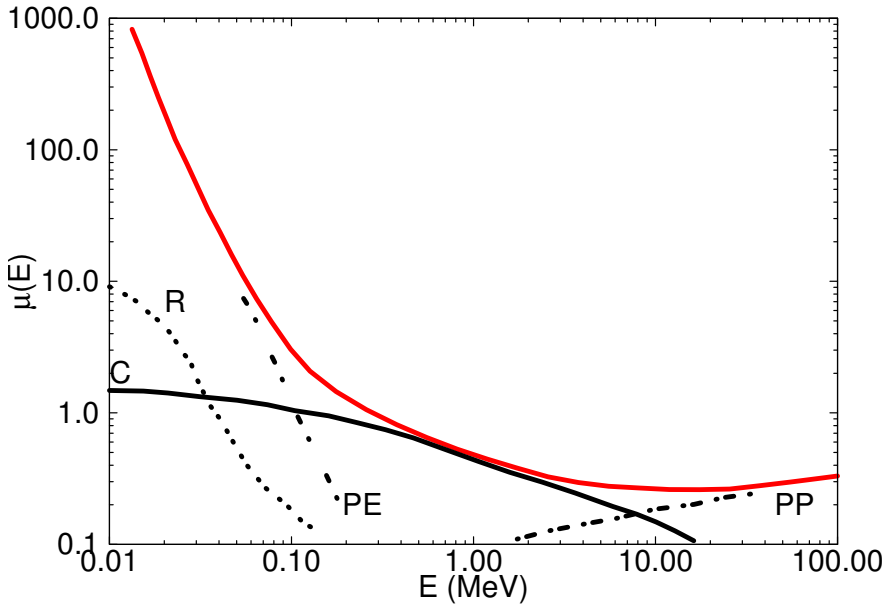

Figure 2.1: Absorption coefficient $\mu(E)$ as a function of incident photon energy.

Sharp peaks correspond to the quantum mechanical effect of absorption of the x-ray by the atom and excitation of an inner core shell electron to the continuum states. The probability of such absorption process is described by Fermi's golden rule. The positions of the absorption peaks are unique to the given absorption atom. The third feature, oscillation of the absorption coefficient is called x-ray absorption fine structure. It contains the information about the inter atomic distances and coordinations number of the absorbing atoms. In the inset, I present a schematic representation of the attenuation of incident electromagnetic beam as a function of thickness of the slab.

2.2 One-electron Fermi's golden rule approximation

The dominant mechanism for x-ray absorption in the region of hard x-rays ($\sim$ 10 keV) is photoelectron excitation (see fig. 2.1). Contributions from Compton scattering is negligible for the region of hard x-ray energies for thin samples and is dominant at higher energies. Only processes which are first order in the electromagnetic field are important, due to the

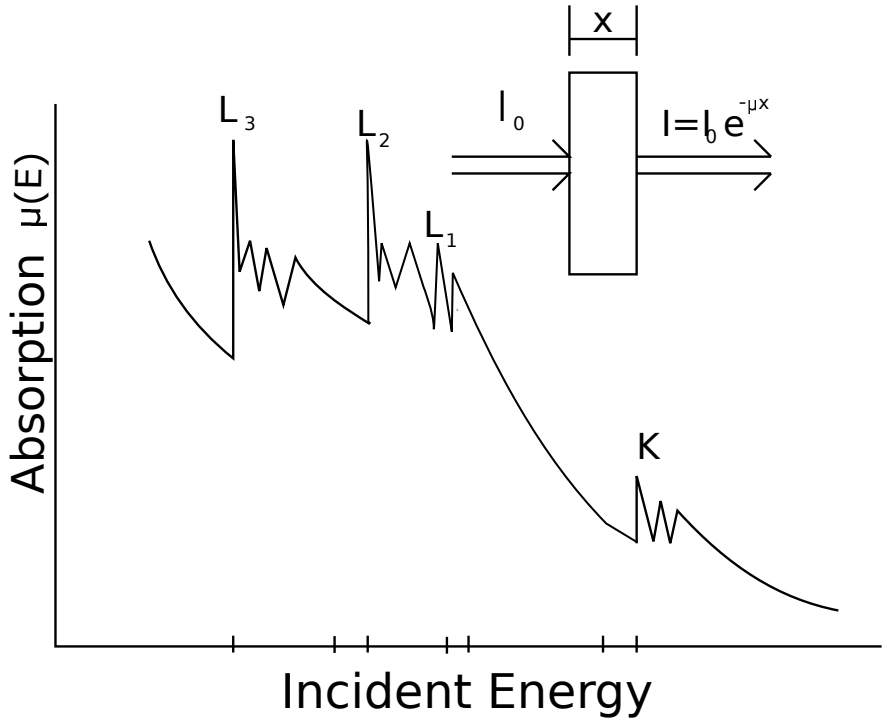

Figure 2.2: Absorption coefficient $\mu(E)$ as a function of incident photon energy. K, L_1,L_2 and L_3 corresponds to n = 1 and n = 2 shells Inset: X-ray absorption in a slab of thickness x, incident intensity of the beam is I_0, in each unit of length $d\,x$ the intensity is attenuated by $d\,I$.

weakness of the electromagnetic field. In one-electron approximation, Fermi's golden rule is given by[68]:

$$\mu \propto \sum_f |\langle \psi_f | \mathbf{p} \cdot \mathbf{A}(\mathbf{r}) | \psi_i \rangle|^2 \delta(E_f - E_i - \omega), \tag{2.2}$$

where ψ_i and ψ_f refer respectively to initial and final eigenstate of effective one electron Hamiltonian H and H′ with energies E_i and E_f respectively. Calculations of ψ_i and ψ_f are calculated using self consistent potentials. In the above equation $\mathbf{p}$ is a momentum operator and $\mathbf{A}(\mathbf{r})$ is a vector potential of the incident electromagnetic field, which can be described using classical polarization. The Vector potential can be approximated for a monochromatic

field of plane wave:

$$\mathbf{A}(\mathbf{r},t) = \hat{\varepsilon} A_0 e^{i\mathbf{k}\cdot\mathbf{r}}, \tag{2.3}$$

where $\hat{\varepsilon}$ is the polarization direction and $\hat{\varepsilon} \perp \hat{\mathbf{k}}$, $\hat{\mathbf{k}}$ is the wave vector, and ω is incident energy of x-ray beam.

For deep-core excitations, spatial dependence of the electromagnetic field can be neglected (vector potential $\mathbf{A}$), and the dipole approximation is used $e^{i\mathbf{k}\cdot\mathbf{r}} \approx 1$ and the momentum form can be replaced by the position form using commutator $[H, \mathbf{r}] = \mathbf{p}/m$.

$$\hat{\varepsilon} \cdot \bigtriangledown \longleftrightarrow \frac{\hbar\omega}{m} \hat{\varepsilon} \cdot \mathbf{r} \tag{2.4}$$

Quadrupole corrections are on the order of $(Z\alpha)^2$, where $\alpha \approx 1/137$ is a fine structure constant and Z is atomic charge seen by the core electron.
In dipole approximation, absorption coefficient is given by equation:

$$\mu \propto \sum_f |\langle \psi_f | \hat{\varepsilon} \cdot \mathbf{r} | \psi_i \rangle|^2 \delta(E_f - E_i - \omega), \tag{2.5}$$

which can be expressed in Green's function using the spectral representation:

$$\sum_f |\psi_f\rangle\langle\psi_f| \delta(E - E_f) = (-1/\pi)\,\mathrm{Im}\, G(E) \tag{2.6}$$

$$\mu(E) \propto \sum_f \langle \psi_i | \hat{\varepsilon} \cdot \mathbf{r} | \psi_f \rangle \langle \psi_f | \hat{\varepsilon} \cdot \mathbf{r} | \psi_i \rangle \delta(E_f - E) = -\frac{2}{\pi}\,\mathrm{Im}\,\langle \psi_i | \hat{\epsilon} \cdot r\, G(\mathbf{r}', \mathbf{r}, E)\, \hat{\epsilon} \cdot r | i \rangle, \tag{2.7}$$

where $G(\mathbf{r}', \mathbf{r}, E)$ is a Fourier transformation of $G(\mathbf{r}', \mathbf{r}; t, t')$ with respect to time. The time dependent Green's function gives relative probability of an electron to propagate from $\mathbf{r}$ at time t to $\mathbf{r}'$ at time t'.

2.3 Real space Green's Functions calculation `FEFF`

The RSGF approach implemented in `FEFF` code. I used the `FEFF` x-ray package to obtain key results of the projects discussed in chapters 4 and 5. For more details one can refer to many prior works [6, 69] and published papers, as well as [83] a user guide.[84]

In the FEFF code, single particle Green's function is used in calculations of absorption coefficient, assuming that the single particle solution to the effective Hamiltonian gives a good description to the many-body effects. In the case of a strongly correlated system such as transition metal oxides (TMO), it is missing a peak which corresponds to the charge transfer state. In chapter 4 we present an *ab initio* approach which extends single particle approximation of x-ray absorption to the case of TMOS.

2.3.1 Self consistent field (SCF) loop

Once the crystal structure of the desired material is loaded from the input file in the initialization step of SCF loop, Dirac-Fock (relativistic generalization of Hartree-Fock) atomic code[7, 105] is used to obtain free atomic densities for each atom (see fig. 2.3). Electronic densities are calculated from occupied atomic orbitals and in the next step the Mattheiss[67] prescription is used to overlap the densities. All the interactions are treated using ground state of exchange-correlation potential [100] in local density approximation(LDA). The total potentials of atoms are treated in muffin tin (MT) approximation: it is flat in the interstitial region and spherically symmetric around atoms $V_{\text{tot}} \approx V_{\text{mt}} + \sum_i V_i(|\mathbf{r} - \mathbf{R}_i|)$. The total potential V_{tot} is separated into the potential of a single scatterer at the absorption site and all other scatterers, e.g., $V_{\text{tot}} = V_c + \sum_{i\neq} V_i$. Green's function $G(\mathbf{r}, \mathbf{r}', t)$ can be separated into central and scattered parts $G = G^c + G^{sc}$.

In the next step, electron density is spherically averaged and the radius of charge neutral atom is estimated. The electron charged density is integrated until the right number of electrons is obtained. This procedure is called Norman prescription and the calculated radii are called MT radii. The MT radii can be overlapped by using input options to reach better empirical results.

In the next step of the SCF loop, Dirac equation is solved for wave function up to Norman radii. An analytical solution with spherical symmetry of free particle state is assumed outside of MT potential. From the solution to Dirac equation and partial phases, Green's function is constructed as a sum for central atom and scattering contributions. Details on

the constructions of scattered part can be found in Ref. [8, 85]. In the next step, density is obtained from the Green's function and the SCF loop is repeated until convergence is reached.

A summary of the work flow of FEFF implementation of RSGF is presented in fig. 2.3.

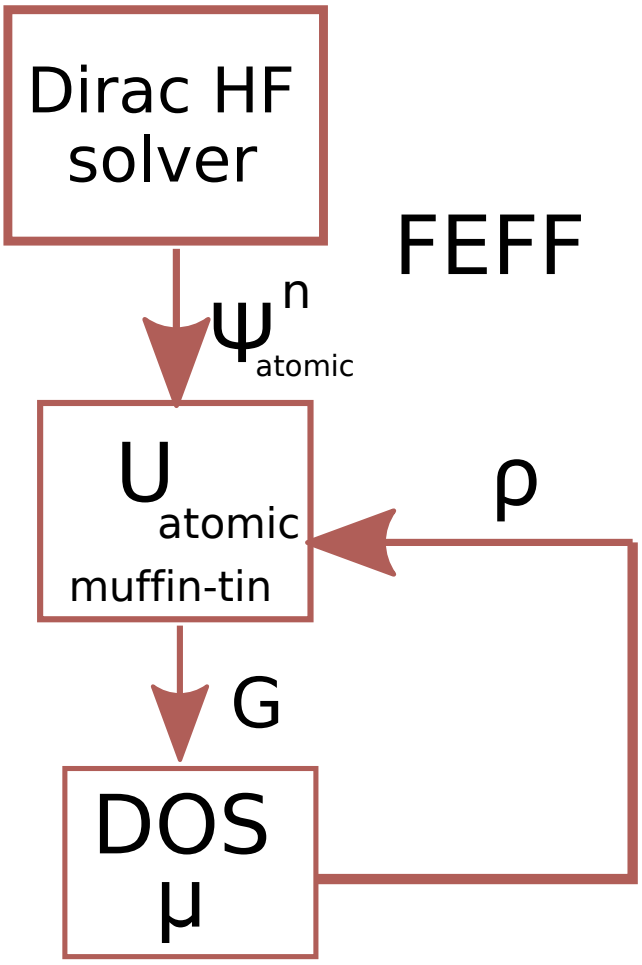

Figure 2.3: FEFF work flow implementation of RSGF approach. Dirac Hartree-Fock solver is solving

In the angular momentum expansion Green's function can be expressed:[79, 80]

$$G(\mathbf{r},\mathbf{r}',E) = -2k \left[\delta_{n,n'} \sum_{L} H^{E}_{Ln}(\mathbf{r}_{>}) \bar{R}^{E}_{Ln'}(\mathbf{r}_{<}) + \sum_{L,L'} R^{E}_{Ln}(\mathbf{r}_n)\, e^{\imath\delta_{Ln}} g^{E}_{Ln,L'n'} e^{\imath\delta_{L'n'}} \bar{R}^{E}_{L'n'}(\mathbf{r}'_{n'}) \right], \tag{2.8}$$

where $R^{E}_{Ln}(\mathbf{r}) = \imath^{l} R^{E}_{Ln}(r) Y_L(\hat{\mathbf{r}})$ is the regular solution of Dirac equation at the site n with angular momentum L and energy E. $H^E = N^E - \imath R^E$, and where N^E is irregular solution of the Dirac equation. The bar denotes complex conjugation, n and n' labels atomic sites nearest to the $\mathbf{r}$ and $\mathbf{r}'$. The full multiple scattering matrix (FMS) in one particle Green's function[8] is:

$$\begin{aligned} g^{E}_{Ln,L'n'} = & \sum_{n_1,L_1} G^{0}_{Ln,L_1n_1} t_{L_1n_1} G^{0}_{L_1n_1,L'n'} \\ & + \sum_{n_1,L_1} \sum_{n_2,L_2} G^{0}_{Ln,L_1n_1} t_{L_1n_1} G^{0}_{L_1n_1,L_2n_2} t_{L_2n_2} G^{0}_{L_2n_2,L'n'} \\ & + \dots \end{aligned}$$

where $G^{0}_{Ln,L'n'} = (1-\delta_{n,n'} G^{0}_{L,L'})(\mathbf{R}_n - \mathbf{R}_{n'})$ is a two center propagator and $t_{Ln} = expi\delta_{Ln} sin\delta_{Ln}$.

Using Green's function one can define density matrix:

$$\rho(\mathbf{r},\mathbf{r}';E) = -\frac{1}{\pi} \operatorname{Im} G(\mathbf{r},\mathbf{r}';E). \tag{2.9}$$

To obtain the real space density matrix one needs to integrate the previous expression over energies:

$$\rho(\mathbf{r},\mathbf{r}') = \int_{\text{occ}} \rho(\mathbf{r},\mathbf{r}';E) dE. \tag{2.10}$$

The diagonal part of the density matrix gives the real space density:

$$\rho(\mathbf{r}) = \rho(\mathbf{r},\mathbf{r}) \tag{2.11}$$

To obtain local density of states (LDOS), one needs to integrate the energy dependent density matrix over the coordinates inside the Norman sphere:

$$\rho_i(E) = \int_{|\mathbf{r}-\mathbf{R}_i|<R_{\text{nrmn}}} \rho(\mathbf{r},\mathbf{r},E) d^3r \tag{2.12}$$

The Fermi energy E_F is determined from conditions of charge neutrality

$$N = \sum_i \int_{E_c}^{E_F} \rho_i(E)\, d\, E, \tag{2.13}$$

where N is the total number of valence electrons, and deeply-bounded core states are frozen in their atomic states. Only valence electron properties are calculated self-consistently. The lower bound in is the core-valence separation energy which is set to 30 eV.

To summarize in this chapter I describe RSGF approach to the solution of Green's function in single particle approximation. I present implementation of RSGF in *ab initio* code `FEFF`.

Chapter 3

MOLECULAR DYNAMIC SIMULATIONS

3.1 Introduction

The aim of molecular dynamics is to model the detailed macroscopic dynamic behaviors of many types of the physical systems. Historically, it dates back to the 1950's, when the first computer simulations become available. Using molecular dynamic simulation techniques, the equilibrium and transport properties of many-body systems become accessible for simulation. The nuclear motion of the system is simulated using laws of classical mechanics. It is a good approximation for molecular systems with atomic number Z $>$ 2 or vibrations with frequencies such that $h\nu < k_b T$. In this approach the movement of ions relative to electrons is assumed to be slow.

3.2 Microcanonical Ensemble

The equation of motion is time invariant under transformation $t \to -t$ and the total energy is constant of motion:

$$\frac{\partial E}{\partial t} = \frac{\partial H(\mathbf{R}^N, \dot{\mathbf{R}}^N)}{\partial t} = 0. \tag{3.1}$$

The properties mentioned above are important in establishing the connection between molecular dynamics and statistical physics. Statistical physics connects microscopic details of the system with the physical observables such as equilibrium thermodynamic properties. Statistical mechanics is based on Gibb's ensemble concept, which is that individual microscopic configurations of a very large system lead to the same macroscopic properties. This means that it is not necessary to know the precise detailed motion of every particle in a system in order to predict its properties.

In the microcanonical ensemble, the number of particles N have energy E and the volume

is fixed. Therefore a dynamic trajectory will generate a series of classical states, corresponding to the microcanonical ensemble. The assumption, that given infinite time, the system will cover the entire constant energy hyper surface is known as the ergodic hypothesis. Thus under the ergodic hypothesis, averages over a trajectory of the system obeying Hamilton's equation are equivalent to averages over microcanonical ensemble.

3.3 Ab initio molecular dynamics

The total ground-state energy of the many body interacting system of electrons with classical nuclei at the positions $\{\mathbf{R}_i\}$ is postulated to equal to the Kohn-Sham energy:

$$\min_{\Psi_0}\{\langle\Psi_0|H_e|\Psi_0\rangle\} = \min_{\psi_0} E^{\mathrm{KS}}[\psi_i]. \tag{3.2}$$

The minimum of Kohn-Sham energy is given by the functional:

$$E^{KS}[\{\psi_i\}] = T_s[\{\psi_i\}] + \int d\mathbf{r}\, V_{\mathrm{ext}}(\mathbf{r})\, n(\mathbf{r}) + \frac{1}{2}\int d\mathbf{r}\, V_H(\mathbf{r})\, n(\mathbf{r}) + E_{xc}[n] + E_{ions}(\mathbf{R}^N), \tag{3.3}$$

which is a functional of auxiliary functions (Kohn-Sham orbitals) $\{\psi_i(\mathbf{r})\}$ which satisfy orthonormality relation:

$$\langle\psi_i|\psi_j\rangle = \delta_{ij}. \tag{3.4}$$

3.4 Born-Oppenheimer Molecular Dynamics

Historically, the Born-Oppenheimer Molecular Dynamics (BOMD) became computationally accessible only after Car-Parinello Molecular Dynamics (CPMD) come in use[18]. This is due to the differences in the approximations of interaction potential. The interaction potential in BOMD is recalculated at each MD step using the Kohn-Sham energy functional. In the case of CPMD, the interaction potential is incorporated using fictitious dynamics of electrons in the equation of motions. Recalculation of interaction potential in the case of BOMD requires more computational resources hence why CPMD was historically used first.

Unlike classical MD, the BOMD uses Kohn-Sham energy for interaction energy $U(\mathbf{R}^n)$ rather than model potential. The Lagrangian for BO is given by:

$$L_{\mathrm{BO}}(\mathbf{R}^N, \dot{\mathbf{R}}^N) = \sum_{I_1}^{N} \frac{1}{2} M_I \dot{\mathbf{R}}_I^2 - \min_{\psi_i} E^{\mathrm{KS}}[\{\psi_i\}; \mathbf{R}^N], \tag{3.5}$$

with constraint on the orthogonality of Kohn-Sham orbitals. Equations of motions are:

$$M_I \ddot{\mathbf{R}}_I = \nabla_I \big[\min_{\{\psi_i\}} E^{\mathrm{KS}}[\{\psi_i\}; \mathbf{R}^N] \big]. \tag{3.6}$$

Forces are given by:

$$\frac{d}{d\mathbf{R}_I} \big[\min_{\psi_i} E^{\mathrm{KS}}[\{\psi_i\}; \mathbf{R}^N] \big]. \tag{3.7}$$

Forces can be calculated from the extended energy functional with the orthogonality constraint:

$$\mathcal{E}^{\mathrm{KS}} = E^{\mathrm{KS}} + \sum_{ij} \Lambda_{ij} (\langle \psi_i | \psi_j \rangle - \delta_{ij}). \tag{3.8}$$

Taking the derivative gives the result:

$$\frac{\mathcal{E}^{\mathrm{KS}}}{d\mathbf{R}_I} = \frac{\partial E^{\mathrm{KS}}}{\partial \mathbf{R}_I} + \sum_{ij} \Lambda_{ij} \frac{\partial}{\partial \mathbf{R}_I} \langle \psi_i | \psi_j \rangle + \sum_i \big[\frac{\partial E^{\mathrm{KS}}}{\partial \langle \psi_i |} + \sum_j \Lambda_{ij} | \psi_j \rangle \big] \frac{\partial \langle \psi_i |}{\partial \mathbf{R}_I}. \tag{3.9}$$

For optimized Kohn-Sham orbitals, the term in the brackets is zero. Thus, the forces simplify to:

$$F^{\mathrm{KS}}(\mathbf{R}_I) = -\frac{\partial E^{\mathrm{KS}}}{\mathbf{R}_I} + \sum_{ij} \Lambda_{ij} \frac{\partial}{\partial \mathbf{R}_I} \langle \psi_i | \psi_j \rangle. \tag{3.10}$$

As we can see, the accuracy of the forces in BOMD depends linearly on the accuracy of Kohn-Sham energy.

3.5 Car-Parrinello unified approach for electrons and ions

R. Car and M. Parinello in the paper "Unified Approach for Molecular Dynamics and Density-Functional Theory"[19] proposed a scheme which combines molecular dynamics (MD) and density-functional theory (DFT). They extended MD beyond usual pair-potential approximation and for the first time made it possible efficiently to simulate both covalent bonded and metallic systems, as well as simulations of large systems.

The original MD simulations were based on empirical pair potential, which is an appropriate approach in the simulation of the systems like rare gases but do not work well in the case of systems with covalent bonds, or in metallic systems. The DFT approach, on the other hand provided accurate description of chemical bonding and ground state properties, but is computationally demanding, which makes in difficult to study very large or disordered systems and to compute interaction forces for MD simulations.

Electronic density of occupied single-particle orthonormal orbitals is given by: $n(\mathbf{r}) = \sum_i |\psi_i(\mathbf{r})|^2$.[50] The point of BO potential is given by the minimum with respect to $\psi_i(\mathbf{r})$ of the energy functional:

$$E[\{\psi_i\}, \{R_I\}, \{\alpha_\nu\}] = \sum_i \int_\Omega d^3r \psi_i^*(\mathbf{r})[-(1/2)\bigtriangledown^2]\psi_i(\mathbf{r}) + U[n(\mathbf{r}), \{R_I\}, \{\alpha_\nu\}], \tag{3.11}$$

where $\{R_I\}$ are nuclear coordinates and $\{\alpha_\nu\}$ is all the possible external constrains. The functional U contains all the interaction terms e.g. Hartree, exchange, correlations contributions, internuclear repulsion and effective electronic potential energy.

Minimization of energy functional Eq. 3.11 and subject to the orthonormality constraint gives self-consistent KS equations:

$$\{-\frac{1}{2}\bigtriangledown^2 + \frac{\partial U}{\partial n(\mathbf{r})}\}\psi_i(\mathbf{r}) = \epsilon_i\psi_i(\mathbf{r}) \tag{3.12}$$

The above equation can be solved by matrix diagonalization. It is required at every MD step. The computational cost of such procedure grows quickly and is expensive in case of disordered systems.

The method proposed by R. Car and M. Parinello[19] unified both MD simulations and DFT calculations of the ground state properties, thereby achieving accurate description of large system behavior. Their only assumption was Born-Oppenheimer (BO) approximation, which separates nuclear and electronic coordinates. In the unified approach proposed by R. Car and M. Parinello KS functional is treated as a complex optimization problem which the authors solve using the concept of simulated annealing.[49] In the simulated annealing approach an objective function $O(\beta)$ is minimized with respect to β using successions of β

which are generated using Boltzmann-like probability distribution ($\propto O(|\{\beta\}|/T)$), which is obtained by using Monte-Carlo procedure. As T goes to 0 the lowest state of $O(|\{\beta\})$ should be reached unless the system is trapped in metastable state.

In the CPMD approach the objective functional is the total energy functional with variational parameters which are the coefficients of the expansion of the KS orbitals, ionic positions and external constrains $\{\alpha_\nu\}$. The parameters $\{\psi_i\}$, $\{R_I\}$ and $\{\alpha_\nu\}$ are time dependent. The Lagrangian considered in CPMD approach:

$$L = \sum_i \frac{1}{2}\mu \int_\Omega d^3r |\dot{\psi_i}|^2 + \sum_i \frac{1}{2} M_I \dot{R}_I^2 + \sum_\nu \frac{1}{2}\mu_\nu \dot{\alpha_\nu}^2 - E[\{\psi_i\}, \{R_I\}, \{\alpha_\nu\}], \tag{3.13}$$

with holonomic constrains:

$$\int_\Omega d^3r \psi_i^*(\mathbf{r}, t)\psi_i(\mathbf{r}, t) = \delta_{ij}. \tag{3.14}$$

M_I are ionic masses and μ_ν are arbitrary parameters. Euler-Lagrange equations gives us equations of motion:

$$M_i \ddot{\mathbf{R}}_I = -\frac{\partial \mathbf{E}}{\partial \mathbf{R}_I} = \mathbf{F}_I[\mathbf{R}_J(t)] \tag{3.15}$$

$$\mathbf{F}_I = -\frac{\partial E}{\partial R_I} \tag{3.16}$$

$$\mu \ddot{\psi_i}(\mathbf{r}, t) = -\frac{\partial E}{\partial \psi_i^*(\mathbf{r}, t)} + \sum_k \Lambda_{ik}\psi_k(\mathbf{r}, t) = -H\psi_i(\mathbf{r}, t) + \sum_k \Lambda_{ik}\psi_k(\mathbf{r}, t) \tag{3.17}$$

$$\mu_\nu \ddot{\alpha}_{nu} = -\frac{\partial E}{\partial \alpha_\nu}. \tag{3.18}$$

The ion dynamics $R_I(t)$ eq. (3.15) have a real physical meaning, whereas dynamics of $\psi_i(\mathbf{r}, t)$ and α_ν are fictitious and do not have real physical meaning, but are only a tools for performing the simulated annealing computation. Λ_{ik} are Lagrange multipliers which were introduced to satisfy the holonomic constrain eq. (3.14). Lagrange eq. (3.13) defines the potential energy E and classical kinetic energy K:

$$K = \sum_i \frac{1}{2}\mu \int_\Omega d^3r |\dot{\psi_i}|^2 + \sum_i \frac{1}{2} M_I \dot{R}_I^2 + \sum_\nu \frac{1}{2}\mu_\nu \dot{\alpha_\nu}^2. \tag{3.19}$$

The equilibrium value of classical kinetic energy can be calculated as temporal average over trajectories of the equation of motion eqs. (3.15), (3.17) and (3.18). By variation of the

velocities $\{\dot{\psi}_i\}$, $\{\dot{R}_I\}$ and $\{\dot{\alpha}\}$, the temperature of the system can be slowly reduced, to reach the equilibrium state minimum of E. At the equilibrium $\ddot{\psi}_i = 0$ and eq. (3.18) is identical to KS eq. (3.12) up to unitary transformation and eigenvalues of Λ are the same as KS eigenvalues. When these conditions are met, the Lagrangian eq. (3.13) describes the real physical system. Thus it is clear that the approach proposed by R. Car and M. Parinello is much more efficient in the large system then direct diagonalizations techniques.

3.5.1 Stationary solution

In the steady state $\ddot{\psi}_i(\mathbf{r}, t) = 0$ all time derivatives are equal to zero and eq. (3.17) leads to the equation:

$$H\psi_i(\mathbf{r}, t) = \sum_k \Lambda_{ik}\psi_k(\mathbf{r}, t). \tag{3.20}$$

It is a usual solution with Λ_{ik} the Lagrange multipliers. Considering matrix elements in the equation eq. (3.20) shows that Λ is the transpose H ($\Lambda_{ik} = H_{ik}$), where H is the KS Hamiltonian. Diagonalizing Λ gives the eigenvalues of KS equations. The solution is self-consistent since the full KS energy is minimized eq. (3.11). The solution is stationary only if the energy is at its variational minimum. Dynamic simulated annealing is a numerical method of finding the minimum of non-linear self-consistent KS eq. (3.12).

Car and Parrinello[19] present a comparison of calculation of the eigenvalues of Si is presented using self-consistent solution of KS equations with local pseudo-potential and local-density approximation. The use the CPMD approach and utilize simulated annealing. The two results converges after initial equilibration time.

Pseudo-potential theory[78] allows one to replace the strong electron ion potential with a much weaker potential, called pseudo-potential, which describes all the valence electron properties moving through the solid, including relativistic effects. The solid is replaced with the pseudo valence electrons and pseudo-ion cores. Pseudo electrons experience exactly the same potential as real electrons in the region outside the core. Inside the core, region pseudo electrons experience much weaker potential, which makes the solution of the Schrdinger

equation much simpler by allowing expansion of the wave function on a much smaller set of plane waves.

3.5.2 Nuclear dynamics

The main benefit of the CPMD approach is the coupled motion of nuclei and electrons. This means that in the *ab initio* electronic structure, calculations of thermal motion, liquids and thermal phase transition can be treated at the same time[58, 60]. The fictitious kinetic energy eq. (3.19) does not represent quantum mechanical energy of electrons; electron dynamics results of the Lagrangian are also fictitious. The advantages of this approach is that it allow the ground state of the electrons to propagate through the space of basis functions and be close to the true ground state.

3.5.3 Challenges in Car-Parrinello unified algorithm

For systems with an energy gap, the CPMD approach works well. The characteristic frequency of the fictitious dynamics of the electron degrees of freedom are proportional to $\propto E_{\text{gap}}/\mu$. However in the case when the fictitious frequency of oscillations is much larger than the nuclear vibrational frequency, the electrons follow nuclei adiabatically.[76] Thus the fictitious mass needs to be set to satisfy adiabatic conditions within acceptable accuracy.[20]

Another challenge of the CPMD approach is the time step, which is governed by fictitious electron degrees of freedom. A typical mass of the electron can be on the order of hundreds of electron mass. This can be a problem in the case of plane wave calculations because they have a property that cut-off frequency can be raised indefinitely to achieve convergence.

When the gap is vanishing and the levels cross in the metal, this can lead to a transfer of energy to fictitious degrees of freedom. This issue can be solved by using a thermostat which pumps energy into the ionic system and renormalizes the kinetic energy of the system.

3.6 Alternative approaches to density functional quantum molecular dynamics

Due to the challenges of the CP schema described above, other methods have emerged. This new methods are based on the idea proposed by Kohn and Sham[50] that local density functional (LDF) theory, when coupled with the Hellman-Feynman theorem[32], can efficiently evaluate forces to allow investigation of electronic and dynamic properties at the same time.

An alternative to the simultaneous update of electrons and ions is the exact calculations of the electronic ground state after each ionic step. The local part of Hamiltonian is diagonal in real space and the kinetic energy part is diagonal in reciprocal space. Evaluation of both is fast when fast Fourier transformations are used.

This approach is not as elegant as described CPMD approach, and the time step is governed by nuclear dynamics, just as in classical MD calculations. It is larger than in the CP case by an order of magnitude, though, which make calculations more efficient then classical MD calculations.

At each ionic state the ground state of the electrons needs to be solved much more accurately, which would require more steps in the self consistent loop. The two approaches require roughly same number of calculations. Iterative methods used for finding eigenvalues and eigenvectors can be chosen to find the eigenvalues and eigenvectors of all the occupied states or only occupied subspace which spans the eigenvectors. The eigenvalues can be used in Fermi function, so treatment of the metals is not problematic.

3.7 Nosé-Hoover thermostat

In classical MD, simulations are done using a microcanonical ensemble where the number of particles, volume and the energy have constant value. However in the experiments, temperature is generally controlled instead of energy. An ensemble in which the number of particles and the volume and temperature is controlled is called a canonical ensemble. Some techniques which are used to control the temperatures are velocity rescaling, the Andersen

thermostat[5] and the Nosé-Hoover thermostat.

Nosé showed that canonical distribution can be generated using a smooth, deterministic, and time reversible trajectory. Time scale parameter s was introduced and its conjugated momentum p_s and parameter Q with Hamiltonian[70]:

$$H_{\text{Nosé}} = \Phi(q) + \sum p^2/2ms^2 + (X+1)kTkns + p_s^2/2Q, \tag{3.21}$$

where X is the number of degrees of freedom of the system coupled to the heat bath (parameters s and p_s). Nosé proved that microcanonical distribution in the augmented set of variables is equivalent to a canonical distribution with variables $q\,, p'$ and p' is a scaled momentum p/s. Equations of motion for the Hamiltonian are:

$$\dot{q} = \frac{p}{ms} \tag{3.22}$$

$$\dot{p} = sF \tag{3.23}$$

$$\dot{s} = \frac{sp_s}{Q} \tag{3.24}$$

$$\dot{p_s} = \sum \frac{p^2}{ms^2} - (X+1)kT, \tag{3.25}$$

where substitution was made $t_{\text{old}} \equiv s\, t_{\text{new}}$. Nosé showed that the phase space distribution resulting from the equation of motion are canonical in in variable $q, p/s$.

3.8 Results of MD simulations using `VASP`

In this section, I present details of MD simulations for Be, Li and Si. I will discuss specific strategies that I used to obtain convergent results and my procedures for ensuring the quality of simulations.

In the preliminary runs, the optimal size of the super cell is investigated first. It should be selected based on the following criteria:

- Convergence of the temperature to that desired, as a functions of the number of steps;
- Convergence of the total energy as a function of the number of steps;

- Convergence of Free energy as a function of the number of steps;
- Number of SCF steps needed to converge at each step MD step; and
- Center of mass of super cell behavior.[1]

3.8.1 Be

Using these criteria, I found that a super cell of orthogonal cell of $4 \times 4 \times 6$ repetitions of the orthogonal 4-atom conventional cell of 4 atoms worked best for us. I found that a symmetric cell of $4 \times 4 \times 4$ repetitions of orthogonal cell would start drifting in certain direction, probably due to the hexagonal closed pack symmetry of the Be. Along z axis, the unit cell is larger than in x and y direction, which makes the super cell non symmetrical.

In the MD simulations, I controlled temperature using the Nosé-Hoover thermostat.[70] For the range of temperatures I studied, I adjusted the thermostat s_{mass} parameter to match the characteristic phonon frequencies, which range from 10 – 18 THz for Be.[96] Calculations were performed at the NERSC computational facility. Each MD simulation was limited by the submission queue time limit. I checked the results for consistency after each of simulation and resubmitted any suspicious calculations.

In the next sequence of graphs, I present some of the metrics which were taken into account while evaluating the convergence of the MD simulations. I present graphs for temperature T, energy E and free energy F as a function of time for simulations of the structure pf Be at: 300 K, 40 K and 850 K.

In fig. 3.1 I present the temperature of the system simulation as a function of time at the temperature of thermostat 300 K. The oscillations of the temperature as a function of time are due to the way the thermostat is coupled to the system. I start with the system which is a perfect crystal with temperature equal to zero. The first spike is due to the differences between the thermostat temperature and the temperature of the system. Eventually the

[1] In some of the cases, the structure starts drifting. In particular, this was observed in non-symmetric super cells size e.g 2 by 3 by 3 of a primitive cell. In the MD of Be, structure started drifting.

temperature of the system and the temperature of thermostat equilibrate to the desired value of 300 K. In fig. 3.2 the spike in the energy at around 2 ps is because of termination

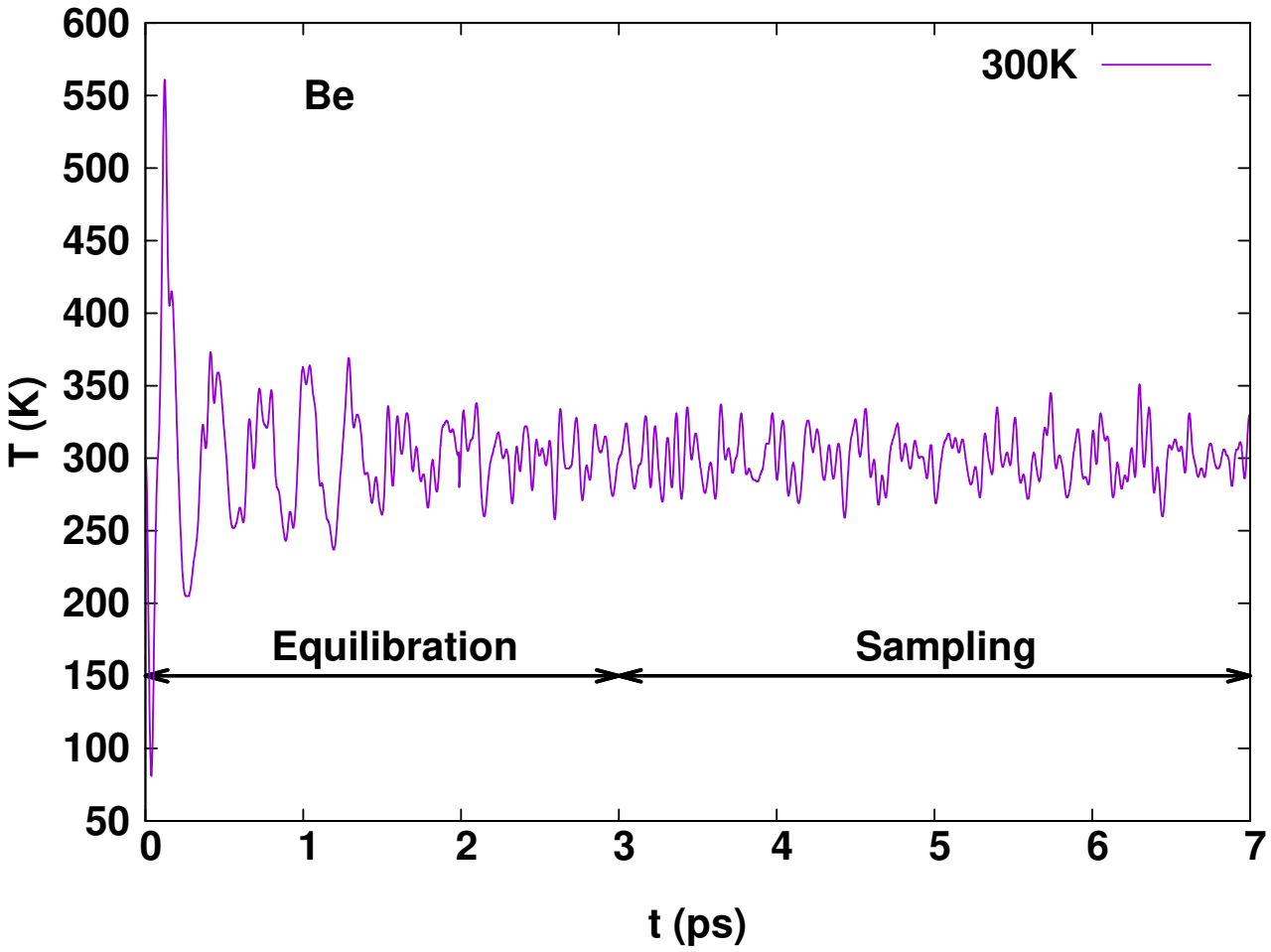

Figure 3.1: Temperature as a function of time simulation for Be at T= 300 K

due to the limit in calculations time. The speed of each atom is update once convergence is reached. In the case of sudden termination the information about the speed is lost. The described behavior does not affect the temperature and free energy of the system see the figs. 3.1 and 3.3. The energy of the system is not fixed in the canonical ensemble. That is why the energy of the system rises as a function of time.

