Carrier Scattering Rates in Zincblende Structure Semiconductors derived from $14 \times 14 \, \vec{k} \cdot \hat{p}$ and $ab \; initio$ Pseudopotential Methods

Bjørnar Karlsen

July 16, 2013

Supervisors:
Jon Andreas Støvneng (NTNU)
Trond Brudevoll (FFI)
Asta Katrine Storebø(FFI)

Master thesis done in collaboration with NTNU and FFI

Abstract

A set of programs for calculating carrier-phonon and carrier-alloy scattering rates have been constructed. Energy bands, gradients of the energy, second derivatives of the energy and eigenvectors of the carrier states were first calculated using a 14×14 $\vec{k} \cdot \hat{p}$-method. Scattering rates were then derived, based upon the calculations of the developed $\vec{k} \cdot \hat{p}$ software.

Furthermore, scattering rates have also been derived using band structures from the $ab\ initio$ pseudopotential code ABINIT. A program for setting up ABINIT calculations on a projector augmented wave (PAW) basis, as well as software for converting the band data into the required input format for scattering rate calculations have been developed.

Finally, the results produced from both band structure methods are discussed and compared. This analysis has been made for the specific case of the zincblende structure $Hg_{(1-x)}Cd_xTe$ at temperatures 77K and 300K

Foreword

The work for this master thesis at the Norwegian Technical and Scientific University (NTNU) where the assignment is given by Norwegian Defense Research Establishment (FFI). The thesis is the fulfilment of my degree in applied physics at NTNU under the guidance of Jon Andreas Støvneng (NTNU), Trond Brudevoll (FFI) and Asta Katrine Storebø(FFI), and I send my sincerest thanks to all of them.

This work is a part of a greater Monte Carlo (MC) program for charge carrier transport in zincblende semiconductors. The Co-authors on this MC program were Ken Vidar Falch and Tore Sivertsen Bergslid both carrying out their Master theses in applied and general physics respectively. I will thank them both for a pleasant collaboration in revising the MC program, especially Tore who created the reader used to extract the band plots from KPBAND. This thesis considers the band structure, its first and second derivatives in k-space and the eigenstates. This has been done by through two methods: The $\vec{k} \cdot \hat{p}$-method the $ab\ initio$ pseudopotential method. Further these calculations have been used to calculate phonon- and alloy scatterings of light holes, heavy holes and conduction band electrons. The specific material considered in this Master thesis is $Hg_{0.72}Cd_{0.28}Te$ at temperatures 77K and 300K.

The programs for the $\vec{k} \cdot \hat{p}$-method and scattering calculations were extracted from an earlier work by Einar Halvorsen, where the model were expanded from a 6x6-model to a 8x8- and a 14x14-model with cubic harmonic corrections of higher order k-terms far from the first Brillouin zone center, and the scattering program has been made compatible with these changes. The program for performing the ab initio pseudopotential calculations is the program ABINIT initiated by the university Universite Catholiqe de Louivain, with the additional programs Atompaw and Atompaw2Abinit for generating and converting the PAW pseudopotentials for the calculations (read more in the Appendix D.3). Programs for converting ABINIT results to a usable format were also created.

In the process of doing this master thesis work I have had to be introduced to a larger program package KPBAND and SCRATES and expanding them, as well as learning a previously to me unknown computer language FORTRAN90. I have also had to get familiarized with the LINUX operative system as well as learning to install and run ABINIT with Atompaw and Atompaw2Abinit. Shortcomings by applying $ab\ initio$ Virtual Crystal Approximations in ABINIT were uncovered. Although the GW is found in ABINIT the alternative methods for solving the band gap problem were very few.

Contents

1. Introduction

Charge carrier transport simulations in semiconductors can be carried out in two very different ways, either by explicitly solving the transport equations, or by performing Monte Carlo (MC) simulations. Whereas the first method builds directly upon equation-solving techniques, the latter method represents a more intuitive, or particle-oriented, approach which solves the Boltzmann equation indirectly.

This thesis is part of a larger effort to develop a full band (FB) MC code able to utilize numeric band data from different sources like $14 \times 14 \, \vec{k} \cdot \hat{p}$ or $ab \, initio$ electronic structure codes. What is presented here only serves as a preparatory work for the actual full band MC program. The purpose of the final FBMC-code is to simulate charge carrier transport in different kinds of materials without resorting to local (in k-space) analytic band descriptions, and instead represent the bands numerically and over the whole Brillouin zone. The programs used for this thesis are based upon an earlier work by Einar Halvorsen [1], and have been considerably expanded in order to be applicable for both electrons and holes and for different types of semiconductors.

MCT ($Hg_{(1-x)}Cd_xTe$) is an extensively used detector material for infrared photon detectors and detector arrays [2]. It will serve as a test material for the codes to be described here. The material itself is an alloy, consisting of the wide band gap semiconductor $CdTe$ and the semimetal $HgTe$. One reason for its popularity is the variable energy gap between the valence band and the conduction band, depending on the ratio between $HgTe$ and $CdTe$ and the temperature of the material. The particular composition assumed in the examples to be presented here was 28% $CdTe$ and 72% $HgTe$, producing gaps of 265meV at 300K and 211meV at 77K.The material also has very unique impact ionization properties making it very attractive for use in highly sensitive Avalanche Photo Diodes (APDs). Carrier-carrier scattering including impact ionization/Auger recombination is discussed in a separate Master thesis by Ken Vidar Falch [3].

The first part of the present work was to calculate energy band structures, k-space energy gradients, k-space energy second derivatives and k-space carrier eigenstates. Methods used were the $\vec{k} \cdot \hat{p}$ and the $ab \, initio$ pseudopotential method, represented by the Halvorsen program KPBAND and ABINIT[4], respectively. In using the ABINIT results a set of complementary codes had to be made in order to produce results compatible with the KPBAND program.

The second part was to calculate rates of the phonon and alloy scattering mechanisms. The program used for this was the Halvorsen SCRATES program, which had to be revised in order to make use of the $14 \times 14 \, \vec{k} \cdot \hat{p}$ expansion introduced in KPBAND. In addition the alloy scattering mechanism was included.

Shortly before the work on the present thesis started, the programs of Halvorsen were prepared for the next step of code development and slightly modernized. Tore Sivertsen Bergslid updated the programs from FORTRAN 77 to FORTRAN 90 and introduced two preliminary, new modules halvorinit.f90 and rateinit.f90 for quick and easy user modification of the input parameters. In a forthcoming Master thesis, Bergslid has already implemented many of the new results produced during the present thesis work into the new FBMC code.

Accomplishments made in the present thesis are the following:

- Expansion and generalization the Halvorsen KPBAND program:

  1. Expanded the $\vec{k} \cdot \hat{\vec{p}}$ model from a strict 6×6 matrix to an 8×8 and a 14×14 matrix.

  2. Generalized the algorithms for easy transfer between the three above mentioned $\vec{k} \cdot \hat{\vec{p}}$ models.

  3. Enabled electron simulations by including conduction bands in the models.

  4. Introduced new parameters necessary for MCT calculations at different lattice temperatures, e g initiated 77K and 300K simulations by halvorinit77K.f90 and halvorinit300K.f90 respectively.

- Revising the SCRATES program:

  1. Generalized the SCRATES program to read eigenvectors from the 6×6, 8×8 and 14×14 matrices.

  2. Implemented alloy scattering calculations in SCRATES as an option.

- Installation and research regarding how to use a rather poorly documented open source *ab initio* electronic structure code: ABINIT. Learning and familiarizing with Linux and FORTRAN.

- Execution of *ab initio* band structure calculations from pseudopotentials using the ABINIT program:

  1. Performed band structure calculations for MCT using projector augmented waves (PAWs) as a basis.

  2. Created a program ABINITGENERATOR.f90 for producing input files for calculating a coarse and a fine mesh band structure in $1/48^{th}$ wedge of the Brillouin zone.

  3. Created the programs ABINITCONVERTER.f90 and ABINITDIFFERENTIATOR.f90 for converting the output from ABINIT and calculating the derivatives of the output from ABINIT to conform with the format used by the corresponding Halvorsen band structure program KPBAND.

- Created MATLAB programs for plotting band structures and scattering rates.

The basis for the construction of the FBMC-program is the paper done by Fischetti and Laux [5]. They concentrated on electron transport in $GaAs$ and Si. According to Halvorsen, this work and the work related to the 6×6-Kohn-Luttinger-Hamiltonian (for $GaAs$) by Hinkley and Singh [6] were the main influences in deriving his model. For the complete revision of the MC program the papers by Bertazzi et al. [7],[8] have been the most influential. Both the present thesis and Bertazzi?s works apply the 14×14 $\vec{k} \cdot \hat{\vec{p}}$ method and a pseudopotential method.

For the $\vec{k} \cdot \hat{\vec{p}}$ method Bertazzi used a rather complicated multi k-point expansion, starting from different points in the Brillouin zone, with interpolation between the resulting band structures. Realizing that the $\vec{k} \cdot \hat{\vec{p}}$ method was only intended for small deviations from the starting k-point, we have contented ourselves with a single $\vec{k} \cdot \hat{\vec{p}}$ expansion

starting at k = 0, but with (rather crude) analytical band corrections for larger values of k. For expansion of the 6×6 Hamiltonian into an 8×8 and a 14×14 Hamiltonian, the models used by Pokatilov et al. [9] (8×8), and Pfeffer and Zawadzki [10] (14×14) have been heavily leaned upon.

For the pseudopotential method Bertazzi basically used empirical pseudopotentials, whereas the present work tries to go one step further and apply *ab initio* pseudopotentials using the ABINIT program.

In Chapter 2 the fundamentals MC simulations are presented, followed by relevant theory, like scattering rates and free flight times, and energy bands. In Chapter 3 the algorithms for carrying out the calculations mentioned in Chapter 2 are presented. Chapter 4 presents the results from the 14×14 $\vec{k} \cdot \hat{\vec{p}}$ method and the *ab initio* pseudopotential method. It also includes a comparison of the resulting band structures. Appendix D contains a user guide for the programs developed in this work.

2. Theory

This chapter gives the theoretical foundations of this work, beginning with the description of the MC simulator. The method is to simulate the trajectory of multiple charge carriers (negative electrons or positive electron holes) through a material, here for semiconductors, and more specifically MCT (however other kinds of materials should also be calculable). The MC simulations require both state- and scattering calculations. The state of the charged carrier evolves based on the external environment as described in eq. (2), which in turn leads to an evolution of the carrier velocity as in eq. (1). However this continuous evolution lasts only for a very short time, then it is scattered into a completely new state with a new continuous state evolution.

The MC approach and scattering calculations are described in section 2.1. In the rest of this chapter the energy-state calculations are described. The starting point is the state (or band) calculations using the $\vec{k}\cdot\hat{\vec{p}}$-method using the 14×14-model, which is described for both the 8×8- and the 14×14-model in section 2.2. The energy bands resulting from this model are however only sufficiently accurate close to the center of the 1st Broullien zone (1BZ), because it tends to deviate strongly towards the zone edges. This is due to the solutions being perturbations around a single point (here center or Γ-point). As the MC simulator is supposed to be a full-zone simulation, the full 1BZ needs to be calculated more realistically. For this very reason cubic harmonics have been added. This is in order to correct the state calculations far from the center of the 1BZ, which is described in section 2.2.3. Further, an alternative band structure calculation have been made through the $ab\ initio$ pseudopotential method using the program ABINIT. This method is of course more complicated, but makes the band structure calculations more accurate, and is described in section 2.3.

2.1. Monte Carlo Method in Electronic Transport

This section describes how the Monte Carlo method is applied in order to account for the charge transport in a semiconductor. It describes the framework of a MC-program, of which this work is to be a part. The part of this thesis which considers results from the $\vec{k} \cdot \hat{\vec{p}}$-method is based on the specialization project [11] presiding this thesis, which in turn was an improvement of the work by Halvorsen [1]. In order to make the band structures sufficiently accurate for performing full-zone scattering calculations, there has been added cubically harmonic correction terms for the light hole, heavy hole and conduction band. In this thesis only the 14×14-model calculations are used, since this is the model which gives the most realistic description of the band structure.

As a further addition to the prior thesis [11] scattering calculations based on $ab\ initio$ pseudopotentials have been carried out. The energy eigenvalues and state eigenvectors are acquired by extracting them from the ABINIT program using the LSDA(local spin denisty approximation) Troullier-Martins method described in their work [12]. From these values the gradient and second derivatives in k-space are calculated by using the formulas in Appendix A. Finally the scattering rates are calculated.

2.1.1. The Monte Carlo (MC) Approach

A strict theoretical approach would in this case be to solve the Boltzmann transport equation for the system. It could of course be done, but is not the intention for this work. The intention however that this work is meant to be used in a MC analysis of the charge carrier transport, where the carriers are described semi-classically, meaning they are described both with a quantum mechanical state and a position. This approach takes into account much of the same theory, but the solutions are made using (pseudo) random numbers on the computer, which introduces an experimental-like element to the solution.

The charge carriers are as mentioned described semi-classically, with a Bloch state and a position. The carriers are scattered between the different Bloch states due to a number of scattering mechanisms such as field affections (eg. ionized impurities) and different kinds of phonon interactions. In a scattering event a carrier will be transferred from an energy band n with wave vector $\vec{k}$, a state $|n, \vec{k}>$, to a similar state $|n', \vec{k'}>$. Each scattering is assumed to occur instantly, and the carriers are assumed to move as described by the semi-classical equations of motion between each scattering. The carrier motion equations are as follows:

$$\frac{d\vec{r}}{dt} = \frac{1}{\hbar} \nabla_{\vec{k}} E_n(\vec{k}) \tag{1}$$

$$\frac{d\vec{k}}{dt} = \frac{q}{\hbar} \vec{F}(\vec{r}) = \frac{q}{\hbar}(\vec{E} + \vec{v} \times \vec{B}) \tag{2}$$

Here $\vec{v}$ is the velocity, $\vec{r}$ is the position, $\vec{p}$ is the momentum, $\vec{F}$ is the force, q is the carrier charge, $\vec{E}$ is the electric field and $\vec{B}$ is the magnetic field.

As has been said, each scattering event is assumed to happen instantly, meaning the charge carrier moves in a so called "free flight time" τ between each scattering. This time is the time interval between each scattering, so it would be the inverse of the scattering rate σ. The scattering rate is described quantum mechanically as the probability for a scattering to occur per unit time, or rather the number of scatterings which occurs per unit time. The total free flight time will following the Matthiessen rule. It says: The inverse of the free flight time equals the sum over the inverse of the free flight time of each scattering mechanism τ_m, or the sum over the scattering rate for each scattering mechanism, σ_m.

$$\frac{1}{\tau} = \sum_{m \in mech.} \frac{1}{\tau_m} = \sum_{m \in mech.} \sigma_m \tag{3}$$

2.1.2. Scattering rates

The carriers may be scattered between two Bloch states through an array of different scattering mechanisms. The mechanisms can be divided into the following main groups:

- Phonon scattering

- Deformation potential phonon scattering

- Impurity scattering

- Alloy scattering

- Carrier-carrier scattering

For the work in this thesis only a few of these mechanisms have been considered. The scattering rates have been calculated through the program SCRATES, and the following scattering mechanisms are available in this program:

- Polar optical phonon scattering (absorption and emission), included

- Acoustic deformation potential phonon scattering (absorption and emission), included

- Non-polar optical phonon scattering (absorption and emission), included

- Ionized impurity scattering, not included

- Alloy scattering, included

The importance of the different scattering mechanisms may vary based on material type and temperature. As mentioned in section 2.1.1, the scattering processes transfer a charged carrier from a Bloch state $|n, \vec{k}>$, to a state $|n', \vec{k}'>$. The transition probability of a scattering process, m, is given by Fermi's golden rule (where spin flip scattering is neglected):

$$P_{nn'}^m(\vec{k}, \vec{k}') = \frac{2\pi}{\hbar}|M_{nn'}^m(\vec{k}, \vec{k}')|^2|G_{nn'}(\vec{k}, \vec{k}')|^2\delta(E_n(\vec{k}) + \Delta E^m - E_{n'}(\vec{k}')) \quad (4)$$

$E_n(\vec{k})$ and $E_{n'}(\vec{k}')$ are the carrier's initial and final state energies and ΔE^m is the energy transfer caused by the scattering. $M_{nn'}^m(\vec{k}, \vec{k}')$ is the interaction matrix which will be more discussed in section 3.2.2. Fermi's golden rule insures both of the required conservation rules, conservation of energy and crystal momentum. The energy is conserved by the Dirac's delta-function, and the momentum in the transition from one reciprocal lattice point to another, $\vec{G}$:

$$\vec{k} - \vec{k}' + \vec{q} = \vec{G} \quad (5)$$

Here $\vec{q}$ is the shift in the wave vector due to the scattering. In the case of phonon scattering, $\vec{q}$ is either absorbed or emitted by the carriers depending on whether the phonon is absorbed or emitted. When $\vec{G} = 0$ the scattering process is known as a normal scattering process, but when $\vec{G} \neq 0$ the carrier is scattered into another Brillouin zone which is called an Umklepp process. Only the normal scattering processes are applicable in SCRATES.

$|G_{nn'}(\vec{k}, \{k'\})|^2$ is the overlap factor. Often, as in this case, this factor is placed outside the squared matrix element. This is done in order to make the calculations more orderly. It is given as follows:

$$G_{nn'}(\vec{k}, \{k'\}) = \int_{cell} d^3 r u^*_{n',\vec{k}'}(\vec{r}) u_{n,\vec{k}}(\vec{r}) exp(\imath \vec{G} \cdot \vec{r}) \tag{6}$$

Here scattering due to spin-flipping is neglected, but the fact that each band has two different spin-states needs to be taken into consideration. This is done by summing over the final sates, and averaging over the initial states. The different spins are denoted by $\mu, \mu' = 1, 2$ which results in:

$$|G_{nn'}(\vec{k}, \vec{k}')|^2 = \frac{1}{2} \sum_{\mu=1}^{2} \sum_{\mu'=1}^{2} |\int_{cell} d^3 r u^*_{n',\vec{k}'}(\vec{r}) u_{n,\vec{k}}(\vec{r}) exp(\imath \vec{G} \cdot \vec{r})|^2 \tag{7}$$

In order to simulate trajectories of the charge carriers, the free flight time in the medium needs to be calculated. This is done by taking the inverse of the total scattering rate. It can be interpreted as the probability for a carrier to be transferred from a state $|n, \vec{k} >$ to any other band or momentum. The scattering rate, $\sigma_n(\vec{k})$, is then calculated as follows:

$$\sigma_n(\vec{k}) = P_n(\vec{k}) = \sum_{m,n'} P^m_{nn'}(\vec{k}) = \sum_{m,n'} \left[\sum_{\vec{k}'} P^m_{nn'}(\vec{k}, \vec{k}') \right] = \frac{1}{\tau_n} \tag{8}$$

The sum in the square parentheses is the probability for a state $|n, \vec{k} >$ to transfer to a band n' for any wave vector. Because the space of allowed $\vec{k}'$ is very dense, it is allowed to perform an integration instead of a summation. Of course in a computer the integral will be calculated as a sum.

2.1.3. Alloy scattering

All of the scattering mechanisms in SCRATES mentioned in the previous section already existed prior to this work, and were all described by Brudevoll [13]. The only exception to this is the alloy scattering mechanism, which was added for this work. This was not included previously since the material in question was $GaAs$. Here however $Hg_{1-x}Cd_xTe$ is analyzed, and proves to be an important mechanism.

The reason why this mechanism occurs, must be seen in the light of a non-alloy material such as $GaAs$. This material is by a good approximation perfectly periodical, meaning that the charge carriers will move in a coherently varying potential. For $Hg_{1-x}Cd_xTe$ the Te atoms may be seen as occurring periodically such as the As atom in $GaAs$ material, but Hg and Cd atoms occur highly aperiodically. This leads to the potential in the material also varying highly aperiodically, which in turn scatters the charge carriers. From the works by Bertazzi et al. [8] and Derelle et al. [14] it is expected that this is a very prevalent scattering mechanism, so it had to be introduced into the SCRATES program. As can be seen in section 4.2.2. it is a strong scattering mechanism.

The complete formula for the rate of alloy scattering is derived by Ridley [15]. The resulting squared matrix element becomes

$$|M(\vec{k}, \vec{k}')|^2 = (DV)^2 \frac{x(1-x)}{N_c} \tag{9}$$

which also may be found in Appendix C. DV is the alloy scattering potential (different for electrons and holes), x is that from the alloy formula $A_xB_{1-x}C$ (where $A = Cd$, $B = Hg$ and $C = Te$) and N_c is the cation concentration.