The free energy F of the system is presented in fig. 3.3. It oscillates as a function of time.

Is is interesting to note that the frequency of oscillations changes as a functions of time. At about 4 ps the oscillations become consistent.

In figs. 3.4 to 3.6 I present results of MD simulations of Be at 40 K. In fig. 3.4 the dependence of the temperature as a function of time. After the two nodes at 0.5 ps and at 2 ps, the system equilibrates at about 2.5 ps.

In fig. 3.5, I present energy of the system as a function of time for temperature 40 K. As a function of time, the energy is decreasing. The system is modeled as a canonical ensemble in which the total energy is not fixed. In fig. 3.6, I present free energy as a function of time. The system equilibrates fairly fast at about 2.5 ps. In figs. 3.7 to 3.9, I present results of the simulation of Be at the temperature of 850 K. I start from the super cell structure which corresponds to 650 K, which is why the system does not have an initial spike in the temperature as a function of time, unlike what the system experienced at 40 K and 300 K. Energy as a function of time is rising because the system is modeled as a canonical ensemble.

3.9 Conclusions

I used *ab initio* MD simulations package `VASP` for a wide range of parameters from room temperature and densities to warm dense matter regime. With proper care it is a powerful tool for prediction of the thermally disordered structure which has many fine tuning parameters. With enough training and careful analysis of results converged calculations can be achieved. Plots of energy and temperature as a function of simulated time are the most important metrics upon which decision of convergence can be made.

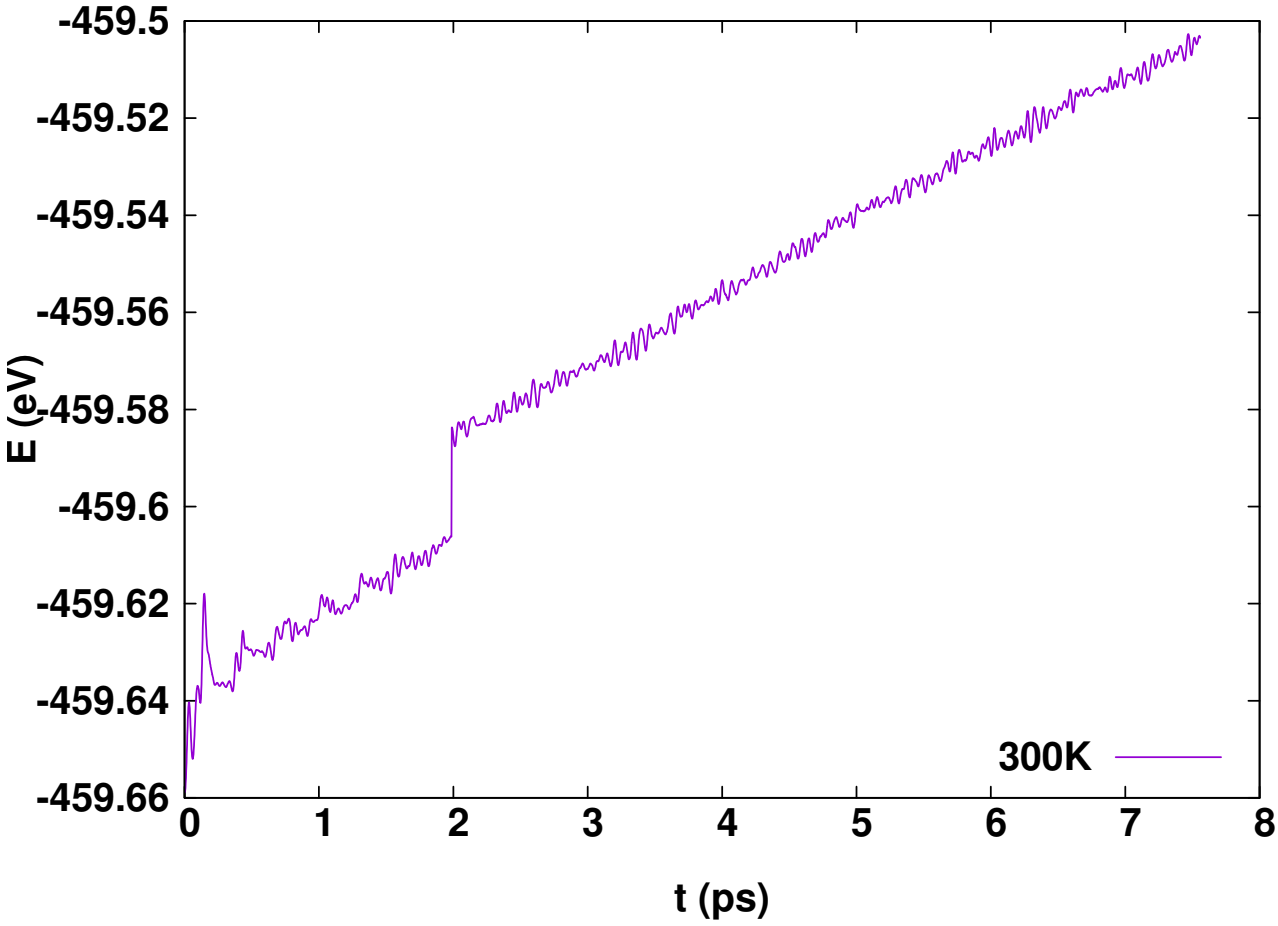

Figure 3.2: Energy as a function of time simulation for Be at T= 300 K. Jump in the energy around 2 ps is due to the computational procedure. In which every calculation is limited by the time.

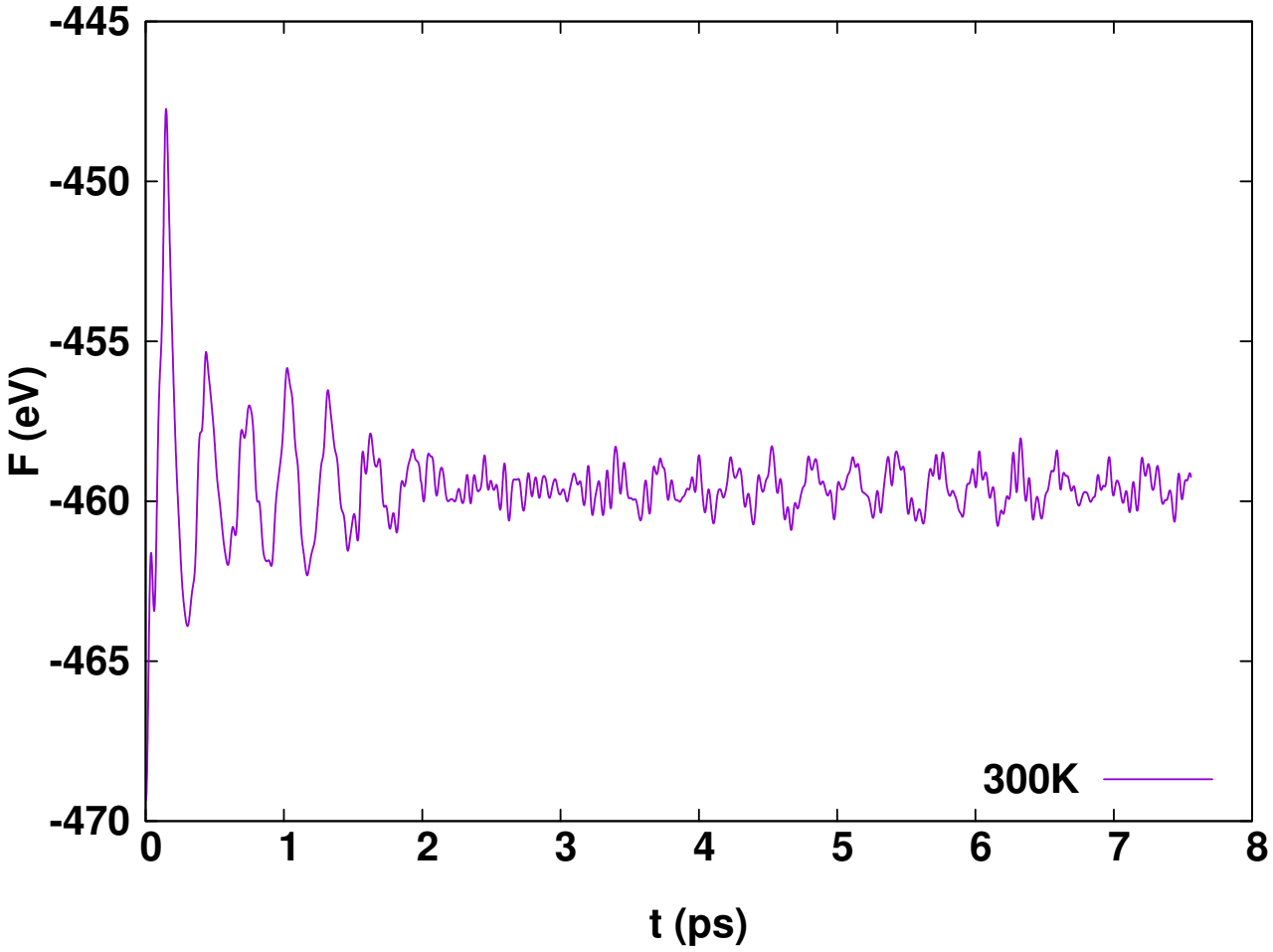

Figure 3.3: Free energy as a function of time simulation for Be at T= 300 K. Oscillations of the free energy is due to the nature of Nosé-Hoover thermostat.

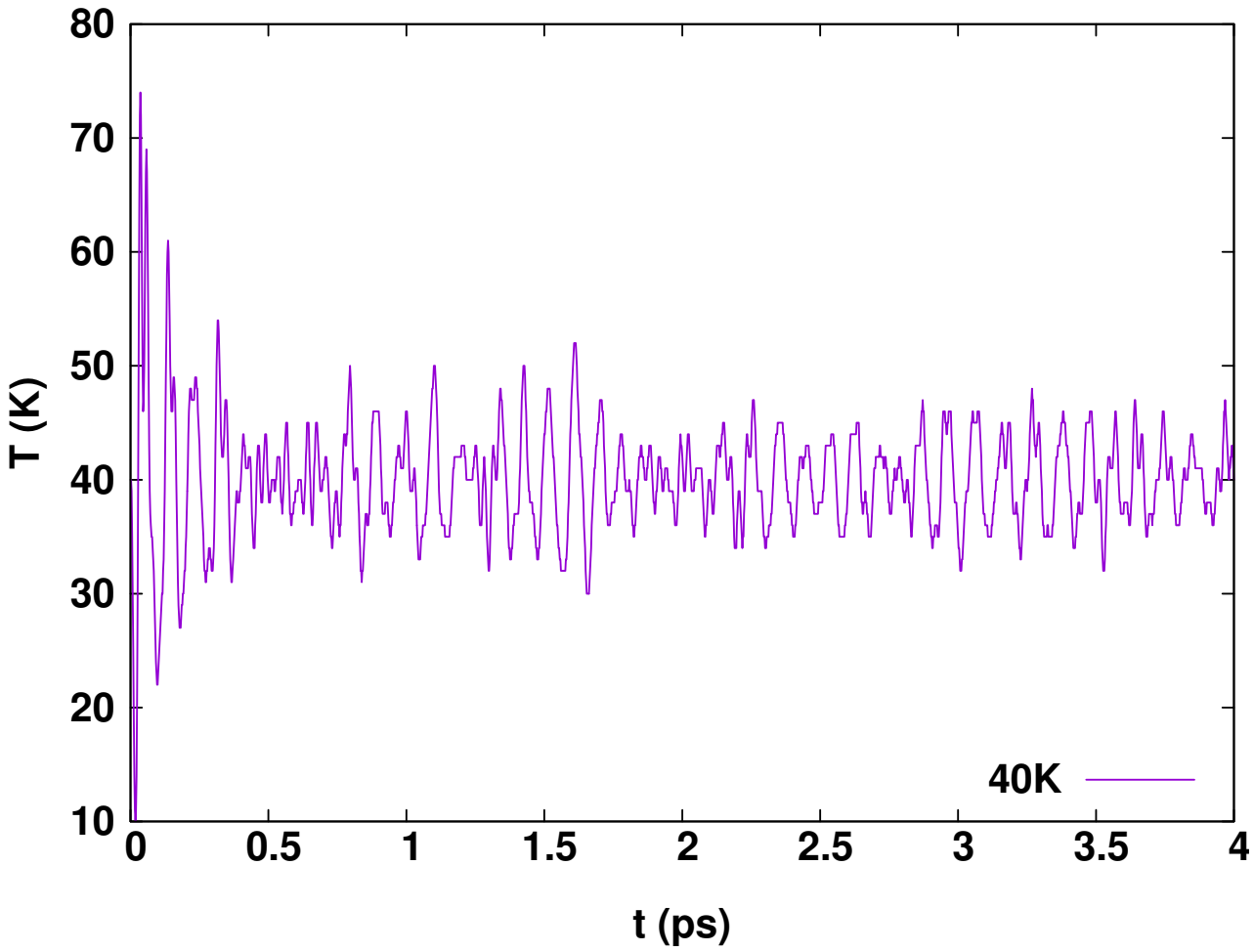

Figure 3.4: Temperature as a function of time simulation for Be at T= 40 K

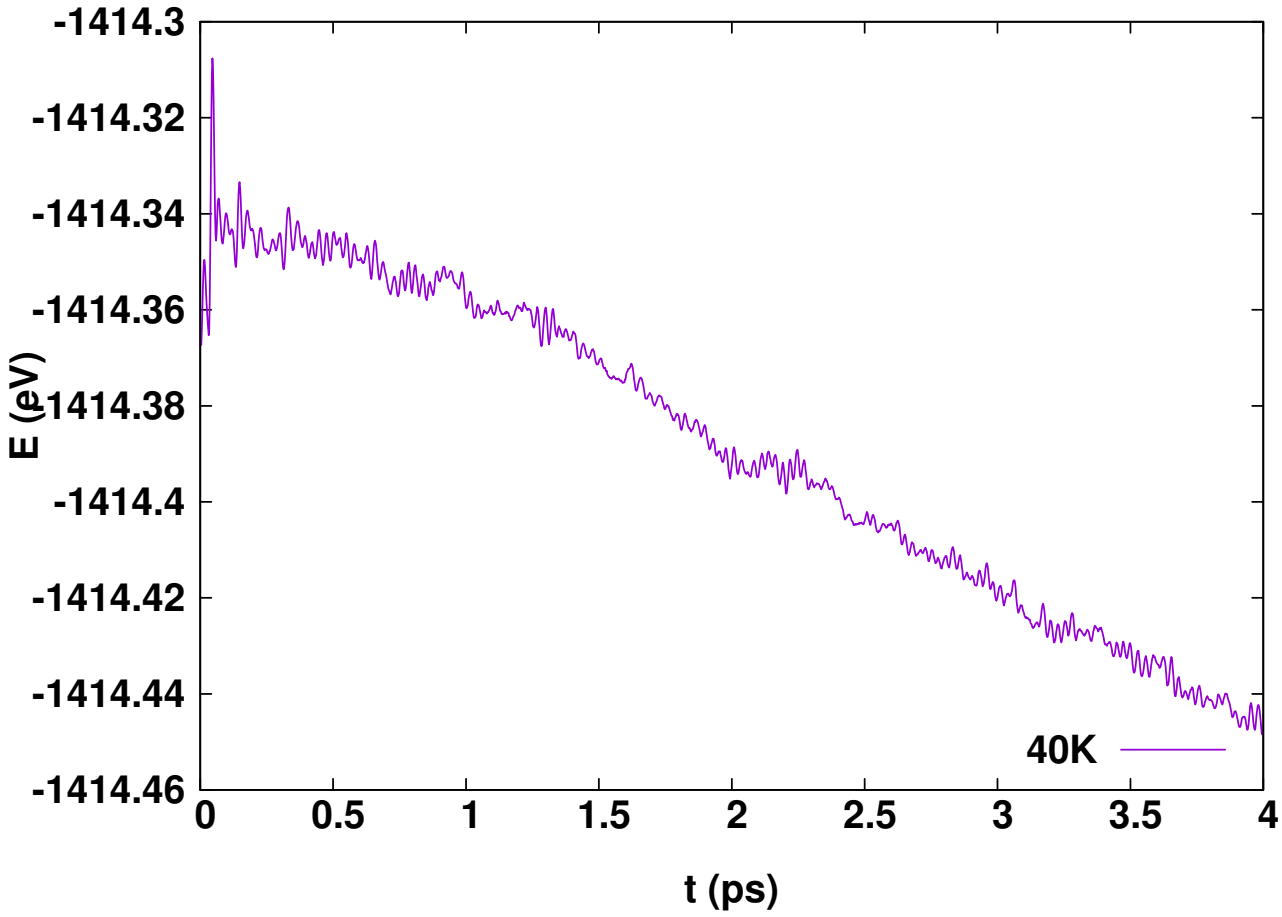

Figure 3.5: Energy as a function of time simulation for Be at T= 40 K.

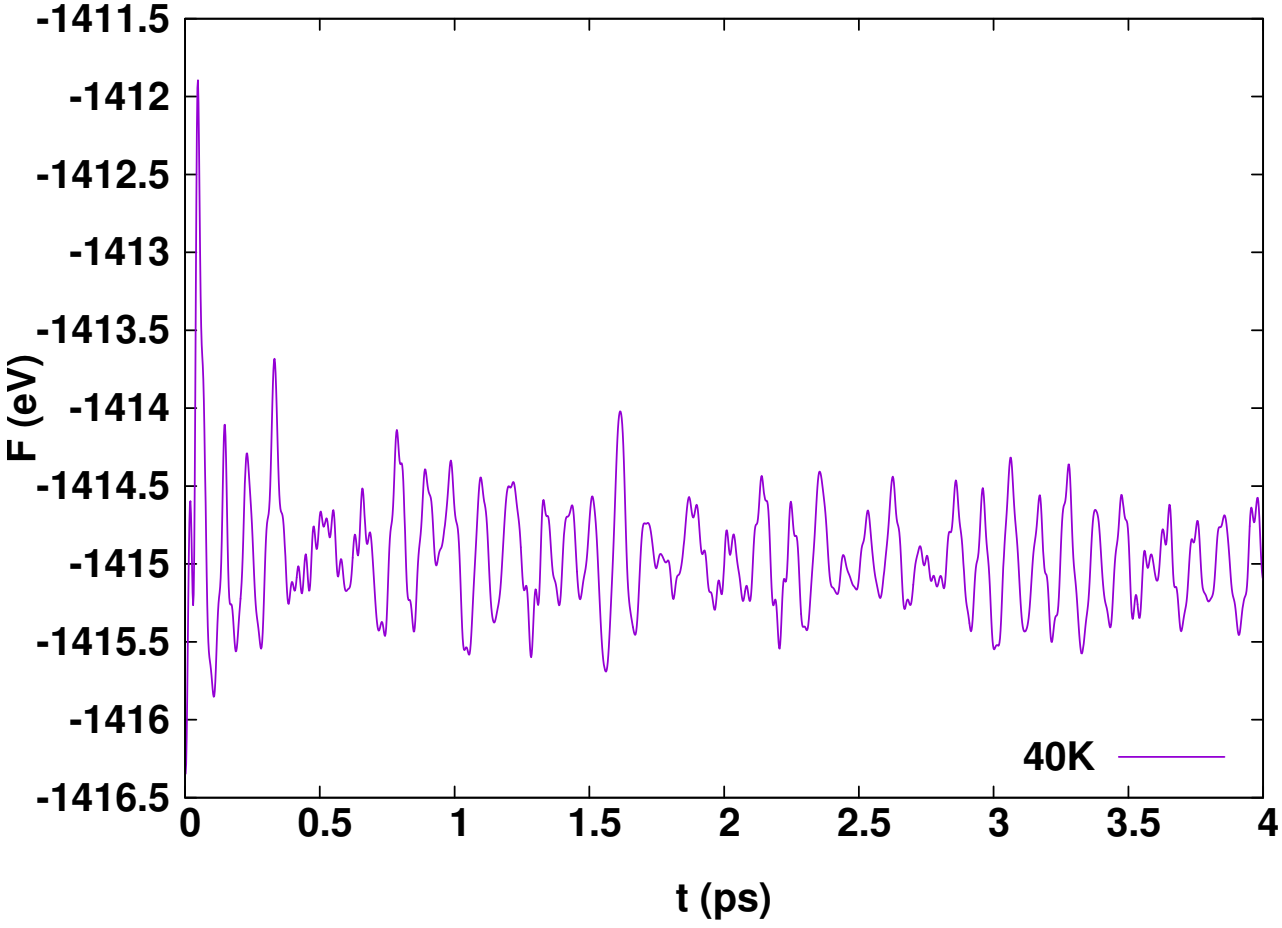

Figure 3.6: Free energy as a function of time simulation for Be at T= 40 K. Oscillations of the free energy is due to the nature of Nosé-Hoover thermostat.

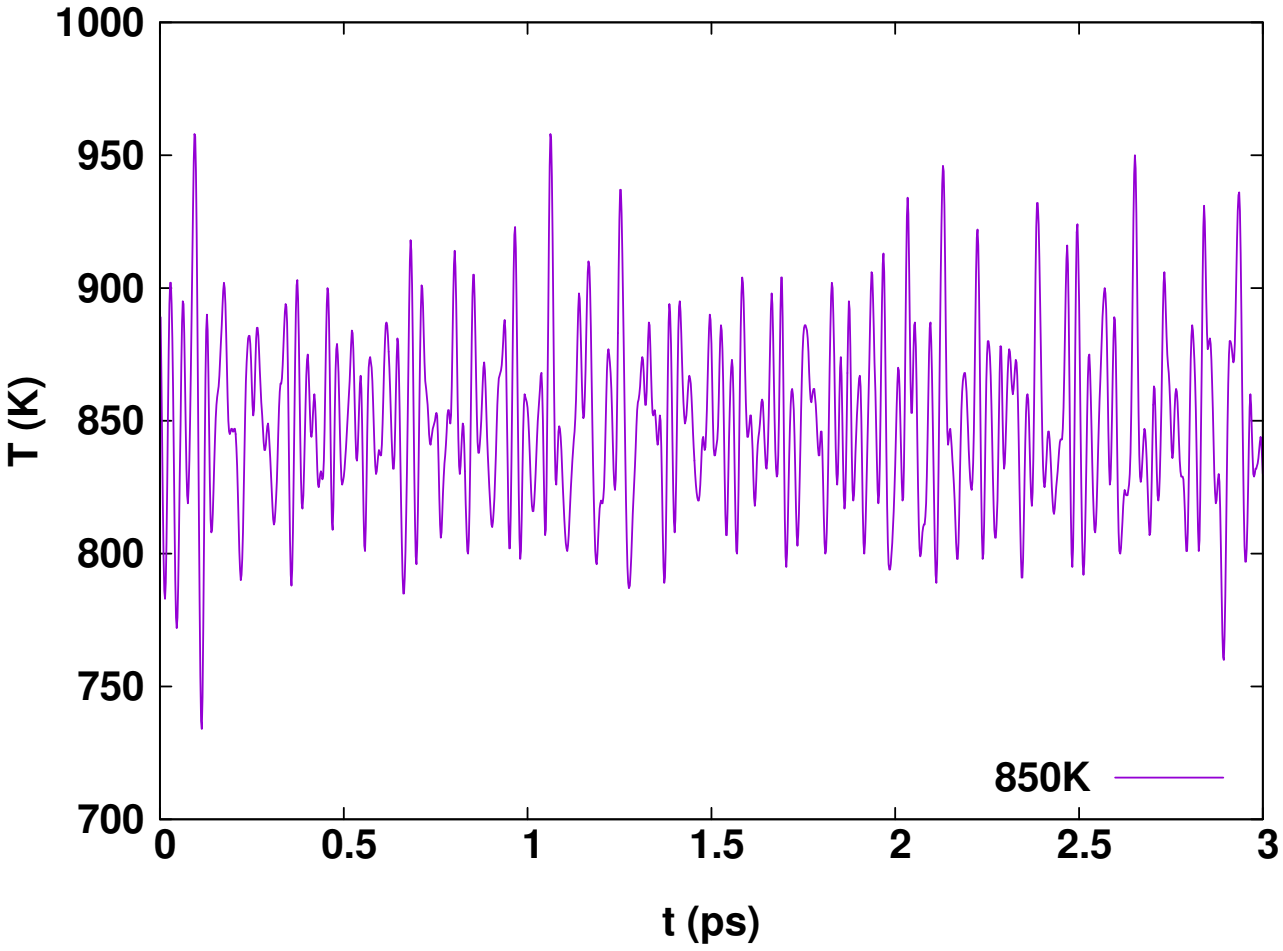

Figure 3.7: Temperature as a function of time simulation for Be at T= 850 K

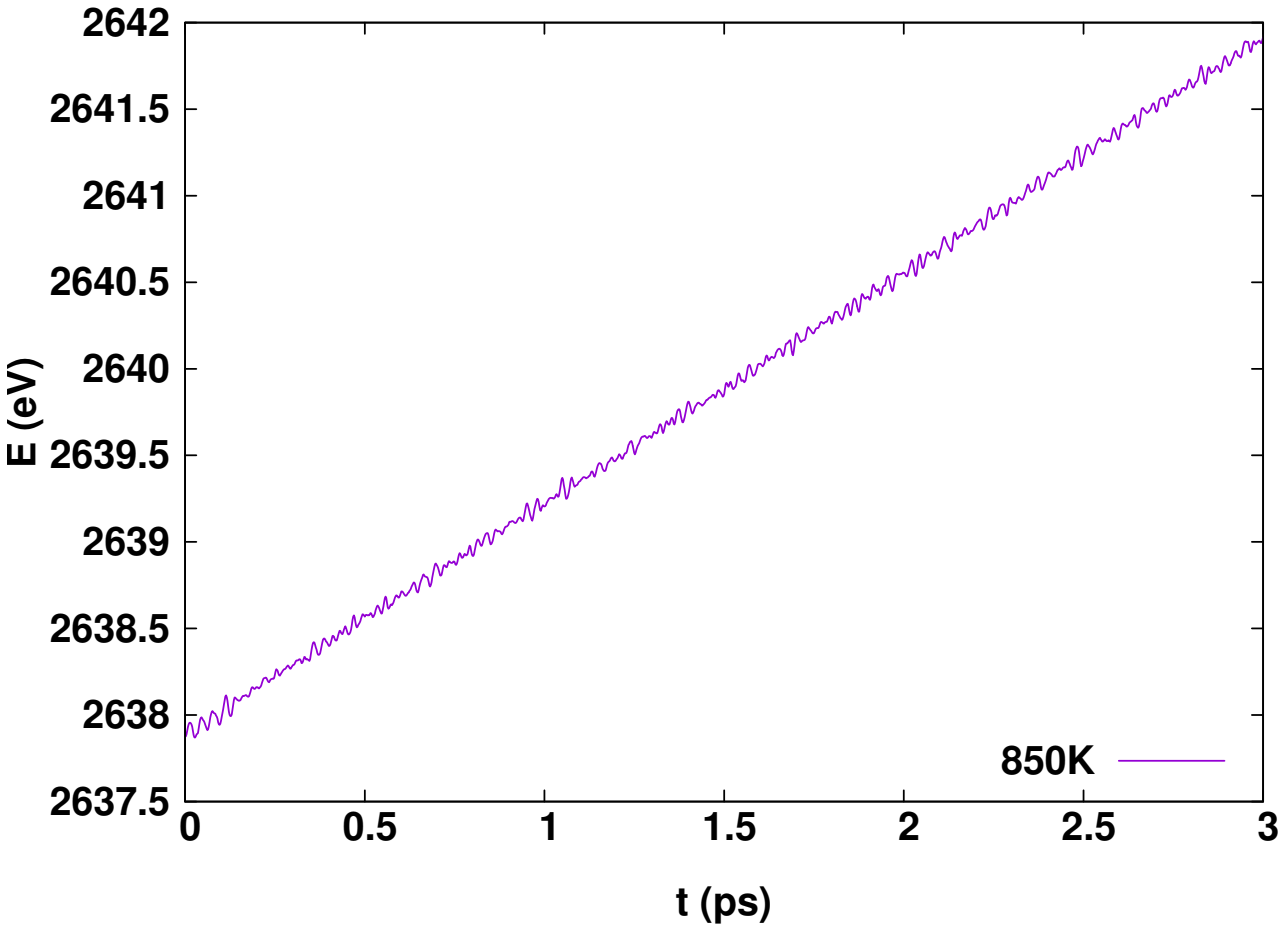

Figure 3.8: Energy as a function of time simulation for Be at T= 850 K.

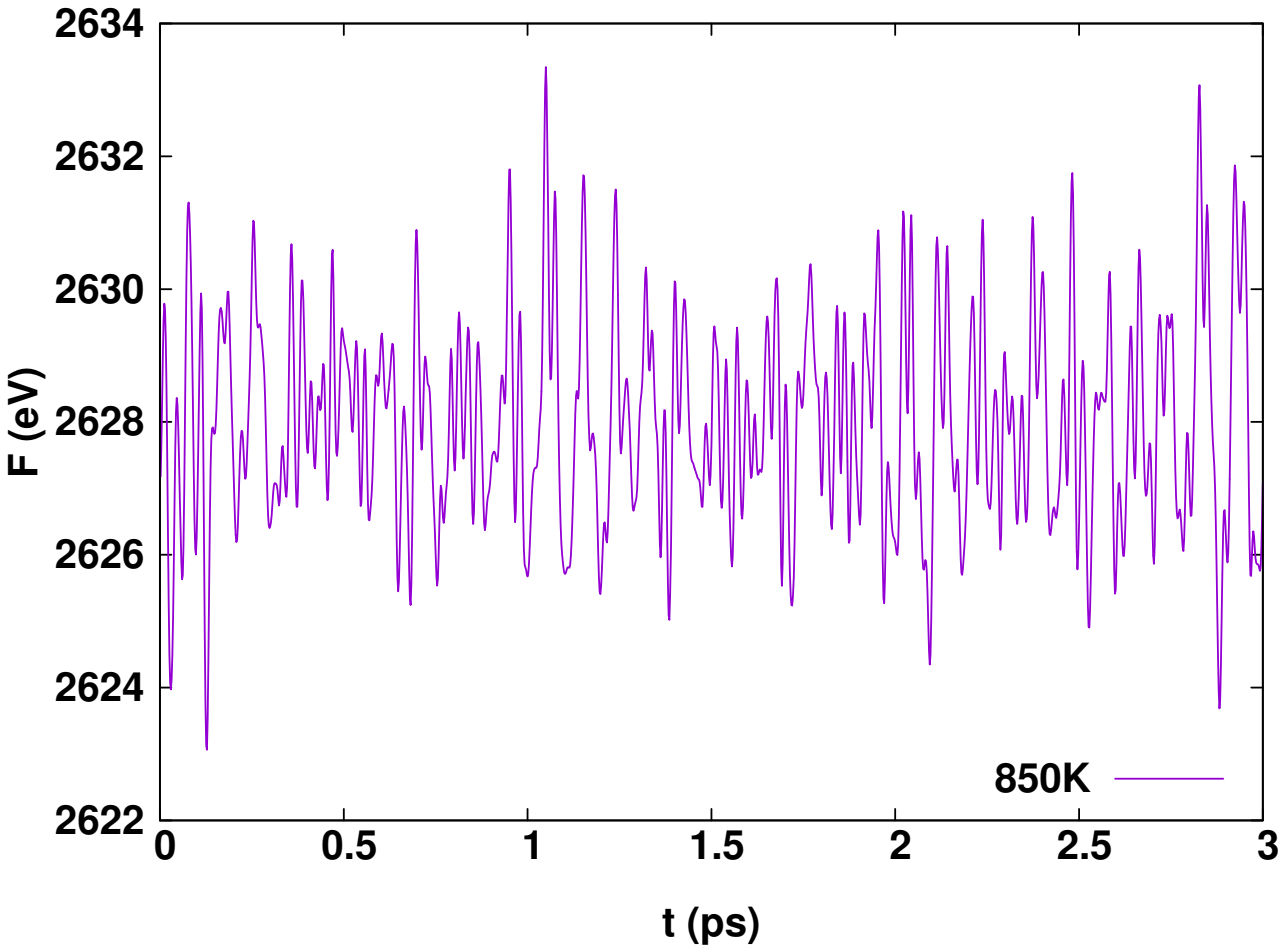

Figure 3.9: Free energy as a function of time simulation for Be at T= 850 K. Oscillations of the free energy is due to the nature of Nosé-Hoover thermostat.

Chapter 4

BEYOND SINGLE PARTICLE APPROXIMATION: CHARGE TRANSFER SATELLITES IN X-RAY SPECTRA OF TRANSITION METAL OXIDES

(*Originally published as*: E. Klevak, J. J. Kas, and J. J. Rehr. Phys. Rev. B 89, 085123 (2014))

Strongly correlated materials such as transition metal oxides (TMOs) often exhibit large satellites in their x-ray photoemission (XPS) and x-ray absorption spectra (XAS). These satellites arise from localized charge-transfer (CT) excitations that accompany the sudden creation of a core hole. Here we use a two-step approach to treat such excitations in a localized system embedded in a condensed system and coupled to a photoelectron. The total XAS is then given by a convolution of a spectral function representing the localized excitations and the XAS of the extended system. The local system is modeled roughly in terms of a simple three-level model, leading to a double-pole approximation for the spectral function that represents dynamically weighted contributions from the dominant neutral and charge-transfer excitations. This method is implemented using a resolvent approach, with potentials, radial wave-functions and matrix elements from the real-space Green's function code `FEFF`, and parameters fitted to XPS experiments. Representative calculations for several TMOs are found to be in reasonable agreement with experiment.

4.1 Introduction

Shake-up excitations in x-ray absorption (XAS) and x-ray photoemission spectra (XPS) have long been of interest.[2, 25, 99] These excitations are generated by the intrinsic response of a system to a suddenly created core hole, and are reflected in satellite peaks in the spectra. Examples in XAS range from edge-singularities in metals,[71] to many-body amplitude factors in x-ray absorption fine structure.[81] Recently such satellites have also been found to explain the extrinsic and intrinsic losses and interference effects in XPS experiments.[38, 95] However, these effects are relatively small in weakly-correlated materials, where the dominant excitations are plasmons. In those cases the satellite amplitudes are of order 10% of the main peak and become negligible near absorption thresholds in the adiabatic limit.[82] Consequently, broadened single-particle theories with a core hole can be good approximations.[37, 42, 47, 83] In contrast, dramatic satellites comparable in strength to the main peak are typically observed in the spectra of correlated materials such as TMOs and high-temperature superconductors. These satellites are often attributed to localized charge-transfer (CT) excitations.[26] In such cases the one-particle approximation fails dramatically in the near-edge region. Several approaches with various degrees of sophistication have been introduced to address this behavior. For example, the "charge-transfer multiplet approach" treats strong correlations locally, with solid-state effects modeled by crystal-field parameters.[24] Configuration interaction techniques have also been applied to small clusters,[45] but these methods are computationally intensive.

Our goal in this work is to develop a simple yet practical, semi-quantitative approach to model both local correlations and solid-state effects to explain these excitations. Our approach is based on a simplified two-step model with a localized system embedded in a solid, and coupled to a photoelectron. The approach incorporates both localized charge-transfer excitations and long-ranged plasmon excitations. In particular, our method combines the model of localized excitations introduced by Lee, Gunnarsson and Hedin (LGH),[61] with the treatment of solid-state effects and other inelastic losses as in the real-space Green's

function approach used in the FEFF9 XAS code.[83] As a justification for this separation we note that the localized and extended excitations are spatially and energetically decoupled. Our main result is an expression for the XAS of charge-transfer systems as a convolution of an effective spectral function $A_L(\omega, \omega')$ that contains the localized CT excitations, and an approximation for the XAS of extended systems $\tilde{\mu}(\omega)$ that builds in long-range, extrinsic inelastic losses

$$\mu(\omega) = \int d\omega' \, A_L(\omega, \omega') \, \tilde{\mu}(\omega - \omega') \equiv A_L * \tilde{\mu}, \tag{4.1}$$

where $A_L(\omega, \omega') \equiv A_{kcL}(\omega')$ and photon energy $\omega = \varepsilon_k - \varepsilon_c$, where ε_k and ε_c are energies of photoelectron state k and core level correspondingly. As discussed by Kas et al.,[48] $\tilde{\mu}(\omega)$ is related to the quasi-particle XAS by an analogous convolution $\tilde{\mu} = A_Q * \mu$. At low energies compared to the plasma frequency, plasmon satellites become negligible and $\tilde{\mu}(\omega) \approx \mu(\omega)$, i.e. the spectra calculated in the presence of a core hole. Within the simplest three-level model for the localized system, the CT spectral function A_L has two energy-dependent peaks separated by a characteristic charge-transfer energy splitting δE which is typically a few eV. Our result in eq. (4.1) is similar to the formulation of Calandra et al.,[17] where the spectral function is taken to be the XPS spectra $A_L(\omega, \omega') = \sigma(\omega')$. In contrast the present approach makes use of an explicit model for the localized system and also approximates dynamic effects, such as the crossover from the adiabatic to the sudden limit. I have applied this method systematically to a number of 3d TMOs, and obtain results in reasonable agreement with experiment and other calculations.[17, 103]

4.2 Theory

4.2.1 LOCAL MODEL

Our model for the localized system is adapted from the three-level tight-binding model of Lee, Gunnarsson and Hedin (referred to here as LGH), [61] which is only briefly summarized here. For clarity we adopt similar notation and some key formulae are reproduced in section 4.4; we refer to original paper [61] for additional details. The LGH model can be extended to a more

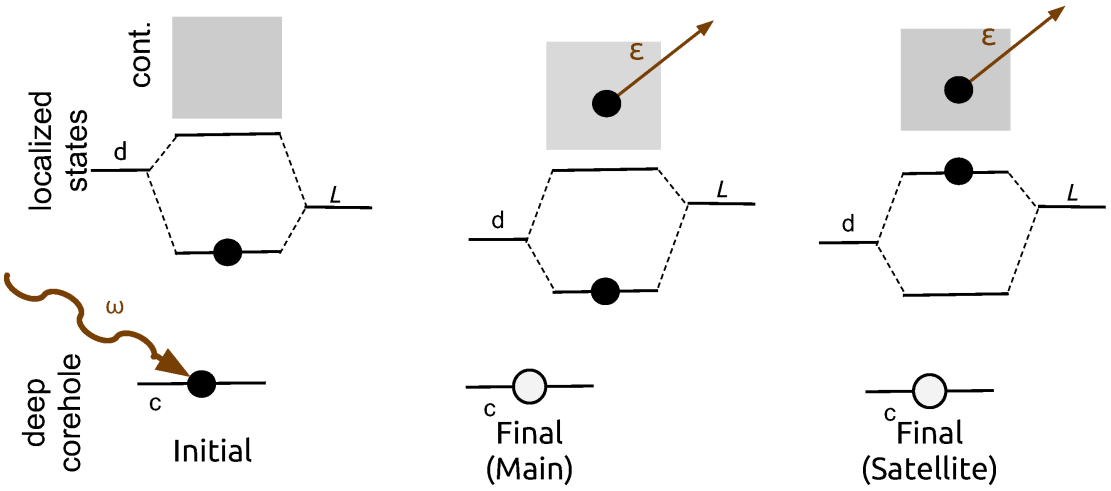

Figure 4.1: Schematic representation of the three-level model (core state and two localized states): the labels are d - transition metal 3d level; L - ligand valence state, cont - continuum states; ε - photoelectron kinetic energy, arrows represent the momentum of scattered electron.

realistic description, for example, using the Haydock recursion scheme[41] applied to a tight-binding Hamiltonian, and keeping only the leading iterations. Nevertheless, the simplified LGH model captures the essential physics of the charge-transfer process. As illustrated in fig. 4.1, the levels include a strongly localized d level, a less localized ligand level L, and a deep core level c. Physically, this local model represents a system in which upon photoecxitation, a localized level d is pulled below the ligand level L due to the Coulomb interaction with the core hole. As a result, there is a finite probability that the electron originally in the L state is transferred to the d state. This process corresponds to the lowest energy main peak ("shake-down") in the photoemission process and strongly screens the core hole. There is also a finite probability that the electron will remain in level L, corresponding to the satellite peak and a less screened core hole. The effective spectral function is determined from the relative probabilities of these two processes.

The Hamiltonian of the full system is separated as

$$\mathcal{H} = \mathcal{H}_0 + T + V + \Delta, \tag{4.2}$$

where $\mathcal{H}_0$ is the Hamiltonian of the local system, T is the kinetic energy of the photoelectron, V is the Coulomb interaction between the photoelectron and the local system, and Δ is the coupling to the x-ray field. In detail

$$\begin{aligned} \mathcal{H}_0 &= \sum_i \varepsilon_i n_i + \sum_l U_l n_c n_l + t(c_d^\dagger c_L + hc), \\ V &= \sum_{\mathbf{k}\mathbf{k}'} \left[\sum_i n_i V^i_{\mathbf{k}\mathbf{k}'} - V^c_{\mathbf{k}\mathbf{k}'} \right] c^\dagger_{\mathbf{k}} c_{\mathbf{k}'}, \\ \Delta &= \sum_{\mathbf{k}} M_{c\mathbf{k}} \left[c^\dagger_{\mathbf{k}} c_c + hc \right]. \end{aligned} \tag{4.3}$$

Here $i = (c, d, L)$, $l = (d, L)$, ε_d, ε_L, and ε_c are, respectively, the bare energies of the d, L and c levels, U_d and U_L represent, respectively, the Coulomb interaction between the core hole and the d and L levels, and t is a hopping matrix element, which we approximate by just one value, k and k' are wave vectors of continuum states. n_i denotes occupation number of each level. Implicit in this model is the constraint $n_d + n_L = 1$, so that $\mathcal{H}_0$ can be simplified in terms of a single Hubbard-like parameter $U = U_d - U_L$. Similarly, $n_c = 0$ when the core hole is present and only the difference in the potentials $V_{sc}(\mathbf{r}) = V_L(\mathbf{r}) - V_d(\mathbf{r})$ is needed in eq. (4.5) to represent the change in potential when the electron hops from the ligand level L to the localized d level. The scattering matrix elements $V_{kk'}$ are given by $\langle \vec{k} | V_{sc}(\mathbf{r}) | \vec{k}' \rangle$, and $U = \int d^3r\, \rho_c(\mathbf{r}) V(\mathbf{r}) \approx V(0)$, since the core charge ρ_c is highly localized. Throughout this paper we will use Hartree atomic units $e = \hbar = m = 1$ unless otherwise specified.

The excited states of this local model $|\Psi^f_{sk}\rangle$, with $s = 1$ and 2, can be calculated exactly within a two-particle basis $|\psi_s\rangle|\psi_k\rangle$, using the resolvent approach of LGH

$$|\Psi^f_{sk}\rangle = \left[1 + \frac{1}{E - \mathcal{H}_0 - T - V - i\eta} V \right] |\psi_s\rangle|\psi_k\rangle, \tag{4.4}$$

where $|\psi_s\rangle$ correspond to the eigenstates of $\mathcal{H}_0$ [see section 4.4] with a full core-hole $n_c = 0$.

Table 4.1: Parameters used in the local model: U, t and δE for MnO, FeO, CoO, NiO and CuO are fit to XAS R_0 is obtained by averaging the distance between absorber and ligand atoms, $\varepsilon_0, \delta E, \phi$ and r_{00} were calculated using eq. (4.17), (4.15) and (4.19).

	U(eV)	t(eV)	R_0(a.u.)	ε_0	δE(eV)	$\phi = \theta$	r_{00}
MnO	11.0	1.6	2.23	3.16	6.3	0.26	3.0
FeO	9.7	1.9	2.14	3.83	6.1	0.33	1.7
CoO	8.9	1.5	2.13	4.38	5.3	0.29	2.3
NiO	10.6	2.1	2.08	3.65	7.4	0.34	1.5
CuO	13.0	1.5	2.23	3.03	7.2	0.21	5.4

Following LGH we also represent the local potential by

$$V_{sc}(\mathbf{r}) = \begin{cases} [-V_{3d}(r) + \frac{1}{R_0}]/\varepsilon_0 & r < R_0 \\ 0 & r > R_0. \end{cases} \tag{4.5}$$

Here $V_{3d}(r)$ is the potential of the d level calculated using 3d wave functions from `FEFF9`,[83] and the $1/R_0$ term crudely represents the potential of the ligand charge shell. $V_{3d}(r)$ potential was calculated with a core-hole, same potential calculated without core hole is closely similar. The constant ε_0 is chosen so that $U = -V_{sc}(0)$. Since $V_{3d}(0) \gg 1/R_0$, $\varepsilon_0 \approx V_{3d}(0)/U$. The parameters used in our sample calculations are given in table 4.1, with details on how they are obtained given in section 4.2.2. The scattering potentials $V_{sc}(r)$ are shown in fig. 4.2. These parameters are calculated using `FEFF` wave functions and values of R_0 from table 4.1 which represent distances between absorber and ligand atoms. The scattering potential is set to zero beyond R_0.

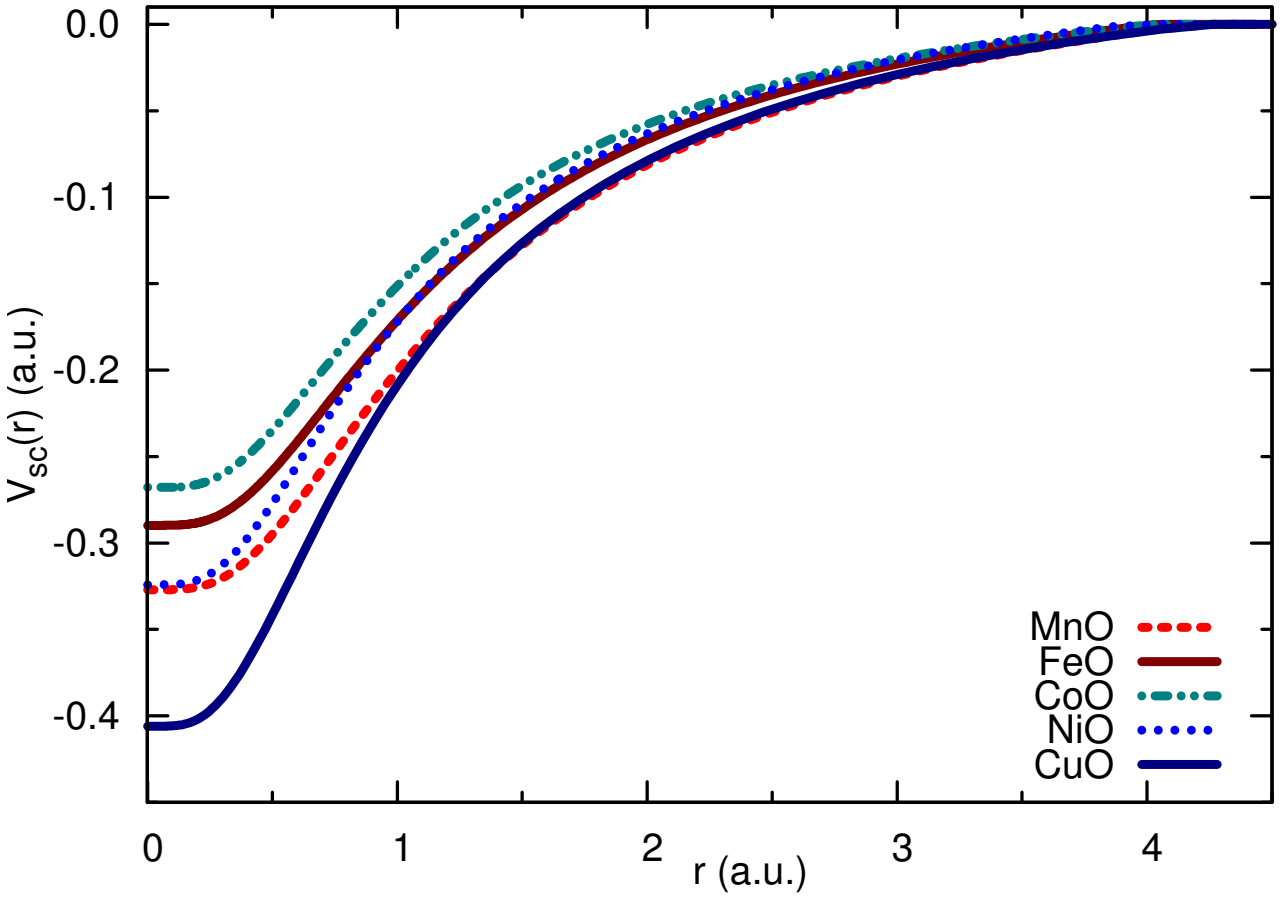

Figure 4.2: Photoelectron scattering potentials $V_{sc}(r)$ see Eq. 4.5 for MnO, FeO, CoO, NiO and CuO

The matrix elements $V_{kk'}$ of the scattering potential are presented in Fig. 4.4 and compared with the analytic form discussed in LGH. Note that the numerical calculations are in good agreement with the analytical form except at low energies. The matrix elements $V_{kk'}$ for NiO and CoO are similar, due to the similarity of their scattering potentials $V_{sc}(r)$ (see Fig. 4.2). The radial transition matrix elements (see Fig. 4.3) are given by

$$M_{ck} \propto \sqrt{(\varepsilon_k - \varepsilon_c)} \int \mathrm{d}r\, r^2\, \psi_c(r)\, r\, u_{lk}(r), \tag{4.6}$$

where the radial wave functions $u_{lk}(r)$ are

$$u_{lk}(r) = \begin{cases} \tilde{u}_{lk}(r) & r < r_{mt} \\ \sqrt{2/L}\,[h_l(kr)e^{i\delta_{lk}} + c.c.] & r > r_{mt}, \end{cases} \tag{4.7}$$

and are calculated from subroutines in `FEFF9`[83]. Here, $\tilde{u}_{lk}(r)$ corresponds to the regular solution for angular momentum l at the origin, matched to solutions beyond r_{mt} in terms of spherical Hankel functions $h_l(kr)$, and δ_{lk} is the partial wave phase shift. Boundary conditions within a sphere of large radius $L = 40$ a.u. and an exponential grid with 0.01 a.u. step are used to obtain smooth results for the matrix elements.

4.2.2 CT PARAMETERS

Due to the simplicity of our model, there is no simple correspondence to the tight-binding parameters of a more realistic system. Nevertheless, the charge-transfer parameters of the three level system can be chosen to fit the main ($s = 1$) and satellite ($s = 2$) peaks in XPS experiments. The energy difference between these two peaks is defined as δE, while w_1/w_2 refers to the ratio of intensity of the main to the satellite peak and is given by Eq. (4.17) and (4.18). For NiO we fit these quantities to the 1s XPS edge data (Fig. 4.6 a) of Calandra et al.[17] Due to band splitting, the main peak is bimodal and asymmetric. Thus we simply fit its strength to two asymmetric Fano-lineshapes, while the satellite peak was fit with a single Gaussian. The plasmon peaks at about -25 eV are ignored, since they are implicitly included in $\tilde{\mu}$ and are only important at high energies. Solving Eq. (4.17) and (4.18) yields estimates for U and t. For CoO we used 3s XPS edge data,[75] since 1s results are not available. As in the case of NiO, we also subtracted the peak near -12 eV. We fit the main peak with a single asymmetric Fano-lineshape while the satellite was fit with a single Gaussian. In the case of MnO, FeO and CuO we used 3s XPS experimental data [34] as 1s are not available. For MnO and FeO we fit the main and satellite peaks with asymmetric Fano-lineshapes, following the same procedure for estimating parameters for the LGH model described above. We also tried to estimate the hopping parameter t from the width of the

projected d-density of states of the absorber as obtained from the `ldos` module in `FEFF`, but these results only agree qualitatively with the fits given in Table 4.1.

4.3 Results

Assuming isotropic XPS and summing over all directions, the XAS is simply related to the XPS photocurrent

$$\mu(\omega) = \sum_{sk} J_k^s(\omega) \propto \sum_s \frac{1}{k_s} |M(s, k_s)|^2, \tag{4.8}$$

where $M(s,k) \equiv \langle \Psi_{sk}^f | \Delta | \Psi_0 \rangle$ is obtained from the resolvent formula in Eq. (4.4)), where $k_s = \sqrt{2(\omega + E_0 - E_s)}$, and E_s are obtained from eigenvalues of the model Hamiltonian from Eq. (4.17). To simplify the discussion of the dynamical effects, we introduce the ratio $R_s(\omega)$ between the calculated photocurrents at a given photon energy, and the photocurrent $J_i^0(\omega)$ in the sudden approximation,

$$R_s(\omega) = \sum_k J_k^s(\omega) / J_s^0(\omega). \tag{4.9}$$

where the sudden-limit is

$$\begin{aligned} J_s^0(\omega) &= \frac{1}{k_s} \sum_k |M_{ck} w_s|^2 \delta(\omega - \epsilon_k + E_0 - E_s) \\ &= w_s^2 \, \tilde{\mu}(\omega - E_s), \end{aligned} \tag{4.10}$$

and $J_k^s(\omega)$ is given by Eq. (4.16). Here $\tilde{\mu}$ implicitly includes the effects of long-range inelastic losses in the XAS, M_{ck} is given by Eq. (4.6), and the weights w_s are given by Eq. (4.18). Thus the total XAS can be written as

$$\begin{aligned} \mu(\omega) &= \sum_{sk} |M(s,k)|^2 \delta(\omega - \epsilon_k + E_0 - E_s) \\ &= \sum_s J_s^0(\omega) R_s(\omega). \end{aligned} \tag{4.11}$$

Fig. 4.5 shows $R_s(\omega)$ for a number of TMOs. Note that there is a significant "overshoot" at low energies, while above about 30 eV, $R_s(\omega)$ tends to the sudden limit. This behavior arises from the interplay between intrinsic and extrinsic effects represented by the first and second terms in Eq. (4.4). The overshoot is a fairly small correction to the adiabatic limit, since the interaction between the scattered electron and the core hole is relatively small as a result of screening of the core hole by the charge transfer process. In our calculations, $\tilde{\mu}(\omega - E_s)$ is the total XAS spectrum calculated from `FEFF9`.[83] Thus one finally obtains the convolution formula of Eq. (4.1) with the spectral function $A_L(\omega, \omega^{'})$ given by

$$A_L(\omega, \omega') = \frac{1}{D} \sum_s w_s^2 R_s(\omega) \delta(\omega' - \omega_s, \tag{4.12}$$

where the normalization constant $D = \sum w_s^2 R_s(\omega)$. Calculations of $R_s(\omega)$ were performed using the resolvent formula (see Eq. (4.4)) with a Hamiltonian matrix with indices s, k and 80 k-points of dimension 160×160. Calculations of directionally averaged XAS for a number of TMOs are presented in Fig. 4.7. Calculations were carried out using both our two step model and, for comparison, a convolution with a multiple Lorentzian fit to the XPS as in Calandra et al.[17] Results are presented in Fig. 4.7 show that the two methods are numerically similar, with small differences arising from differences in broadening between XAS and XPS experiments. In the case of method described by Calandra et al. resulting spectra depends on XPS experimental broadening. That is the reason why in case of FeO and MnO, one can notice that broadening is overestimated. In our two step approach,

to account for differences in broadening ($\Delta\Gamma$) of the main and satellite peaks in CoO, which can be seen in XPS experimental data (see Fig. 4.6), broadening of the satellite peak was carried out with value $\Delta\Gamma = 0.4$ eV. Results for CuO and NiO are presented in Fig. 4.7 (a, b) using values of $\Delta\Gamma$ 0.6 and 1 eV respectively. Calculations of XAS for FeO and MnO are presented in Fig. 4.7 (d, e), values of $\Delta\Gamma$ are 1.46 and 1.1 eV respectively. In case of NiO Fig. 4.7 (b) one can notice difference in position of the satellite peaks between two methods, this is due to the structure of the main peak in XPS see (see Fig. 4.6 (a)). The effect of CT satellites are clearly seen in the Fig. 4.7. In particular the satellite peak transfers oscillator

strength from the main peak and fills in missing spectral weight in one particle calculations above the main peak. On the other hand, the treatment here is only semi-quantitative as the satellite peaks seen in the experimental XAS are more broadened than can be represented by a two-peaked spectral function. For MnO, the parameters for the LGH model were fit to the XPS experiments shown in Fig. 4.6, and are given in Table 4.1. Following Calandra et al. we calculate K edge of CuO polarized XAS and choose to present calculation for couple of angles together with experimental data,[15] the three angles specified in Fig. 4.9 are the rotation angles of goniometer.[15] In both calculations, presented in Fig. 4.9 we used 1s XPS CuO data[1] referenced in Caladra et al. From the Fig. 4.9 one can see clear improvement of pre-edge peaks, the sattelite peak however are overestimated, this is due to the simplicity of our model which does not take into account any polarization dependent effects. For the other polarization directions, discussed in Clandra et al., we get very similiar results, with small differences due to the single particle spectra. From the Fig. 4.9 one can see that the two methods give numerically similar result, this is due to the broadening of XPS,[1] which matches broadening in XAS.[15]

4.4 Some more details on Local model

Here we briefly summarize the details of the sudden approximation for the local model, closely following the methodology and notation of LGH.[61] The initial state $|\Psi_0\rangle$ is the ground state of $\mathcal{H}_0$ with $n_c = 1$

$$|\Psi^0\rangle = -\sin(\theta)|\psi_c\rangle|\psi_a\rangle + \cos(\theta)|\psi_c\rangle|\psi_b\rangle. \tag{4.13}$$

The final states $|\Psi^f_{sk}\rangle$ are given by Eq. (4.4), where $|\psi_1\rangle$ and $|\psi_2\rangle$ are the eigenstates of $\mathcal{H}_0$ with $n_c = 0$, and can be parameterized conveniently in terms of mixing angles θ and ϕ

$$|\psi_s\rangle \quad = \quad \begin{cases} \cos(\phi)|\psi_a\rangle - \sin(\phi)|\psi_b\rangle, \ \ s = 1, \\ \sin(\phi)|\psi_a\rangle + \cos(\phi)|\psi_b\rangle, \ \ s = 2, \end{cases} \tag{4.14}$$

where

$$\begin{aligned} \tan(2\theta) &= 2t/(\varepsilon_a - \varepsilon_b + U), \\ \tan(2\phi) &= 2t/(\varepsilon_b - \varepsilon_a), \end{aligned} \tag{4.15}$$

and a and b correspond to the metal d- and oxygen p-levels. The photocurrent is then calculated using[3]

$$J_k^s(\omega) = |\langle \Psi_{sk}^f | \Delta | \Psi^0 \rangle|^2 \delta(\omega - \varepsilon_k + E_0 - E_s). \tag{4.16}$$

The spectrum of the model Hamiltonian is characterized by the parameters

$$\begin{aligned} E_s &= \frac{1}{2}(\varepsilon_a + \varepsilon_b) \mp \delta E/2, \\ \delta E &= \sqrt{(\varepsilon_a - \varepsilon_b)^2 + 4t^2}. \end{aligned} \tag{4.17}$$

We consider only the symmetric case with $\varepsilon_a = \varepsilon - U/2$ and $\varepsilon_b = \varepsilon$. The weights of main and satellite levels are then

$$w_s = \begin{cases} -\sin(\phi + \theta), & s = 1 \\ \cos(\phi + \theta), & s = 2. \end{cases} \tag{4.18}$$

In the sudden[61] limit there is no interaction between the photoelectron and the electron on the outer level, so that the ratio of the main to the satellite peak intensities is

$$r_{00} = \lim_{\omega \to \infty} \frac{\sum_k J_k^2(\omega)}{\sum_k J_k^1(\omega)} = \cot^2(\phi + \theta). \tag{4.19}$$

where r_{00} represents Taking into account the analytical form of dipole matrix elements in the sudden[61] limit one obtains

$$r(\omega) = r_{00} \left[\frac{\tilde{\omega}}{\tilde{\omega} + \delta E} \right]^{\frac{3}{2}} \left[\frac{1 + (\tilde{\omega} + \delta E)/\tilde{E}_d}{1 + \tilde{\omega}/\tilde{E}_d} \right]^2, \tag{4.20}$$

here $\tilde{E}_d$ is characteristic energy of the analytic dipole matrix element M_k, $\tilde{\omega} = \omega - \omega_{th} = \omega - (E_2 - E_1)$ and δE is given in (4.17).

4.5 Exact treatment

Finally, from Eq. (4.4) the transition matrix element $\langle \Psi^f_{sk}|\Delta|\Psi_0\rangle$ for the two state model is

$$M(s,k) \equiv \langle \Psi^f_{sk}|\Delta|\Psi_0\rangle = \langle\psi_k|\langle\psi_s|\Bigg[1 + V\frac{1}{\epsilon_k + E_s - \mathcal{H}_0 - T - V + i\eta}\Bigg]\Delta|\Psi_0\rangle. \tag{4.21}$$

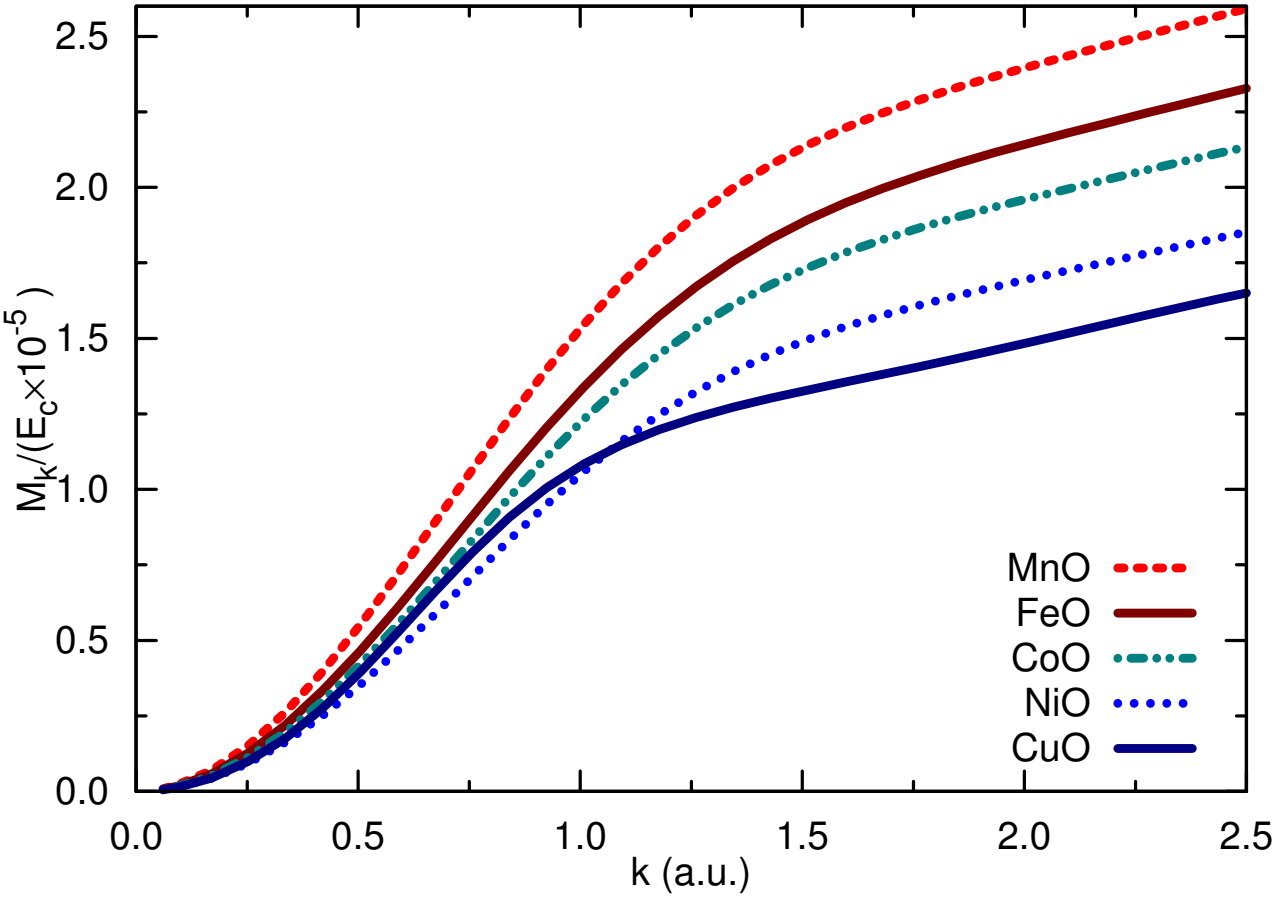

Figure 4.3: Dipole matrix elements M_{ck} in Eq. (4.6) for MnO, FeO, CoO, NiO and CuO, calculated from `FEFF`.

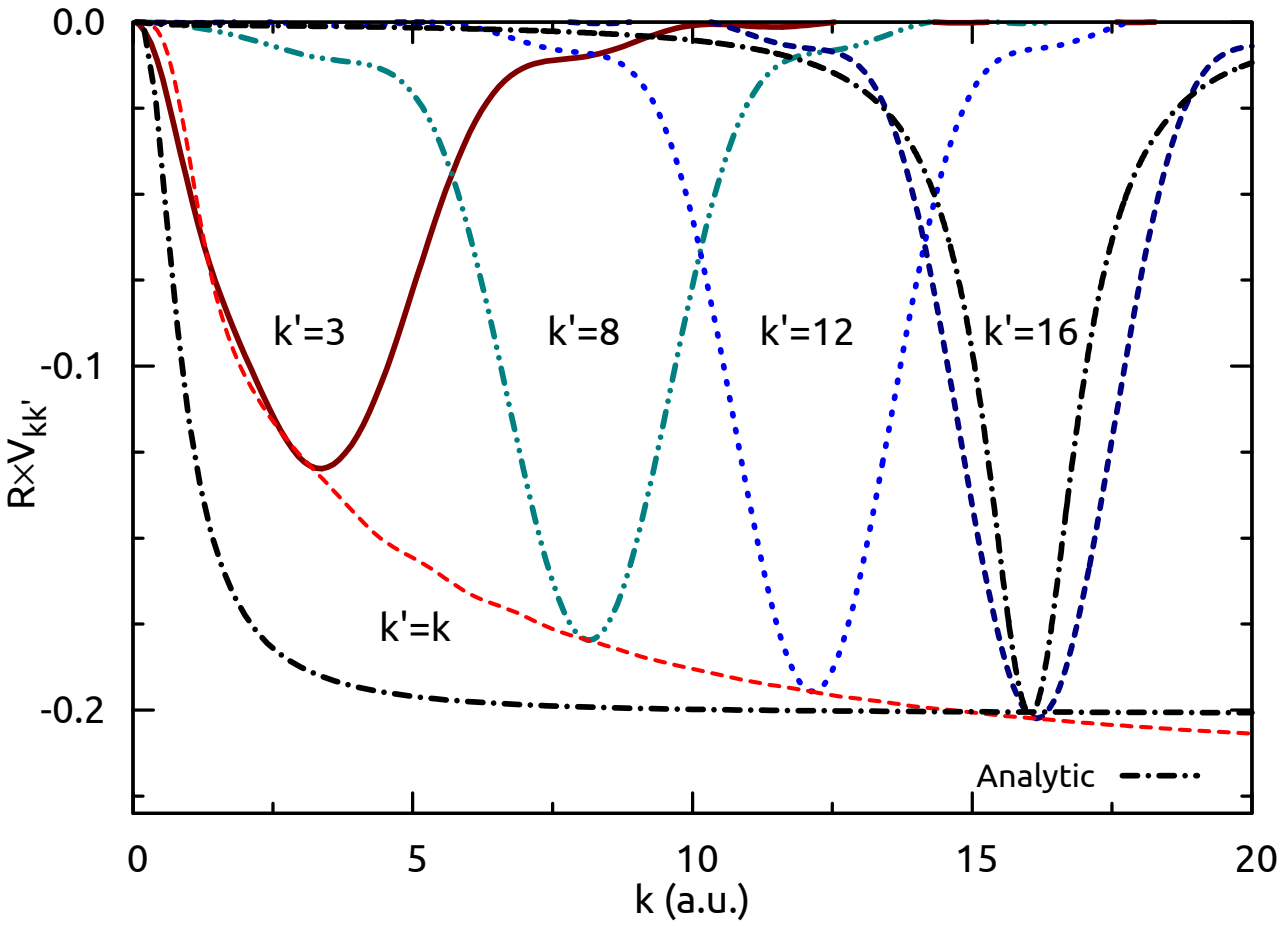

Figure 4.4: Matrix elements $V_{kk'}$ for CoO. For comparison, the LGH analytical form [Ref. [61]] of $V_{kk'}$ for $k = k^{'}$ and $k^{'} = 16$ au is also shown, calculated with parameters: $\tilde{V} = -0.79$ a.u., $R_{sd} = 1.01$ au and $R_s = 3.97$ au.

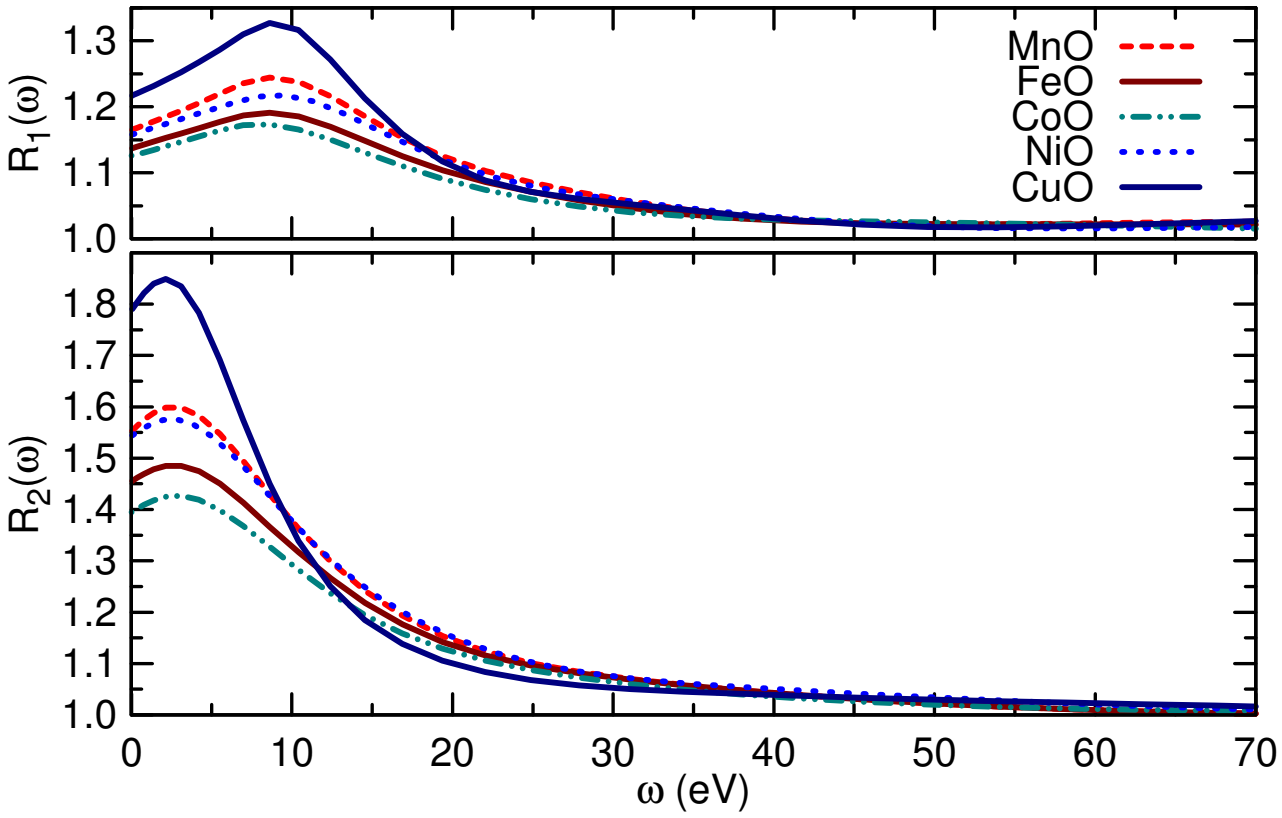

Figure 4.5: Ratios between full photocurrent Eq. (4.16) and photocurrent in the sudden limit Eq. (4.10) for the main $R_1(\omega)$ (top) and satellite $R_2(\omega)$ (bottom) peaks Eq. (4.9) for MnO, FeO, CoO, NiO and CuO. Note the significant dynamic variation at low energies while the adiabatic limit is reached around 30 eV.

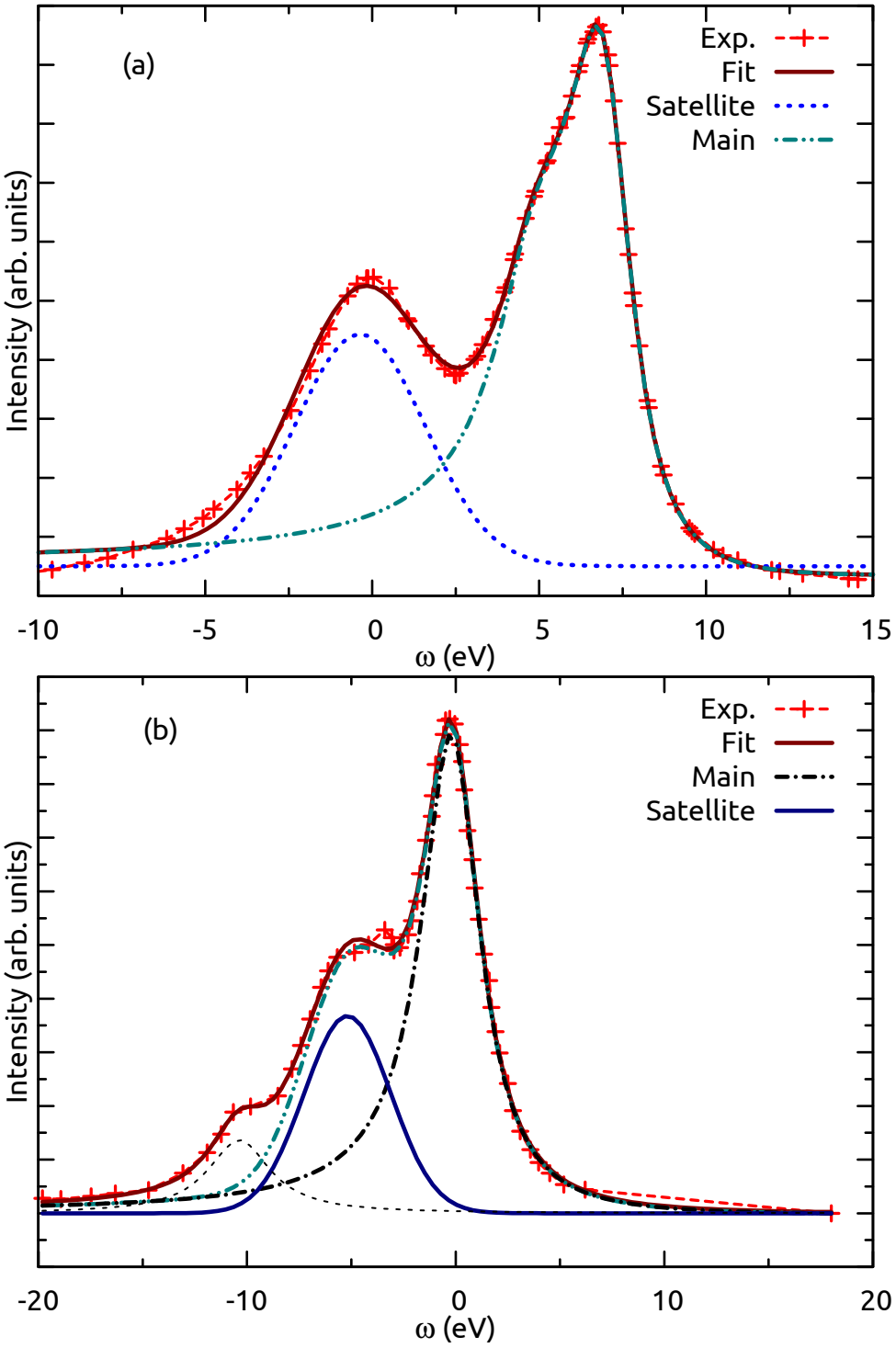

Figure 4.6: (a) Recent XPS experiment [Ref. [17]] of NiO 1s edge, with subtracted plasmon background. (b) XPS spectra [Ref. [75]] for CoO.