2.1.4. MC selection of scattering mechanism

It has been discussed how scattering mechanisms lead to changes in the Bloch-states, but it has not yet been discussed how the trajectory of the charge carriers are affected by them. In section 2.1.1 the semi-classical equations of motion were described in eqs. (1) and (2), which the carriers follow between each scattering. The free flight time, τ was derived in eqs. (3) and (8), but for a specific initial state $|n, \vec{k}>$ that means $\tau = \tau_n(\vec{k})$. Further the assumption is that after each time interval, τ_n a new scattering occurs, which is chosen randomly. The probability for choosing a scattering mechanism has to be proportional to the scattering rate. The scattering rates will for programming purposes be arbitrarily enumerated, so that the probability for choosing one of the m first scattering mechanisms is

$$P_m(\vec{k}) = \sum_{i=1}^{m} P_n^i(\vec{k}) \tag{10}$$

From this the selection is made by generating a flatly distributed value $r \in [0, P_n(\vec{k}))$. From this it is seen that scattering mechanism nr. m is choose if a r satisfying $P_{m-1} \leq r < P_m$, where $P_0 = 0$.

2.1.5. Final state selection

After a scattering mechanism is chosen, the carrier ends up in a final state $|n', \vec{k}'>$. The requirement is that the energy and crystal momentum in the process are conserved (see eqs. (4) and (5)). Both band and wave vector must be selected accordingly, and there will be a range of possible final states. This thesis does not contain a scheme for performing such a selection, but it does however contain an analyzis of the magnitude of the scatterings. The analysis is however only supposed to give a clue to the importance of the scattering mechanisms, and to which carriers are the most characteristic for the material.

2.2. The $\vec{k} \cdot \hat{p}$-method for the 8×8 and 14×14 cases

In this section it is presented how to calculate the energy and states of the different bands by the 8×8 and 14×14 $\vec{k} \cdot \hat{p}$-methods. This model is an extension of the 6×6 model made by Halvorsen [1], where in the 8×8 the first conduction band (s-like states) is added, and in the 14×14 also higher conduction bands similar to the valence bands (p-like states) are added. Although the 8×8-model is not used in this work, it is however constructed in relation to it, so it still has to be a part of the presentation.

The linear terms in $\vec{k}$ are neglected in the 6×6 part of the matrix, just as was done by Halvorsen, although they could be considered as well which was done by Pfeffer and Zawadzki [10] and by Rössler et al. [16],[17]. In the coupling between the valence

bands and the s-like conduction states only the linear term in $\vec{k}$ are considered, but here also the second order terms could be taken into account. In the 14×14-model only linear $\vec{k}$ terms are included.

The $\vec{k} \cdot \hat{p}$-method is an approximated perturbation method, and the model is not accurate enough far from the expansion point (here the Γ-point). This problem is partially solved by introducing analytical corrections based on cubic harmonics, which is more closely described in section 2.2.4. The expansions were primarily made in order to make the model valid for narrow band gap semiconductors such as $HgCdTe$, which as stated is the topic for this thesis.

Only states which interact strongly are directly considered and are classified as category A states. All other states that interact weakly are only indirectly considered and are classified as category B states. The model should also be valid for wide band gap materials such as $GaAs$ (which was done by Pfeffer and Zawadzki [10]), which Halvorsen worked with, where the calculations should be more accurate, and should also be valid further away from the band extrema.

KPBAND is also able to calculate the band structure and states for strained semiconductors in the 6×6-model, but this is not further discussed in this thesis.

2.2.1. Derivation of basis state interaction

The electron- and hole states are described by the time-independent Schrödinger equation for cases without external fields , and at first without spin-orbit interaction, which will be introduced in section 2.2.3:

$$\hat{H}\Psi = \left[\frac{\hat{\vec{p}}^2}{2m} + V(\vec{r}) \right] \Psi = E\Psi \tag{11}$$

$V(\vec{r})$ is the periodic potential caused by the lattice and electrons. The equation is then solved under the assumption that Ψ can be described as Bloch states (states consisting of a wave part $\exp(\imath \vec{k} \cdot \vec{r})$ and a periodic part $u_{n,\vec{k}}(\vec{r})$):

$$\Psi_{n,\vec{k}}(\vec{r}) = \exp(\imath \vec{k} \cdot \vec{r}) \cdot u_{n,\vec{k}}(\vec{r}) \tag{12}$$

This is a state of band n with the wave vector $\vec{k}$, which has an energy denoted $E_n(\vec{k})$. These state-functions are assumed to be normalized to unity in the crystal.

A further assumption about the properties of the carrier-states is that they start to mix when removed from the expansion point in $\vec{k}$-space, $\vec{k}_0$. The periodic part then becomes a super position of these parts at the expansion point, which means they take the following form:

$$\Psi_{n,\vec{k}}(\vec{r}) = \exp(\imath(\vec{k} - \vec{k}_0) \cdot \vec{r}) \sum_j \Psi_{j,\vec{k}_0}(\vec{r}) C_j^n(\vec{k}) \tag{13}$$

The basis states are chosen to be orthonormal at $\vec{k}_0$ and therefor also at an arbitrary $\vec{k}$. The expansion of the state-functions in eq. (13) leads to the following modification of the Hamiltonian. $\hat{H}'$ in eqs. (14) and (15) is named the Bloch-Hamiltonian:

$$\hat{H}\Psi_{n,\vec{k}}(\vec{r}) = \exp(\imath(\vec{k} - \vec{k}_0) \cdot \vec{r})\hat{H}' \sum_j \Psi_{j,\vec{k}_0}(\vec{r})C_j^n(\vec{k}) \tag{14}$$

$$\hat{H}' = \hat{H} + \frac{\hbar^2}{2m}(\vec{k} - \vec{k}_0)^2 + \frac{\hbar}{m}(\vec{k} - \vec{k}_0) \cdot \hat{\vec{p}} \tag{15}$$

In order to solve the Schrödinger equation, it is normed by a state $\Psi_{n',\vec{k}}$. The most beneficial way to do so is to introduce the state-function as a vector product:

$$\Psi_{n,\vec{k}}(\vec{r}) = \exp(\imath(\vec{k} - \vec{k}_0) \cdot \vec{r}) \sum_j \Psi_{j,\vec{k}_0}(\vec{r})C_j^n(\vec{k}) = \exp(\imath(\vec{k} - \vec{k}_0) \cdot \vec{r})\mathcal{F}^T \cdot \mathcal{C}_n(\vec{k}) \tag{16}$$

Here $\mathcal{F}$ and $\mathcal{C}_n(\vec{k})$ are column vectors, where $\mathcal{F}$ consists of the basis functions and is the same for all states, and $\mathcal{C}_n(\vec{k})$ consists of the contribution constants for the basis function for a given state $|n, \vec{k} >$. This allows for choosing easily treatable basis functions for $\mathcal{F}$, which means only the vectors $\mathcal{C}_n(\vec{k})$ will have to be solved in order to acquire the state functions. Further the equation is divided by the exponential term, and multiplied by the complex conjugated basis vector, $\mathcal{F}^*$. Lastly the normalization is completed by integrating over the crystal. The result is a matrix eigenvalue problem, which easily is solved by the computer.

$$\int_{crystal} d^3r \mathcal{F}^* \hat{H}' \mathcal{F}^T \mathcal{C}_n(\vec{k}) = \mathcal{H}\mathcal{C}_n(\vec{k}) = E_n(\vec{k})\mathcal{C}_n(\vec{k}) \tag{17}$$

The problem has now gone from solving an eigenfunction problem for a Hamilton operator, $\hat{H}$, to solving an eigenvalue problem for a Hamilton matrix $\mathcal{H}$. The elements in the matrix are as follows:

$$\begin{aligned}
\mathcal{H}_{ij} &= \left[E_i(\vec{k}_0) + \frac{\hbar^2}{2m}(\vec{k} - \vec{k}_0)^2 \right]\delta_{ij} + \frac{\hbar}{m}(\vec{k} - \vec{k}_0) \cdot \vec{p}_{ij} \\
&+ \frac{\hbar^2}{2m}\sum_{\alpha,\beta}(\vec{k} - \vec{k}_0)_\alpha(\vec{k} - \vec{k}_0)_\beta \left[\sum_{l \in B} \frac{p_{il}^\alpha p_{lj}^\beta + p_{il}^\beta p_{lj}^\alpha}{E_i(\vec{k}_0) - E_l(\vec{k}_0)} \right]
\end{aligned} \tag{18}$$

Here $(\alpha, \beta) \in (x, y, z)$. The two last terms need some clarifications. As has been said, only few states interacts strongly, and they are included in the collection of basis-functions. These category A states are easily united with the momentum operator which constitutes the first of the terms. The last of the terms is the momentum operator applied on the category B states. As these states only interacts weakly, they are included through a perturbation approach. This makes a transfer from an A-state to a B-state and then back again to an A-state. Such terms are known as re-normalized terms, and are in this thesis only included for the valence bands.

2.2.2. Matrix setup for the case without spin-orbit interaction

This section is only meant as a mid-section in order to set up the complete calculation. The extrema are located at the Γ-point ($\vec{k}_0 = 0$). For the valence bands there is a

threefold degeneracy, which contributes the basis-functions $|X>$, $|Y>$ and $|Z>$ which transform in x, y and z direction respectively. These functions are all chosen to be real. The first conduction band gives the basis function $|S>$ which transform spherically and is chosen imaginary. In the 14×14 model there also is a threefold degeneracy which gives the basis-functions $|X'>$, $|Y'>$ and $|Z'>$. These are similar to the valence bands functions, but are all chosen imaginary. These choices were made in order to maximize the number of real value parameters. The following is the no-spin Hamiltonian matrix pertaining to the 14×14-model, where the lower right 4×4 part of the matrix pertains to the 8×8-model:

$$
\begin{bmatrix}
E'_c + E_k & 0 & 0 & -P_1 k_x & 0 & Q k_z & Q k_y \\
0 & E'_c + E_k & 0 & -P_1 k_y & Q k_z & 0 & Q k_x \\
0 & 0 & E'_c + E_k & -P_1 k_z & Q k_y & Q k_z & 0 \\
P_1 k_x & P_1 k_y & P_1 k_z & E_c + E_k & P_0 k_x & P_0 k_y & P_0 k_z \\
0 & Q k_z & Q k_y & P_0 k_x & L_x + M_{yz} & N_{xy} & N_{xz} \\
Q k_z & 0 & Q k_z & P_0 k_y & N_{xy} & L_y + M_{xz} & N_{yz} \\
Q k_y & Q k_x & 0 & P_0 k_z & N_{xz} & N_{yz} & L_z + M_{xy}
\end{bmatrix}
$$

$$(19)$$

The order of the coefficients in the eigenvector are according to the order of basis states: $|X'>$, $|Y'>$, $|Z'>$, $|S>$, $|X>$, $|Y>$ and $|Z>$. E'_c and E_c are the energies at the expansion points respectively for the higher and first conduction bands. The energy scale is set such that the extreme for the valence band is at zero. E_k is the kinetic term:

$$
E_k = \frac{\hbar^2 \vec{k}^2}{2m} \tag{20}
$$

P_0, P_1 and Q are the intraband momentum matrix elements which are valued real, imaginary and real respectively, and are defined as:

$$
P_0 = \frac{\hbar}{m} < S|p_x|X > \tag{21}
$$

$$
P_1 = \frac{\hbar}{m} < S|p_x|X' > \tag{22}
$$

$$
Q = \frac{\hbar}{m} < X|p_y|Z' > \tag{23}
$$

The notations L_i, M_{ij} and N_{ij} were made to save space in the matrix, and are defined as follows($i, j \in (x, y, z)$):

$$
L_i = L k_i^2, M_{ij} = M(k_i^2 + k_j^2), N_{ij} = N k_i k_j \tag{24}
$$

The constants L, M and N are the constants of the perturbed momentum, and are defined as (E_v is the energy at the extreme of the valence band which is defined as zero):

$$L = \frac{\hbar^2}{2m} + \frac{\hbar^2}{m^2} \sum_{l \in B} \frac{|<X|p_x|l>|^2}{E_v - E_l(0)} \tag{25}$$

$$M = \frac{\hbar^2}{2m} + \frac{\hbar^2}{m^2} \sum_{l \in B} \frac{|<X|p_y|l>|^2}{E_v - E_l(0)} \tag{26}$$

$$N = \frac{\hbar^2}{m^2} \sum_{l \in B} \frac{<X|p_x|l><l|p_y|Y> + <X|p_y|l><l|p_x|Y>}{E_v - E_l(0)} \tag{27}$$

2.2.3. Matrix setup for the case with spin-orbit interaction

In order to make more realistic calculations, spin-orbit interaction effects need to be added to the Hamiltonian:

$$\hat{H} = \hat{H}_0 + \hat{H}_{SO} = \left[\frac{\hat{p}^2}{2m} + V(\vec{r}) \right] + \left[\frac{\hbar}{4m^2c^2} (\nabla V \times \hat{\vec{p}}) \cdot \vec{\sigma} \right] \tag{28}$$

Where $\hat{H}_0$ is the no-spin-Hamiltonian (see eq. (11)), and $\hat{H}_{SO}$ is the spin-orbit-Hamiltonian. This results in the following new Bloch-Hamiltonian:

$$\hat{H}' = \hat{H} + \frac{\hbar^2}{2m}(\vec{k} - \vec{k}_0)^2 + \frac{\hbar}{m}(\vec{k} - \vec{k}_0)\hat{\vec{\pi}} \tag{29}$$

$$\hat{\vec{\pi}} = \hat{\vec{p}} + \frac{\hbar}{4m^2c^2}(\vec{\sigma} \times \nabla V) \tag{30}$$

The operators $\hat{H}_0$ and $\hat{H}_{SO}$ are made into matrices $\mathcal{H}$ and $\mathcal{H}_{SO}$ separately exactly as in eqs. (16) and (17). However, due to the different spin orientations $\pm 1/2$, the collection of basis-functions is expanded to the double. The new basis-functions are denoted α for spin $+1/2$, and β for spin $-1/2$:

$$|X^{(\prime)}\alpha>, |Y^{(\prime)}\alpha>, |Z^{(\prime)}\alpha>, |S\alpha>, |X^{(\prime)}\beta>, |Y^{(\prime)}\beta>, |Z^{(\prime)}\beta>, |S\beta> \tag{31}$$

Here $|X^{(\prime)}\alpha>$ represents the two functions $|X\alpha>$ from the valence band and $|X'\alpha>$ from the upper conduction bands, and also the same for $|Y^{(\prime)}>$ and $|Z^{(\prime)}>$.

In addition to considering symmetries of the coordinates as in section 2.2.2, also the spin orientations have to be considered(they must be the same) in order to pick out nonzero elements for the matrices. If the functions in eq. (31) are used in the basis-vector $\mathcal{F}$ as in section 2.2.2, the following spin-orbit Hamiltonian is obtained:

$$\mathcal{H}_{SO} = \frac{\Delta_{0/1}}{3} \begin{bmatrix} 0 & -\imath & 0 & 0 & 0 & 1 \\ \imath & 0 & 0 & 0 & 0 & -\imath \\ 0 & 0 & 0 & -1 & \imath & 0 \\ 0 & 0 & -1 & 0 & \imath & 0 \\ 0 & 0 & -\imath & -\imath & 0 & 0 \\ 1 & \imath & 0 & 0 & 0 & 0 \end{bmatrix} \tag{32}$$

The spin-orbit interaction works within the valence states, within the upper conduction states, and across these states. The s-like states do not contribute to the spin-orbit interaction. In order to diagonalize these matrices, the expanded Kohn-Luttinger basis is introduced, where the valence states and the upper conduction states are constructed in the same manner:

$$|\frac{1}{2},+\frac{1}{2}>_c = |S\alpha> \tag{33}$$

$$|\frac{1}{2},-\frac{1}{2}>_c = |S\beta> \tag{34}$$

$$|\frac{3}{2},+\frac{3}{2}>_{v(c)} = \frac{1}{\sqrt{2}}(|X^{(\prime)}\alpha> + \imath|Y^{(\prime)}\alpha>) \tag{35}$$

$$|\frac{3}{2},+\frac{1}{2}>_{v(c)} = \frac{1}{\sqrt{6}}(\imath|X^{(\prime)}\beta> - |Y^{(\prime)}\beta> - 2\imath|Z^{(\prime)}\alpha>) \tag{36}$$

$$|\frac{3}{2},-\frac{1}{2}>_{v(c)} = \frac{1}{\sqrt{6}}(|X^{(\prime)}\alpha> - \imath|Y^{(\prime)}\alpha> + 2|Z^{(\prime)}\beta>) \tag{37}$$

$$|\frac{3}{2},-\frac{3}{2}>_{v(c)} = \frac{1}{\sqrt{2}}(\imath|X^{(\prime)}\beta> + |Y^{(\prime)}\beta>) \tag{38}$$

$$|\frac{1}{2},+\frac{1}{2}>_{v(c)} = \frac{1}{\sqrt{3}}(|X^{(\prime)}\beta> + \imath|Y^{(\prime)}\beta> + |Z^{(\prime)}\alpha>) \tag{39}$$

$$|\frac{1}{2},-\frac{1}{2}>_{v(c)} = \frac{1}{\sqrt{3}}(-\imath|X^{(\prime)}\alpha> - |Y^{(\prime)}\alpha> + \imath|Z^{(\prime)}\beta>) \tag{40}$$

These transformations results in the following spin-orbit matrix:

$$\mathcal{H}_{SO} = \Delta_{0/1}\begin{bmatrix} \frac{1}{3} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{3} & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{3} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{3} & 0 & 0 \\ 0 & 0 & 0 & 0 & -\frac{2}{3} & 0 \\ 0 & 0 & 0 & 0 & 0 & -\frac{2}{3} \end{bmatrix} \tag{41}$$

Further it is beneficial to do some changes in the energy scale: $E_v + \Delta_0/3 = 0$, $E_c = E_0$ and $E_c' - 2\Delta/3 = E_0 + E_1$. As the momentum operator $\hat{\vec{p}}$ has been changed to $\hat{\vec{\pi}}$, the momentum matrix elements need to be adjusted. The resulting Hamiltonian matrices for the 8×8 ($\mathcal{H}_{8\times8}$) and the 14×14-model ($\mathcal{H}_{14\times14}$) are as follows:

$$F = -\gamma_1 k^2 - \gamma_2(k^2 - 3k_z^2) \tag{42}$$

$$G = -\gamma_1 k^2 + \gamma_2(k^2 - 3k_z^2) \tag{43}$$

$$H = \imath 2\sqrt{3}\gamma_3 k_z k_- \tag{44}$$

$$I = -\sqrt{3}\gamma_2(k_x^2 - k_y^2) - i2\sqrt{3}\gamma_3 k_x k_y \tag{45}$$

The parameters γ_1, γ_2 and γ_3 are the Luttinger parameters which are experimental values. However, if the spin-orbit coupling is weak, the new momentum operator $\hat{\vec{\pi}}$ approaches the normal momentum operator $\hat{p}$, and the constants L, M and N from section 2.2.2 are usable to make the following approximations:

$$\gamma_1 = -\frac{L + 2M}{3} \tag{46}$$

$$\gamma_2 = -\frac{L - M}{6} \tag{47}$$

$$\gamma_3 = -\frac{N}{6} \tag{48}$$

It should be noted that KPBAND does not use the Luttinger parameters directly, but rather the values A, B and D, which are defined as follows:

$$A = -\gamma_1 \tag{49}$$

$$B = -2\gamma_2 \tag{50}$$

$$D = -2\sqrt{3}\gamma_3 \tag{51}$$

Due to the lack of space the following simplified notations are used:

$$E_{c1} = E_0 + \frac{\hbar^2 \vec{k}^2}{2m} \tag{52}$$

$$E_{c2} = E_0 + E_1 + \frac{\hbar^2 \vec{k}^2}{2m} \tag{53}$$

$$
\mathcal{H}_{8\times8} =
\begin{pmatrix}
E_{c1} & 0 & \frac{P_0 k_+}{\sqrt{2}} & -\frac{\imath\sqrt{2}P_0 k_z}{\sqrt{3}} & \frac{P_0 k_-}{\sqrt{6}} & 0 & \frac{P_0 k_z}{\sqrt{3}} & -\frac{\imath P_0 k_-}{\sqrt{3}} \\
0 & E_{c1} & 0 & \frac{\imath P_0 k_+}{\sqrt{6}} & \frac{\sqrt{2}P_0 k_z}{\sqrt{3}} & \frac{\imath P_0 k_-}{\sqrt{2}} & \frac{P_0 k_+}{\sqrt{3}} & \frac{\imath P_0 k_z}{\sqrt{3}} \\
\frac{P_0 k_-}{\sqrt{2}} & 0 & F & H & I & 0 & \frac{\imath}{\sqrt{2}}H & -\imath\sqrt{2}I \\
\frac{\imath\sqrt{2}P_0 k_z}{\sqrt{3}} & -\frac{\imath P_0 k_-}{\sqrt{6}} & H^* & G & 0 & I & \frac{\imath}{\sqrt{2}}(G-F) & \frac{\imath\sqrt{3}}{\sqrt{2}}H \\
\frac{P_0 k_+}{\sqrt{6}} & \frac{\sqrt{2}P_0 k_z}{\sqrt{3}} & I^* & 0 & G & -H & -\frac{\imath\sqrt{3}}{\sqrt{2}}H^* & \frac{\imath}{\sqrt{2}}(G-F) \\
0 & -\frac{\imath P_0 k_+}{\sqrt{2}} & 0 & I^* & -H^* & F & -\imath\sqrt{2}I^* & -\frac{\imath}{\sqrt{2}}H^* \\
\frac{P_0 k_z}{\sqrt{3}} & \frac{P_0 k_-}{\sqrt{3}} & -\frac{\imath}{\sqrt{2}}H^* & -\frac{\imath}{\sqrt{2}}(G-F) & \frac{\imath\sqrt{3}}{\sqrt{2}}H & \imath\sqrt{2}I & \frac{1}{2}(F+G)+\Delta_0 & 0 \\
\frac{\imath P_0 k_+}{\sqrt{3}} & -\frac{\imath P_0 k_z}{\sqrt{3}} & \imath\sqrt{2}I^* & -\frac{\imath\sqrt{3}}{\sqrt{2}}H^* & -\frac{\imath}{\sqrt{2}}(G-F) & \frac{\imath}{\sqrt{2}}H & 0 & \frac{1}{2}(F+G)+\Delta_0
\end{pmatrix}
\tag{54}
$$

The order of the coefficients in the eigenvector according to the order of basis states: $|\frac{1}{2}, +\frac{1}{2}>_c$(s-like), $|\frac{1}{2}, -\frac{1}{2}>_c$, $|S>$(s-like), $|\frac{3}{2}, +\frac{3}{2}>_v$, $|\frac{3}{2}, +\frac{1}{2}>_v$, $|\frac{3}{2}, -\frac{1}{2}>_v$, $|\frac{3}{2}, -\frac{3}{2}>_v$, $|\frac{1}{2}, +\frac{1}{2}>_v$ and $|\frac{1}{2}, -\frac{1}{2}>_v$

$$
\begin{bmatrix}
E_{c2}+\Delta_1 & 0 & 0 & 0 & 0 & 0 & -\frac{P_1 k_-}{\sqrt{2}} & 0 & \frac{\Delta'}{3} & -\frac{Q k_+}{\sqrt{3}} & -\frac{\imath Q k_z}{\sqrt{3}} & 0 & -\frac{\imath Q k_+}{\sqrt{6}} & -\frac{\sqrt{2} Q k_z}{\sqrt{3}} \\
0 & E_{c2}+\Delta_1 & 0 & 0 & 0 & 0 & -\frac{\imath\sqrt{2}P_1 k_z}{\sqrt{3}} & \frac{\imath P_1 k_-}{\sqrt{6}} & -\frac{Q k_-}{\sqrt{3}} & \frac{\Delta'}{3} & 0 & -\frac{\imath Q k_z}{\sqrt{3}} & 0 & -\frac{\imath Q k_+}{\sqrt{2}} \\
0 & 0 & E_{c2}+\Delta_1 & 0 & 0 & 0 & -\frac{P_1 k_+}{\sqrt{6}} & -\frac{\sqrt{2}P_1 k_z}{\sqrt{3}} & \frac{\imath Q k_z}{\sqrt{3}} & 0 & \frac{\Delta'}{3} & \frac{Q k_+}{\sqrt{3}} & \frac{\imath Q k_-}{\sqrt{2}} & 0 \\
0 & 0 & 0 & E_{c2}+\Delta_1 & 0 & 0 & 0 & \frac{\imath P_1 k_+}{\sqrt{2}} & 0 & \frac{\imath Q k_z}{\sqrt{3}} & \frac{Q k_-}{\sqrt{3}} & \frac{\Delta'}{3} & \frac{\sqrt{2} Q k_z}{\sqrt{3}} & \frac{\imath Q k_-}{\sqrt{6}} \\
0 & 0 & 0 & 0 & E_{c2} & 0 & -\frac{P_1 k_z}{\sqrt{3}} & -\frac{P_1 k_-}{\sqrt{3}} & \frac{\imath Q k_-}{\sqrt{6}} & 0 & -\frac{\imath Q k_+}{\sqrt{2}} & \frac{\sqrt{2} Q k_z}{\sqrt{3}} & -\frac{2\Delta'}{3} & 0 \\
0 & 0 & 0 & 0 & 0 & E_{c2} & -\frac{\imath P_1 k_+}{\sqrt{3}} & \frac{\imath P_1 k_z}{\sqrt{3}} & -\frac{\sqrt{2} Q k_z}{\sqrt{3}} & \frac{\imath Q k_-}{\sqrt{2}} & 0 & -\frac{\imath Q k_+}{\sqrt{3}} & 0 & -\frac{2\Delta'}{3} \\
\frac{P_1 k_+}{\sqrt{2}} & -\frac{\imath\sqrt{2}P_1 k_z}{\sqrt{3}} & \frac{P_1 k_-}{\sqrt{6}} & 0 & \frac{P_1 k_z}{\sqrt{3}} & -\frac{\imath P_1 k_-}{\sqrt{3}} & & & & & & & & \\
0 & \frac{\imath P_1 k_+}{\sqrt{6}} & \frac{\sqrt{2}P_1 k_z}{\sqrt{3}} & \frac{\imath P_1 k_-}{\sqrt{2}} & \frac{P_1 k_+}{\sqrt{3}} & \frac{\imath P_1 k_z}{\sqrt{3}} & & & & & & & & \\
-\frac{\Delta'}{3} & -\frac{Q k_+}{\sqrt{3}} & -\frac{\imath Q k_z}{\sqrt{3}} & 0 & -\frac{\imath Q k_+}{\sqrt{6}} & -\frac{\sqrt{2} Q k_z}{\sqrt{3}} & & & & & & & & \\
-\frac{Q k_-}{\sqrt{3}} & -\frac{\Delta'}{3} & 0 & -\frac{\imath Q k_z}{\sqrt{3}} & 0 & -\frac{\imath Q k_+}{\sqrt{2}} & & & \mathcal{H}_{8\times 8} & & & & & \\
\frac{\imath Q k_z}{\sqrt{3}} & 0 & -\frac{\Delta'}{3} & \frac{Q k_+}{\sqrt{3}} & \frac{\imath Q k_-}{\sqrt{2}} & 0 & & & & & & & & \\
0 & \frac{\imath Q k_z}{\sqrt{3}} & \frac{Q k_-}{\sqrt{3}} & -\frac{\Delta'}{3} & \frac{\sqrt{2} Q k_z}{\sqrt{3}} & \frac{\imath Q k_-}{\sqrt{6}} & & & & & & & & \\
\frac{\imath Q k_-}{\sqrt{6}} & 0 & -\frac{\imath Q k_+}{\sqrt{2}} & \frac{\sqrt{2} Q k_z}{\sqrt{3}} & \frac{2\Delta'}{3} & 0 & & & & & & & & \\
-\frac{\sqrt{2} Q k_z}{\sqrt{3}} & \frac{\imath Q k_-}{\sqrt{2}} & 0 & -\frac{\imath Q k_+}{\sqrt{6}} & 0 & \frac{2\Delta'}{3} & & & & & & & &
\end{bmatrix}
\tag{55}
$$

The order of the coefficients in the eigenvector according to the order of basis states: $|\frac{3}{2},+\frac{3}{2}>_c$(p-like), $|\frac{3}{2},+\frac{1}{2}>_c$(p-like), $|\frac{3}{2},-\frac{1}{2}>_c$(p-like), $|\frac{3}{2},-\frac{3}{2}>_c$(p-like), $|\frac{1}{2},+\frac{1}{2}>_c$(p-like), $|\frac{1}{2},-\frac{1}{2}>_c$(p-like), $|\frac{1}{2},+\frac{1}{2}>_c$(s-like), $|\frac{1}{2},-\frac{1}{2}>_c$, $|S>$(s-like), $|\frac{3}{2},+\frac{3}{2}>_v$, $|\frac{3}{2},+\frac{1}{2}>_v$, $|\frac{3}{2},-\frac{1}{2}>_v$, $|\frac{3}{2},-\frac{3}{2}>_v$, $|\frac{1}{2},+\frac{1}{2}>_v$ and $|\frac{1}{2},-\frac{1}{2}>_v$

2.2.4. Cubic harmonic polynomials for corrections

In order to produce more realistic band structures from the $\vec{k} \cdot \hat{\vec{p}}$ method, corrections consisting of cubic harmonic polynomials are used. The corrections are supposed to affect the bands far from the Γ-point, as it is here where the deviations are the greatest. However each direction ((100), (110) and (111)) in the 1BZ needs to be affected with different strengths. For this reason the cubic harmonic functions are weighted differently. The the corrected bands, $E'_n(\vec{k})$ become:

$$E'_n(\vec{k}) = E_n(\vec{k}) + f_n(\vec{k}) \tag{56}$$

where $f_n(\vec{k})$ is the correction function of a specific band:

$$f_n(\vec{k}) = p_n(\vec{k}) \cdot \sum_{i \in 0,4,6} u_i(\vec{k}) \tag{57}$$

The first term, $p_n(\vec{k})$, is the polynomial

$$p_n(\vec{k}) = \sum_{i=i_{min}}^{i_{max}} a_i^n k^{2i} \tag{58}$$

where a_i^n are coefficients, and it and both i_{min} and i_{max} are optimized for the specific cases. The second term of eq. (5) are the cubic harmonic functions. These functions are chosen such that u_0 affects the (100)-(and (110)- and (111)-) direction, u_4 affects the (110)-(and (111)-) direction, and u_6 affects the (111)-direction only.

$$\begin{aligned} u_0 &= 1 \\ u_4 &= \sum_{i,j \in x,y,z} k_i k_j, \quad i \neq j \\ u_6 &= k_x k_y k_z \end{aligned} \tag{59}$$

Here all k_i are direction cosines of $\vec{k}$, so only the a_i^n need to be adjusted in the polynomials of eq. (58) in the optimization. In Figure 7 a comparisons between the $\vec{k} \cdot \hat{\vec{p}}$ for the affected bands are made for the cases with corrections at 77K and 300K. As will be argued in chapter 4 only the light holes (LH), heavy holes (HH) and the first conduction band electrons (CB) are interesting in doing simulations, so only theses bands are corrected in order to obtain bands similar to those of the *ab initio* pseudopotential calculations.