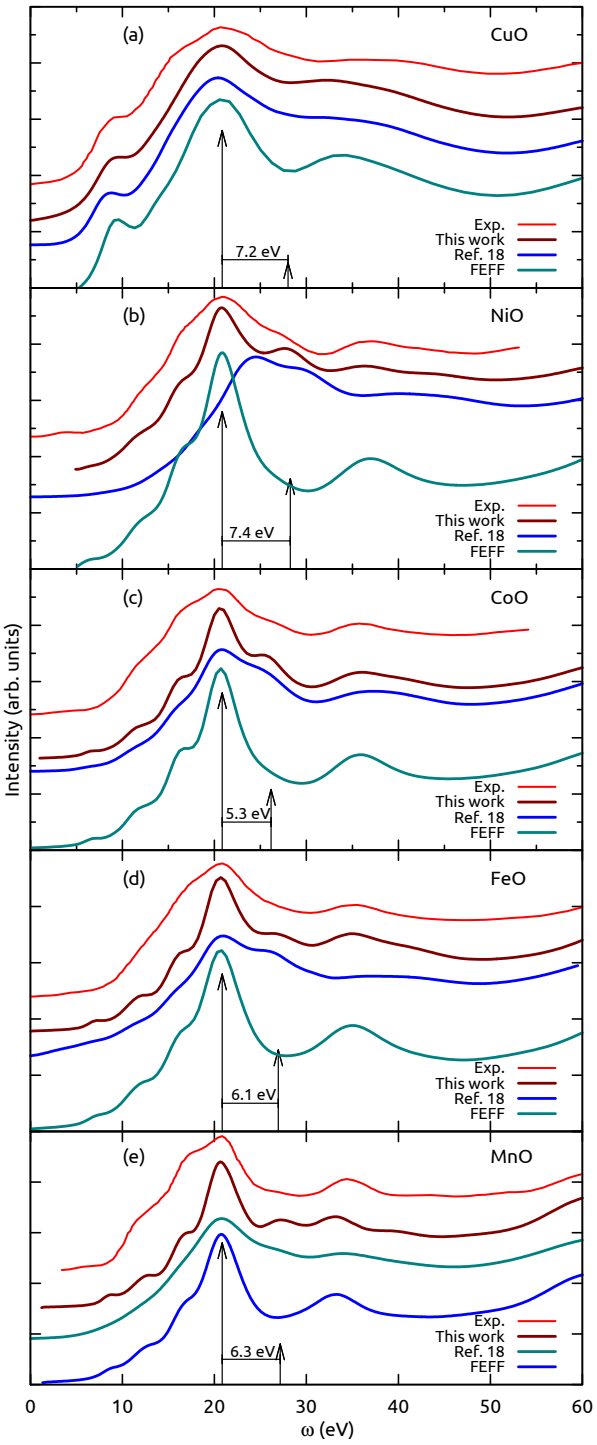

Figure 4.7: Comparison of the experimental XAS [Ref. [17, 103]], convolution using three level model (LGH), to single particle (FEFF) calculation and Calandra approach [Ref. [17]] for (a) MnO, (b) FeO, (c) CoO, (d) NiO and (e) CuO. Main and satellite peak position and intensities are shown with arrows. Intensity ratio is equal to r_{00}, see Table 4.1

.

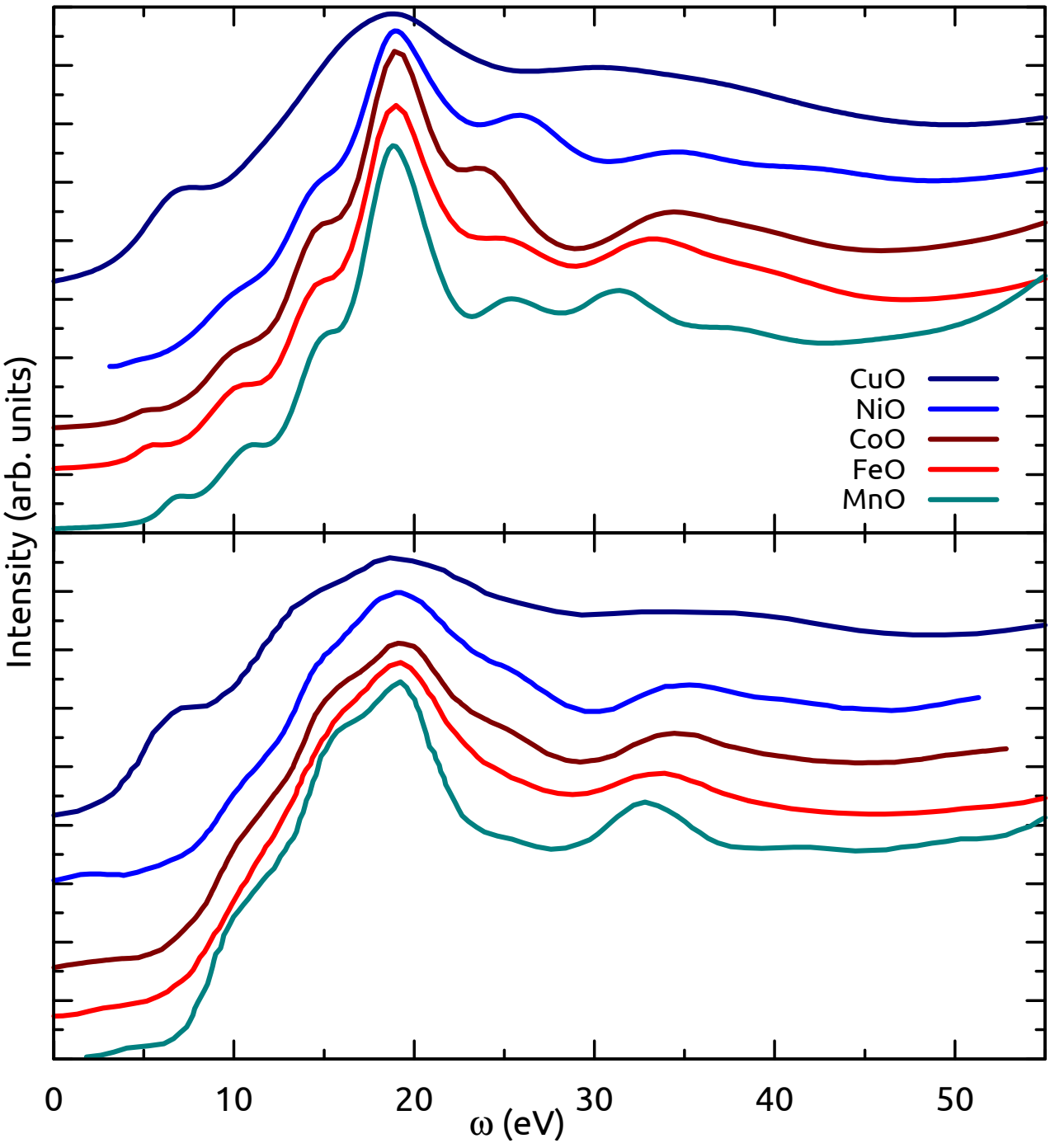

Figure 4.8: Comparison of spectra calculated using three level system (LGH) (top) and experimental [Ref. [17, 103]] XAS data (bottom) for a number of TMOs.

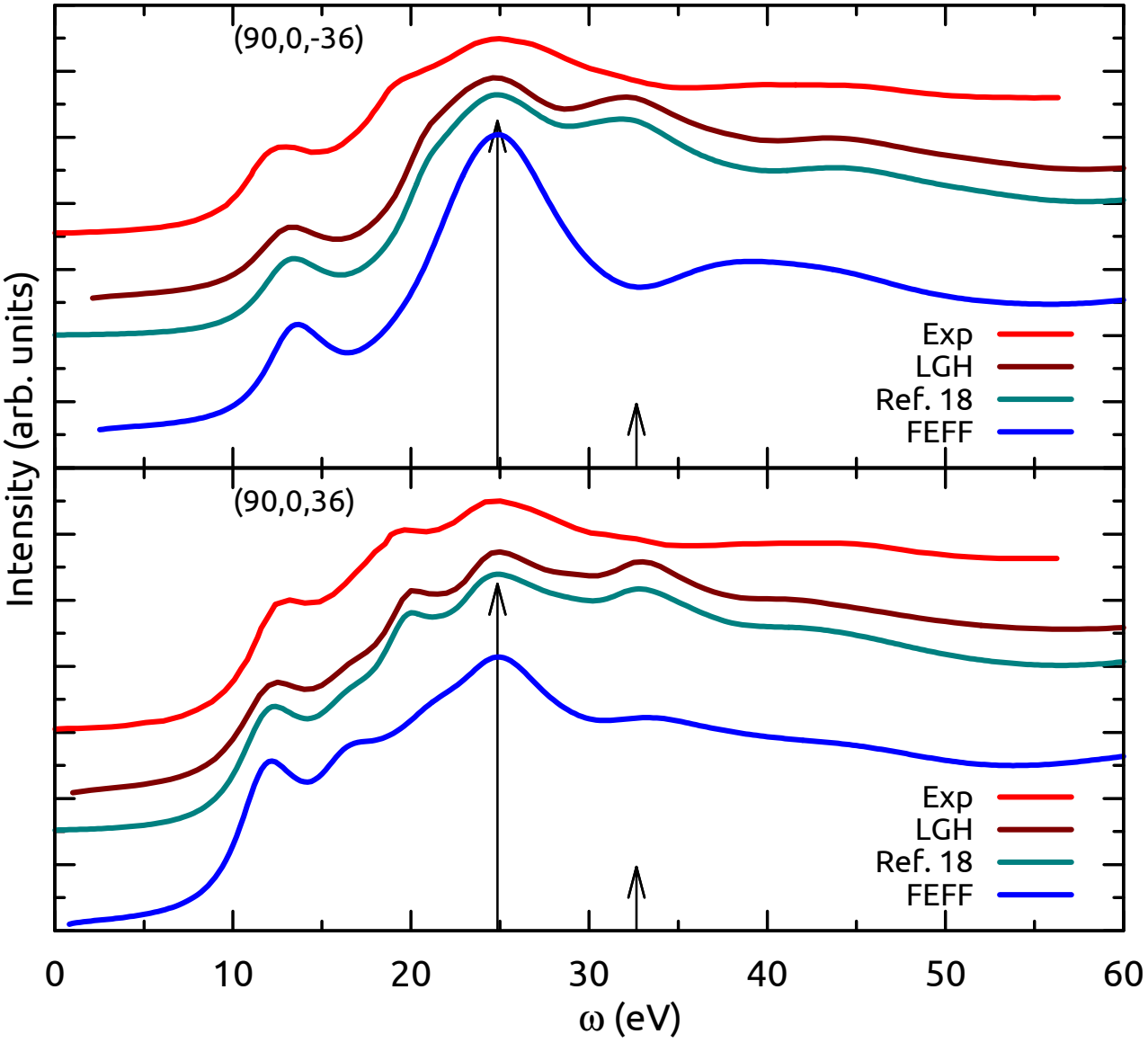

Figure 4.9: Comparison of the experimental polarized K edge XAS [Ref. [15]] of CuO, convolution using three level model (LGH), to single particle (FEFF) calculation and Calandra approach [Ref. [17]] for two polarizations. In the parenthesis are the angles of rotation of goniometer.

Chapter 5

FINITE TEMPERATURE CALCULATIONS OF THE COMPTON PROFILE OF BE, LI AND SI.

We present real space, full scattering Green's function calculations of the Compton profile for thermally disordered Beryllium, Lithium and Silicon. This is in principle a qualitative test case for study of final temperature effects on valence electron ground state. We find good agreement with experimental results and confirm that thermal disorder and thermal expansion are important in Compton profile calculations and both need to be taken into account in studies of valence electron ground state.

The high resolution Compton profile (CP) is widely used to study the electronic and chemical properties of materials.[13, 22, 39, 40, 73, 89] For example, for metallic systems it can be used to probe the Fermi surface and its behavior as a function of temperature and density. In materials which undergo phase transitions, the CP has proven a useful technique to distinguish different phases. For instance, in the case of the solid to liquid phase transition in Si, where it revealed persistent covalent bonding in the liquid phase.[72, 73]

A variety of methods have been used previously to compute CPs. For example, band structure methods have been widely used for periodic systems, but their application to studies of electronic properties in thermally disordered systems, such as the Si case mentioned above, is more challenging. Early theoretical calculations of CP in thermally disordered materials used Boltzmann statistical averages over atomic configurations generated from Gaussian distributions along the harmonic phonon modes. This approach was used to study thermal disorder effects in Li with a frozen, eight-atom supercell, predicting a broadening of the CP.[29] In contrast, a later study found a narrowing of the CP for Al, Li and Be as a result

of thermal expansion.[21, 97, 98] Another widely used theoretical method for the calculation of CPs is the KKR approach,[51] which has been applied to systems such as substitutionally disordered alloys. KKR uses a coherent potential approximation, where the disordered alloy is replaced with an ordered system and an averaged effective site potential.[11] Although this simplification is in principle a limitation of this method, excellent agreement with experiment was reported in a study of LiF where the quasiharmonic approximation with zero point motion was used.[31] The same KKR approach with correlation corrections[59] was used in the study of Li.[90, 92] Discrepancies between theory and experiment were attributed to the self-energy effects connected to the excitation of plasmaron modes. Another approach used to study temperature dependence of the CP of Li is modified augmented plane-wave method,[16] theory moderately well predicts temperature dependents of experiment. GW approximation with effects of electron correlation produced results which are very close to the experiment for Li.[57] More recent studies have used Molecular Dynamic (MD) simulations to generate structural configurations for the different thermally disordered materials, in an attempt to obtain better electronic structure properties. In the case of water ice,[87, 88] the CP anisotropy was studied using maximally localized Wannier functions, coupled with Car-Parrinello molecular dynamics[18] (CPMD) simulations. Other recent studies used the same CPMD approach to investigate the solid-liquid phase transition in Si and B,[72, 73] where differences in CP between the two phases were attributed to the dominant role of covalent bonds breaking over the thermal disorder of Si sites.[73] In contrast to the behavior reported in the cases of Al, Be and Li,[21, 97, 98] the CP width of Si did not change much in the solid to liquid phase transition as density increased.

Given the seemingly contradictory effects of disorder, expansion and choice of methodology discussed above, in this paper we report *ab initio* calculations of the CP and electron momentum density (EMD) in the impulse approximation (IA), using real space full multiple scattering (FMS) techniques. We focus on thermally disordered Be, Li and Si, where the thermally disordered structures are generated using finite temperature *ab initio* MD simulations, and compare our results to experiments[43, 73, 98] and other theories, including

recent CPMD calculations.[72, 73] The real space Green's function (RSGF) approach has the advantage of not being limited to crystalline systems, as is the case for band structure calculations. It allows for the efficient study of the electronic properties of materials as a function of thermal disorder and expansion, making it ideally suited to investigate liquids and solids in the Warm Dense Matter (WDM) regime.

5.1 Compton Scattering Theory

X-ray Thomson scattering (XRTS) in the limit of large momentum transfer is directly related to the CP. Experimentally, the system is probed by a narrow x-ray beam incident on the sample and the intensity of scattered radiation is measured as a function of energy cross section and scattering angle. In this two-photon process, the double differential scattering cross section (DDSC) $d^2\sigma\,(\mathbf{q},\,\omega)/d\Omega d\omega_2$ is measured, where Ω is the measured solid angle and ω_2 is the detected energy of radiation, $\mathbf{q}$ and ω are, respectively, the momentum and energy transfered to the sample. Thus, the DDSC determines the relative probability of transferring momentum q and energy ω to the sample in the scattering process. The incident energy in the XRTS experiment is far larger than any binding energy in the sample. In this regime, first order perturbation shows that the main contribution arises from the dominant $\mathbf{A}^2$ term in the interaction Hamiltonian, where $\mathbf{A}$ is the vector potential.[93] In the non-relativistic limit, the DDSC is given by

$$\frac{d^2\sigma}{d\,\Omega\,d\,\omega_2} = \left(\frac{d\,\sigma}{d\,\Omega}\right)_{\mathrm{Th}} S(\mathbf{q},\,\omega). \tag{5.1}$$

Here the Thompson scattering cross section is

$$\left(\frac{d\,\sigma}{d\,\Omega}\right)_{\mathrm{Th}} = \frac{\omega_2}{\omega_1} r_0^2 (\hat{\epsilon}_1 \cdot \hat{\epsilon}_2^*)^2, \tag{5.2}$$

where ω_1 is incident energy, and the dynamic structure factor is

$$S(\mathbf{q}, w) = \sum_F \left| \langle F | \sum_j \exp(i\mathbf{q}\cdot\mathbf{r}_j) | I \rangle \right|^2 \delta(E_F - E_I - \omega). \tag{5.3}$$

In the above equations, $r_0 = \alpha^2$ is the classical electron radius; $\hat{\epsilon}_1$ and $\hat{\epsilon}_2$ are incoming and outgoing photon polarizations, respectively; $|I\rangle$ and $|F\rangle$ are the initial and final states with energies E_I and E_F; and $\mathbf{r}_j$ is the position operator for the jth electron. Throughout this work we adopt atomic units where $\hbar = m = e = 1$ and $c = 1/\alpha$.

For large energy transfer relative to the binding energy of a given electronic orbital of the target species, *i.e.* in the impulse approximation (IA), Platzman and Eisenberg have shown that the XRTS can be described approximately as a Doppler-broadened CP.[30] The energy change is given by

$$\omega = E_f - E_i = \frac{1}{2}(\mathbf{p} + (\mathbf{k_1} - \mathbf{k_2}))^2 - \frac{\mathbf{p^2}}{\mathbf{2}} = \frac{\mathbf{|q|^2}}{\mathbf{2}} + \mathbf{q}\cdot\mathbf{p}, \tag{5.4}$$

where $\mathbf{p}$ is the initial momentum of the electron and $\mathbf{q}$ is the momentum transfered to it. The CP factor $J(p_q)$ has a direct relation to the dynamic structure $S(\mathbf{q},\omega)$, given by

$$S(\mathbf{q},\omega) \quad = \quad (1/q)J(p_q), \tag{5.5}$$

where

$$J(p_q) \equiv \int d^3\,p\,\rho(\mathbf{p})\,\delta(p_q - (\omega/q - q/2)). \tag{5.6}$$

Here $\rho(\mathbf{p})$ is the EMD with momentum $\mathbf{p}$, and the δ function ensures energy conservation with $p_q \equiv \omega/q - q/2$. The EMD can be calculated from the real space density matrix by taking the Fourier transform

$$\rho(\mathbf{p}) = \int d^3r\,d^3r' e^{\imath\mathbf{p}\cdot(\mathbf{r}-\mathbf{r}')}\rho(\mathbf{r},\mathbf{r}'). \tag{5.7}$$

In the RSGF approach, the density matrix is given by

$$\rho(\mathbf{r},\mathbf{r}') = -\frac{2}{\pi}\mathrm{Im}\int_{E_c}^{\infty} dE\,G(\mathbf{r},\mathbf{r}',E)f_T(E), \tag{5.8}$$

where $f_T(E)$ is the Fermi function, and the real space Green's function $G(\mathbf{r},\mathbf{r}',E)$ in the muffin-tin approximation is given by[80]

$$\begin{aligned} G(\mathbf{r},\mathbf{r}',E) = -2k\Bigg[&\delta_{n,n'}\sum_L H^E_{Ln}(\mathbf{r}_>)\bar{R}^E_{Ln'}(\mathbf{r}_<) \\ &+ \sum_{L,L'} R^E_{Ln}(\mathbf{r}_n)\,e^{\imath\delta_{Ln}} g^E_{Ln,L'n'} e^{\imath\delta_{L'n'}}\bar{R}^E_{L'n'}(\mathbf{r'}_{n'})\Bigg]. \end{aligned} \tag{5.9}$$

Additional details of this formulation have been reported elsewhere.[66]

In the homogeneous case, the EMD is directly related to the CP averaged over the angles:

$$\rho(p) = -\frac{1}{2\pi p}\frac{d}{dp_q}J(p_q)|_{p_q=p}, \tag{5.10}$$

and the following important sum rule holds for N, the number of the valence electrons per unit cell:

$$N = 2\pi \int dp\, p^2\, \rho(p) = 2\pi\, \rho_0 \int dp\, p^2\, f(p). \tag{5.11}$$

5.2 Calculation Details

5.2.1 MD calculations

All MD simulations were performed with VASP (Viena Ab Initio Simulation Package).[54, 55] PAW potentials[14, 56] were used throughout this work, with plane wave cutoffs of 308.8 eV for Be, 140 eV for Li, and 245.7 eV for Si. For Be, the simulation cell consisted of $4 \times 4 \times 6$ repetitions of the orthogonal 4-atom conventional cell for a total of 384 atoms. For Li, the simulation cell had 250 atoms originating from $5 \times 5 \times 5$ repetitions of the orthogonal 2-atom conventional cell. Finally, the Si simulation cell was generated from $4 \times 4 \times 4$ repetitions of the 8-atom conventional cell. Temperature was controlled using the Nosé-Hoover thermostat.[70]

For the range of temperatures used in this work, the thermostat s_{mass} parameter was adjusted to match the characteristic phonon frequencies, which range from 10 – 18 THz in Be,[96] 2 – 8 THz in Li,[101] and 4 – 15 THz in Si.[27] The Be and Li MD runs used a time step of 2 fs and, for temperatures below melting, were thermalized for 8 ps. For solid Si, the time step was reduced to 1 fs and the runs were thermalized for 6 ps. For liquid Si, the system was first run at 3000 K for about 1.7 ps to ensure that the structure would be in the liquid phase. Next, the temperature was brought down to 1757 K in about 0.4 ps and thermalized for about 1.2 ps until equilibrium was reached. Given that the Fermi energies of these systems are much larger than the temperatures studied here, we set the electronic temperature to 0 K in both the MD dynamics described above and the CP calculations described in the following section.

5.2.2 Compton Profile calculations

The Compton profile calculations were performed using the real space code FEFF, which does not require periodicity of the system. It implements built-in self-consistent potentials and quasiparticle effects and is widely used for x-ray spectroscopy. For a detailed description of the methodology see Refs. [83] and [66]. To get a sampling of the disordered system, we used 24 MD snapshots at temperatures above 400 K and 12 MD snapshots at temperatures below 400 K. We chose snapshots 50 to 70 fs apart in the thermalized regime. The snapshots were used to construct cluster models for the FEFF RSGF calculations, where the radii for the SCF potentials and FMS Green's functions parts of the calculations were set so that the clusters would have $\sim$ 170 atoms, respectively. Number of atoms in the cluster gives the size of the cluster which sets the uncertainty of momentum. I choose uncertainty in the momentum equal to 0.1 a.u.. These calculations used 13 unique self-consistent muffin-tin potentials, which roughly account for the absorber and its nearest neighbors. All other atoms outside the first coordination shell shared the same muffin-tin potential. The final CP was then calculated as an average over the CP from each structural snapshot, and the vertical bars displayed in this paper correspond to errors in the mean unless otherwise stated. One of the most important parameters for convergence test is angular momentum l in fig. 5.1, I present results of convergence test for Be. I use value of angular momentum $l = 3$ for simplicity's sake.

One of the most important parameters determining the quality of the calculated CP is the maximum angular momentum $l_{\max}$ used in Eq. (5.9). Given that the computational demand of the simulations depends greatly on this parameter, we looked for the lowest possible value that ensures converged results. Fig. 5.1 shows the percent error between the CPs computed with increasingly larger $l_{\max}$ ($l_{\max} = 1 - 6$) to those obtained with a large $l_{\max} = 7$. In the case of Li and Be, $l_{\max} > 3$ gives errors lower than 30% for the range of momentum values of interest. For Si, the errors are still larger for $l_{\max} \leq 4$, thus we use $l_{\max} = 5$.

Finally, the CP is rescaled to give the proper number of valence electrons (Eq. 5.11),

although the scaling is usually very small (1%) and arises from momentum density that lies beyond the integrations bounds. This causes small errors (usually ∼5%) in the numerical integration of the electron density matrix $\rho(\mathbf{r}, \mathbf{r}')$ used to obtain the CP with Eq. (5.6). The CP is broadened by $\pi/z'_{\max}$ due to the convolution theorem, where $z'_{\max}$ is set by the FMS radius. In the cases of Be and Li, the integration bound $z'_{\max} \approx 30$ a.u. which results in a broadening on the order of 0.1 a.u..

5.3 Results and Discussion

5.3.1 Structural Properties

The pair distribution functions $g(r)$ for Be, Li and Si as a function of temperature and density are presented in Fig. 5.2. The density of each system was adjusted to that of the specific temperature of the simulation, in order to account for thermal expansion. In Fig. 5.2(a) pair distribution function for Be at 300 K has all the shell peaks pronounced. At the 650 K and 850 K the second shell peak is smeared out, which should be observed in the CP. For Be we performed a test simulations at ambient density and temperatures up to 2000 K, phase transition to the liquid phase occurs between 1800 K and 2000 K. With expanded volume the melting point in the MD simulations should be close to the experimental value for Be. In the case of Li Fig. 5.2 (b), the melting transition occurs at lower temperature than in the case of Be and Si, due to smaller Debye temperature. The second shell peak which is observed at 95 K is smeared out at higher temperatures and this change in the structure should contribute to the valence electron state probed in CP experiment. Results of MD simulations of Si are presented in the Fig. 5.2 (c). Pair distribution function at 3000 K is a preliminary run to ensure that the starting configuration for simulation at the temperature of 1787 K is in liquid phase. The density at which the system was simulated corresponds to the density of Si at 1787 K.[73]

5.3.2 Role of disorder vs. thermal expansion

Beryllium

Fig. 5.3 presents the CP profile of thermally disordered Be at finite temperature. For comparison we also show experimental results from Huotari *et al*,[43] as well as their theoretical results[43] obtained using pseudopotential theory with disorder. The coefficients for these pseudopotentials were fit to match the experiment.

For temperatures 650 K and 850 K, our calculations agree well with the experiment. The position of the "umklapp" peak around 2.4 a.u. corresponds to a non-zero contribution of reciprocal lattice vector and matches experimental results.[43] As the temperature is increased, the "umklapp" peak at temperatures 650 K and 850 K is smeared out, due to the thermal disorder effect. Our results are missing amplitude but give a good agreement for the position of the "umklapp" peak. In Fig. (5.4) we present CP differences normalized to the amplitude of the CP at zero momentum and compare to the experiment.[43] The agreement between the theory and the experiment is within the error bars.

We do not observe the broadening of the CP reported in the case of Na and Li.[29, 33] In all of the calculations of CP, temperature in the Fermi function was set to 0 K, because the Fermi energy is much larger then the studied temperatures.

In Fig. 5.5 we present temperature dependence of the "umklapp peak" for temperatures 650 K and 850 K as a function of thermal disorder with and without thermal expansion. The curves are labeled "ambient" for ambient density and "expanded" for thermally expanded volume. The peak is broadened in the case of expanded volume due to the larger amplitudes of ion oscillations.

The overall effect of lattice modulation has only a small effect on the Be valence electron wave function. This is likely due to Be's relatively high Debye temperature. In Fig. 5.4, we present differences between the CP calculations at 300 K and 650 K, and at 300 K and 850 K, together with experimental results for the direction of the momentum transfer q = [110] and the pseudopotential theory reported in the same work of Huotari et al.. To summarize, we

find that thermal expansion of Be has a moderate effect on the CP calculations, as reported in Huotari et al.,[43] and thermal disorder at moderate temperature smears out the intensity of the umklapp peak and is a "second order" effect on the CP.

Lithium

We present the CP for Li as a function of temperature in Fig. 5.6, where the crystal structure was expanded using the coefficient thermal expansion $\alpha = 46$ K^{-1}. In contrast to previous reports,[29, 33] we do not observe broadening of the CP. The inset in Fig. 5.6 shows the behavior of the "umklapp" peak as a function of both thermal expansion and thermal disorder. Although it behaves similarly to Be, the peak smears at a temperature of 400 K, much lower than that found for Be. This is due to Li being a softer material with a much lower Debye temperature.

In the Fig. 5.6 we present experimental results[90, 91] together with KKR[92] and GWA[57] calculations. Our calculations are very close to the KKR one with correlation correction.[59] The two methods are based on LDA approach and have no many body effects implemented, thus we believe that discrepancy between calculations and the results are due to the many body effects. Schulke et al.[92] attributed this discrepancies to the self energy effect with excitation of plasmaron mode. GWA results[57] are in a very close agreement with the experiment, in the same work LDA calculations with correlation effects[9] have been performed which reduced discrepancies between LDA and the experiment. The results are better in the case of GWA then in the case of LDA calculations due to the differences in the occupation number density $N_{b,b}(\mathbf{k}) = \pi^{-1} \int_{-\infty}^{\mu} ImG_{b,b}(\mathbf{k}, E)dE$, in LDA approach it was obtained using Fermi gas model and it is different obtained in GWA approximation in the case of Li.

In order to study the CP anisotropy, the lower panels of Fig. 5.6 present differences between directional CP, labeled accordingly together withe experimental results.[92] We find that oscillation amplitudes decrease as a function of temperature with the largest amplitude observed at 95 K, and with the largest anisotropy occurring in the $[110] - [111]$ direction

differences. Moreover, as the temperature grows the amplitude of the anisotropy is smeared out and higher momentum peaks disappear. Our results follow the oscillation pattern in the experiment, however discrepancies between the theory and experiment which are due to the many body effects are still noticeable. With the increase of temperature our calculations are generally closer to the experiment.

The CP differences between the CP calculated at 95 K and 300 K are presented in Fig. 5.7 and compared to experimental measurements of thermally disordered Li for the $q = [110]$ direction of momentum transfer.[98] With the exception of a small discrepancy in the position of the peak, which is likely due to the use of a constant value for the thermal expansion coefficient, our results are in good agreement with the experiment, falling within the measured error bars.

Silicon

Finally, we studied the solid and liquid phase CP of Si (Fig. 5.8). Our results are in reasonable agreement with recent experiments[63, 73] for a broad range of p. In the region around $p = 1.1$ a.u., we have less then 1 percent disagreement with Okada et al..[73]

In Fig. 5.9 we present our calculations of CP differences, between solid Si at temperatures 298 K and liquid Si at 1787 K for two values of the cutoff of angular momentum $l = 4$ and $l = 5$, together with the experimental results[63, 73] and the theory reported in the same work. Our calculations agree reasonably well with experimental results. Differences between the two theories may be due to differences in CP calculation using the RSGF approach and momentum based theory and due to the differences in MD simulations.

In the Fig. 5.10, we present calculations of EMD for solid structure at 298 K and in liquid at 1787 K and a compare with experimental results,[63, 73] obtained from CP experimental profiles. Differences in the EMD slope in the experiments versus theory may be due to unaccounted broadening in the experimental results.

5.4 Conclusions

We present *ab initio* CP calculations of MD simulated disorder of Be, Li and Si for a number of temperatures. We study thermal disorder and thermal expansion effects on high momentum components of the CP. To calculate CP we use the RSGF approach and the real space code FEFF. We find that thermal lattice expansion has a moderate effect on the CP in both Be and Li as reported previously.[43] Thermal disorder has an effect of smearing the high momentum umklapp peak in the case of Be and Li. We find good agreement with the experimental results[43, 63, 73, 98] and to a good precision we predict changes in the CP of polycrystalline Si which undergoes solid to liquid phase transition. We observe that in the case of Li, strong directional anisotropy, which was discussed in the recent work on LiF[31] as a function of thermal disorder, becomes weaker.

For temperatures close to the melting point, both thermal disorder and thermal expansion play an important role and need to be taken into account when studying the electronic properties of Be and Li. For Si structure, which undergoes a solid – liquid phase transition, disorder has an important role and MD simulations coupled with real space theory can reasonably predict the influence of the disorder on the CP.

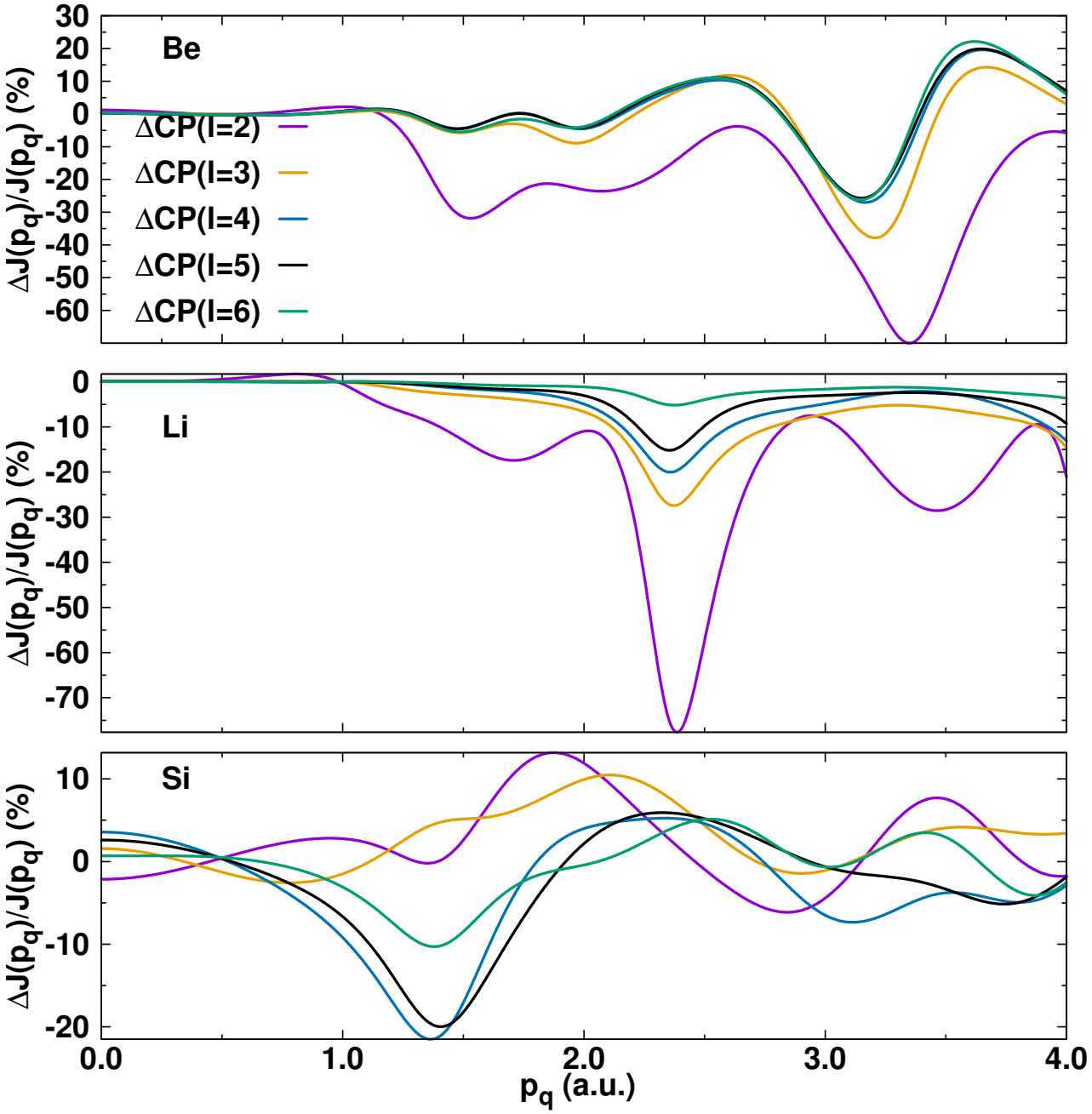

Figure 5.1: CP percent difference relative to $l = 7$ for Be, Li and Si for q = [100].

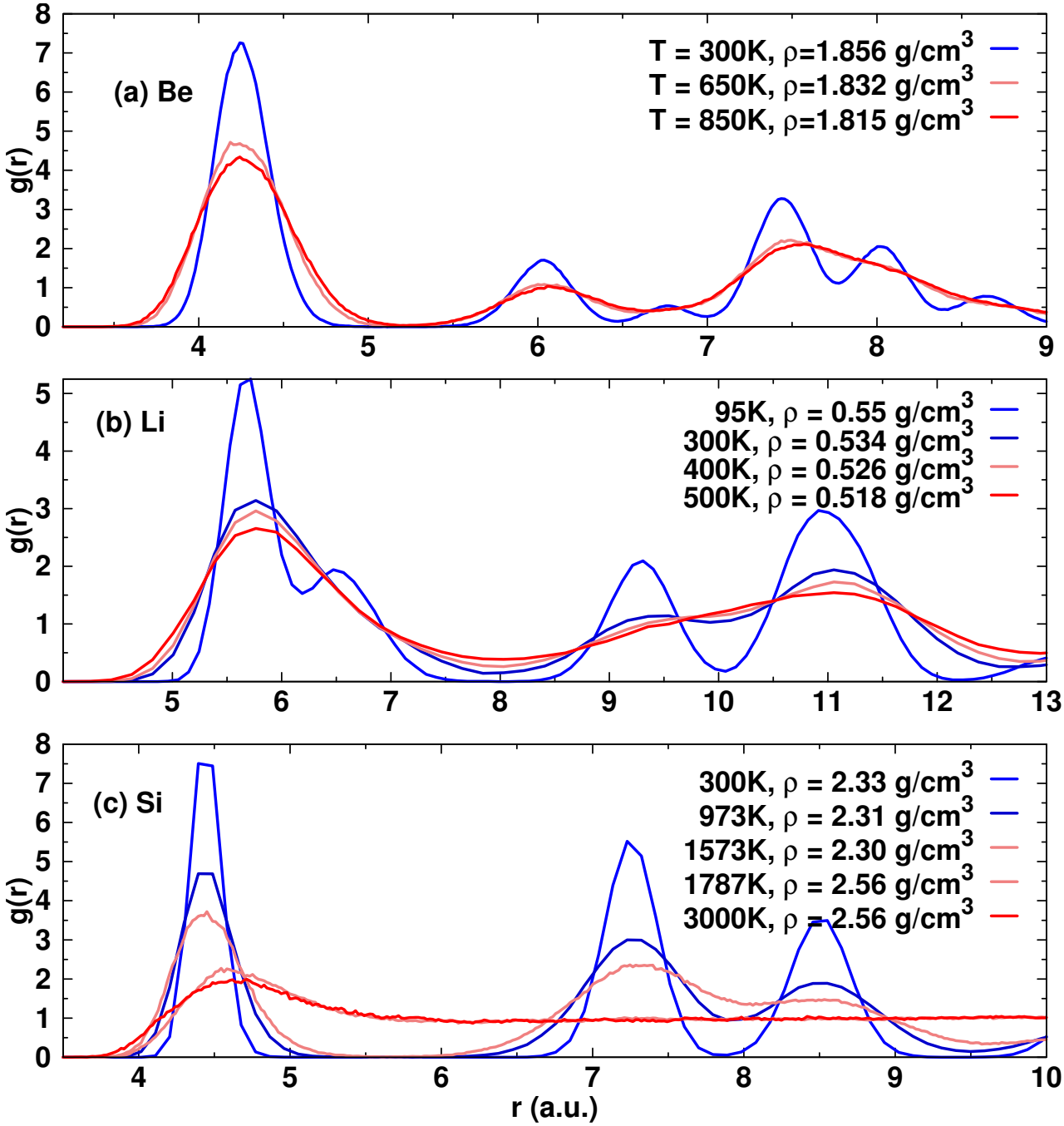

Figure 5.2: Pair distribution functions $g(r)$ of Be, Li and Si as a function of temperature and adjusted densities. (a) $g(r)$ for Be, (b) $g(r)$ for Li, where the density was adjusted using its linear thermal expansion coefficient ($\alpha = 46$ K^{-1}), (c) $g(r)$ for Si.

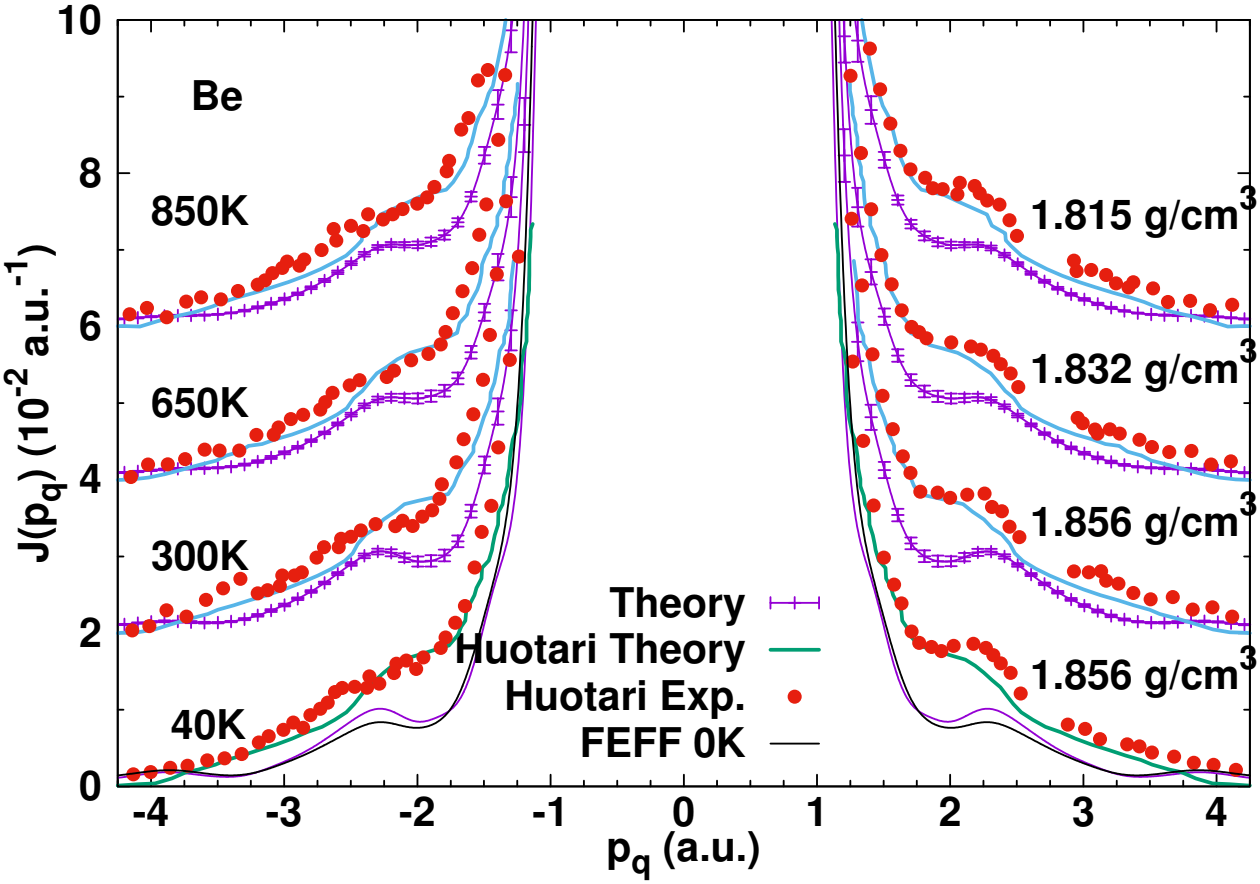

Figure 5.3: Be CP calculations for direction q = [110], experiment and pseudopotential theory with thermal disorder[43] at a number of temperatures and densities.[46] For clarity of presentation, each data set was shifted by 0.02 electrons/a.u.. Theoretical results were broadened to match the experimental resolution of 0.16 a.u.[43]

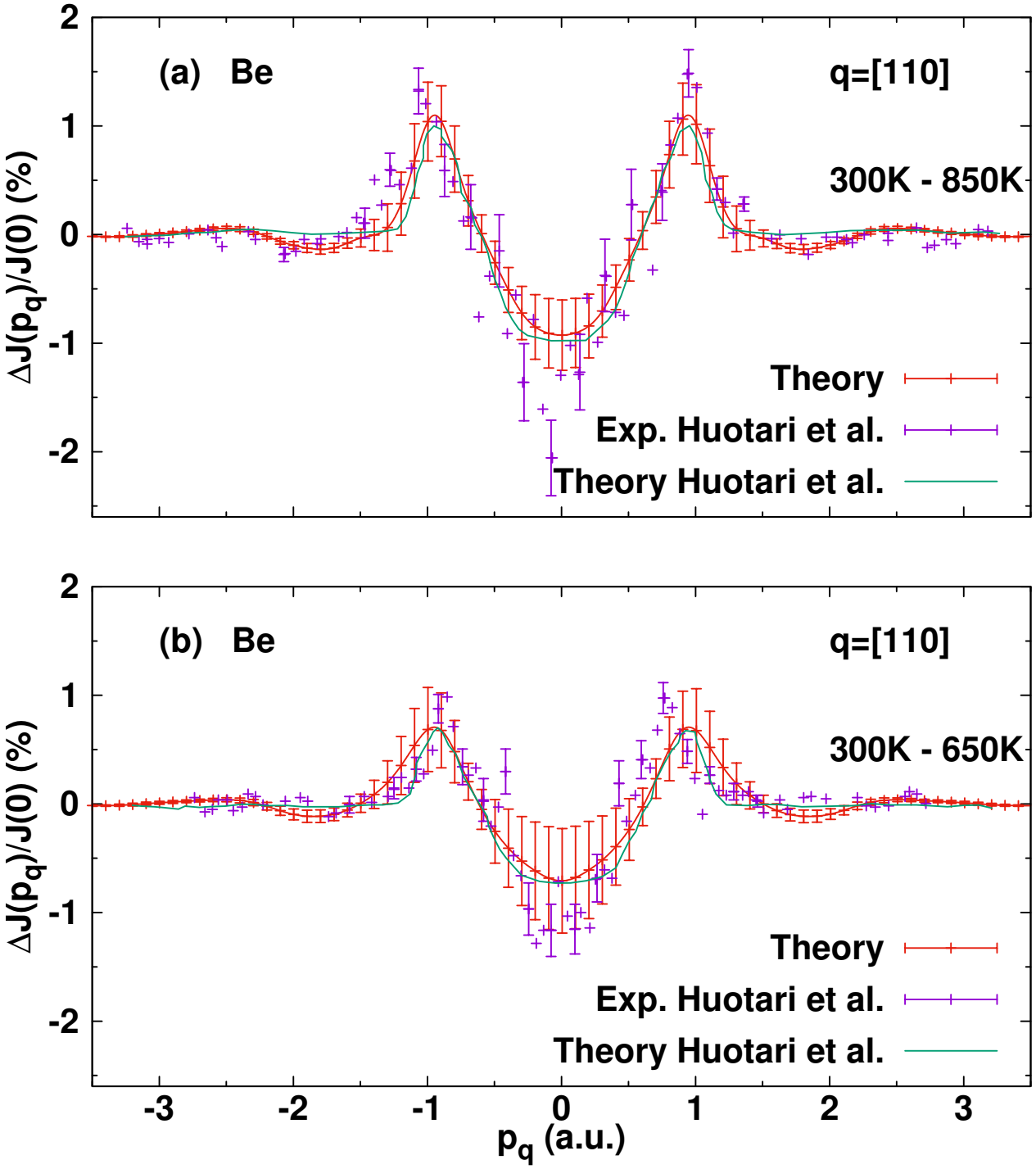

Figure 5.4: Be CP differences calculations vs. experiment and pseudopotential theory[43] for direction q = [110]. (a) Differences in CP between 300 K and 850 K. (b) Differences in CP between 300 K and 650 K.

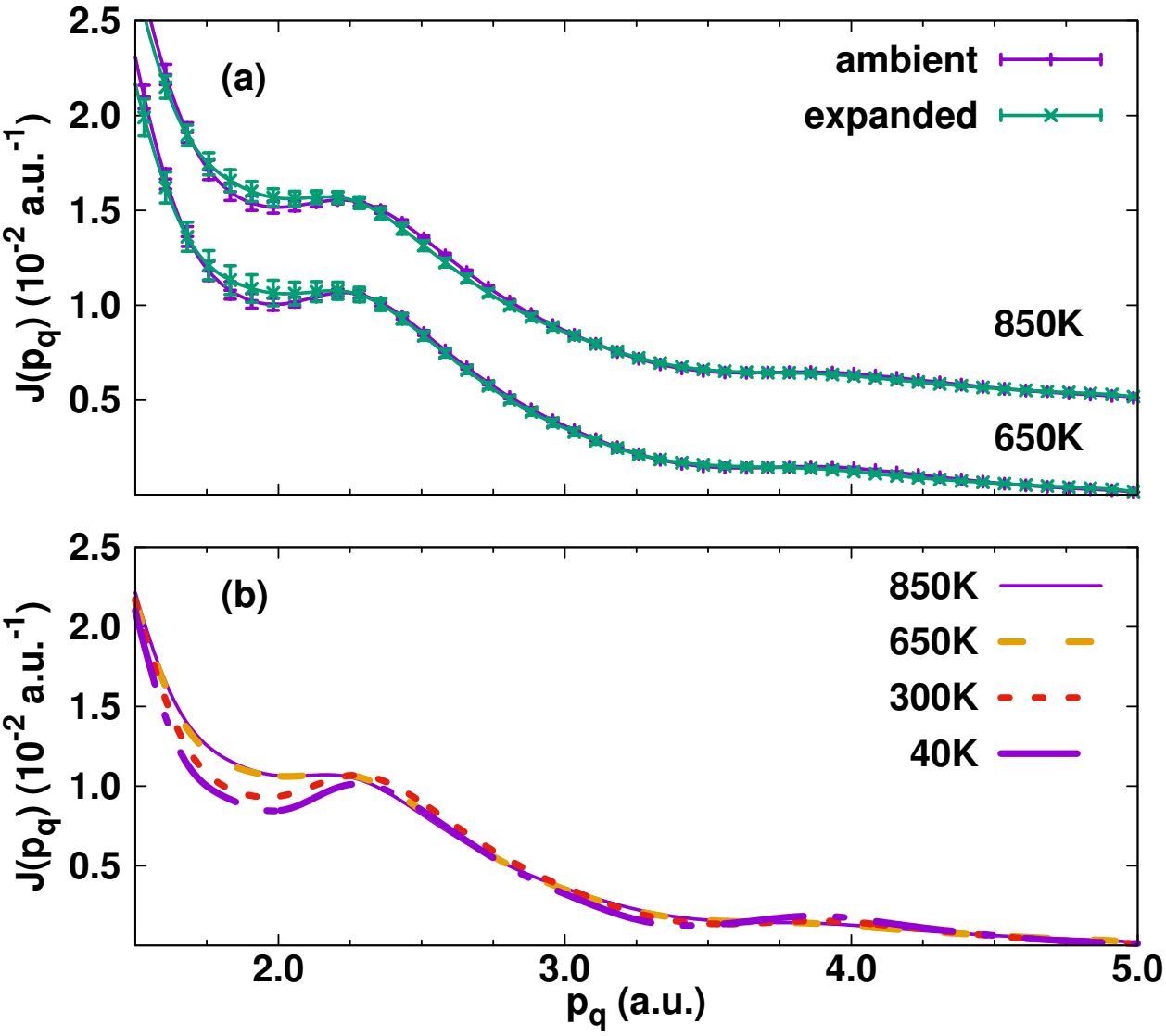

Figure 5.5: (a) Region of "umklapp" peak. CP calculation of expanded and ambient density with disorder. For clarity, results were shifted by 0.05 electrons/a.u.. (b) Comparison of Be CP calculations with disorder.

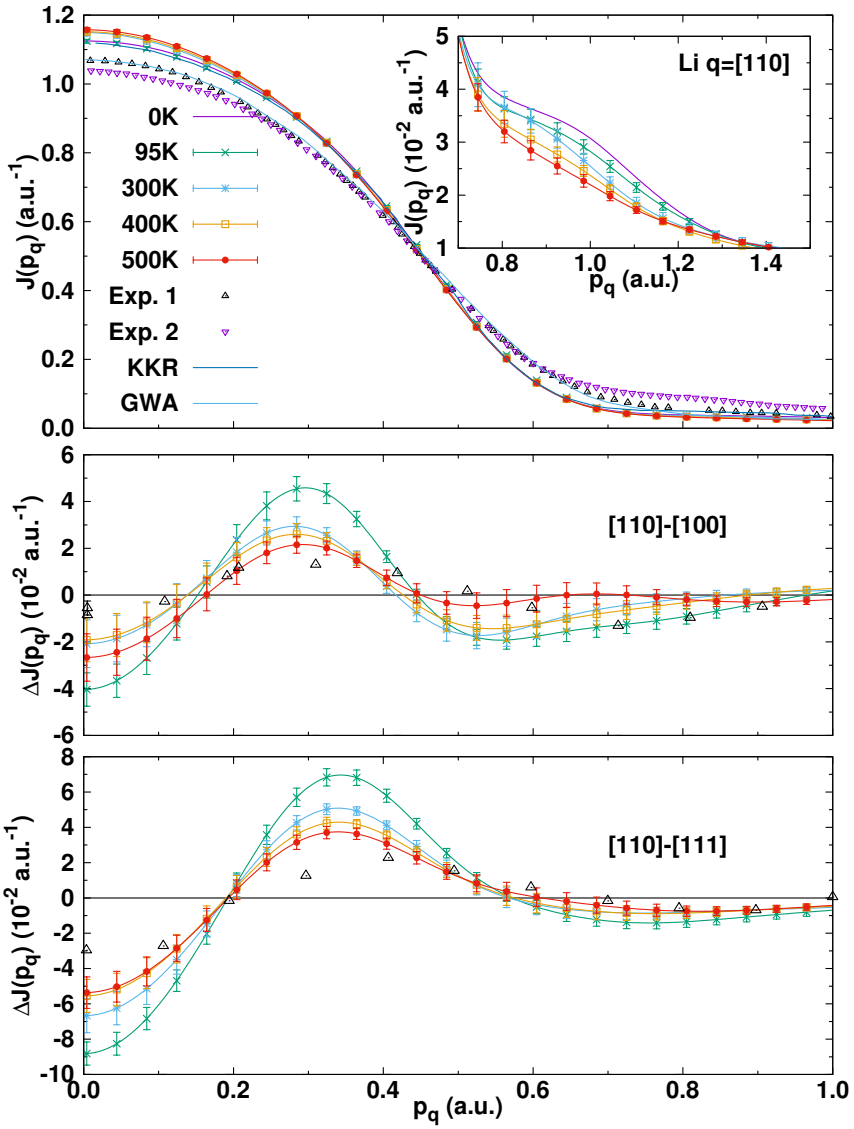

Figure 5.6: Li CP calculations with thermal disorder for a number of temperatures and differences in CP between directions. Comparison with Sakurai et al. "Exp. 1"[90] and Schulke et al. "Exp. 2"[91] and KKR[92] and GWA[57] theories. Upper panel: CP for [110] direction. Inset: "umklapp" peak amplitude at a number of temperatures and lattice expansion. Lower panels: Differences between CP for a number of directions for a range of temperatures and comparison to the experiment[92]. T = 0 K is the calculation without disorder with ambient lattice constant.

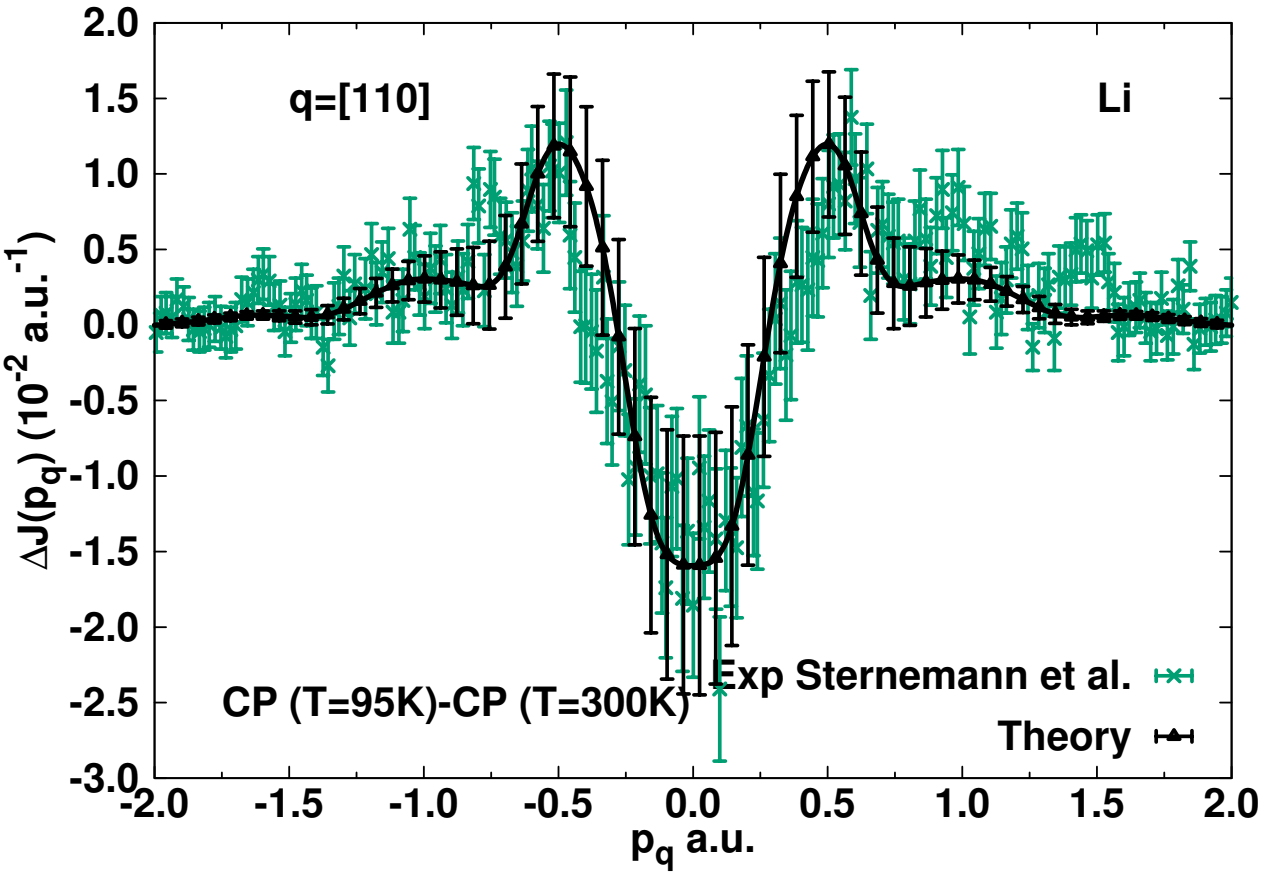

Figure 5.7: Differences in calculated Li CP for q = [110] between T = 95 K and T = 300 K and comparison with experiment.[98]

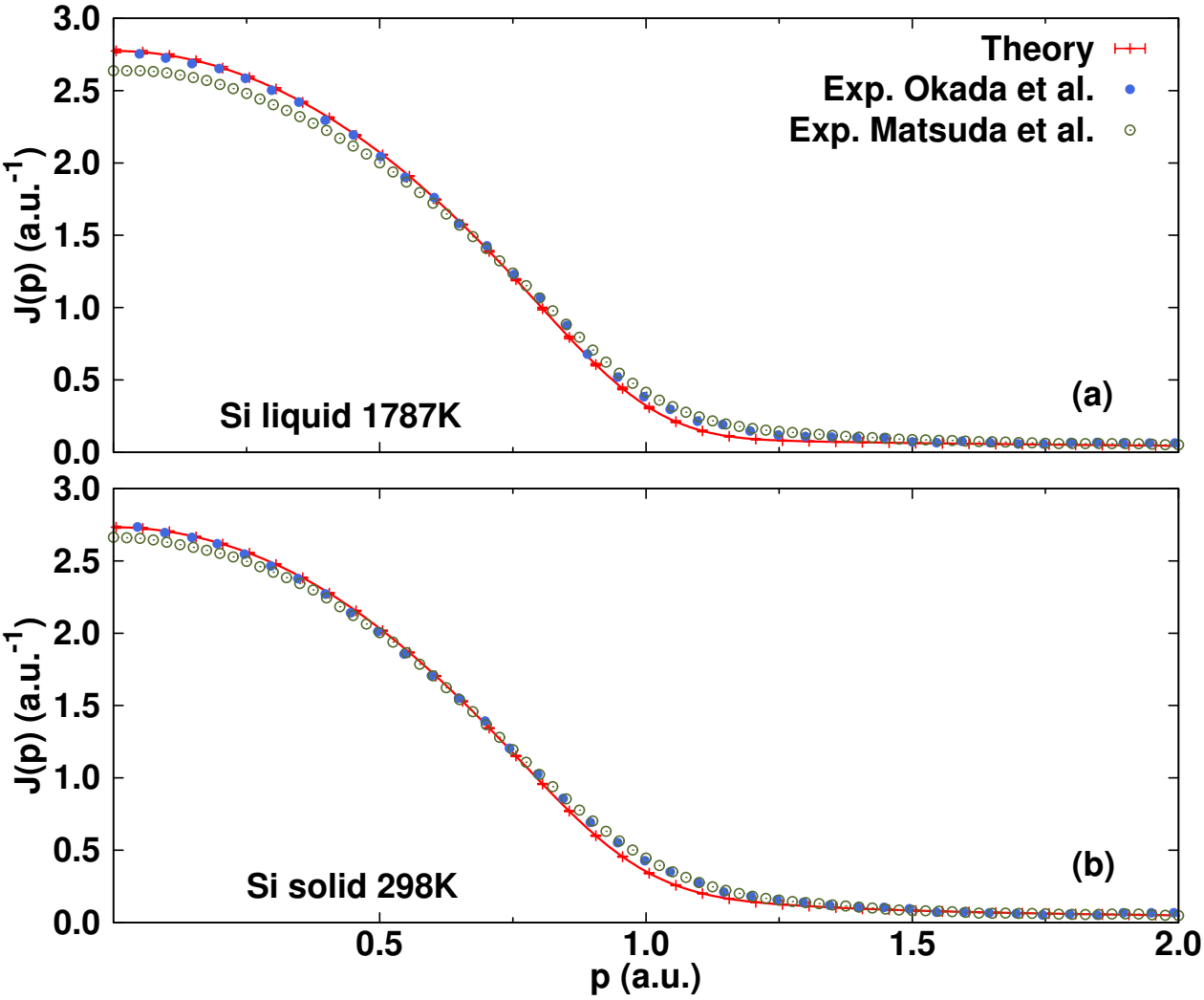

Figure 5.8: Si CP calculations and comparison with experiments.[63, 73] (a) CP of polycrystalline Si in liquid phase calculations and experiments (b) CP of polycrystalline Si in solid phase calculations and experiments.

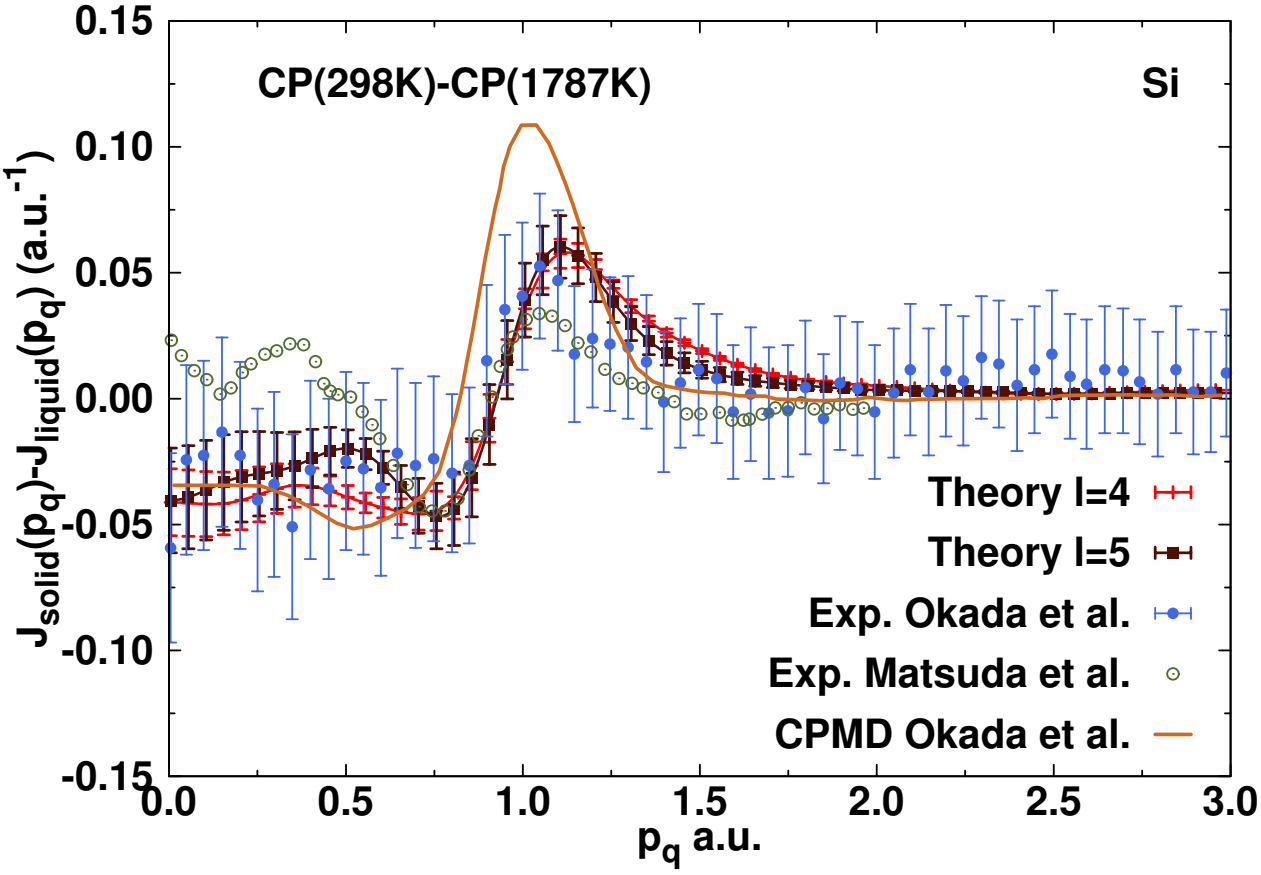

Figure 5.9: Differences in polycrystalline Si CP calculations for angular momentum $l = 4$ and $l = 5$ between T = 300 K (solid) and T = 1757 K (liquid) and CPMD calculations, and comparison with experiments.[63, 73]

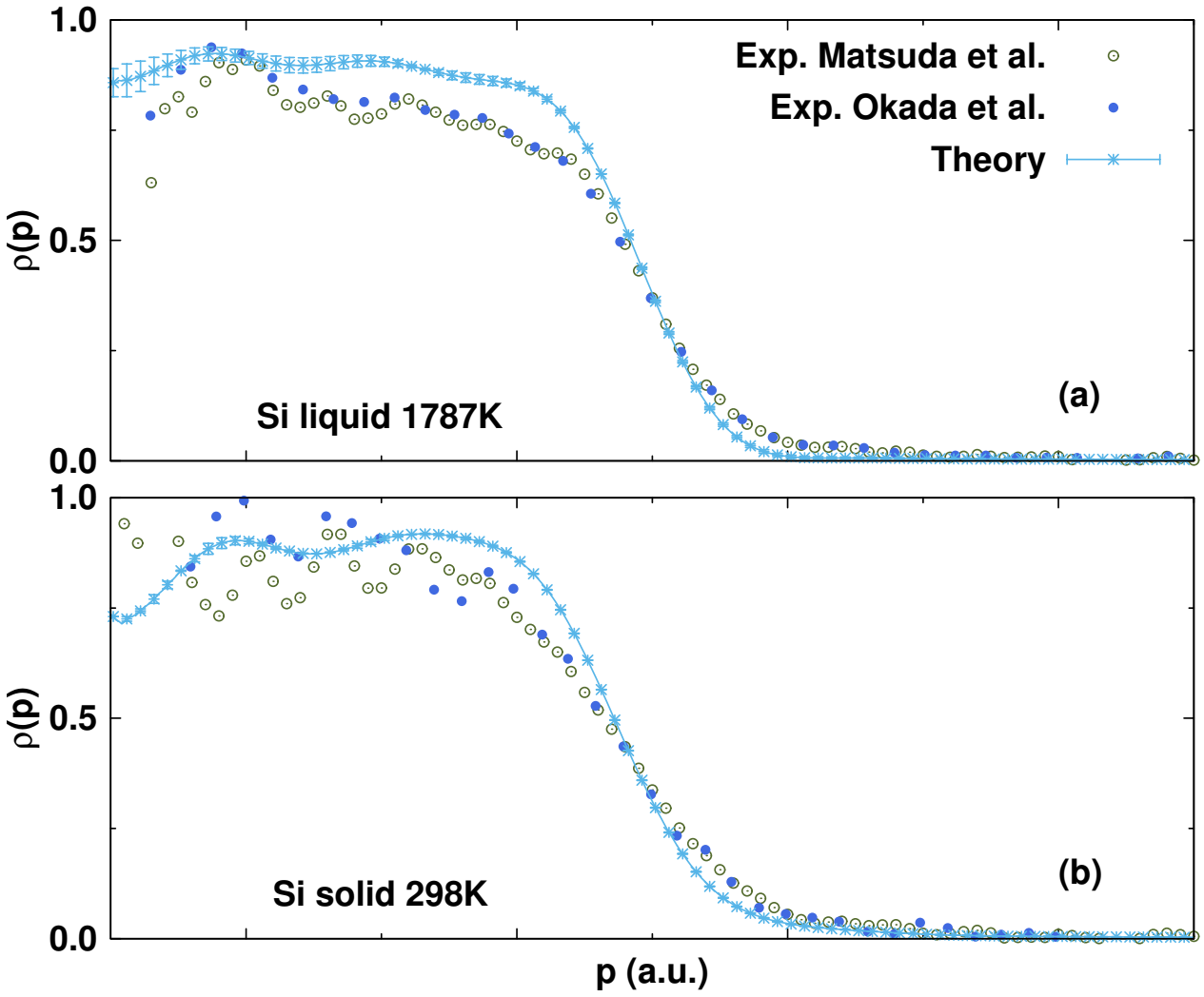

Figure 5.10: (a) Valence EMD $\rho(p)$ for Si at T $= 298K$ and comparison with experiments.[63, 73] (b) Valence EMD $\rho(p)$ for Si at T $= 1787K$ and comparison with experiments.[63, 73]

Chapter 6

WARM DENSE MATTER REGIME

I present calculations of CP and electron momentum density (EMD) for Be in the Warm Dense Matter (WDM) regime at elevated densities and temperatures; I compare results to disordered and ordered systems. As discussed in the recent work by Mattern et al.[64–66], present experimental practice for inferring state variables from x-ray Thomson scattering (XRTS) experiments ignores the dominant role of the interaction through orthogonalization of the valence (free) and bound electrons. In this work, I find that the results of Mattern et al. generalize reasonably well to disordered systems and solid state effects must be considered in the interpretation of XRTS studies.

6.1 Introduction

Warm Dense Matter (WDM) has solid-like densities and temperatures which can be significantly higher than the Fermi temperature of a system. Studies of matter in this state can be of special relevance to planetary and stellar conditions,[4, 44, 86, 102] novel phases such as electrides and new experiments in which the WDM regime is reachable, for example in an Inertial confinement fusion (ICF) experiment.[62] WDM states exist for a very short period of time - on the order of nanoseconds or less. In order to estimate thermodynamic parameters, physical properties need to be probed, for which x-ray techniques are favorable. X-ray Thomson Scattering (XRTS) is widely used for inferring thermodynamic parameters in the WDM regime. However, a number of objections to present methodology have been raised, in particular core-valence orthogonalization.

In fig. 6.1, parameter space of high density and high energy is presented; the picture is

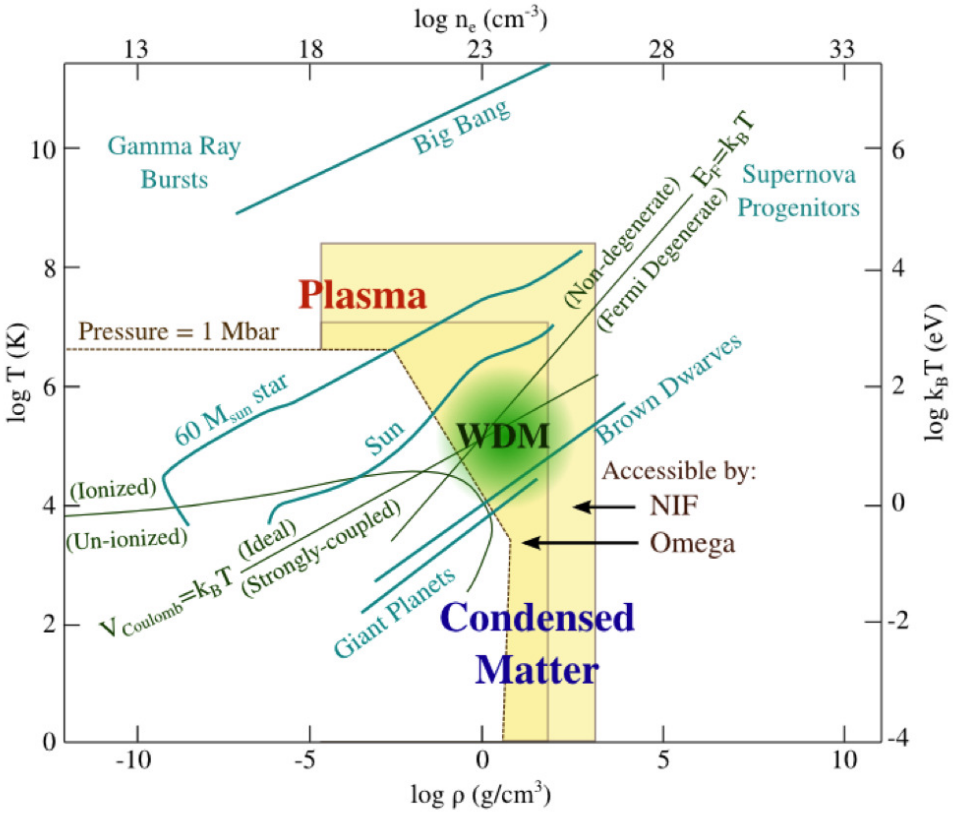

Figure 6.1: Density and pressure parameter space region of interest of WDM. WDM is defined as regime in which temperature reaches outer electrons ionization values ($\sim$ 1 eV). Figure is redrawn from Ref. [23]

redrawn from [23]. Both the horizontal and vertical axes are on logarithmic scale. Along the horizontal axis, the density of matter is presented at the bottom, and at the top approximate density of electrons is plotted. On the vertical axis, logarithmic temperature is plotted, on the left side in Kelvins and on the right side in eV. Above the dashed line labeled "Pressure = 1 Mbar" is the region of high-energy-density physics (HEDP), this region is defined as the region where applied pressure is comparable to the internal energy of room-temperature atoms and molecules [28]. The shape of the boundary of 1 Mbar pressure is set due to the

fact that at low density pressure is radiation dominated. As density is increased, the thermal pressure becomes dominant; once the Fermi-pressure of electrons reaches 1 Mbar, the curve becomes vertical.

The curve labeled "ionized/unionized" presents boundaries of non-ionized matter. At low density, it is roughly set by binding energy of the outer electrons ($\sim$ 1 eV). As density increases, overlapping Coulomb potential delocalizes the outermost electrons, forming a conduction band. The next curve is labeled "$V_{coulomb} = k_B T$". On the high temperature side, interactions are weak and can be treated perturbatively. The last bounding curve, labeled "Non-degenerate/Fermi degenerate", corresponds to temperature $T \sim E_F$ in the system, bounding the region where quantum statistics is important.

The region labeled WDM is much less studied. It is in between two well studied regions, plasma and Condensed Matter region. The definition of WDM is loosely defined as matter at solid-like or higher density, with temperature comparable to Fermi energy and partially ionized electrons and partially degenerate. It is an important regime in laser-driven compression and heating experiments, at least as a transient en route to the high density and temperature needed for fusion[35, 36, 62]. As discussed in chapter 5, my goal is to establish the use of non-resonant inelastic scattering to determine state variables in WDM, via the CP directly related to NRIXS.

To this end, I report *ab initio* calculations of the Compton Profile (CP) in impulse approximation (IA) and electron momentum density (EMD) calculations using real-space full-scattering techniques of thermally disordered Be at elevated densities and temperatures simulated using Molecular Dynamics(MD), an important test case in the WDM regime. I find that at both high temperatures and densities solid state effects greatly influence the state of EMD and one needs to account for them when inferring the state variable of the system. To answer the question of the role of ionic disorder in CP in WDM regime, I compare perfect crystal calculations, which have electronic temperature, with calculations, which have ionic and electronic temperature of CP and valence electron momentum density. I use Fermi function to fit EMD and extract corresponding temperature (T^*) for ionic and

electronic disorder and perfect crystal calculations with electronic disorder. Results of this analysis are presented in fig. 6.10. Differences between the estimated parameters for two sets of calculations show that solid state effects are significant and simplified models of degenerate Fermi gas in the WDM regime is not a good model to use, independent of the disorder of the system.

6.2 Methods

The two main codes which I used for calculations were `VASP` and `FEFF`, which were developed for simulations and *ab initio* calculations at conventional condensed matter temperatures. In order to use them at elevated densities and temperatures, I performed diagnostics to establish a parameter space in which reliable results could be obtained. In `VASP` calculations, to save on computational time, only ionic temperature was set to the desired level. However, I compared the pair distribution function of the calculation with only ionic temperature with the calculation with ionic and electronic temperature set to desired value and found no differences between the results of the two calculations.

6.2.1 MD simulations

I used *ab initio* simulation package `vasp` (Viena Ab Initio Simulation Package),[54, 55] with PAW potentials.

To test the available potentials in `VASP`, I used a simple Be dimer model. Distance between the atoms was varied and in fig. 6.2, total energy of the system as a function distance between the ions is plotted for two *ab initio* programs `Gaussian` and `VASP`. Differences in pair potential which starts at about 3 eV signifies differences in the approximations of ionic potentials between two codes. Thus I only simulated Be for temperatures up to 2 eV; beyond that temperature more analysis is required in order to proceed.

For the temperature-dependent MD simulations, a Nosé-Hoover thermostat[70] was used. The super cell used in the MD simulation has 384 atoms, which consist of $4\times4\times6$ orthogonal cell of 4 atoms. For the range of temperatures, nosé parameter s_{mass} was adjusted to match

the characteristic phonon frequency, which is $\sim$ 16.4 THz.[96] I used a smaller time step of 0.5 fsec for the simulations. Equilibrium with the thermostat was reached in about 5000 steps.