2.3. *Ab initio* pseudopotential method

In this section it is described, how pseudopotentials (and pseudowave-functions) are used to solve the Schrödinger equation. The pseudopotentials may be what is called *empirical* and *ab initio*. *Ab initio* pseudopotentials are derived from the atomic calculations of the material in question, so it is almost totally unbiased. Empirical pseudopotentials are given by empirically derived potentials, which are imposed on the system. This approach provides a good description of the band structures of a semiconductor, and due to it being simpler it is much used, among others by Bertazzi [7]. However, this method of imposing an empirical pseudopotential rather than deriving one extracted

from the material calculations, makes the empirical pseudopotential method less rigorous. For this reason the *ab initio* pseudopotential method has been chosen for this thesis. Solving this must then be done with an *ab initio* program. For this reason the calculations are extracted from the ABINIT-program using the Local Spin Density Approximation (LSDA) Troulier-Martins method [12].

2.3.1. Kohn-Sham auxiliary system

The first step in describing the electron states of the atoms is to solve the many-body problem for each of the materials involved, Hg, Cd and Te. Solving a many-body problem demands extensive computer power for such heavy elements, and is only applicable for the lightest elements such as H, He and Li. In this occasion the density functional theory (DFT) is introduced in order to simplify the calculations, so the resulting problem is to solve a Kohn-Sham auxiliary system.

$$(H_{KS}^\sigma - \epsilon_i^\sigma)\psi_i^\sigma(\vec{r}) = 0 \tag{60}$$

Where ϵ_i^σ are the eigenvalues, and H_{KS}^σ is the Kohn-Sham hamiltonian defined in Hartree atomic units

$$H_{KS}^\sigma(\vec{r}) = -\frac{1}{2}\nabla^2 + V_{KS}^\sigma(\vec{r}) \tag{61}$$

with,

$$V_{KS}^\sigma(\vec{r}) \begin{array}{l} = V_{ext}(\vec{r}) + \frac{\delta E_{Hartree}}{\delta n(\vec{r},\sigma)} + \frac{\delta E_{xc}}{\delta n(\vec{r},\sigma)} \\ = V_{ext}(\vec{r}) + V_{Hartree}(\vec{r}) + V_{xc}^\sigma(\vec{r}) \end{array} \tag{62}$$

where V_{ext} is the external potential, $V_{Hartree}$ is the Hartree potential, V_{xc}^σ is the exchange-correlation potential and density n. The latter potential is the most difficult one to accurately describe. However, great progress has been made to obtain these through the local density approximation (LDA) and generalized gradient approximation (GGA) described in section 2.3.2. The collective potential $V_{KS}^\sigma(\vec{r})$ is replaced with a pseudopotential $V^{PP}(\vec{r})$. In deriving this potential, a simpler and smoother radial pseudowavefunction $R_l^{PP}(r)$ is proposed, which disregards the affects of the core electrons. This new function has no nodes, and is unequal to the all-electron wavefunction up to the core radius r_{cl}, from where they become equal. The pseudowave-function must provide the same energy eigenvalue as the all-electron wavefunction. From Troullier and Martins [12] the following pseudowave-functions are proposed:

$$R_l^{PP}(r) = \begin{cases} r^l \exp[p(r)], r < r_{cl} \\ R_l^{AE}, r \geq r_{cl} \end{cases} \tag{63}$$

where,

$$p(r) = \sum_{i=0}^{6} c_{2i} r^{2i} \tag{64}$$

The seven coefficients are decided from seven rules:

- (i) Norm-conservation of charge inside r_{cl}

- $(ii - vi)$ Continuity of R_l^{PP} and its first four derivatives at $r = r_{cl}$

- (vii) Zero curvature of the screened pseudopotential at $r = 0$

This pseudopotential is derived from the radial Schrödinger equation for this pseudowavefunction

$$\left[-\frac{1}{2}\frac{d^2}{dr^2} + \frac{l(l+1)}{2r^2} + V_l^{PP}(r) \right] r R_l^{PP}(r) = \epsilon_l r R_l^{PP}(r) \tag{65}$$

where the potential is derived by rearranging the equation to

$$V_l^{PP}(r) = \epsilon_l - \frac{l(l+1)}{2r^2} + \frac{1}{2r R_l^{PP}(r)}\frac{d^2}{dr^2}[r R_l^{PP}(r)] \tag{66}$$

As can be seen the pseudopotentials are decided by how the functions R_l^{PP} are chosen. These functions are calculated from factors such as core charges, core radii and valence band occupations. For this work the actual generation of these potentials are viewed more as 'black box' procedures, and are generated by the program ATOM-PAWv2.2 for a PAW(projector augmented wave) basis.

2.3.2. Local (spin) density approximation (L(S)DA) and generalized gradient approximation (GGA)

Kohn and Sham proposed that that solids could be view as a homogenous electron gas. This leads to the exchange and correlation to be local in character, which leads to the LDA or LSDA where the exchange and correlation energy is the space integral over the exchange correlation energy density of a homogenous electron gas.

$$E_{xc}^{LSDA}[n^{\uparrow}, n^{\downarrow}] = \int d^3 r n(\vec{r}) \epsilon_{xc}^{hom}(n^{\uparrow}(\vec{r}), n^{\downarrow}(\vec{r})) \tag{67}$$

The success of L(S)DA has lead to the development of various generalized gradient approximations (GGAs). The GGA is an improvement of the calculation accuracy by also including the density gradients in the exchange correlation energy density. It was originally an intention to use GGA, but due to lack of time to experiment with it, it was left for the successors of this work. However it should still be presented. A further difference from the LSDA is that the exchange and correlation energy density can not directly be assumed from a homogeneous electron gas, but indirectly it can be described as the exchange energy density of a homogeneous electron gas.

$$E_{xc}^{GGA}[n^{\uparrow}, n^{\downarrow}] = \begin{array}{c} \int d^3 r n(\vec{r}) \epsilon_{xc}(n^{\uparrow}, n^{\downarrow}, |\nabla n^{\uparrow}|, |\nabla n^{\downarrow}|, ...) \\ \int d^3 r n(\vec{r}) \epsilon_x^{hom}(n) F_{xc}(n^{\uparrow}, n^{\downarrow}, |\nabla n^{\uparrow}|, |\nabla n^{\downarrow}|, ...), \end{array} \tag{68}$$

where F_{xc} is the dimensionless exchange and correlation enhancement factor, and $\epsilon_x^{hom}(n)$ is the exchange energy density of a homogeneous electron gas. Further descriptions of the L(S)DA and GGA are not very interesting for this work, but it may be found in more depth in R.M. Martin's Electronic Structure [18] chapter 8 and the article

"Comparison Shopping for a Gradient-Corrected Density Functional" by Perdew and Burke [19].

2.3.3. LDA+U and Scissors operator

One of the greatest problems with the Kohn-Sham approach is that the functionals for exchange and correlation are not systematically developed. This is especially a problem where the electrons are more localized and the interactions are strong, as in compounds such as HgCdTe. In order to compensate for this, the "LDA+U" method [20] is introduced. The "U" is a potential which drags down chosen localized orbitals relative to the other orbitals. For example, d-orbitals lying too high can be pulled down in energy and compactified around the core. The resulting screening of the core charge elevates especially the omnipresent s states of the conduction band. By adjusting this "U" an energy gap equal to that from empirical data is obtained as a side effect. This will generally overdo the adjustment of the d-states, but such effects can be tolerated since the d bands are not important for our purposes here.

The "U" potential was found to operate less smoothly in ABINIT than in the augmented plane wave code WIEN2K [21]. This deficiency might have a few different causes. The implementation of the method could be less thorough in ABINIT, leaving some weaknesses in the calculations. Alternatively the starting pseudopotentials may not have been properly constructed. As will be mentioned in section 3.2.1., there were no generator recipes for neither Cd, Hg nor Te on the homepage of ATOMPAW [22]. This leads to the suspicion that no version of ATOMPAW can handle these elements very well. Consequently, in order to solve the band gap problem temporarily, the Scissors operator had to be applied.

The Scissors operator is, although much used, only an emergency solution to the energy gap problem. Where as LDA+U is a pre-solving operator on the system, the Scissors operator is certainly a post-solving operator. That means, the eigenvalue problem is first solved, then the band gap is adjusted. It then follows that resulting changes in the charge density are not accounted for in the solution, leading to an inaccurate solution where the band structure does not change when changing the band gap. Applying the Scissors operator means to brutally cut and paste the band gap to agree with the correct one, which is a rather ugly attempt at solving this problem.

Alternative solutions to this problem could be either the Exact Exchange Correlated Electrons (EECE) [23],[24] and the modified Becke-Johnson method (mBJ)[25],[24]. Unfortunately these options were not implemented in ABINIT, and therefore not used for this thesis. The array of *ab initio* programs is a minefield where the choice of one may not include desired methods found in another program.

2.3.4. Bandstructure and wavefunction in a solid

The pseudopotentials described in the previous section is used to solve the energy eigenvalue problem for the case of a solid. However, the pseudopotentials used for solving the Kohn-Sham equation is not the exact potentials from the previous section, but a re-calculation of them based on solving the Poisson equation for a "good" initial guess for the solution of the Kohn-Sham equation. It is then to be noted that this is an it-

erative approach, so the solving of the Kohn-Sham equation and the re-calculation of the pseudopotential is done multiple times (as is the case for the Hartree-Fock method). The eigenfunctions of the electrons states are in the simplest form assumed be given as plane-waves Fourier series,

$$\Psi_{\vec{k}}(\vec{r}) = \sum_j c(\vec{k} + \vec{G}_j) e^{i(\vec{k}+\vec{G}_j)\cdot\vec{r}} \tag{69}$$

where $c(\vec{k} + \vec{G}_j)$ are the plane-wave coefficients and $\vec{G}_j$ are reciprocal lattice vectors. The electron states are still Bloch states, where the Bloch functions are collections of exponents:

$$u_{\vec{k}}(\vec{r}) = \sum_j c(\vec{k} + \vec{G}_j) e^{i\vec{G}_j\cdot\vec{r}} \tag{70}$$

In reality the number of Fourier components should go to infinity, but in a numerical calculation the components are limited to an energy-cutoff E_{cut}, where $\frac{1}{2}(\vec{k} + \vec{G}_j)^2 \leq E_{cut}$. In order to reduce computational requirements, it is crucial to decrease the number of higher Fourier components, so a pseudopotential as smooth as possible should be chosen.