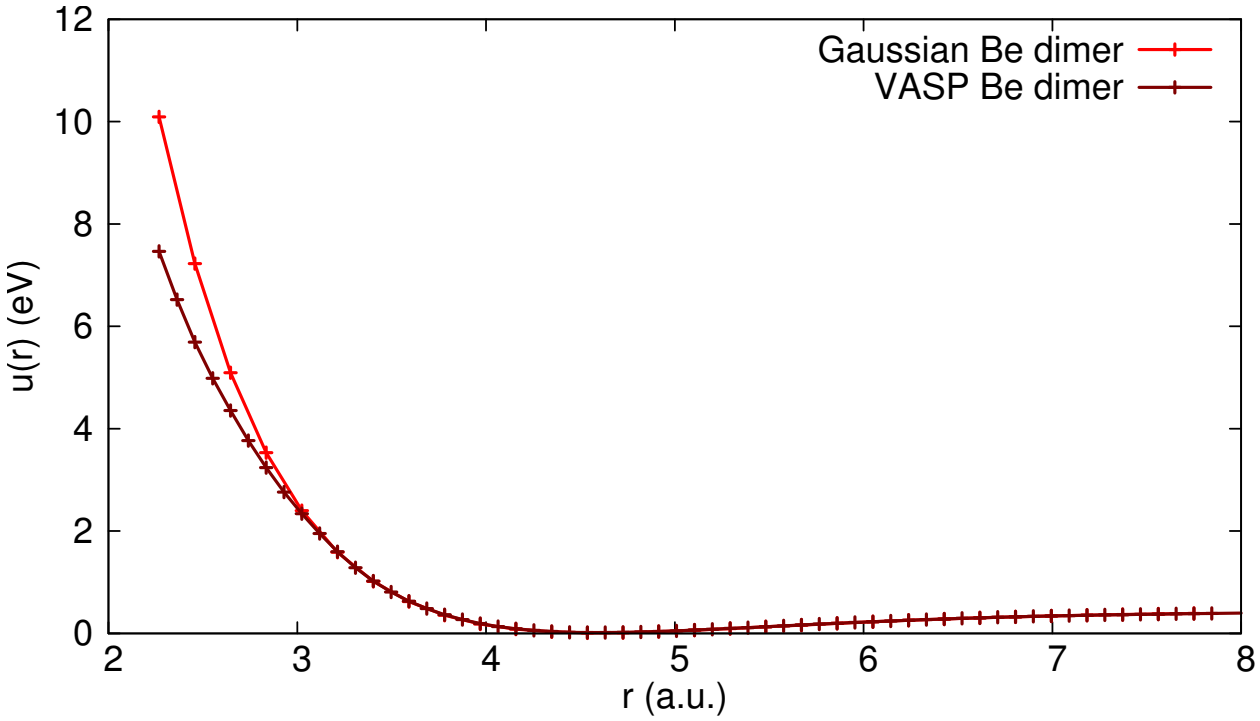

Figure 6.2: Comparison of pair potential $U(r)$ of Be dimer as a functions of distance between ions calculated using Gaussian and VASP. At about 3 eV the two curves start diverging.

The pair distribution functions for a number of densities and temperatures are presented in Fig.(6.3), calculated from MD simulations at specific temperatures and densities. At the densities 0.57 of ambient density V_0 and T= 2 eV the second shell peak is still present, at V=V_0 at T= 2 eV the second shell peak is not as pronounced, and at 1-eV simulations at 0.57 of ambient volume V_0 has pronounced first and second shells peaks. The behavior of the pair distribution function $g(r)$ follows density and temperature changes in the system. With such significant changes, the valence electron properties should be influenced accordingly.

6.2.2 RSGF CP calculations

For CP calculation in disordered systems, I used the real space code `FEFF` [66, 83]. To get a sampling of the disordered system, I used 24 randomly chosen atoms from a number of time frames toward the end of MD simulations. The number of potentials used in the `FEFF` calculations is 13, which corresponds to the number of nearest neighbor plus central atom; all the atoms outside the first shell had the same potential. In the self consistent loop and full scattering matrix calculations, I used $\sim$ 170 atoms, which sets the size of the cluster. Number of atoms in the cluster gives the size of the cluster which sets the uncertainty of momentum. I choose uncertainty in the momentum equal to 0.1 a.u.. These calculations used 13 unique self-consistent muffin-tin potentials, which roughly account for the absorber and its nearest neighbors. All other atoms outside the first coordination shell shared the same muffin-tin potential. The final CP was then calculated as an average over the CP from each structural snapshot, and the vertical bars displayed in this paper correspond to errors in the mean unless otherwise stated. One of the most important parameters for convergence test is angular momentum l in fig. 5.1, I present results of convergence test for Be. I use value of angular momentum l = 3 for simplicity's sake. Momentum resolution is inversely proportional to the size of the cluster, and in all of the calculations momentum resolution was kept on the order of 0.1 a.u.. To calculate CP for the crystal at ambient conditions I used lattice constants $a = 4.3289$ a.u. and $c = 6.7675$ a.u.. To obtain isotropic CP I use Lebedev quadrature and averaged over 50 angular directions. Temperature of electrons was set using Fermi function and is described in chapter 2. Integral over the CP gives number of valence electrons. Every CP was renormalized to give the right number of electrons in Be 4.

6.3 Results and discussion: Role of disorder and estimation of state parameters crystalline structure approximations.

To answer an important question:"how big is the role of disorder, or can one satisfactorily estimate state parameters using crystalline calculations?", I present two sets of calculations of CP and EMD: first, RSGF calculations using prefect crystal as the lattice and, second, sys-

tems with thermally disordered atoms and electronic temperature using MD+RSGF method described in previous chapters.

Results of the first type of the calculations of isotropic CP (averaged over angles) at ambient density are presented in fig. 6.4. Broadening of the CP as the temperature is increased is due to the valence electron temperature set to the corresponding values, given in the legend of fig. 6.4.

Results of calculations of the second type for disordered MD simulated Be of isotropic CP and EMD, both with error bars, are presented in figs. 6.5 and 6.6. In fig. 6.5 calculations of isotropic CP are done at temperatures of 1 eV and 2 eV at ambient density and compared to the calculations of CP at 2 eV at 0.57 of ambient density V_0. The error bars are set by the number of CP calculations, which gives representation of the thermal disorder in Be. In fig. 6.6 calculations of EMD ($n(p)$) were done using equation:

$$\rho(p) = -\frac{1}{2\pi p}\frac{d}{dp_q}J(p_q)|_{p_q=p}, \tag{6.1}$$

at T = 1eV for ambient density and 1.73 of ambient volume, together with the fit of Fermi function. Corresponding effective temperature is labeled T^*, where $f(p) = 1/\rho_0\,\rho(p)$ and $\rho_0 = V/(2\pi)^3$, where V is the volume per valence electron.

The EMD of Be was derived from CP of first type calculation for densities: ambient density V_0 0.57 of ambient density V_0 and 1.72 V_0. Results of the EMD calculations are presented in figs. 6.7 to 6.9. The EMD as a function of temperature in the WDM regime is dominated by the Fermi function, as a function of temperature. Solid state effects have an important role, as they can be seen from the tails of the CP extending far beyond Fermi momentum of the system, unlike the case of CP in degenerate Fermi gas. The abscissa was renormalized by the Fermi momentum. It is important to note that at T= 0 eV, effective temperature parameter $T^* = 3.9$ eV, which shows that simple model of the Fermi function fitted to calculations of $n(p)$, over-estimates the temperature of the system.

To quantitatively compare the results of the two methods and to answer the second question: "Do simple degenerate gas models used in plasma community give a good description

to the valence electron ground state in WDM regime and solid state effects are dominated by Fermi distribution?", I used results of EMD figs. 6.6 to 6.9 calculation together with Fermi function $f(E,T^*) = \frac{1}{1+e^{(E-E_F)/kT^*}}$, to estimate effective temperature T*. The fit procedure was done with the condition on the normalization of EMD, which is $n = \int dp\, p^2\, n(p)$, with the error at each point treated as a weight.

Results of the analysis are presented in fig. 6.10. On the abscissa I plot temperature set in the simulation MD and RSGF. On the ordinate, I plot effective temperature T*, which is obtained from fitting procedure. I present results for ambient and expanded structure. Points represent results of the second type of calculations, where both thermal disorder and electronic temperature are presented. Crosses are results of the first type of calculations, where crystal structure is kept constant and electronic temperature is set to the desired value.

I would like to begin by discussing the first question, which was: "How big is the role of disorder, or can one satisfactorily estimate state parameters using crystalline calculations?". Differences between the first and second type of calculations are obvious, even though the Fermi function does not fit ideally in the case of MD+RSGF calculations. With this information, I can conclude that thermal disorder is important and CP is not dominated by the Fermi function. That being said, the general agreement of established effective temperature T* shown in fig. 6.10 still raises the question of whether ordered calculations can be used to interpret EMD in a disordered system. If pragmatically functional, this would greatly simplify calculations and the task of establishing the state parameters.

In regard to my second question, using simple Fermi function to extract state parameters is an appealing approach, but one needs to keep in mind that, due to the solid state effects, there is an offset in the T* value. In principle, this simple approach can be used and further investigation is needed in order to develop theoretical model which would relate effective temperature and parameters of state, which can have a complex dependence on density, temperature and ionization.

The next step of this study should be to perform more calculations in available parameter

space. With that information, one could deduce necessary improvements to the theory and have a reliable model for estimation of state variables using modified Fermi function.

6.4 Conclusions

I presented calculation of isotropic CP of Be for thermally disorder MD simulated system in fig. 6.5 and derived EMD presented in fig. 6.6. In addition I present the calculation of isotropic CP of Be in an ordered system with only electronic temperature set by Fermi function, and derived EMD for a number of temperatures and densities. Available parameter space was carefully investigated and I concluded that with the *ab initio* package `VASP`, in the WDM regime temperatures up to 2 eV are trustworthy despite reported results of MD simulations at temperatures up to 10 eV see fig. 6.2. For high density and temperature RSGF calculation using *ab initio* code `FEFF`, I was limited by convergence performance. In the cases when ions were too close, self consistent loop failed to converge.

I compared calculation of isotropic CP of Be for thermally disorder MD simulated system and derived EMD presented to calculations of CP and EMD in ordered system in order to answer the question whether perfect crystal calculation can be used in order to establish state parameters. The possibility still can not be ruled out, and more calculations at various temperature and densities are need in order to decide.

I used an effective parameter T^* to quantify the differences between the first and second class of calculations, as well as to answer two questions. First, whether non-solid state degenerate gas models are good enough to use for establishing state parameter. Second, whether we can develop an empirical model combined with solid state calculations which would make it possible to establish state parameters. I conclude that degenerate gas models do not reasonably predict temperature, due to not including solid state effects which produce tails beyond Fermi momentum. Model which can take into account temperature, density and ionization of the system needs to be developed. It will be an excellent tool for inferring state parameters from experimental data in WDM regime.

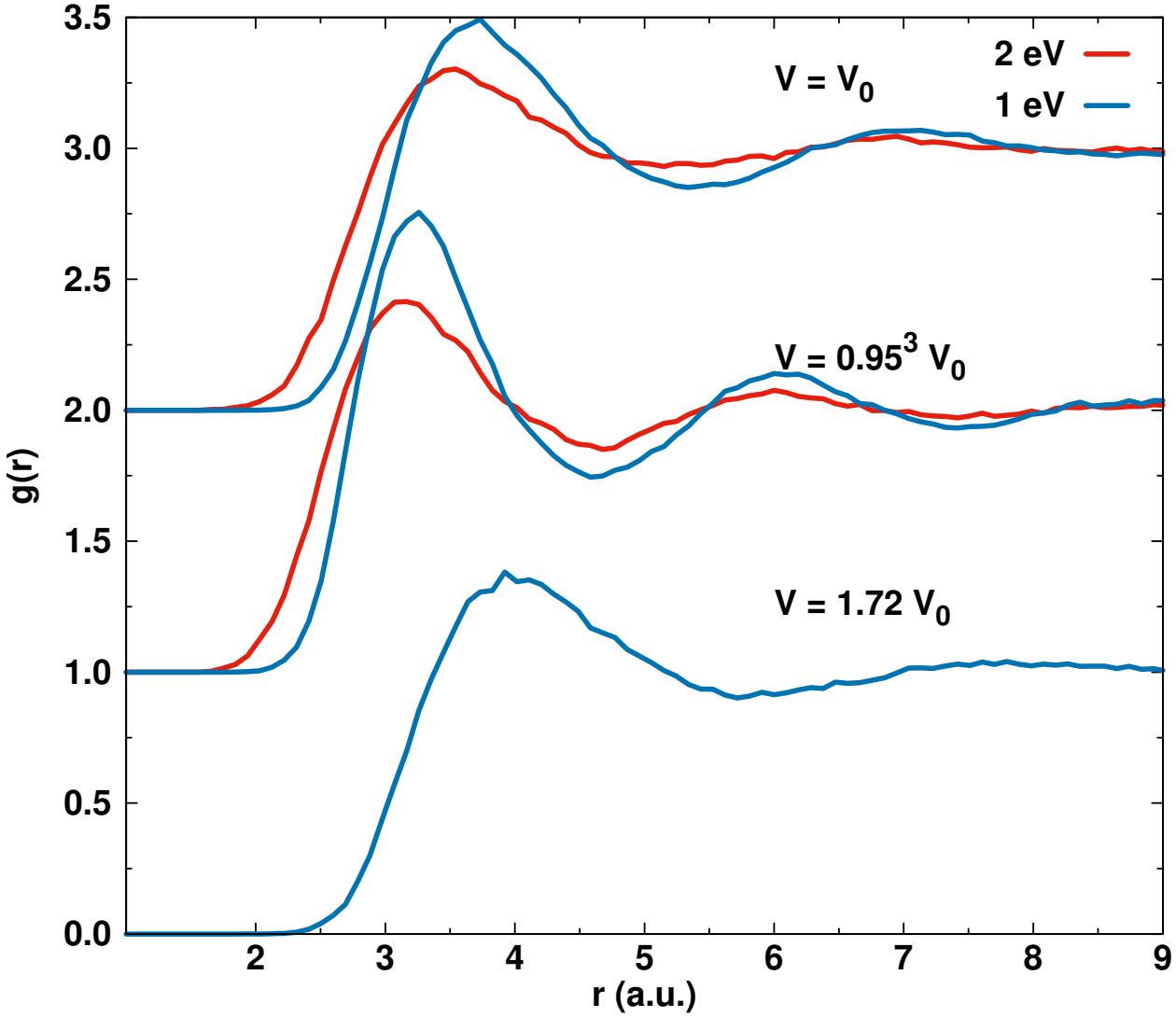

Figure 6.3: Pair distribution function for 1 eV and 2 eV ambient density (V_0) and 1 eV and 2 eV for 0.57 of ambient density V_0 and 1.72 of ambient density V_0, off set by 1 for clarity of presentation.

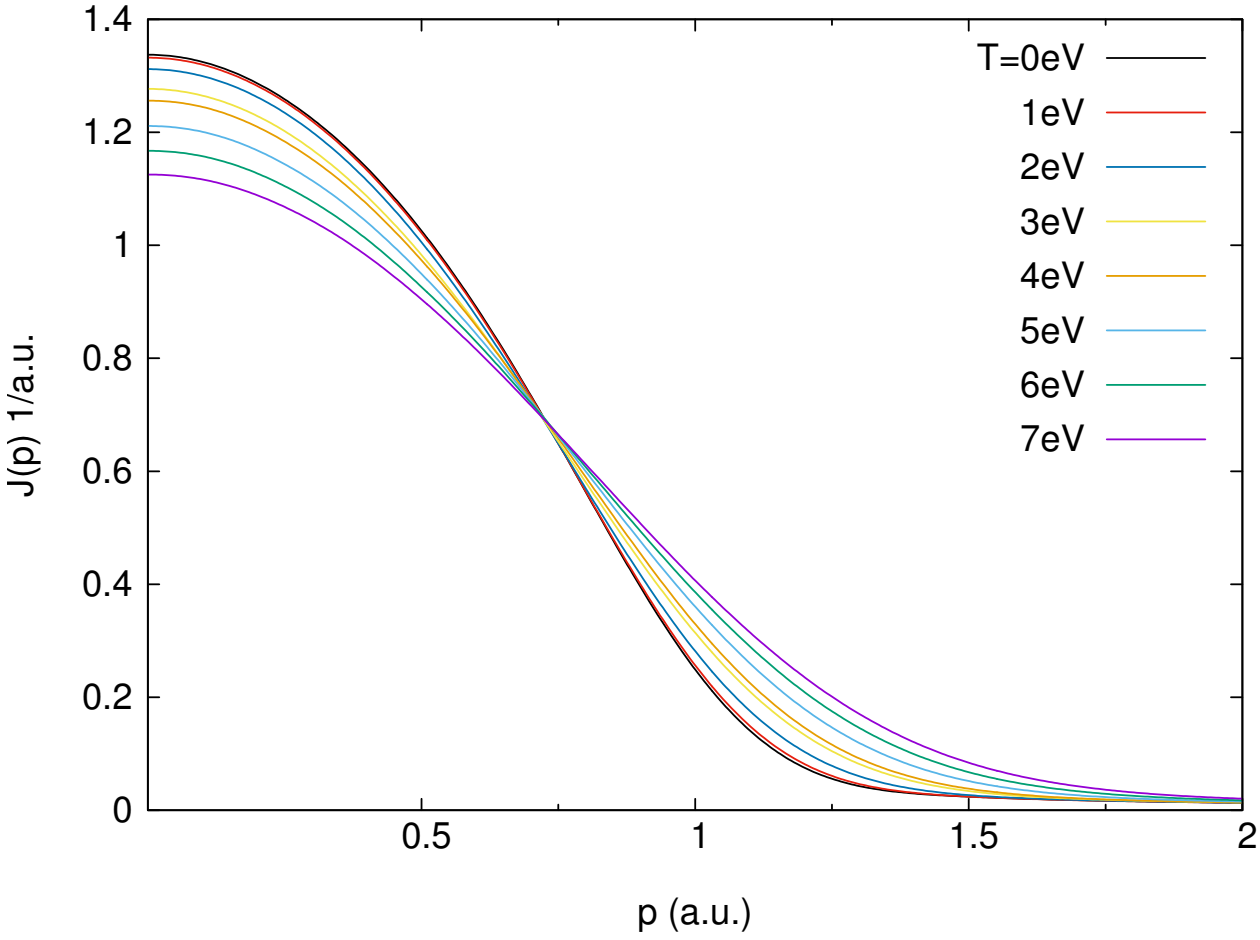

Figure 6.4: Directionally averaged calculations of CP for number of temperatures of crystalline structure of Be.

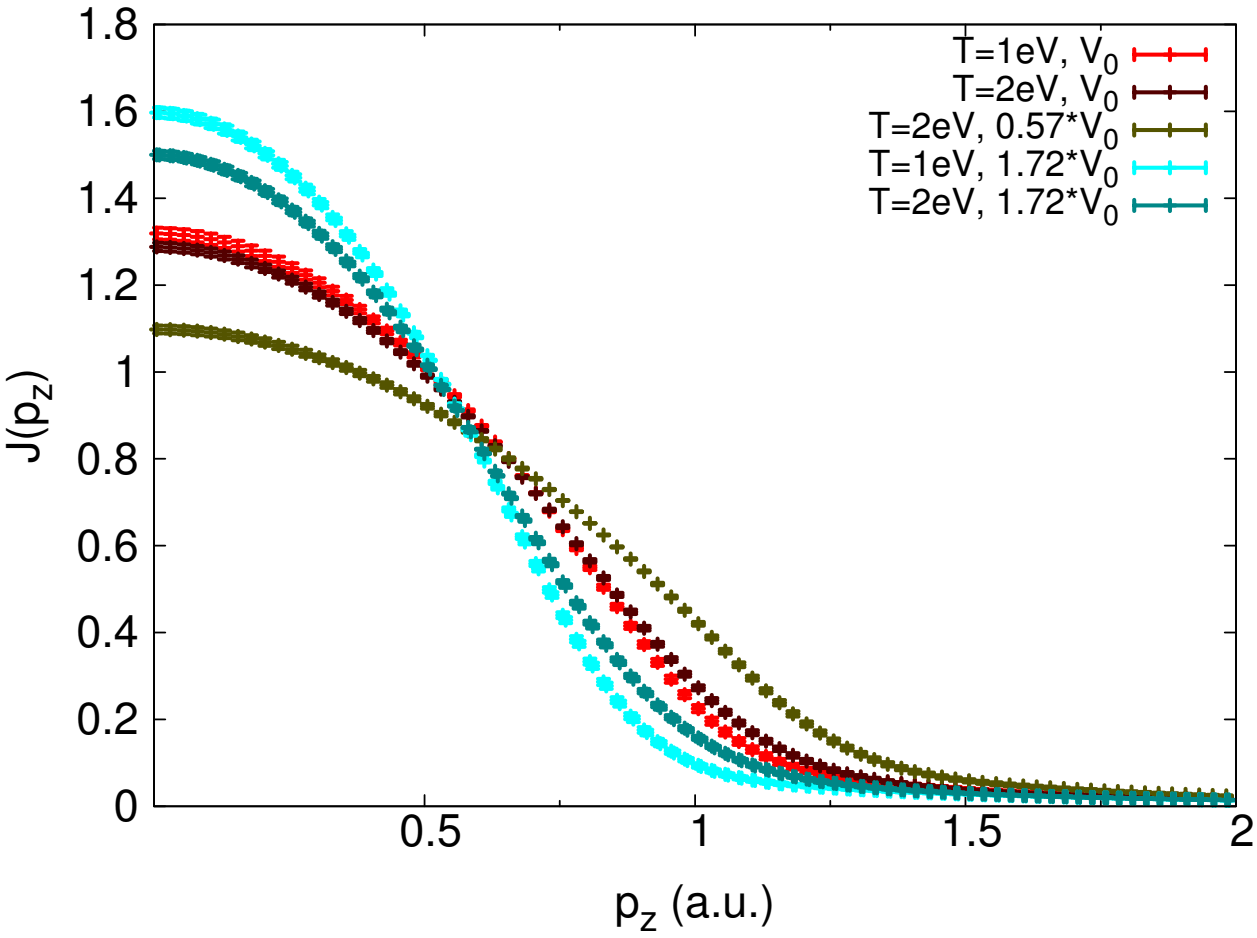

Figure 6.5: Directionally averaged CP calculation (MD+FEFF) at ambient density normalized to number of valence electrons for a number of temperatures and densities

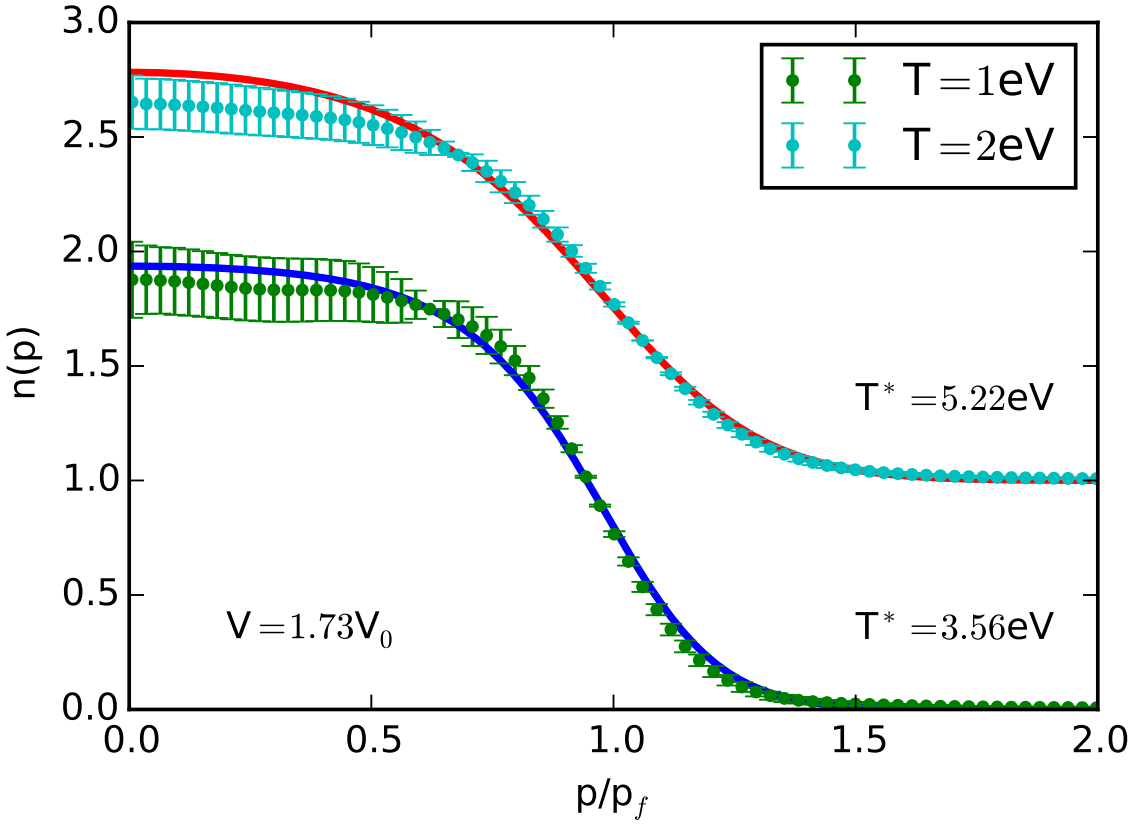

Figure 6.6: Density distribution function $f(p)$ at T= 1eV, comparison of FEFF calculations of MD simulated structure (ionic and electronic temperature). The red curve and blue curve Fermi function fit, together with temperature and chemical potentials for Fermi function.

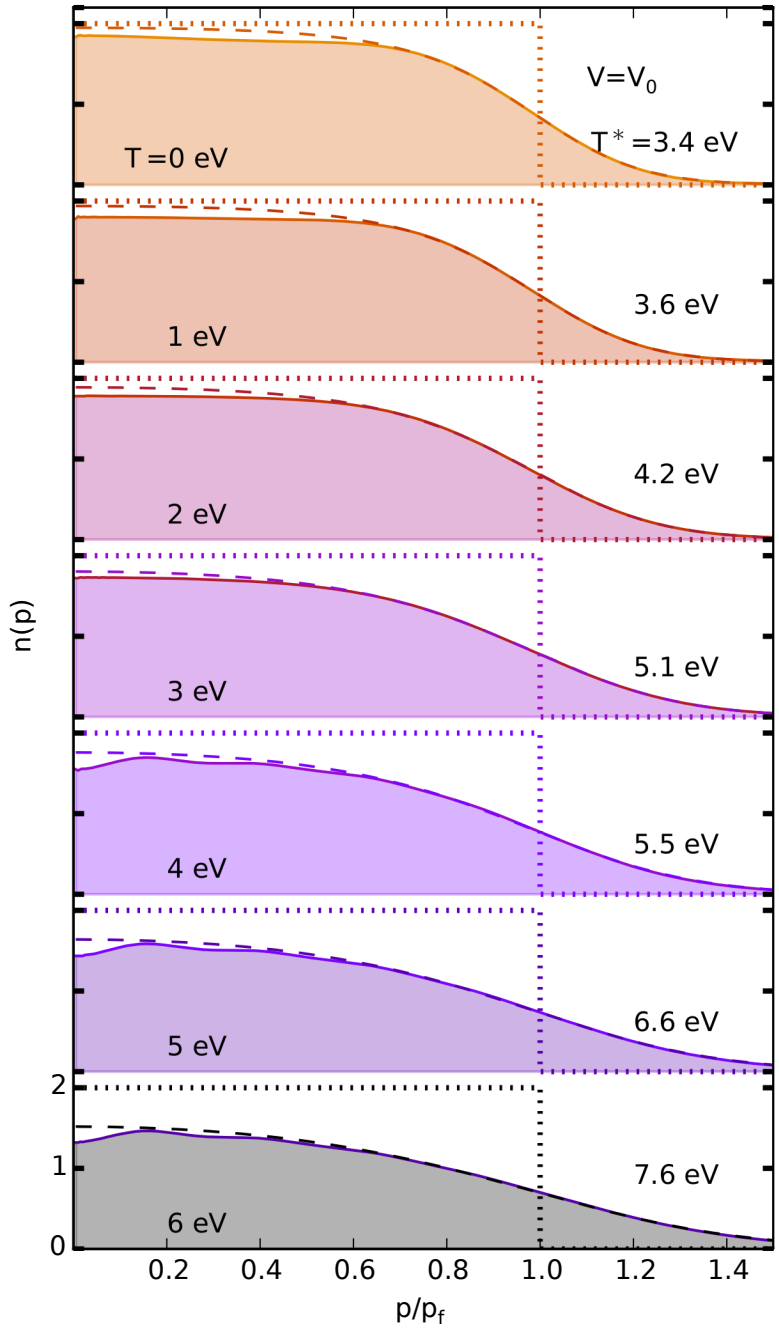

Figure 6.7: Calculations of electron momentum density $n(p)$ of the crystalline structure of Be for a number of temperatures, labeled T, and ambient density V=V_0 and the best fit of the Fermi function, together with the Fermi temperature, labeled T^*. Abscissa is normalized to corresponding Fermi momentum.

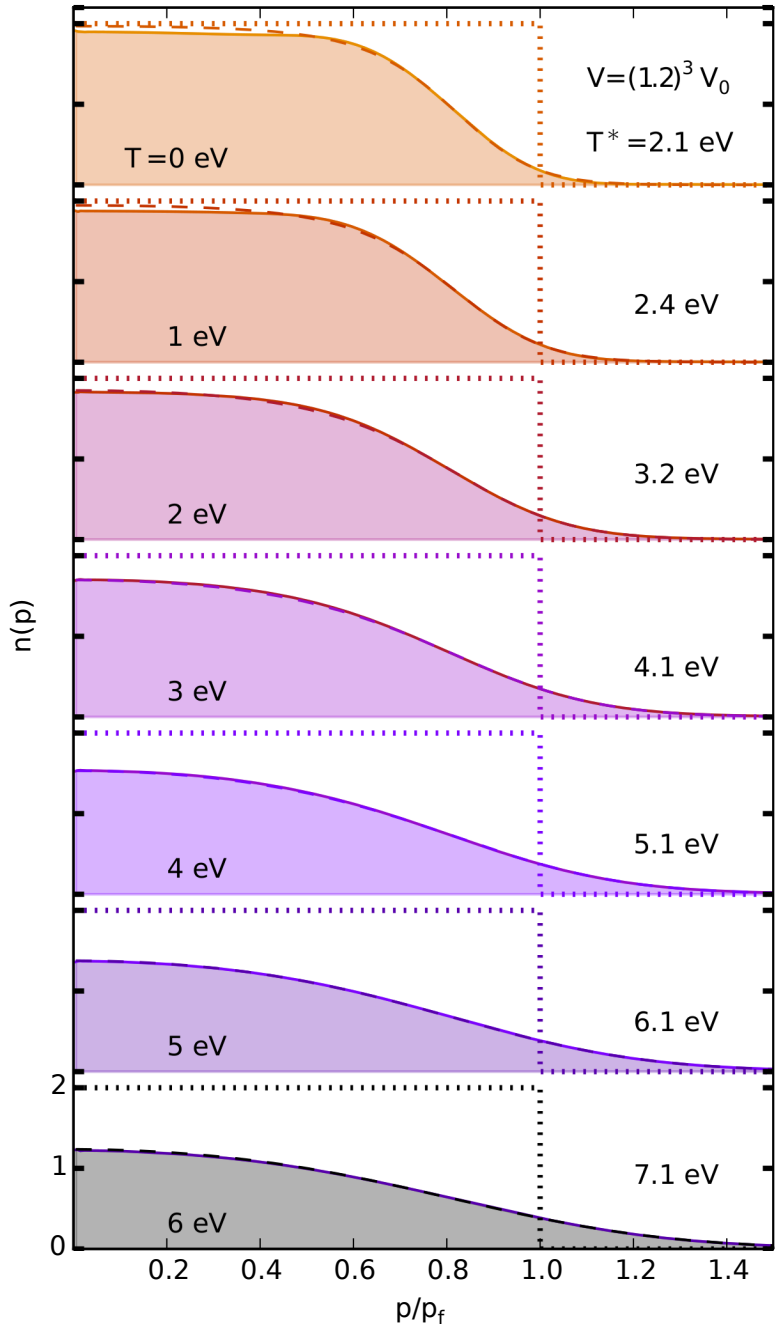

Figure 6.8: Calculations of electron momentum density $n(p)$ of the crystalline structure of Be for a number of temperatures, labeled T, expanded volume V= 1.2^3V$_0$ and the best fit of the Fermi function, together with the Fermi temperature, labeled T*. Abscissa is normalized to corresponding Fermi momentum.

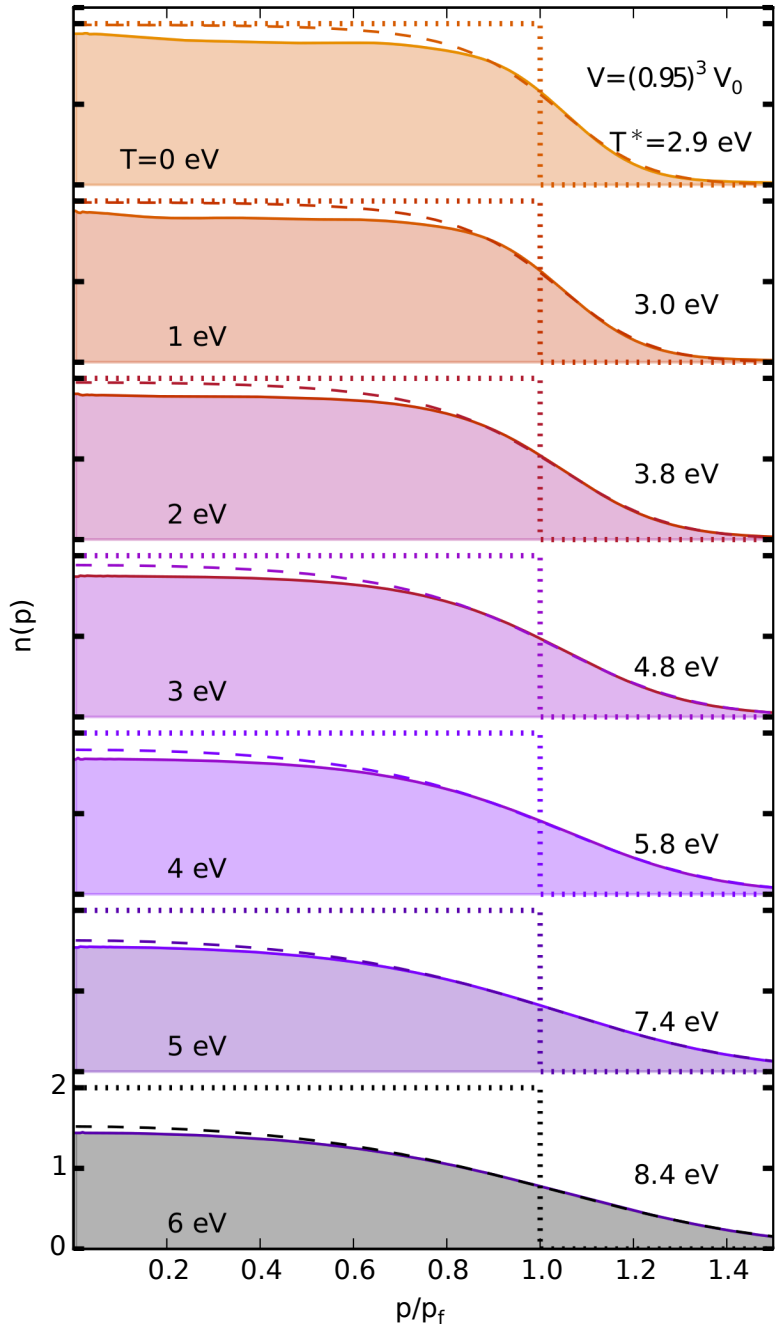

Figure 6.9: Calculations of electron momentum density $n(p)$ of the crystalline structure of Be for a number of temperatures, labeled T, contracted volume V= 0.95^3V$_0$ and the best fit of the Fermi function, together with the Fermi temperature, labeled T*. Abscissa is normalized to corresponding Fermi momentum.

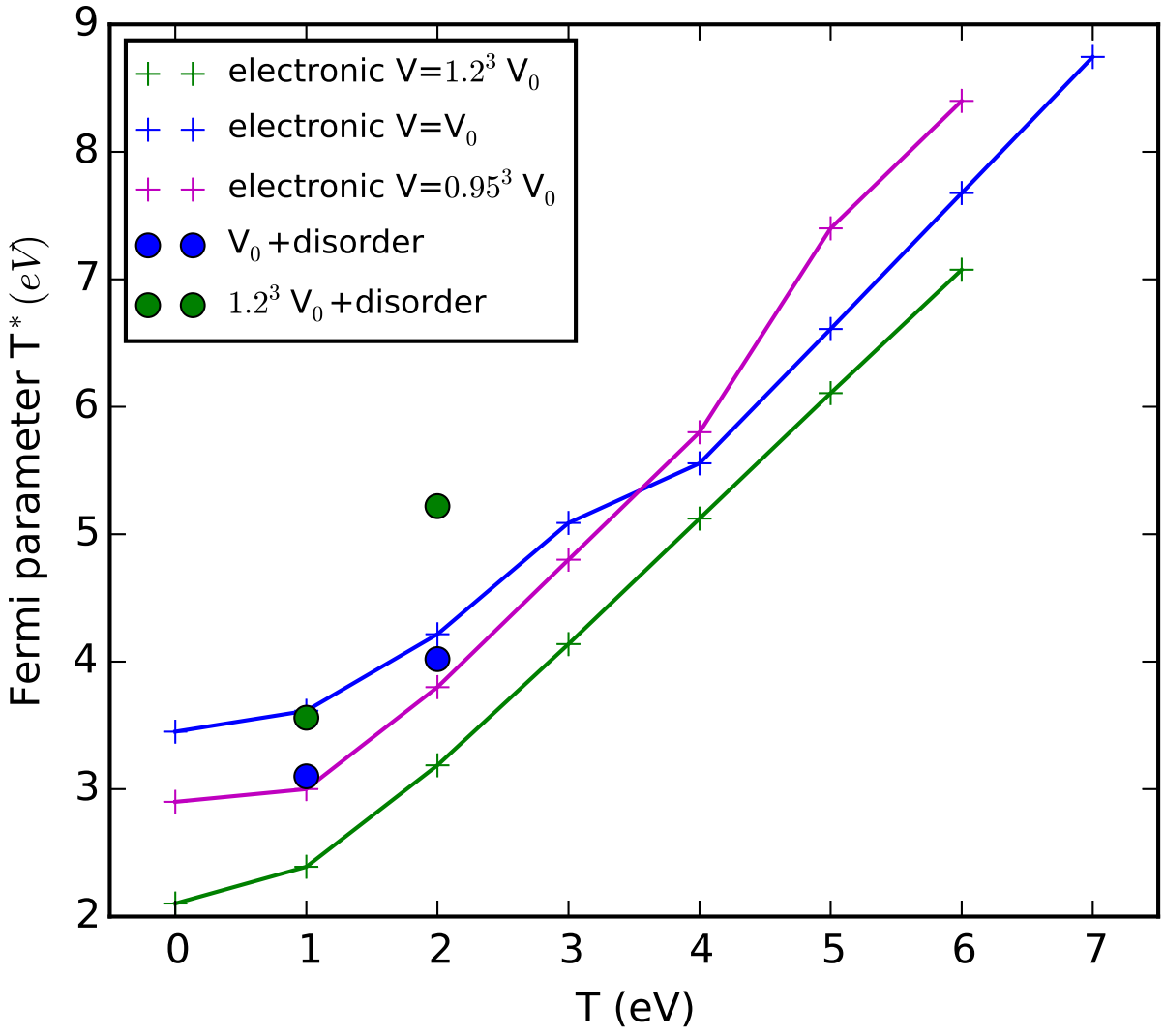

Figure 6.10: Effective temperature T^* based on Fermi-Dirac fit versus temperature T set in the calculations, for disordered and ordered system. Full lines with crosses ordered; blue and green points disordered.

Chapter 7

CONCLUSIONS

7.1 Beyond single particle approximation in TMOs XAS calculations

XAS and XPS are used in the study of electronic and geometric properties of materials. There is strong evidence that in the transition metal oxides (TMOs) there is a second channel for electrons to scatter. Specifically, in the XPS of NiO, CuO, MnO and FeO, a second peak within $\sim$ 5 eV of the main peak is observed. XAS and XPS describe the same scattering process and are directly related. Within the single particle picture, XAS of TMOs are missing a key feature which is clearly observed in the XPS experiment.