The energy eigenvalue problem for this method is solved in a similar way as for the $\vec{k} \cdot \hat{p}$-method. The eigenfunctions are defined as a coefficient vector $\mathcal{C}$ and an exponent vector $\mathcal{F}$:

$$\Psi_{\vec{k}}(\vec{r}) = \sum_j e^{i(\vec{k}+\vec{G}_j)\cdot\vec{r}} c(\vec{k} + \vec{G}_j) = \mathcal{F}^{\mathcal{T}}\mathcal{C} \tag{71}$$

The Hamilton operator is then applied on the vector representation of the states. Then the adjoint of the exponent vector is applied on the equation, and it is integrated over $\vec{r}$. This produces a matrix eigenvalue problem with the energy band structure as the eigenvalue, and the coefficient vector as the eigenvector. The matrix elements are derived as follows:

$$\hat{H}\Psi_{\vec{k}}(\vec{r}) = \hat{H} \sum_j e^{i(\vec{k}+\vec{G}_j)\cdot\vec{r}} c(\vec{k} + \vec{G}_j) = \epsilon(\vec{k}) \sum_j e^{i(\vec{k}+\vec{G}_j)\cdot\vec{r}} c(\vec{k} + \vec{G}_j) \tag{72}$$

$$\int d^3 r e^{i(\vec{k}+\vec{G}_i)\cdot\vec{r}} \hat{H}\Psi_{\vec{k}}(\vec{r}) = \sum_j \mathcal{H}_{ij} c(\vec{k} + \vec{G}_j) = \epsilon(\vec{k}) c(\vec{k} + \vec{G}_i) \tag{73}$$

Here $\mathcal{H}_{ij}$ are the matrix elements of the Hamiltonian matrix $\mathcal{H}$, which are expressed as

$$\mathcal{H}_{ij}(\vec{k}) = \frac{1}{2}(\vec{k} + \vec{G}_j)^2 \delta_{ij} + V_{local}(\vec{G}_i - \vec{G}_j) + \sum_l V_{nonlocal,l}(\vec{G}_i + \vec{k}, \vec{G}_j + \vec{k}) \tag{74}$$

Further descriptions of this equation is not very interesting for this thesis, so it is also view as a "black box" procedure. It can be found more thoroughly described in the paper by Troullier and Martins [12].

2.3.5. Orthogonalized plane wave (OPW) approach

The OPW method is the presiding approach for the PAW method, which is the method used in this work. This section is only meant as a building block upon which the the next section and the *ab initio* part of this work is built. The method was introduced by Herring in 1940 [26],[27], as the first quantitative method for calculating bands other than sp-bonded metals [28]. This method provides the understanding that Si is an indirect semiconductor [29],[30].

The OPW states are constructed from a basis of valence states with the form

$$\chi_{\vec{q}}^{OPW}(\vec{r}) = \frac{1}{\Omega} \left\{ \exp^{\imath \vec{q} \cdot \vec{r}} - \sum_j <u_j|\vec{q}> u_j(\vec{r}) \right\}, \tag{75}$$

where

$$<u_j|\vec{q}> = \int d\vec{r} u_j(\vec{r}) \exp^{\imath \vec{q} \cdot \vec{r}}, \tag{76}$$

and it follows that (OPW) is orthogonal to each function u_j. The only requirement for the $u_j(\vec{r})$ is that they are localized around each nucleus. These functions are to be chosen so that eq. (73) is divided into a smooth and a localized part.

In making the problem more easily solvable the valence states are chosen to be of the orthogonalized form. For this reason the states may be labeled by the angular momentum lm, so that all functions making up a state must have the exact same lm. It then follows that the states must take the form

$$\psi_{lm}^v(\vec{r}) = \tilde{\psi}_{lm}^v(\vec{r}) + \sum_j B_{lmj} u_{lmj}(\vec{r}) \tag{77}$$

where ψ is the valence state, $\tilde{\psi}$ is the smooth part, and the second term is the localized part. All quantities can be expressed in terms of the original OPWs by Fourier transforms:

$$\psi_{lm}^v(\vec{r}) = \int d\vec{q} c_{lm}(\vec{q}) \chi_{\vec{q}}^{OPW} \tag{78}$$

$$\tilde{\psi}_{lm}^v(\vec{r}) = \int d\vec{q} c_{lm}(\vec{q}) \exp^{\imath \vec{q} \cdot \vec{r}} \tag{79}$$

$$B_{lmj}^v(\vec{r}) = \int d\vec{q} c_{lm}(\vec{q}) <u_j|\vec{q}> \tag{80}$$

It is then advantageous to express (75) as a transformation from ψ to $\tilde{\psi}$.

$$|\psi_{lm}^v> = \mathcal{T}|\tilde{\psi}_{lm}^v> \tag{81}$$

This transformation in a linear form is the exact method for recovering ψ in the PAW approach.

The most straightforwards approach is to choose the localized states to be core orbitals $u_{lmi} = \psi_{lmi}^c$. This means choosing the potential to be the actual spherical potential near the nucleus, so the ψ_{lmi}^c are the lowest eigenstates of the Hamiltonian.

Note that the OPWs are not orthonormal as each wave has a norm smaller than unity.

$$< \chi_{\vec{q}}^{OPW} | \chi_{\vec{q}}^{OPW} >= 1 - \sum_{j} | < u_j | \vec{q} > |^2 \tag{82}$$

This means that the equations using these waves are generalized eigenvalue problems which include an overlap matrix.

2.3.6. Projector Augmented Wave (PAW) basis

PAW is an improvement of the OPW approach where more modern techniques for calculating total energy, forces and stress are used. Similar to the OPW approach, the PAW approach consists of integrals of the (rapidly varying) all-electron wavefunctions close to the nucleus and are evaluated as a combination of integrals of smooth functions. So just as in the OPW approach, a smooth part of the valence wavefunction ($\tilde{\psi}$) is defined, which is related back to the all-electron valence wavefunctions (ψ) by a linear transform ($\mathcal{T} = 1 + \mathcal{T}_0$). Adopting the Dirac notation, and assuming that the smooth functions can be expressed in partial waves m, gives the expression

$$|\tilde{\psi} >= \sum_{m} c_m |\tilde{\psi}_m >, \tag{83}$$

and the corresponding all-electron function

$$|\psi >= \mathcal{T}|\tilde{\psi} >= \sum_{m} c_m |\psi_m >, \tag{84}$$

which result in the following:

$$|\psi >= |\tilde{\psi} > + \sum_{m} c_m \{|\psi_m > - |\tilde{\psi}_m >\}. \tag{85}$$

Further it is a usual requirement that the transformation is linear, and that the coefficients must be given by a projection in each sphere

$$c_m =< \tilde{p}_m |\tilde{\psi} > \tag{86}$$

for some set of projection operators $\tilde{p}$. Further it is assumed that they satisfy the biorthogonality condition,

$$< \tilde{p}_m |\tilde{\psi}_{m'} >= \delta_{mm'} \tag{87}$$

By applying eq. (86) on eq. (85) it is obtained

$$|\psi >= \mathcal{T}|\tilde{\psi} >= |\tilde{\psi} > + \sum_{m} \{|\psi_m > - |\tilde{\psi}_m >\} < \tilde{p}_m |\tilde{\psi} >, \tag{88}$$

where

$$\mathcal{T} = 1 + \sum_{m} \{|\psi_m > - |\tilde{\psi}_m >\} < \tilde{p}_m |. \tag{89}$$

This operator translates a general operator $\hat{A}$ applicable on the all-electron waves $|\psi_m>$ to an operator $\tilde{A}$ for the smooth waves $|\tilde{\psi}_m>$, where the expression becomes as follows:

$$\tilde{A} = \mathcal{T}^\dagger \hat{A} \mathcal{T} = \hat{A} + \sum_{mm'} |\tilde{p}_m > \{< \psi_m|\hat{A}|\psi_{m'}> - < \tilde{\psi}_m|\hat{A}|\tilde{\psi}_{m'} >\} < \tilde{p}_{m'}| \quad (90)$$

The purpose of these translations of states and operators is to make the descriptions more exact. Although $\tilde{A}$ may be a general translated operator the operator in view is the Hamiltonian operator. Both eigenvalue and the smooth part of the eigenstate are then more accurately calculated. The complete eigenstates is then obtained by applying the translator $\mathcal{T}$ on the smooth wave.

2.3.7. Virtual Crystal Approximation (VCA)

Constructing a VCA for empirical pseudopotentials is a relatively simple task, which has been done by Bertazzi among others [7]. In his work the potential was weighted with the contribution of $CdTe$ and $HgTe$. Then the alloy disorder was included as a distribution of the potential difference of the two materials.

In order to make true *ab initio* pseudopotential calculations of $HgCdTe$, and more specifically for $Hg_{0.72}Cd_{0.28}Te$, a new fictitious $Hg-Cd$ atom should be made. However ATOMPAW and ABINIT do currently have some problems to respectively generating and running these atoms, so for this work $HgTe$ has been used together with $Hg_{0.72}Cd_{0.28}Te$ parameters in order to make calculations similar to those for MCT. I.e. the MCT lattice constant and using the "LDA+U" and Scissors operators to produce band gap as for MCT. Everywhere else $HgTe$ descriptions have been used. If in the future this should be possible for ATOMPAW and ABINIT a true VCA of MCT should be made, but this is left for successors of this work.

Based on communication with Torrent [31] it was uncovered that ATOMPAW as of today is not able to use floating point occupation numbers, but future versions will most likely be able to do so. The same conversation also states that ABINIT is only able to make VCAs for norm conserving potentials. PAW have potentials which are norm conserving.