In this work I have developed a simplified, semi-empirical model which describes the effects of charge-transfer excitations in XAS, thus extending the formulation of Lee, Gunnarsson, and Hedin.[61] The spectra are modeled by a localized three-state system coupled to a photoelectron, and implemented using `FEFF` real-space Green's function *ab initio* code to include solid state effects and extrinsic losses. The final spectrum is a convolution of a single-particle XAS calculated using `FEFF`, with a frequency-dependent spectral function consisting of two delta functions that represents the localized charge-transfer excitations.

I find fairly good agreement between my results and the XAS for the metal K edges for a number of TMOs (see Fig. (4.8)). In these spectra, the presence of charge transfer satellites can clearly be seen by comparing the total XAS to the single-particle XAS, because such peaks are missing in the latter. The convolution of the single-particle spectra obtained using *ab initio* code `FEFF` with the CT spectral function $A_L(\omega, \omega')$ reproduces fairly well the peaks at higher energies with an energy splitting δE. However, the CT satellite peaks in the model spectra are sharper, which is likely an artifact of the two-delta function model for the spectral function. From Fig. (4.8) one can see that there is a noticeable discrepancy

between experimental and calculated spectra in the pre-edge region: in some cases there are missing peaks, while in others the intensities are smaller. These differences might be due to the spherical muffin-tin approximation of the scattering potential, as full potential single particle calculations[17] for CoO and NiO have greater intensities in the pre-edge region. However the main goal of this study was to approximate the CT satellite peaks. Further work is needed to obtain *ab initio* values of the parameters used in the model. In principle these could be found using constrained DFT or constrained RPA methods.

7.2 Temperature dependent CP calculations of Be, Li and Si near melting point

In the second part of this work, I present *ab initio* CP calculations of MD simulated disorder of Be, Li and Si at a number of temperatures. I study thermal disorder and thermal expansion effects on high momentum components of the CP. To calculate CP, I use the RSGF approach and the real space code `FEFF`. I find that thermal lattice expansion has a moderate effect on the CP in both Be and Li, as previously reported. Thermal disorder has the effect of smearing the high momentum umklapp peak in the case of Be and Li. I find good agreement with the experimental results and, to a good degree of precision, I predict changes in the CP of polycrystalline Si, which undergoes a solid to liquid phase transition. I observe that in the case of Li, strong directional anisotropy, which was discussed in the recent work on LiF as a function of thermal disorder, becomes weaker. For temperatures close to the melting point, both thermal disorder and thermal expansion play an important role and need to be taken into account when studying the electronic properties of Be and Li. For Si structure, which undergoes a solid to liquid phase transition, disorder has an important role and MD simulations coupled with real space theory can reasonably predict the influence of the disorder on the CP.

To summarize, I present ab initio CP calculations using the RSGF approach coupled with MD simulations, which makes the study of thermal disorder a straightforward task. I test the proposed approach on Be, Li and Si and find good agreement with the recent

experiment. I discuss in detail both the effect of thermal expansion and disorder on high and low components of CP, both of which are important and need to be taken into account. My RSGF approach coupled with MD simulations is not limited to the crystalline systems and it allows efficient study of valence electron properties of the materials as a function of thermal disorder and thermal expansion. It is ideally suited for studies of the materials across a wide range of temperatures, from zero to the WDM regime. A theory which can take solid state effects into account and quantitatively predict the changes in the CP in the WDM regime is going to be a powerful tool. I attribute differences in CP of Li as a function of temperature between LDA approach and the experiment to the many body effects, e.g. electron-electron interaction in Li. To get a better direct match of CP in the case of Li between the experiment and this current approach, further work is needed which would take into account the many body effects in the RSGF approach.

7.3 Ground state properties of Be in WDM regime

In the WDM regime I study effects of thermal disorder and electronic temperature on the valence electron ground state energy of Be. Be is of special interest due to its use in the ICF experiment[62]. The ground state properties of Be are key to the estimation of the state parameters from the NRIXS experiment. NRIXS is directly related to the CP and is perfectly suited for probing the state of matter due to its great penetration length and temporal resolution, which is on the order of picoseconds. My goals in the last part of this thesis were to:

- Study the role of disorder on valence electron ground state properties;
- Understand the interplay between ionic and electronic temperatures; and
- Develop a method which would be used in prediction of parameters of equation of state in WDM regime.

First, I investigated accessible parameter space for MD simulation program `VASP`. I did a series of simple calculations in which I varied the Be dimer distance. Energy as a function of total energy is plotted in fig. 6.2. I found that as the distance between the atoms is decreased, energy starts diverging ($\sim$ 2.5 eV) between `Gaussian` and `VASP`, which is due to the differences in atom potential approximations. This means that MD simulations above T= 2.5 eV might not be trustworthy and need to be rechecked, as should the percent of ions which get too close to neighboring atoms. Once the MD simulations were obtained, I performed CP calculations. Calculations using `FEFF` at such high temperatures and densities were never performed previously. Further investigation showed that in the case of high density, the self consistence loop fails when the atoms get too close. These two technical constraints, which can be fixed, must be kept in mind when running calculations.

Next, I compared calculations of perfect crystal with electronic temperature for a sequence of densities. I fitted a simple model of Fermi electron gas to extract temperature parameters for both perfect crystal calculations and systems with temperature disorder. The results of the analysis are presented in fig. 6.10. The differences between perfect crystal calculations and those for thermally disordered systems show that solid state effects in the WDM regime are important and simplified plasma models, which do not take into account effects of high momentum components on the valence electron ground state, are a simplification.

In future studies, the next step would be to investigate the parameter space between 1 eV and 2 eV for a number of densities which are available for the proposed MD + RSGF method. With the availability of more data points on the space diagram fig. 6.10 (system temperature versus effective temperature T^*), it would be possible to predict the temperature of the system by fitting the Fermi function to the electron momentum density.

Appendices

In the file `resolv.f90` I provide implementation of resolvent approach of Lee-Gunarson-Hedin[61].

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!Author Egor Klevak
!The code is based of the paper by Lee, Gunarson, Hedin resolvent approach
!PhysRevB.60.8034
!link:http://journals.aps.org/prb/pdf/10.1103/PhysRevB.60.8034
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
Module ResolvMod
! Multiple parameters which have been introduced in the paper
real*8,     parameter, public :: Ea    = ( 1.96d0 - 10.58d0 /2.d0 ) / 27.2d0 !
    This is in hatree
!Eb = epsilon from LEE hedin pg 8040 Ea = epsilon - U/2
! For CUCl2 Epsilon = 1.96 wich units is not clear I asume it is eV
real*8,     parameter, public :: Ec    = -155.4d0/ 27.2d0! This is hatree !
   Core hole energy
real*8,     parameter, public :: Eb    = 1.96d0 / 27.2d0
real*8,     parameter, public :: U     = 10.58d0 / 27.2d0! W treshold in
   hartree is 7.24275
real*8,     parameter, public :: t     = 1.84d0 / 27.2d0 ! W treshold in
   hartree is 7.24275
real*8,     parameter, public :: dE    = 6.45d0 ! Electron volts
real*8,     parameter, public :: phi   = 3.d-1
complex*16, parameter, public :: conim = (0,1)


CONTAINS

!Computes function M(s,k) Eq. (56)
!Input parameters:
!Mk - matrix element precalculated using Eq. (15) and wavefunctions from FEFF
!Vk - matrix element precalculated using Eq. (14) and wavefunctions from FEFF
!ne - size of the Mk and Vk matrix
```

```
subroutine resolv(Mk, Vk, ne, em, f1)
implicit none

integer, intent(in)                    :: ne
real*8, intent(in),dimension(ne, ne)   :: Vk
real*8, intent(in),dimension(ne)       :: Mk


real*8, intent(in), dimension(ne)      :: em !Matrix to calculate momentum

real*8, dimension( 2*ne, 2*ne )        :: H, Horgn, Vij
real*8, dimension( 2*ne )              :: W
real*8, dimension(:), allocatable      :: WORK
real*8, dimension( 2, 2 )              :: H0
complex*16, dimension( 2*ne )          :: M1, M2
real*8, dimension( 2 * ne )            :: msk
real*8, dimension( 2 )                 :: ws
real*8, dimension(2)                   :: v, E
real*8, dimension(ne)                  :: p2, omega
real*8                                 :: tmp, tmp1,tmp2, tmp3,tmp4, tmp5, tmp6,
    eta, tmpim1, tmpim2, M1int, M2int, Mkint
integer                                :: i, j, s, k1, k2, s1, s2, s3, nu, N,
   LDE, LWORK, info,ie1,ie2,ie3
character*30,intent(in)                :: f1
character                              :: JOBZ, UPLO
integer, parameter                     :: resolv_unit = 45
integer, parameter                     :: resolv_unit1 = 44
integer, parameter                     :: diag = 47
integer, parameter                     :: original1 = 48
integer, parameter                     :: original2 = 49
integer, parameter                     :: Mmatrix1 = 50
integer, parameter                     :: Mmatrix2 = 54
integer, parameter                     :: Mmatrix3 = 55
```

```
integer, parameter                    :: one = 51

open (unit = resolv_unit, file=f1, action = "write", status="replace")
open (unit = resolv_unit1, file="test", action = "write", status="replace")
open (unit = diag, file="diagonal", action = "write", status="replace")
open (unit = original1, file="Original1", action = "write", status="replace")
open (unit = original2, file="Original2", action = "write", status="replace")
open (unit = Mmatrix1, file="M_matrix.txt", action = "write", status="replace"
   )
open (unit = one, file="one", action = "write", status="replace")
open (unit = Mmatrix2, file="M_matrix_s2.txt", action = "write", status="
   replace")
open (unit = Mmatrix3, file="M_matrix_s3.txt", action = "write", status="
   replace")

JOBZ = 'V'  ! N  Compute eigenvalues only, V eigenvalues and eigenvectors
UPLO = 'U'  ! upper part of the triangle
N    = 2 * ne
LDE  = 2 * ne
LWORK = 3 * N -1
eta = 1.d-1

ALLOCATE(WORK(LWORK))

H0(1,1) = Ea * COS(phi)**2 + Eb * SIN(phi)**2 - t * SIN(2.d0*phi)
H0(2,2) = Ea * SIN(phi)**2 + Eb * COS(phi)**2 + t * SIN(2.d0*phi)
H0(1,2) = SIN(phi*2.d0) * (Ea - Eb) / 2.d0 + t * COS(2.d0*phi)
H0(2,1) = H0(1,2)
ws(1) = - SIN( 2.d0 * phi )
ws(2) =   COS( 2.d0 * phi )

E(1) = (Ea + Eb) / 2.d0 - DSQRT((Ea - Eb)**2 + 4 * t**2) / 2.d0
E(2) = (Ea + Eb) / 2.d0 + DSQRT((Ea - Eb)**2 + 4 * t**2) / 2.d0
```

```
89  v(1) = -SIN(phi)
    v(2) = COS(phi)
91
    p2 = em
93  !DO i = 1, ne
    !   p2(i) = REAL(SQRT(2*(em(i) - eref)))
95  !END DO

97  DO i = 1, 2*ne
      if (i.gt.ne) then
99      s1 = 2
        ie1 = i - ne
101   else
        s1 = 1
103     ie1 = i
      endif
105    DO j = 1, 2*ne
        if (j.gt.ne) then
107       s2 = 2
          ie2 = j - ne
109     else
          s2 = 1
111       ie2 = j
        endif
113
       if( i == j ) then
115     H(i,j) = H0(s1,s2) + Vk(ie1,ie2)*v(s1)*v(s2) + p2(ie2)**2/2.d0
        Horgn(i,j) = H(i,j)
117     Vij(i,j) = Vk(ie1,ie2)*v(s1)*v(s2)
       else
119     H(i,j) = H0(s1,s2) + Vk(ie1,ie2)*v(s1)*v(s2)
        Horgn(i,j) = H(i,j)
121     Vij(i,j) = Vk(ie1,ie2)*v(s1)*v(s2)
```

```
    endif

  END DO
END DO
DO i = 1, 2*ne
  DO k1 = 1, 2*ne
    write (original1, '(1(E10.3,TR1))', advance = 'NO'), Horgn(i,k1)
  END DO
  write (original1, *)
END DO
  CALL DSYEV(JOBZ, UPLO, N, H, LDE, W, WORK, LWORK, INFO)
  call wlog('Info return from diagonalization')

DO i = 1, 2*ne
 write (resolv_unit, '(E12.6)'), W(i)

  DO j = 1, 2*ne
    write (resolv_unit1, '(2(E10.3,TR1))', advance = 'NO'), H(i,j)
    tmp = 0.0d0
    tmp1 = 0.0d0
    DO k1 =1, 2*ne
      tmp = tmp + H(i,k1)*W(k1)*H(j,k1)  ! H_old = U * D * U^(T), D - diagonal
     matrics with eigen values
      DO k2 = 1, 2*ne
        tmp1 = tmp1 + H(k1,i)*Horgn(k1,k2)*H(k2,j) ! Orthogonal (i,j) element
    of the matrics  H_diag = U^(T)*H_old*U
      END DO
    END DO
    write (original2, '(1(E10.3,TR1))', advance = 'NO'), tmp
    write (diag, '(1(E10.3,TR1))', advance = 'NO'), tmp1

  END DO
```

```
153
    write (resolv_unit1, *)
155    write (diag, *)
    write (original2, *)
157 end do
DO k1 = 1, 2*ne
159 tmp3 = 0.0d0
tmp4 = 0.0d0
161 tmp5 = 0.0d0
tmp6 = 0.0d0
163   DO i =1, 2*ne
      tmp3 = tmp3 + Horgn(i,k1)*H(2,i)
165     tmp4 = tmp4 + Horgn(k1,i)*H(2,i)
      tmp5 = tmp5 + Horgn(i,k1)*H(i,2)
167     tmp6 = tmp6 + Horgn(k1,i)*H(i,2)
  END DO
169
    write (one, '(6(E15.6,TR1))'), tmp3/W(2), tmp4/W(2), H(2,k1), tmp5/W(2),
     tmp6/W(2), H(k1,2)
171 END DO

173 DO s = 1, 2
  DO i = 1, ne
175     k1 = (s-1)*ne + i
      msk(k1) = Mk(i) * ws(s)
177   END DO
END DO
179
DO i = 1, ne
181   s1 = 1
    s3 = 2
183   ie1 = (s1-1) * ne + i
    ie3 = (s3-1) * ne + i
```

```
185   tmpim1 = (0.0d0,0.0d0)
      tmpim2 = (0.0d0,0.0d0)
187   DO j = 1, 2 * ne
        DO nu = 1, 2 * ne
189       DO k1 = 1, 2 * ne
            if (k1.gt.ne) then
191           s2 = 2
              ie2 = k1 - ne
193         else
              s2 = 1
195           ie2 = k1
            endif
197
            tmpim1 = tmpim1 + ( Vij(ie1,j) * H(j,nu) * H(k1,nu) * Mk(ie2) * ws(
        s2)) / ((p2(i)**2) / 2.d0 + E(s1) - W(nu) + conim * eta ) !c^{nu}_1 is
        matrix = U
199         tmpim2 = tmpim2 + ( Vij(ie3,j) * H(j,nu) * H(k1,nu) * Mk(ie2) * ws(
        s2)) / ((p2(i)**2) / 2.d0 + E(s3) - W(nu) + conim * eta ) !c^{nu}_1 is
        matrix = U
          END DO
201     END DO
      END DO
203   M1(i) = Mk(i) * ws(s1) + tmpim1
      M2(i) = Mk(i) * ws(s3) + tmpim2
205   omega(i) = ((p2(i)**2) / 2.d0)*27.2d0
      write (Mmatrix1, '(6(E14.6,TR1))' ), M1(i), M2(i), p2(i), ((p2(i)**2) / 2.d0
        )*27.2d0
207 !   write (Mmatrix2, '(3(E14.6,TR1))' ), M2(i),((p2(i)**2) / 2.d0)*27.2d0
    END DO
209

    tmp = 0.d0
211

    DO i = 1, ne
```

```
213   tmp = omega(i)+ dE
      call terpc(omega, M1, ne, 3, tmp, M1int)
215   call terpc(omega, M2, ne, 3, tmp, M2int)
      call terp(omega, Mk, ne, 3, tmp, Mkint)
217   write (Mmatrix2, '(9(E14.6,TR1))' ), M1int, M2int, M1(i), M2(i), p2(i),
       DSQRT((2.d0 * tmp )), omega(i)
      write (Mmatrix3, '(2(E14.6,TR1))' ), (DSQRT((2.d0 * tmp ))/DSQRT((2.d0 *
       omega(i) )))*((Mkint*SIN(2*phi))/(Mk(i)*COS(2*phi)))**2, omega(i)
219 END DO

221 close(resolv_unit)
    close(resolv_unit1)
223 close(original1)
    close(original2)
225 close(diag)
    close(Mmatrix1)
227 close(one)
    close(Mmatrix2)
229 close(Mmatrix3)
    DEALLOCATE(WORK)
231 END subroutine resolv
    END MODULE ResolvMod
```

resolv.f90

This code listing contains class definition for generating `FEFF` input file from VASP output file.

```
1 # Authors: B. Mattern, E. Klevak

3 import numpy as np
  from itertools import cycle
5 from scipy.interpolate import splrep, splint
  import sys as sys
7 import copy as copy
```

```
"""
Contains useful class definitions for converting VASP output to FEFF input
    file
"""
class Lattice(object):
  def __init__(self, a, b, c, alpha=0, beta=0, gamma=0, x=1, y=1, z=1, n=10,
    tag="", filename=""):
     self.a = a
     self.b = b
     elf.c = c
     self.x = x
     self.y = y
     self.z = z
     self.alpha = alpha*np.pi/180.0
     self.beta = beta*np.pi/180.0
     self.gamma = gamma*np.pi/180.0
     self.n = n;
     self.tag = tag;
     self.filename = filename;

  def fractional_coordinates_transform_matrix(self):
     R = np.zeros((3,3))
     sgamma = np.sin(self.gamma)
     cgamma = np.cos(self.gamma)
     sbeta  =  np.sin(self.beta)
     cbeta  =  np.cos(self.beta)
     salpha =  np.sin(self.alpha)
     calpha =  np.cos(self.alpha)

     V = np.abs(1-calpha**2-cbeta**2-cgamma**2+2*calpha*cbeta*cgamma)**(0.5)

     R[0,0] = self.a
```

```
39      R[0,1] = self.b * cgamma
        R[0,2] = self.c * cbeta
41      R[1,0] = 0.0
        R[1,1] = self.b * sgamma
43      R[1,2] = self.c * (calpha-cbeta*cgamma)/sgamma
        R[2,0] = 0.0
45      R[2,1] = 0.0
        R[2,2] = self.c*V/sgamma
47
        return R
49
    def lattice(self,structure='hcp',B=0.00):
51      self.atoms=[]
        R = self.fractional_coordinates_transform_matrix()
53      dx,dy,dz = np.dot(R, np.array([self.x,self.y,self.z]))
        #dx,dy,dz=(0.0,0.0,0.0)
55      sigma = np.sqrt(B/(8.0*(np.pi)**2))
        m=0
57      potNum=0
        if B>0:
59        #np.random.seed(1)
          np.random.seed()
61        distr = np.random.normal(0.0,sigma,(np.floor(2.0*(4.0/3.0)*np.pi*(self.n
    )**3),3))
     #H, xedges, yedges = np.histogram2d(distr[:,:1], x, bins=(50,50))
63      for i in range(-self.n,self.n+1):
         for j in range(-self.n,self.n+1):
65        for k in range(-self.n,self.n+1):
           if (i**2 + j**2 + k**2)**(0.5) <= self.n:
67          xl,yl,zl = np.dot(R, np.array([i,j,k]))
            if B > 0.0:
69            #dx,dy,dz+=distr[m]
              xl +=distr[m][0]
```

```
71              yl +=distr[m][1]
                zl +=distr[m][2]
73              m += 1;
            #else:
75          #dx,dy,dz = np.dot(R, np.array([self.x,self.y,self.z]))
            #print dx,dy,dz
77          if (i,j,k) ==(0,0,0):
                self.atoms.append(Atom(
79      x = xl,
                y = yl,
81              z = zl,
                pot = 0,
83              tag = self.tag,
                r = ((xl)**2+(yl)**2+(zl)**2)**(0.5),
85              n = np.abs(i) + np.abs(j) + np.abs(k)))

87              xl,yl,zl = np.dot(R, np.array([i,j,k]))
                if B > 0.0:
89                #dx,dy,dz+=distr[m]
                  xl +=distr[m][0]*self.x
91                yl +=distr[m][1]*self.y
                  zl +=distr[m][2]*self.z
93                m += 1;

95              self.atoms.append(Atom(
            x = xl+dx,
97              y = yl+dy,
                z = zl+dz,
99              pot = 1,
                tag = self.tag,
101             r = ((xl+dx)**2+(yl+dy)**2+(zl+dz)**2)**(0.5),
                n = np.abs(i) + np.abs(j) + np.abs(k)))
103
```

```
        else:
105         self.atoms.append(Atom(
              x = xl,
107           y = yl,
              z = zl,
109           pot = 1,
              tag = self.tag,
111           r = ((xl)**2+(yl)**2+(zl)**2)**(0.5),
              n = np.abs(i) + np.abs(j) + np.abs(k)))
113
            xl,yl,zl = np.dot(R, np.array([i,j,k]))
115         if B > 0.0:
              #dx,dy,dz+=distr[m]
117           xl +=distr[m][0]*self.x
              yl +=distr[m][1]*self.y
119           zl +=distr[m][2]*self.z
              m += 1;
121


123         self.atoms.append(Atom(
            x = xl+dx,
125           y = yl+dy,
              z = zl+dz,
127           pot = 1,
              tag = self.tag,
129           r = ((xl+dx)**2+(yl+dy)**2+(zl+dz)**2)**(0.5),
              n = np.abs(i) + np.abs(j) + np.abs(k)))
131
    self.atoms.sort(key=attrgetter('r'))
133 num = 0
    for a in self.atoms:
135     if B >0.0:
    a.pot = num if  (num < 14) else 13
```

```
137             num+=1
            else:
139       a.pot = num if  (num < 1) else 1
                num+=1
141
          f = open(self.filename + '.xyz', 'w')
143       g = open(self.filename + '.inp', 'a')
          i = 0
145       for a in self.atoms:
            a.n = i
147         g.write(str(a)+"\n")
            f.write(a.xyz()+"\n")
149         i+=1
          g.write("END\n")
151     def random(self,d):
          return (self.x*random.uniform(-d,d),self.y*random.uniform(-d,d),self.z*
        random.uniform(-d,d))
153
155     def volume(self,V=0):
          sgamma = np.sin(self.gamma)
157       cgamma = np.cos(self.gamma)
          sbeta  =  np.sin(self.beta)
159       cbeta  =  np.cos(self.beta)
          salpha =  np.sin(self.alpha)
161       calpha =  np.cos(self.alpha)
          self.V = self.a*self.b*self.c*sgamma*self.c*(1-cbeta*(((calpha/(cbeta)-
        cgamma)*1.0/sgamma)**2+1.0)**(0.5))**(.5)#/(const.BOHR)**3
163       V=self.V
          return self.V
165
    class Atom(object):
167     def __init__(self, x, y, z, pot=0, tag='', r=0.0, n=0):
```

```
    self.x = x
    self.y = y
    self.z = z
    self.pot = pot
    self.tag = tag
    self.r = r
    self.n = n

  def __str__(self):
    return " %10.5f %10.5f %10.5f  %3d  %10s %10.5f % 5d" % (self.x,self.y,
     self.z,self.pot,self.tag,self.r,self.n)

  def xyz(self):
    return " %2s %10.5f %10.5f  %10.5f" % (self.tag,self.x,self.y,self.z)

  def __repr__(self):
    return "xray.feff.Atom(%10.5f, %10.5f, %10.5f)" % (self.x, self.y, self.z)

class AtomXYZ(object):
  def __init__(self, x, y, z, r):
    self.x = x
    self.y = y
    self.z = z
    self.r = r

  def xyz(self):
    return " %2s %10.5f %10.5f  %10.5f" % (self.tag,self.x,self.y,self.z)

  def __repr__(self):
    return "xray.feff.AtomXYZ(%10.5f, %10.5f, %10.5f)" % (self.x, self.y, self
    .z)
```

```
199 class CalculateRadialDistribution(object):
    """
201     Calculate pair distribution function, from the XDATCAR file,
        from the specified frame. Saves the *.png file
203 """
    def __init__(self, filename, initial = 1, frameBegin=None, frameEnd=None,xyz
     =None, pos=None,cutoff=None, fileSave=None, title = None,nbins=None):
205     if filename:
          if xyz == None and pos == None:
207         #print cutoff
            self.load(filename, initial, cutoff, frameBegin, frameEnd, fileSave,
     title, nbins)
209       elif pos != None:
            self.loadPos(filename, initial, cutoff, frameBegin, frameEnd, fileSave
     , title, nbins)
211       elif xyz != None:
            self.loadXYZ(filename,frameBegin, frameEnd)
213
    def loadXYZ(self, filename, frameBegin=None, frameEnd=None):
215
      self.filename = filename
217   self.atoms = []
      self.atoms_initial = []
219   self.atoms_final = []
      self.scale = 1
221   self.natoms = 0

223   blankCount = 0
      dist1 = []
225   l = 1
      self.xvector = (1,1,1)
227   self.yvector = (1,1,1)
      self.zvector = (1,1,1)
```

```
229     if frameBegin == None:
            frameBegin = 0
231     if frameEnd == None:
            frameEnd = sys.maxint
233
        with open(filename,"r") as f:
235       for line in f:
              if line.strip() != '' and len(line.split()) > 3 and frameEnd >=
        blankCount >= frameBegin:
237               pieces = line.split()
                  self.atoms.append(AtomXYZ(
239                   x = float(pieces[1]),
                      y = float(pieces[1+1]),
241                   z = float(pieces[1+2]),
                      r = 0))
243
              if line.strip() == '':
245               blankCount += 1
                  #if len(self.atoms()) > 0:
247               dist1.extend(self.calculate(self.atoms,cutoff=5,xyz=1))
                  self.atoms = []
249
        #print len(dist1)
251     dist1.extend(self.calculate(self.atoms,cutoff=5,xyz=1))


253
        f.close()
255     self.nframes = blankCount
        w =[]
257     for i in range(0,200):
            w.add(1.0/(4.0*pi*(i*(cutoff/200.0))**2)*cutoff/200.0)
259     hist1 =plt.hist(dist1,200,normed=true,weights=w,color='yellow' )
        plt.show()
```

```
261
    def load(self, filename, initial = 1, cutoff=None, frameBegin=None, frameEnd
     =None, fileSave=None, title = None, nbins = None):
263
      self.filename = filename
265   self.atoms = []
      if initial:
267     self.atoms_initial = []

269   self.atoms_final = []
      self.scale = 1
271   self.natoms = 0

273   if frameBegin == None:
        frameBegin = 0
275   if frameEnd == None:
        frameEnd = sys.maxint
277
      i = 0
279   blankCount = 0
      dist1 = []
281   dist2 = []
      l = 0
283   with open(filename,"r") as f:
        for line in f:
285       if i == 1:
            pieces = line.split()
287         self.scale = float(pieces[l])
          if i == 2:
289         pieces = line.split()
            self.xvector = (float(pieces[l]),float(pieces[l+1]),float(pieces[l
     +2]))
291       if i == 3:
```

```
            pieces = line.split()
293         self.yvector = (float(pieces[l]),float(pieces[l+1]),float(pieces[l
    +2]))
        if i == 4:
295         pieces = line.split()
            self.zvector = (float(pieces[l]),float(pieces[l+1]),float(pieces[l
    +2]))
297     if i == 6:
            pieces = line.split()
299         self.natoms = float(pieces[l])

301     if initial and i > 7 and i < 7 + self.natoms:
            pieces = line.split()
303         self.atoms_initial.append(AtomXYZ(
                x = float(pieces[l]) * self.xvector[0],
305             y = float(pieces[l+1]) * self.yvector[1],
                z = float(pieces[l+2]) * self.zvector[2],
307             r = 0))
            #self.atoms.append(AtomXYZ(
309         #     x = float(pieces[l]) * self.xvector[0],
            #     y = float(pieces[l+1]) * self.yvector[1],
311         #     z = float(pieces[l+2]) * self.zvector[2],
            #     r = 0))
313
    if line.strip() != '' and line.startswith("Direct")!=1 and i >= 7 +
    self.natoms and frameEnd >= blankCount >= frameBegin:
315         pieces = line.split()
            self.atoms.append(AtomXYZ(
317             x = float(pieces[l]) * self.xvector[0],
                y = float(pieces[l+1]) * self.yvector[1],
319             z = float(pieces[l+2]) * self.zvector[2],
                r = 0))
321
```

```
        if line.strip() == '' or line.startswith("Direct"):
323             blankCount += 1

325         if line.strip() == '' or line.startswith("Direct") and i > 8 and
    frameEnd >= blankCount >= frameBegin:
                #print blankCount
327             #print "Len self atoms: ",len(self.atoms)
                dist1.extend(self.calculate(self.atoms,cutoff))
329             k = 0
                self.atoms = []
331         #print blankCount
            i += 1
333
    dist1.extend(self.calculate(self.atoms,cutoff))
335
    if initial:
337   dist2.extend(self.calculate(self.atoms_initial,cutoff))

339 self.atoms = []

341 f.close()
    #for p in dist1: print '%0.5e' % p
343
    self.nframes = i - blankCount - 8
345 if (nbins != None):
        bins = nbins
347 else:
        bins = 200
349
    w =[]
351 dr = cutoff/bins

353 for element in dist1:
```

```
        w.append(1.0/(4.0*pi * element**2 * dr))
355
357     ht1 = plt.hist(dist1, bins, normed=True, weights = w, color='yellow' )
359     if initial:
          hist2 = plt.hist(dist2, bins, normed=True, color='red')
361
        if (True and initial):
363       for i in range(0,bins):
            print ht1[1][i],ht1[0][i],hist2[0][i]
365     elif (True):
          for i in range(0,bins):
367         print ht1[1][i],ht1[0][i]
369     plt.xlim(0,1)
        if fileSave != None:
371             if cutoff != None:
              plt.xlim((1.5,cutoff))
373       plt.ylim((0,2))
          plt.title(title)
375       fig = plt.gcf()
          plt.savefig(fileSave+".png")
377
        plt.show()
379
      def loadPos(self, filename, cutoff=None, frameBegin=None, frameEnd=None,
        fileSave=None, title = None, nbins = None):
381
        self.filename = filename
383     self.atoms = []
        self.scale = 1
385     self.natoms = 0
```

```
        self.xsumm = []
387     self.ysumm = []
        self.zsumm = []
389     self.xpos = []
        self.ypos = []
391     self.zpos = []


393
        if frameBegin == None:
395         frameBegin = 0
        if frameEnd == None:
397         frameEnd = sys.maxint

399     i = 0
        blankCount = 0
401     dist1 = []
        dist2 = []
403     l = 0
        num = 0
405     with open(filename,"r") as f:
          for line in f:
407         if i == 1:
                pieces = line.split()
409             self.scale = float(pieces[l])
            if i == 2:
411             pieces = line.split()
                self.xvector = (float(pieces[l]),float(pieces[l+1]),float(pieces[l
    +2]))
413         if i == 3:
                pieces = line.split()
415             self.yvector = (float(pieces[l]),float(pieces[l+1]),float(pieces[l
    +2]))
            if i == 4:
```

```
417             pieces = line.split()
                self.zvector = (float(pieces[l]),float(pieces[l+1]),float(pieces[l
    +2]))
419         if i == 6:
                pieces = line.split()
421             self.natoms = float(pieces[l])

423         if i > 7 and i < 7 + self.natoms:
                pieces = line.split()
425             self.xpos.append(float(pieces[l]) * self.xvector[0])
                self.ypos.append(float(pieces[l+1]) * self.xvector[1])
427             self.zpos.append(float(pieces[l+2]) * self.xvector[2])

429             self.xsumm.append(float(pieces[l]) * self.xvector[0])
                self.ysumm.append(float(pieces[l+1]) * self.xvector[1])
431             self.zsumm.append(float(pieces[l+2]) * self.xvector[2])

433         if line.strip() != '' and i >= 7 + self.natoms and frameEnd >=
    blankCount >= frameBegin:
                pieces = line.split()
435             self.xpos.append(float(pieces[l]) * self.xvector[0])
                self.ypos.append(float(pieces[l+1]) * self.xvector[1])
437             self.zpos.append(float(pieces[l+2]) * self.xvector[2])

439             self.xsumm[k]+=(float(pieces[l]) * self.xvector[0])
                self.ysumm[k]+=(float(pieces[l+1]) * self.xvector[1])
441             self.zsumm[k]+=(float(pieces[l+2]) * self.xvector[2])

443             k += 1
                if k == 10:
445                 num += 1


447
```

```
        if line.strip() == '':
449             blankCount += 1
                k = 0
451
        i += 1
453     f.close()
        self.nframes = i - blankCount - 8
455     if (nbins != None):
            bins = nbins
457     else:
            bins = 200
459
    def calculate(self, atoms, cutoff=None, xyz=None):
461     if cutoff == None:
           cutoff = min([self.xvector[0],self.yvector[1],self.zvector[2]])
463        if cutoff > 5:
              cutoff = 4
465
        dist = []
467     m = 0
        d = 0
469     if xyz==None:
          for atom1 in atoms:
471          for atom2 in atoms:
                for ix in range(-1,2):
473               dx = (atom1.x - (ix * self.xvector[0] + atom2.x))**2
                  if (dx**(0.5) < cutoff):
475                 for iy in range(-1,2):
                      dy = dx + (atom1.y - (iy * self.yvector[1] + atom2.y))**2
477                   if (dy**(0.5) < cutoff):
                        for iz in range(-1,2):
479                       d =(dy +(atom1.z - (iz * self.zvector[2] + atom2.z))**2)
      **(0.5)
```

```
481                         if d != 0 and d < cutoff:
                              dist.append(d)
483                           m += 1

485     else:
          for atom1 in atoms:
487         for atom2 in atoms:
                    d = 0.0
489                 d += (atom1.x - atom2.x)**2
                    d += (atom1.y - atom2.y)**2
491                 d += (atom1.z - atom2.z)**2

493                 d = d **(0.5)

495                 if d != 0 and d < cutoff:
                      dist.append(d)
497                   m += 1

499     return dist

501 def vasp_atoms(feff_input, bounds):
    xmin,xmax = bounds[0]
503 ymin,ymax = bounds[1]
    zmin,zmax = bounds[2]
505
    dx = xmax - xmin
507 dy = ymax - ymin
    dz = zmax - zmin
509
    # limit to atoms within bounds
511 atoms_in_box = [a for a in feff_input.atoms if
                    xmin <= a.x <= xmax and
```

```
513                     ymin <= a.y <= ymax and
                        zmin <= a.z <= zmax]
515
    # scale to box coordinates
517   vatoms = []
    for a in atoms_in_box:
519     x = (a.x - xmin) / dx
      y = (a.y - ymin) / dy
521     z = (a.z - zmin) / dz
      vatoms.append((x,y,z))
523
525   print "Comment"
    print "     1.000"
527   print "     %10.5f %10.5f %10.5f" % (dx,0,0)
    print "     %10.5f %10.5f %10.5f" % (0,dy,0)
529   print "     %10.5f %10.5f %10.5f" % (0,0,dz)
    print "Selective dynamics"
531   print "Direct"
    for va in vatoms:
533     print "     %10.5f %10.5f %10.5f F F F" % va
535
# class for making potential part of feff.inp file using XDATCAR file from
    VASP
537 class vaspXDATCARToFeff(object):

539   # constructor of the object, which takes the XDATCAR filename and the frame
    number and
    # generates pos array consisting of objects of Atom class
541   def __init__(self, fileName, frame=None, cutOff=None):
      if (fileName and cutOff):
543       pos = self.loadPos(fileName,frame, cutOff)
```

```
    self.writeFeff(pos)
545
    # loads position of the atoms as from XDATCAR, returns array of Atom in feff
      form
547 def loadPos(self, filename, frame, cutOff):
      self.filename = filename
549   self.atoms = []
      self.atomLabel = []
551
      self.atoms_final = []
553   self.scale = 1
      self.natomsArray = []
555   self.natoms = 0

557   i = 0
      blankCount = 0
559   l = 0
      self.atomCounter = 0
561
      with open(filename,"r") as f:
563     for line in f:
        # Read in the thirst six lines of the file
565     if i == 0:
            pieces = line.split()
567         for atom in pieces:
              self.atomLabel.append(atom)
569     if i == 1:
            pieces = line.split()
571         self.scale = float(pieces[l])
        if i == 2:
573         pieces = line.split()
            self.xvector = (float(pieces[l]),float(pieces[l+1]),float(pieces[l
    +2]))
```

```
575         if i == 3:
                pieces = line.split()
577             self.yvector = (float(pieces[l]),float(pieces[l+1]),float(pieces[l
    +2]))
        if i == 4:
579             pieces = line.split()
                self.zvector = (float(pieces[l]),float(pieces[l+1]),float(pieces[l
    +2]))
581     if i == 6:
            pieces = line.split()
583   for numberOfAtoms in pieces:
              self.natomsArray.append(float(numberOfAtoms))
585           self.natoms += int(numberOfAtoms)

587     #if (len(self.natoms) != len(self.atomLabel)):
        #   print "Number of atom entries and number of atoms is different"
589
        if line.strip() != '' and frame == blankCount:
591         #if atomCounter > numberOfAtoms:
            #   print "Too many atom entries in the frame: ", atomCounter
593
    if (self.atomCounter == 0):
595             pieces = line.split()

597             self.atomInit = Atom(
                    x = self.xvector[0]*float(pieces[l]),
599                 y = self.yvector[1]*float(pieces[l+1]),
                    z = self.zvector[2]*float(pieces[l+2]),
601                 pot = 0,
                    tag = self.atomLabel[0],
603                 r = 0,
                    n = self.atomCounter)
605
```

```
                self.atoms.append(Atom(
607                 x = 0,
                    y = 0,
609                 z = 0,
                    pot = 0,
611                 tag = self.atomLabel[0],
                    r = 0,
613                 n = self.atomCounter))
                self.atomCounter += 1
615
                for ix in range(-1,2):
617                     dist = 0.0
                        xtran =  self.xvector[0] * (ix + float(pieces[l])) -
    self.atomInit.x
619                     #xtran =  self.xvector[0] * ix
                        dist += (xtran)**2
621                     if (dist**(0.5) < cutOff):
                           #print dist**(0.5)
623                        for iy in range(-1,2):
                              ytran =  self.yvector[1] * (iy + float(pieces[l
    +1])) - self.atomInit.y
625                           #ytran =  self.yvector[1] * iy

627                           if ((dist+(ytran)**2)**(0.5) < cutOff):
                              dist += (ytran)**2
629                           for iz in range(-1,2):
                                 ztran =  self.zvector[2] * (iz + float(
    pieces[l+2])) - self.atomInit.z
631                              #ztran =  self.zvector[2] * iz
                                 #dist += (ztran)**2
633
                                 if (dist + (ztran)**2)**(0.5) < cutOff:
635                                 dist += (ztran)**2
```

```
                                        self.atoms.append(Atom(
637                                         x = xtran,
                                            y = ytran,
639                                         z = ztran,
                                            pot = 1,
641                                         tag = self.atomLabel[0],
                                            r = dist**(0.5),
643                                         n = self.atomCounter))
                                        self.atomCounter+=1
645
            else:
647             pieces = line.split()

649           dist = 0.0
              for ix in range(-1,2):
651                 dist = 0.0
                    xtran =  self.xvector[0] * (ix + float(pieces[1])) -
    self.atomInit.x
653                 #xtran =  self.xvector[0] * ix
                    dist += (xtran)**2
655                 if (dist**(0.5) < cutOff):
                       for iy in range(-1,2):
657                       ytran =  self.yvector[1] * (iy + float(pieces[1
    +1])) - self.atomInit.y
                          #ytran =  self.yvector[1] * iy
659
                          if ((dist+(ytran)**2)**(0.5) < cutOff):
661                          dist += (ytran)**2
                             for iz in range(-1,2):
663                             ztran =  self.zvector[2] * (iz + float(
    pieces[1+2])) - self.atomInit.z
                                #ztran =  self.zvector[2] * iz
665                             #dist += (ztran)**2
```

```
667                                 if (dist + (ztran)**2)**(0.5) < cutOff:
                                        dist += (ztran)**2
669                                     self.atoms.append(Atom(
                                            x = xtran,
671                                         y = ytran,
                                            z = ztran,
673                                         pot = 1,
                                            tag = self.atomLabel[0],
675                                         r = dist**(0.5),
                                            n = self.atomCounter))
677                                     self.atomCounter+=1

679 # Update the frame counter
        if line.strip() == '':
681         blankCount += 1

683     # Update the counter of the line number
        i += 1
685
    f.close()
687
    self.atoms.sort(key=attrgetter('r'))
689 return self.atoms

691 def writeFeff(self,pos):
        i = 0
693     for atom in pos:
          print ' {0:+.5f} {1:+.5f} {2:+.5f} {3:3d} {4} {5:+.5f} {6}'.format(
    atom.x, atom.y, atom.z, atom.pot, atom.tag, atom.r, i-1)
695         i += 1
```

feff.py

In this listing I have program which created input files for Lebedev quadrature calulations of CP.

```
#!/usr/bin/env python
import os, sys
import subprocess
import lebedev.grids as grids

"""
Used in the pbs driver script for setting up calculations of multiple angles
using Lebedev quadrature
"""

COMPTON_PATH = "compton"

skel=""" run compton module?
           1
pqmax, npq
   5.000000             1000
ns, nphi, nz, nzp
  32  32  32 192
smax, phimax, zmax, zpmax
      {smax}      6.28319      {zmax}      {zpmax}
jpq? rhozzp? force_recalc_jzzp?
 T F T
window_type (0=Step, 1=Hann), window_cutoff
           1  0.0000000E+00
temperature (in eV)
      0.00000
set_chemical_potential? chemical_potential(eV)
 F  0.0000000E+00
rho_xy? rho_yz? rho_xz? rho_vol? rho_line?
 F F F F F
qhat_x qhat_y qhat_z
```

```
  {qhat}
"""

def write_compton_input(qhat, scale=1.0):
  smax = 3.5 * scale
  zmax = 3.5 * scale
  zpmax = 16 * scale

  qhatstr = ' '.join(['%.10f'%f for f in qhat])
  inpstr = skel.format(smax=smax, zmax=zmax, zpmax=zpmax, qhat=qhatstr)
  with open("compton.inp", "w") as f:
    f.write(inpstr)

def run_compton(nprocs, nodefile, tag):

  cmd = ["mpirun", "-np", str(nprocs), "-machinefile", nodefile, COMPTON_PATH]
  ret = subprocess.call(cmd)

  if ret != 0:
    print("Error. Aborting.")
    return False

  ret = subprocess.call(["cp", "compton.dat", "compton.{tag}.dat".format(tag=
    tag)])

  if ret != 0:
    print("Error copying. Aborting.")
    return False

  return True
def copy(tag, qhat, npts):
  res = subprocess.call(["ls","-l","compton.{tag}.dat".format(tag=tag)])
  if (res == 0):
```

```
        return True
  for j in grids.grid_sizes:
    if j < npts:
      grid2 = grids.grids[j]
      for n,(x2,y2,z2,w2) in enumerate(grid2):
        if qhat == [x2,y2,z2]:
          tag2 = "L{j}.{n}".format(j=j, n=n)
          res = subprocess.call(["ls","-l","compton.{tag2}.dat".format(tag2=
    tag2)])
          if (res == 0):
            subprocess.call(["cp", "compton.{tag2}.dat".format(tag2=tag2), "
    compton.{tag}.dat".format(tag=tag)])
            with open("compton.{tag}.dat".format(tag=tag), "a") as myfile:
              myfile.write("# Original file compton.{tag2}.dat\n".format(tag2=
    tag2))
            return True
  return False

nprocs = sys.argv[1]
nodefile = sys.argv[2]
totpts = 0
for npts in grids.grid_sizes:
  if totpts > 2000: break
  grid = grids.grids[npts]
  for i,(x,y,z,w) in enumerate(grid):
      if (npts > 40 and npts < 51):
        qhat = [x,y,z]
        tag = "L{npts}.{i}".format(npts=npts, i=i)
        if (copy(tag, qhat, npts) == False):
          write_compton_input(qhat)
          subprocess.call(["cp", "compton.inp", "compton.{tag}.inp".format(tag
    =tag)])
          with open("compton.{tag}.dat".format(tag=tag), "w") as f:
```

```
93              f.write("")

95            ret = run_compton(nprocs, nodefile, tag)
              if not ret:
97              exit(1)
              totpts += 1
```

run_lebedev.py

Auxilary file for averaging over multiple CPs calculatet calculated using Lebedev Quadrature.

```
1 #!/usr/bin/env python
  import os, sys
3 import subprocess
  import lebedev.grids as grids
5 import numpy as np
  """
7 Calculates averaged over the angles using Lebedev quadrature
  """
9 totpts = 0
  pq = []
11 J_avg = []
  density = ['100']
13 for m in density:
     for npts in grids.grid_sizes:
15    if npts==50:
        grid = grids.grids[npts]
17      favg = "compton.L{npts}.{m}.avg.dat".format(npts=npts,m=m)
        pq = []
19      J_avg = []
        for i,(x,y,z,w) in enumerate(grid):
21        filename = "compton.L{npts}.{i}.dat".format(npts=npts, i=i)
          li_num = 0
23

          for line in open(filename,"r"):
```

```
25          li=line.strip()

27          if not li.startswith("#"):
              word = li.rsplit(" ")
29
              if i == 0:
31              pq.append(float(word[0]))
                J_avg.append(float(word[len(word)-1])*w)
33
              else:
35              J_avg[li_num]+=float(word[len(word)-1])*w
                li_num+=1
37        totpts += 1
      print len(J_avg), len(pq)
39    with open(favg,"w") as f:
       for  i in range(-len(J_avg)+1,len(J_avg)):
41       if i < 0 :
           print >> f,'%.15e'%(-pq[-i]),'%.15e'%(J_avg[-i])
43       else:
           print >> f,'%.15e'%(pq[i]),'%.15e'%(J_avg[i])
```

lebedev_avg.py

This file listing contains program which averages over frames.

```
1 #!/usr/bin/env python
  import os, sys
3 import subprocess
  import lebedev.grids as grids
5 import numpy as np
  import pandas as pd
7
  pd.set_option('display.precision',25)
9 totpts = 0
  pq = []
```

```
J_avg = []
J=np.array([])
J_f=np.array([])
ftot = 25
df = pd.DataFrame()
for i in range(1,ftot):
  favg = "frame{i}/compton.L50.100.avg.dat".format(i=i)
  df[i] = pd.Series(np.loadtxt(favg)[:,1])
  J=np.append(J,np.loadtxt(favg)[:,1])
  if i == 1:
   pq = np.append(pq,np.loadtxt(favg)[:,0])

#df = pd.DataFrame(J.reshape(1999,ftot-1))
#print df.to_string()
#print df.shape
#print df.to_string()
mean = pd.DataFrame(pd.DataFrame(pq))
mean[1] = pd.Series(df.mean(1))
mean[2] = pd.Series(df.std(1))
#print mean
if (True):
  print "# frames ",ftot
  print "# Generated using python pandas and average_frames.py"
  print "# pq mean std"
  print mean.to_string(index = False, header = False, float_format='{:,.15e}'.
   format)
```

average_frames.py

This listing contains varius conversion functs from CP as well as function which calculates electron momentum density of CP.

```
#!/usr/bin/env python2
"""
Compton Profile Convertor
```

```
# Authors: B. Mattern, E. Klevak
Usage:
  {cmd} jpq2sqw <filename> <theta> <energy> [options]
  {cmd} sqw2jpq <filename> <theta> <energy> [options]
  {cmd} jpq2rhop <filename> [options]
  {cmd} xrts2sqw <filename> <energy> [options]
  {cmd} summ_rule_compton <filename> [options]
  {cmd} -h

Options:
  -h --help                Show this screen
  -w --omega=<omega>       Energy transfer "start,end[,step]" (e.g.,
    "0,1001,0.5")
  -s --scattered           <energy> is scattered photon energy instead of
                           incident photon energy
  -n --num-electrons=N     Number of electrons to normalize J(pq) to.
  -a --atomic              Input file is atomic units
  -o --output=<filename>   Filename to save output to [default: stdout]
  -S --symmetrize          Symmetrize CP
  -c --column=<column>     Column in file containing data (only in xrts2sqw)
  -d --debug               Show debugging output
"""
from docopt import docopt
import sys, os
import numpy as np
from schema import Schema, Or, And, Use
from xray import compton, analysis, const

is_positive = lambda x: x > 0
open_output = lambda o: sys.stdout if o == 'stdout' else open(o, 'w')

def parse_omega(omega):
  if omega == None:
```

```
    w = np.linspace(0,1500,1501)
  else:
    tokens = omega.split(',')
    wstep = 1.0
    if len(tokens) == 3:
      wstep = float(tokens.pop())

    if len(tokens) < 2:
      raise ValueError("--omega must be of the form 'start,end[,step]'")

    w1,w2 = [float(t) for t in tokens]
    w = np.arange(w1,w2+wstep/2,wstep)

    return w



opts_schema = Schema({
  '<filename>': Schema(os.path.exists, error='Input file does not exist.'),
  '<theta>': Or(None, And(Use(float), lambda x: 0 <= x <= 180,
                  error='Invalid theta value')),
  '<energy>': Or(None, And(Use(float), is_positive,
                   error='Invalid energy')),
  '--omega': Use(parse_omega, error='Invalid omega'),
  '--num-electrons': Or(None,
                        And(Use(float), is_positive)
                       ),
  '--scattered': bool,
  '--atomic': bool,
  '--symmetrize': bool,
  '--column': Or(None, And(Use(int))),
  '--output': Use(open_output),
  '--debug': bool,
```

```
    '--help': bool,
    'jpq2sqw': bool,
    'sqw2jpq': bool,
    'jpq2rhop': bool,
    'xrts2sqw': bool,
    'summ_rule_compton': bool
})


DEBUGGING = False
def debug(msg):
  if DEBUGGING:
    sys.stderr.write(str(msg))
    sys.stderr.write("\n")

def jpq2sqw(args):
    filename = args['<filename>']
    theta = args['<theta>']*np.pi/180
    energy = args['<energy>']
    scattered = args['--scattered']
    num_electrons = args['--num-electrons']
    atomic_units = args['--atomic']
    out = args['--output']

    cp = compton.ComptonProfile.from_file(filename, num_electrons=
    num_electrons, atomic_units=atomic_units)

    w = np.linspace(0,1500,1501)

    energy_key = 'E2' if scattered else 'E1'
    kwargs = {energy_key: energy}
    sqw = cp.to_sqw_theta(w, theta, **kwargs)
```

```
101     header = ("# S(q,w) generated from J(p_q)\n"
                  "# Source: {filename:s}\n"
103               "# theta: {theta:.2f}\n"
                  "# {which_energy:s}: {energy:.2f}\n"
105               "#\n# w  S  q\n").format(
                      filename=os.path.abspath(filename),
107                   theta=args['<theta>'],
                      which_energy=energy_key,
109                   energy=energy)
      out.write(header)
111   sqw.save(out)

113 def sqw2jpq(args):

115   filename = args['<filename>']
      theta = args['<theta>']
117   energy = args['<energy>']
      scattered = args['--scattered']
119   num_electrons = args['--num-electrons']
      atomic_units = args['--atomic']
121   out = args['--output']

123   sqw = analysis.Curve.from_file(filename)

125   if np.abs(np.log(energy) - np.log(sqw.x).mean()) < 1:
        sys.stderr.write("Input file appears to be in terms of scanned energy
       instead of energy transfer. Treating it as such.\n")
127     if scattered: # scattered energy is fixed, sqw.x is incident
          sqw.x = (sqw.x - energy)[::-1]
129     else:
          sqw.x = (energy - sqw.x)[::-1]
131   sqw.y = sqw.y[::-1]
```

```
133   if scattered:
        E1, E2 = energy + sqw.x, energy
135   else:
        E1, E2 = energy, energy - sqw.x
137
      q = compton.momentum_transfer(E1, E2, theta*np.pi/180)
139   cp = compton.sqw_to_jpq(sqw.y, q, sqw.x)

141   if num_electrons:
        # XXX check that tails are present. otherwise warn...
143     cp = cp.normalize_integral(num_electrons)

145   if atomic_units:
        cp.x *= const.BOHR
147     cp.y /= const.BOHR

149   cp.save(out)


151
    def jpq2rhop(args):
153     filename = args['<filename>']
        num_electrons = args['--num-electrons']
155     atomic_units = args['--atomic']
        symmetrize = args['--symmetrize']
157     out = args['--output']

159     cp = compton.ComptonProfile.from_file(filename, num_electrons=
        num_electrons, atomic_units=atomic_units)

161     if symmetrize:
          cp = cp.symmetrize()
163
        rhop = cp.to_rhop()
```

```
165
    header = ("# rho(p) generated from J(p_q)\n"
167             "# Source: {filename:s}\n"
                "#\n# p         rho(p)\n").format(
169                 filename=os.path.abspath(filename),
                    )
171     out.write(header)
        rhop.save(out)
173
def xrts2sqw(args):
175     filename = args['<filename>']
        w1 = args['<energy>']
177     column = args['--column'] or 1
        out = args['--output']
179


181     data = analysis.Curve.from_file(filename, y=column)

183     # switch to energy tranfer
        w2 = data.x
185     w = (w1 - w2)[::-1]
        S = (data.y * (w1/w2)**2)[::-1]
187
        sqw = analysis.Curve(w, S)
189
        header = ("# S(q,w) generated from XRTS output\n"
191             "# Source: {filename:s}\n"
                "#\n# w         S\n").format(
193                 filename=os.path.abspath(filename),
                    )
195     out.write(header)
        sqw.save(out)
197
```

```
def summ_rule_compton(args):
    filename = args['<filename>']
    column = args['--column'] or 1
    symmetrize = args['--symmetrize']
    out = args['--output']
    #print args

    data = analysis.Curve.from_file(filename, y=column)

    if symmetrize:
      data = data.extend_symmetric()
    #print data
    print analysis.summ_rule(data)


if __name__ == "__main__":
  usage = __doc__.format(cmd=os.path.basename(sys.argv[0]))
  args = docopt(usage, version="Compton 0.1)")

  if args['--debug']: DEBUGGING = True
  debug(args)

  try:
    args = opts_schema.validate(args)
  except Exception as e:
    #sys.stdout.write(usage)
    #sys.stdout.write("\n")
    sys.exit(e.code)
    #sys.stderr.write(str(e) + "\n")

  cmds = ['jpq2sqw', 'sqw2jpq', 'jpq2rhop', 'xrts2sqw', 'summ_rule_compton']

  for cmd in cmds:
```

```
231     if args[cmd]:
          locals()[cmd](args)
233       exit()
```

compton.py

VITA

Egor Klevak was born in 1984 in the city of Kishinev. As a kid, he spent summers on the Danube river fishing, swimming and kayaking. Egor moved to Saint-Petersburg in 1998 to study at a boarding school affiliated with Saint-Petersburg State University (SPSU), where he greatly enjoyed being student and studying Physics and Mathematics. In 2005, he graduated from SPSU with a B.S. in Physics. That same year, he started the Masters program at SPSU, graduating with an M.S in Physics in 2008. Egor moved to Rhode Island later that year to reunite with his family, who had already immigrated to the US. After a short and pleasant time spent working in the University of Pennsylvania's Bioinformatics Group, Egor moved to Seattle in 2010 to start the physics PhD program at the University of Washington (UW). Egor completed his PhD in physics in 2016 and, having fallen in love with the Pacific Northwest, intends to remain in the Seattle area.

BIBLIOGRAPHY

[1] 1s x-ray photoemission collected at the Hiroshima Synchrotron radiation center at. http://www.hsrc.hiroshima-u.ac.jp/cuprates.htm. Accessed: 2014-01-15.

[2] Teijo Åberg. Theory of x-ray satellites. *Phys. Rev.*, 156:35–41, Apr 1967.

[3] Carl-Olof Almbladh and L. Hedin. *Handbook on synchrotron radiation*, volume 1B. North-Holland, 1983.

[4] N Amadou, E Brambrink, A Benuzzi-Mounaix, G Huser, F Guyot, S Mazevet, G Morard, T de Resseguier, T Vinci, K Myanishi, et al. Direct laser-driven ramp compression studies of iron: A first step toward the reproduction of planetary core conditions. *High Energy Density Phys.*, 9(2):243–246, 2013.

[5] Hans C. Andersen. Molecular dynamics simulations at constant pressure and/or temperature. *The Journal of Chemical Physics*, 72(4):2384–2393, 1980.

[6] A. L. Ankudinov. *Self-energy models in XAFS, XANES and XMCD*. PhD thesis, University of Washington, 1996.

[7] A. L. Ankudinov and J. J. Rehr. Relativistic calculations of spin-dependent x-ray-absorption spectra. *Phys. Rev. B*, 56:R1712–R1716, Jul 1997.

[8] AL Ankudinov, B Ravel, JJ Rehr, and SD Conradson. Real-space multiple-scattering calculation and interpretation of x-ray-absorption near-edge structure. *Phys. Rev. B*, 58(12):7565, 1998.

[9] Lundqvis. B. SINGLE-PARTICLE SPECTRUM OF DEGENERATE ELECTRON

GAS .3. NUMERICAL RESULTS IN RANDOM PHASE APPROXIMATION. *Phys. Konden. Mater.*, 7(2), 1968.

[10] A. Bansil, B. Barbiellini, S. Kaprzyk, and P.E. Mijnarends. Electron momentum density and compton profile in disordered alloys. *J. Phys. Chem. Solids*, 62(12):2191 – 2197, 2001.

[11] A. Bansil, R. S. Rao, P. E. Mijnarends, and L. Schwartz. Electron momentum densities in disordered muffin-tin alloys. *Phys. Rev. B*, 23:3608–3616, Apr 1981.

[12] A. Bansil, R. S. Rao, P. E. Mijnarends, and L. Schwartz. Electron momentum densities in disordered muffin-tin alloys. *Phys. Rev. B*, 23:3608–3616, Apr 1981.

[13] B. Barbiellini, Ch. Bellin, G. Loupias, T. Buslaps, and A. Shukla. How the hydrogen bond in nh_4F is revealed with compton scattering. *Phys. Rev. B*, 79:155115, Apr 2009.

[14] P. E. Blöchl. Projector augmented-wave method. *Phys. Rev. B*, 50:17953–17979, Dec 1994.

[15] S. Bocharov, Th. Kirchner, G. Dräger, O. Šipr, and A. Šimůnek. Dipole and quadrupole contributions to polarized cu ¡b¿¡i¿k¡/i¿¡/b¿ x-ray absorption near-edge structure spectra of cuo. *Phys. Rev. B*, 63:045104, Jan 2001.

[16] Helmut Bross. Electronic structure of li with emphasis on the momentum density and the compton profile. *Phys. Rev. B*, 72:115109, 2005.

[17] M. Calandra, J. P. Rueff, C. Gougoussis, D. Céolin, M. Gorgoi, S. Benedetti, P. Torelli, A. Shukla, D. Chandesris, and Ch. Brouder. K-edge x-ray absorption spectra in transition-metal oxides beyond the single-particle approximation: Shake-up many-body effects. *Phys. Rev. B*, 86:165102, Oct 2012.

[18] R. Car and M. Parrinello. Unified approach for molecular dynamics and density-functional theory. *Phys. Rev. Lett.*, 55:2471–2474, Nov 1985.

[19] R. Car and M. Parrinello. Unified approach for molecular dynamics and density-functional theory. *Phys. Rev. Lett.*, 55:2471–2474, Nov 1985.

[20] R Car, M Parrinello, and M Payne. Comment on 'error cancellation in the molecular dynamics method for total energy calculations'. *Journal of Physics: Condensed Matter*, 3(47):9539, 1991.

[21] K.J Chen, V Caspar, Ch Bellin, and G Loupias. Investigation of temperature dependence of compton profiles in lithium. *Solid State Commun.*, 110(7):357 – 361, 1999.

[22] Malcolm Cooper, Peter Mijnarends, Nobuhiro Shiotani, Nobuhiko Sakai, and Arun Bansil. *X-ray Compton scattering*. OUP Oxford, 2004.

[23] R Davidson, D Arnett, J Dahlburg, P Dimotakis, D Dubin, G Gabrielse, D Hammer, T Katsouleas, W Kruer, R Lovelace, et al. Frontiers in high energy density physics: the x-games of contemporary science, 2003.

[24] F. M. F. de Groot. Multiplet effects in x-ray spectroscopy. *Coor. Chem. Rev.*, 249(1):31–63, 2005.

[25] F. M. F. de Groot, J. C. Fuggle, B. T. Thole, and G. A. Sawatzky. 2 p x-ray absorption of 3 d transition-metal compounds: An atomic multiplet description including the crystal field. *Phys. Rev. B*, 42:5459–5468, Sep 1990.

[26] F. M. F. de Groot and Akio Kotani. *Core Level Spectroscopy of Solids*. CRC Press, 2008.

[27] G Dolling. Lattice vibrations in crystals with the diamond structure. In *Inelastic Scattering of Neutrons in Solids and Liquids. V. II. Proceedings of the Symposium on Inelastic Scattering of Neutrons in Solids and Liquids*, 1963.

[28] R Paul Drake. *High-energy-density physics: fundamentals, inertial fusion, and experimental astrophysics*. Springer Science & Business Media, 2006.

[29] SB Dugdale and T Jarlborg. Thermal disorder versus correlation in compton profiles from alkali metals. *Solid State Commun.*, 105(5):283–287, FEB 1998.

[30] P. Eisenberger and P. M. Platzman. Compton scattering of x rays from bound electrons. *Phys. Rev. A*, 2:415–423, Aug 1970.

[31] A. Erba, J. Maul, M. Itou, R. Dovesi, and Y. Sakurai. Anharmonic thermal oscillations of the electron momentum distribution in lithium fluoride. *Phys. Rev. Lett.*, 115:117402, Sep 2015.

[32] R. P. Feynman. Forces in molecules. *Phys. Rev.*, 56:340–343, Aug 1939.

[33] Claudia Filippi and David M. Ceperley. Quantum monte carlo calculation of compton profiles of solid lithium. *Phys. Rev. B*, 59:7907–7916, Mar 1999.

[34] V. R. Galakhov, S. Uhlenbrock, S. Bartkowski, A. V. Postnikov, M. Neumann, L. D. Finkelstein, E. Z. Kurmaev, A. A. Samokhvalov, and L. I. Leonyuk. X-ray photoelectron 3s spectra of transition metal oxides. *eprint arXiv:cond-mat/9903354*, March 1999.

[35] SH Glenzer, G Gregori, RW Lee, FJ Rogers, SW Pollaine, and OL Landen. Demonstration of spectrally resolved x-ray scattering in dense plasmas. *Physical review letters*, 90(17):175002, 2003.

[36] Siegfried H Glenzer and Ronald Redmer. X-ray thomson scattering in high energy density plasmas. *Reviews of Modern Physics*, 81(4):1625, 2009.

[37] O. Gunnarsson and B. I. Lundqvist. Exchange and correlation in atoms, molecules, and solids by the spin-density-functional formalism. *Phys. Rev. B*, 13:4274–4298, May 1976.

[38] Matteo Guzzo, Giovanna Lani, Francesco Sottile, Pina Romaniello, Matteo Gatti, J. J. Kas, J. J. Rehr, Mathieu G. Silly, Fausto Sirotti, and Lucia Reining. Valence electron

photoemission spectrum of semiconductors: *Ab Initio* description of multiple satellites. *Phys. Rev. Lett.*, 107:166401, Oct 2011.

[39] M Hakala, K Nygård, S Manninen, S Huotari, T Buslaps, A Nilsson, LGM Pettersson, and K Hämäläinen. Correlation of hydrogen bond lengths and angles in liquid water based on compton scattering. *J. Chem. Phys.*, 125(8):084504, 2006.

[40] M Hakala, K Nygård, J Vaara, M Itou, Y Sakurai, and K Hämäläinen. Charge localization in alcohol isomers studied by compton scattering. *J. Chem. Phys.*, 130(3):034506, 2009.

[41] R. Haydock and M. J. Kelly. Surface densities of states in the tight-binding approximation. *Surf. Sci.*, 38(1):139148, 1973.

[42] K. Hermann, L.G.M. Pettersson, M.E. Casida, C. Daul, A. Goursot, A. Koester, E. Proynov, A. St-Amant, D.R. Salahub, V. Carravetta, H. Duarte, C. Friedrich, N. Godbout, C. Jamorski J. Guan, M. Leboeuf, M. Leetmaa, M. Nyberg, S. Patchkovskii, L. Pedocchi, F. Sim, L. Triguero, and A. Vela. Stobe-demon version 3.2, 2013.

[43] S. Huotari, K. Hämäläinen, S. Manninen, C. Sternemann, A. Kaprolat, W. Schülke, and T. Buslaps. High-momentum components and temperature dependence of the compton profile of beryllium. *Phys. Rev. B*, 66:085104, 2002.

[44] G Huser, M Koenig, A Benuzzi-Mounaix, E Henry, T Vinci, B Faral, M Tomasini, B Telaro, and D Batani. Temperature and melting of laser-shocked iron releasing into an lif window. *Phys. Plasmas*, 12(6):060701, 2005.

[45] Hidekazu Ikeno, F. M. F. de Groot, Eli Stavitski, and Isao Tanaka. Multiplet calculations of l 2,3 x-ray absorption near-edge structures for 3d transition-metal compounds. *J. Phys.: Condens. Matter*, 21(10):104208, 2009.

[46] Jason R. Jeffries. personal communication.

[47] R. O. Jones and O. Gunnarsson. The density functional formalism, its applications and prospects. *Rev. Mod. Phys.*, 61:689–746, Jul 1989.

[48] J. J. Kas, A. P. Sorini, M. P. Prange, L. W. Cambell, J. A. Soininen, and J. J. Rehr. Many-pole model of inelastic losses in x-ray absorption spectra. *Phys. Rev. B*, 76(19):195116, 2007.

[49] Scott Kirkpatrick, Mario P Vecchi, et al. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.

[50] W. Kohn and L. J. Sham. Self-consistent equations including exchange and correlation effects. *Phys. Rev.*, 140:A1133–A1138, Nov 1965.

[51] J Korringa. On the calculation of the energy of a bloch wave in a metal. *Physica*, 13(6):392 – 400, 1947.

[52] Nobuhiro Kosugi, Yoshinori Tokura, Hidenori Takagi, and Shinichi Uchida. Cu K-edge x-ray-absorption near-edge structure and electronic structure of $nd_{2-x}ce_xcuo_{4-y}$ and $la_{2-x}sr_xcuo_4$. *Phys. Rev. B*, 41:131–137, Jan 1990.

[53] G. Kotliar, S. Y. Savrasov, K. Haule, V. S. Oudovenko, O. Parcollet, and C. A. Marianetti. Electronic structure calculations with dynamical mean-field theory. *Rev. Mod. Phys.*, 78:865–951, Aug 2006.

[54] G. Kresse and J. Hafner. *Ab initio* molecular dynamics for liquid metals. *Phys. Rev. B*, 47:558–561, Jan 1993.

[55] G. Kresse and J. Hafner. *Ab initio* molecular-dynamics simulation of the liquid-metal˘amorphous-semiconductor transition in germanium. *Phys. Rev. B*, 49:14251–14269, May 1994.

[56] G. Kresse and D. Joubert. From ultrasoft pseudopotentials to the projector augmented-wave method. *Phys. Rev. B*, 59:1758–1775, Jan 1999.

[57] Yasunori Kubo. Effects of electron correlations on compton profiles of li and na in the gw approximation. *J. Phys. Soc. of Jpn.*, 66(8):2236–2239, 1997.

[58] Thomas D Kühne, Matthias Krack, Fawzi R Mohamed, and Michele Parrinello. Efficient and accurate car-parrinello-like approach to born-oppenheimer molecular dynamics. *Physical review letters*, 98(6):066401, 2007.

[59] L. Lam and P. M. Platzman. Momentum density and compton profile of the inhomogeneous interacting electronic system. i. formalism. *Phys. Rev. B*, 9, 1974.

[60] Sébastien Le Roux, Anita Zeidler, Philip S. Salmon, Mauro Boero, Matthieu Micoulaut, and Carlo Massobrio. Structural properties of liquid ge_2se_3: A first-principles study. *Phys. Rev. B*, 84:134203, Oct 2011.

[61] J. D. Lee, O. Gunnarsson, and L. Hedin. Transition from the adiabatic to the sudden limit in core-level photoemission: A model study of a localized system. *Phys. Rev. B*, 60:8034–8049, Sep 1999.

[62] John D Lindl, Peter Amendt, Richard L Berger, S Gail Glendinning, Siegfried H Glenzer, Steven W Haan, Robert L Kauffman, Otto L Landen, and Laurence J Suter. The physics basis for ignition using indirect-drive targets on the national ignition facility. *Phys. Plasmas*, 11(2):339–491, 2004.

[63] K. Matsuda, T. Nagao, Y. Kajihara, M. Inui, K. Tamura, J. Nakamura, K. Kimura, M. Yao, M. Itou, Y. Sakurai, and N. Hiraoka. Electron momentum density in liquid silicon. *Phys. Rev. B*, 88:115125, Sep 2013.

[64] B. Mattern. *Compton Scattering and Warm Dense Matter Thermometry*. PhD thesis, University of Washington, 2013.

[65] Brian A Mattern, Gerald T Seidler, and Joshua J Kas. Condensed phase effects on the electronic momentum distribution in the warm dense matter regime. *arXiv preprint arXiv:1308.2990*, 2013.

[66] Brian A. Mattern, Gerald T. Seidler, Joshua J. Kas, Joseph I. Pacold, and John J. Rehr. Real-space green's function calculations of compton profiles. *Phys. Rev. B*, 85:115135, Mar 2012.

[67] L. F. Mattheiss. Energy bands for solid argon. *Phys. Rev.*, 133:A1399–A1403, Mar 1964.

[68] Albert Messiah. Quantum mechanics. vol. i.., 1958.

[69] Alexey I Nesvizhskii. *Theory and Interpretation of L-shell X-ray Absorption Spectra*. PhD thesis, University of Washington, 2001.

[70] S. Nose. A unified formulation of the constant temperature molecular-dynamics methods. *J. Chem. Phys.*, 81(1):511–519, 1984.

[71] P. Nozières and C. T. De Dominicis. Singularities in the x-ray absorption and emission of metals. iii. one-body theory exact solution. *Phys. Rev.*, 178:1097–1107, Feb 1969.

[72] J. T. Okada, P. H.-L. Sit, Y. Watanabe, B. Barbiellini, T. Ishikawa, Y. J. Wang, M. Itou, Y. Sakurai, A. Bansil, R. Ishikawa, M. Hamaishi, P.-F. Paradis, K. Kimura, T. Ishikawa, and S. Nanao. Visualizing the mixed bonding properties of liquid boron with high-resolution x-ray compton scattering. *Phys. Rev. Lett.*, 114:177401, Apr 2015.

[73] J. T. Okada, P. H.-L. Sit, Y. Watanabe, Y. J. Wang, B. Barbiellini, T. Ishikawa, M. Itou, Y. Sakurai, A. Bansil, R. Ishikawa, M. Hamaishi, T. Masaki, P.-F. Paradis, K. Kimura, T. Ishikawa, and S. Nanao. Persistence of covalent bonding in liquid silicon probed by inelastic x-ray scattering. *Phys. Rev. Lett.*, 108:067402, Feb 2012.

[74] J. T. Okada, P. H.-L. Sit, Y. Watanabe, Y. J. Wang, B. Barbiellini, T. Ishikawa, M. Itou, Y. Sakurai, A. Bansil, R. Ishikawa, M. Hamaishi, T. Masaki, P.-F. Paradis, K. Kimura, T. Ishikawa, and S. Nanao. Persistence of covalent bonding in liquid silicon probed by inelastic x-ray scattering. *Phys. Rev. Lett.*, 108:067402, Feb 2012.

[75] Fulvio Parmigiani and Luigi Sangaletti. Fine structures in the x-ray photoemission spectra of mno, feo, coo, and nio single crystals. *J. Electron Spectrosc. and Relat. Phenom.*, 98:287–302, 1999.

[76] G. Pastore, E. Smargiassi, and F. Buda. Theory of *ab initio* molecular-dynamics calculations. *Phys. Rev. A*, 44:6334–6347, Nov 1991.

[77] L.-M. Peng, G. Ren, S. L. Dudarev, and M. J. Whelan. Debye–Waller Factors and Absorptive Scattering Factors of Elemental Crystals. *Acta Crystallogr. Sect. A*, 52(3):456–470, May 1996.

[78] James C. Phillips. Energy-band interpolation scheme based on a pseudopotential. *Phys. Rev.*, 112:685–695, Nov 1958.

[79] M. P. Prange, J. J. Rehr, G. Rivas, J. J. Kas, and John W. Lawson. Real space calculation of optical constants from optical to x-ray frequencies. *Phys. Rev. B*, 80:155110, Oct 2009.

[80] M. P. Prange, J. J. Rehr, G. Rivas, J. J. Kas, and John W. Lawson. Real space calculation of optical constants from optical to x-ray frequencies. *Phys. Rev. B*, 80:155110, Oct 2009.

[81] J. J. Rehr and R. C. Albers. Theoretical approaches to x-ray absorption fine structure. *Rev. Mod. Phys.*, 72:621–654, Jul 2000.

[82] J. J. Rehr, J. J. Kas, M. P. Prange, A. P. Sorini, L. W. Campbell, and F. D. Vila. Inelastic losses and multi-electron excitations in x-ray spectra. In *AIP Conf. Proc.*, volume 882, page 85, 2007.

[83] J. J. Rehr, J. J. Kas, M. P. Prange, A. P. Sorini, Y. Takimoto, and F. D. Vila. *Ab initio* theory and calculations of x-ray spectra. *Comp. Ren. Phys.*, 10(6):548–559, 2009.

[84] J. J. et al Rehr. Feff user guide. http://leonardo.phys.washington.edu/feff/Docs/feff9/feff90/feff90_users_guide.pdf. Accessed: 2016-05-19.

[85] John J Rehr and RC Albers. Theoretical approaches to x-ray absorption fine structure. *Reviews of modern physics*, 72(3):621, 2000.

[86] Bruce A Remington, R Paul Drake, and Dmitri D Ryutov. Experimental astrophysics with high power lasers and z pinches. *Rev. Mod. Phys.*, 78(3):755, 2006.

[87] AH Romero, PL Silvestrelli, and M Parrinello. Compton anisotropy from wannier functions in the case of ice ih. *Phys. Status Solidi B*, 220(1):703–708, 2000.

[88] Aldo H. Romero, Pier Luigi Silvestrelli, and Michele Parrinello. Compton scattering and the character of the hydrogen bond in ice ih. *J. Chem. Phys.*, 115(1):115–123, 2001.

[89] Y Sakurai, M Itou, B Barbiellini, PE Mijnarends, RS Markiewicz, S Kaprzyk, J-M Gillet, S Wakimoto, M Fujita, S Basak, et al. Imaging doped holes in a cuprate superconductor with high-resolution compton scattering. *Science*, 332(6030):698–702, 2011.

[90] Y. Sakurai, Y. Tanaka, A. Bansil, S. Kaprzyk, A. T. Stewart, Y. Nagashima, T. Hyodo, S. Nanao, H. Kawata, and N. Shiotani. High-resolution compton scattering study of li: Asphericity of the fermi surface and electron correlation effects. *Phys. Rev. Lett.*, 74, 1995.

[91] W. Schülke, G. Stutz, F. Wohlert, and A. Kaprolat. Electron momentum-space densities of li metal: A high-resolution compton-scattering study. *Phys. Rev. B*, 54:14381–14395, Nov 1996.

[92] W. Schülke, G. Stutz, F. Wohlert, and A. Kaprolat. Electron momentum-space densities of li metal: A high-resolution compton-scattering study. *Phys. Rev. B*, 54, 1996.

[93] Winfried Schülke. *Electron dynamics by inelastic X-ray scattering*. OUP Oxford, 2007.

[94] J. C. Slater. Wave functions in a periodic potential. *Phys. Rev.*, 51:846–851, May 1937.

[95] J. Söderström, N. Mårtensson, O. Travnikova, M. Patanen, C. Miron, L. J. Sæthre, K. J. Børve, J. J. Rehr, J. J. Kas, F. D. Vila, T. D. Thomas, and S. Svensson. Nonstoichiometric intensities in core photoelectron spectroscopy. *Phys. Rev. Lett.*, 108:193005, May 2012.

[96] R. Stedman, Z. Amilius, R. Pauli, and O. Sundin. Phonon spectrum of beryllium at 80 k. *J. Phys. F: Met. Phys.*, 6(2):157–166, 1976.

[97] C. Sternemann, T. Buslaps, A. Shukla, P. Suortti, G. Döring, and W. Schülke. Temperature influence on the valence compton profiles of aluminum and lithium. *Phys. Rev. B*, 63:094301, Jan 2001.

[98] C. Sternemann, G. Doring, C. Wittkop, W. Schulke, A. Shukla, T. Buslaps, and P. Suortti. Influence of lattice dynamics on electron momentum density of lithium. *J. Phys. Chem. Solids*, 61(3):379–382, 2000.

[99] M. A. van Veenendaal and G. A. Sawatzky. Nonlocal screening effects in $2p$ x-ray photoemission spectroscopy core-level line shapes of transition metal compounds. *Phys. Rev. Lett.*, 70:2459–2462, Apr 1993.

[100] Ulf von Barth and Lars Hedin. A local exchange-correlation potential for the spin polarized case. i. *Journal of Physics C: Solid State Physics*, 5(13):1629, 1972.

[101] DUANE C. WALLACE. Pseudopotential calculation of phonon frequencies and grüneisen parameters for lithium. *Phys. Rev.*, 178:900–901, Feb 1969.

[102] Hugh F Wilson and Burkhard Militzer. Rocky core solubility in jupiter and giant exoplanets. *Phys. Rev. Lett.*, 108(11):111101, 2012.

[103] Z. Y. Wu, D. C. Xian, T. D. Hu, Y. N. Xie, Y. Tao, C. R. Natoli, E. Paris, and A. Marcelli. Quadrupolar transitions and medium-range-order effects in metal k-edge x-ray absorption spectra of 3d transition-metal compounds. *Phys. Rev. B*, 70:033104, jul 2004.

[104] RWG Wyckoff. Crystal structures 1 (7–83). *American Mineralogist Crystal Structure Database*, 1963.

[105] S. I. Zabinsky, J. J. Rehr, A. Ankudinov, R. C. Albers, and M. J. Eller. Multiple-scattering calculations of x-ray-absorption spectra. *Phys. Rev. B*, 52:2995–3009, Jul 1995.