The $Hg-Cd$ atom should be constructed based on a common core for Hg and Cd, which is Kr. Further valence bands should be those of the heavier element, Hg, and all the way down to the Kr core. The occupations of these bands should be given by weighting the contribution of Hg and Cd in the compound. The following would be the occupation table of the valence orbitals for Cd, Hg and $Hg_{0.72}Cd_{0.28}$:

Valence orbitals	Cd occupations	Hg occupations	$Hg_{0.72}Cd_{0.28}$ occupations
5s	2	2	0.28*2+0.72*2=2.00
6s	0	2	0.28*0+0.72*2=1.44
5p	0	6	0.28*0+0.72*6=4.32
6p	0	0	0.28*0+0.72*0=0.00
4d	10	10	0.28*10+0.72*10=10.00
5d	0	10	0.28*0+0.72*10=7.20
4f	0	14	0.28*0+0.72*14=10.08

3. Numerical models and algorithms

This chapter discusses how the scattering rates are calculated numerically. This calculations need the energies, energy gradients, and in most cases also the carrier eigenfunctions of the bands considered. The first and the last are found by solving the eigenvalue problems based on the 14×14-matrix of eq. (55), or using the ABINIT program with a few complementary programs made for this work. The second will be shown in section 3.1.3. for the $\vec{k} \cdot \hat{p}$-method and section 3.2.4 for the use of ABINIT, and both are shown in depth in Appendix A. For the finished MC program, also the second derivative is needed, so it is included in the programs for calculating the bandstructure, however it is not relevant for the work in this thesis.

The programs produced are written in FORTRAN 90, and are run using an Intel compiler with the "Math Kernel Library" (MKL).

3.1. Band structure via the $14 \times 14 \ \vec{k} \cdot \hat{p}$ method

The program KPBAND has been revised from a strict 6×6 model valid only for wide band gap semiconductors like $GaAs$ to handle both 8×8 and 14×14 models valid also for narrow band gap semiconductors such as $Hg_{0.72}Cd_{0.28}Te$. The calculations are based on the $\vec{k} \cdot \hat{p}$-method as described in section 2.2 using the matrices in eqs. (54) and (55), however only eq. (55) is used for this thesis. The program is set up to be able to calculate the following quantities:

- Energy bands, $E_n(\vec{k})$

- Eigenvectors $C_n(\vec{k})$ corresponding to the energy bands

- Gradients of the energy bands, $\nabla_{\vec{k}} E_n(\vec{k})$

- Second derivatives of the energy bands, $\frac{\partial^2 E_n}{\partial_\alpha \partial_\beta}$, where $(\alpha, \beta) \in (x, y, z)$

The eigenvectors are only useful in calculating the overlap factor described by eqs. (6) and (7) numerically. The program calculating scattering rates may use analytical solutions for this factor, or even assuming it integrates to unity. However the numerical calculation is the only valid approach when considering strain in the semiconductor, but this is outside the scope of this work.

The KPBAND program is initialized using the program KPINI.f90. This program enables the user to make choices of what to calculate and prints this information to an initialization file KPBAND.INI. Both the initialization and the calculation program KPBAND2mkl.f90 are using the module halvorinit.f90 to load the parameter values used in the Hamiltonian matrices. Only this module has to be modified in order to change the materials parameters and the models size (6×6, 8×8 or 14×14). The parameters used are look-up values based on experiments or $ab \ initio$ calculations.

The calculations done by KPBAND are as mentioned intended to be used to calculate scattering rates. As the the $\vec{k} \cdot \hat{p}$-method is an approximate method, the calculations are only reality-near in a fraction of 1st Brillouin zone (1BZ). As mentioned in section 2.2.4 this problem has been (partially) solved by introducing cubic harmonic corrections. This
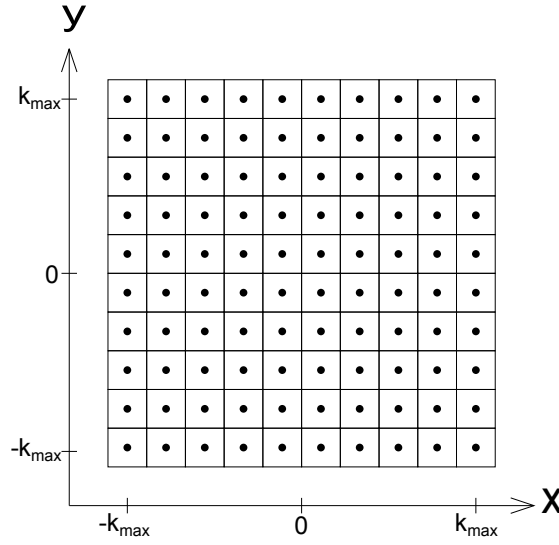
Figure 1: 2D representation of the 3D $\vec{k}$-mesh

leads to the full 1BZ being calculable and is divided into i a 3-dimensional cubic mesh as shown 2-dimensionally in Figure (1). In order to make the calculations more accurate near the Γ-point, it is optional to make a finer grid around it. Both fraction and mesh-number, and whether a finer mesh is to be included, are all selectable from KPINI.f90.

3.1.1. Energies and eigenvectors

The energies with corresponding eigenvectors for the 14×14-model at 77K and 300K are calculated by solving the eigenvalue problems of the Hamiltonian matrix from eq. (55) in a cubic mesh over more than the complete 1BZ. This is done by solving this problem for every cube centroid in the $\vec{k}$-meshes chosen in KPINI.f90. The eigenvalue problems are solved by the subroutine ZHEEV form the MKL-library. If it is desired to make use of the strained semiconductor calculations inherent in the program, it may be done, but only for the 6×6-model. It is important to note, that the bands and vectors will start to mix if the strain results in crossing of the bands. This is due to the fact that ZHEEV-algorithm returns the eigenvalues with eigenvectors ordered by the magnitude of the eigenvalues, and the solver in KPBAND2mkl.f90 expects that this crossing does not happen. This problem is more closely discussed by Halvorsen [1] where the same problem occurs, only that there the NAG math library was used.

In Figure (2) the band structures for a fraction of 1BZ are shown for 77K (blue lines) and 300K (red lines). It can be seen that the structures differ with more than just that the band gap is different. This is not the case for the ABINIT calculations, due to the
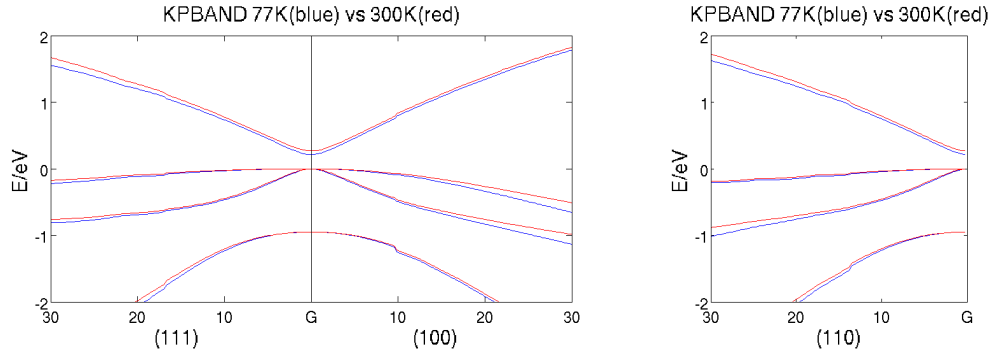
Figure 2: Comparison of close to Γ-point band structures in (100)-, (110)- and (111)-direction using KPBAND. X-axis values given in 1/mm

Scissors operator being used there.

The 77K calculation uses the following parameters:

- $E_0 = 0.265eV$, $E_1 = 3.75eV$, $\Delta_0 = 0.95eV$, $\Delta_1 = 0.25eV$, $\Delta' = \imath 0.0eV$

- $\gamma_1^{14} = -0.623$, $\gamma_2^{14} = 1.080$ and $\gamma_3^{14} = -1.164$

- $P_0 = 8.0eV\,\text{Å}$, $P_1 = \imath 6.5eV\,\text{Å}$, $Q = 7.7eV\,\text{Å}$

The 300K calculation uses the following parameters:

- $E_0 = 0.211eV$, $E_1 = 3.75eV$, $\Delta_0 = 0.95eV$, $\Delta_1 = 0.25eV$, $\Delta' = \imath 0.0eV$

- $\gamma_1^{14} = -0.980$, $\gamma_2^{14} = 1.300$ and $\gamma_3^{14} = -2.425$

- $P_0 = 8.0eV\,\text{Å}$, $P_1 = \imath 6.5eV\,\text{Å}$, $Q = 7.7eV\,\text{Å}$

The definitions of these parameters are found in eqs. (21)-(27) and (46)-(51). It is worth to note that γ_1, γ_2 and γ_3 are not the same for the different model sizes, and they are to be rescaled by $\hbar^2/2m$. For 6×6 model these are the normal Luttinger parameters γ_i^6. For the 8×8 and 14×14 models they are modified as follows(from Pfeffer and Zawadzki [10]):

$$\gamma_1^8 = \gamma_1^6 - \frac{E_{P_0}}{3E_0} \tag{91}$$

$$\gamma_2^8 = \gamma_2^6 - \frac{E_{P_0}}{6E_0} \tag{92}$$

$$\gamma_3^8 = \gamma_3^6 - \frac{E_{P_0}}{6E_0} \tag{93}$$

$$\gamma_1^{14} = \gamma_1^8 - \frac{E_Q}{3(E_0 + E_1)} - \frac{E_Q}{3(E_0 + E_1 + \Delta_1)} \tag{94}$$

$$\gamma_2^{14} = \gamma_2^8 + \frac{E_Q}{6(E_0 + E_1)} \tag{95}$$

$$\gamma_3^{14} = \gamma_3^8 - \frac{E_Q}{6(E_0 + E_1)} \tag{96}$$

$$E_P = 2m\frac{|P|^2}{\hbar^2}, P \in \{P_0, P_1, Q\} \tag{97}$$

These modifications are due to the fact that in the 8×8 model the s-like conduction band is included in category A instead of B, and in the 14×14 model also the p-like upper conduction bands are included in category A instead of B.

The normal Luttinger parameters are obtained by experimentally measuring the effective masses of the different states in different directions(reorganized from Long et al. [32]):

$$\gamma_1^6 = \frac{1}{2}\left(\frac{m}{m_{hh}^*[100]} + \frac{m}{m_{lh}^*[100]}\right) = \frac{1}{2}\left(\frac{m}{m_{hh}^*[111]} + \frac{m}{m_{lh}^*[111]}\right) \tag{98}$$

$$\gamma_2^6 = \frac{1}{4}\left(\frac{m}{m_{lh}^*[100]} - \frac{m}{m_{hh}^*[100]}\right) \tag{99}$$

$$\gamma_3^6 = \frac{1}{4}\left(\frac{m}{m_{lh}^*[111]} - \frac{m}{m_{hh}^*[111]}\right) \tag{100}$$

1.000 of the 1BZ-radius in (100)-direction is used with 40 mesh-points for the coarse mesh, while 0.100 with 40 mesh-points is used fine mesh.

3.1.2. Band gap models and effects for MCT

One of the more interesting features of the MCT is that the band gap E_g which equals the bottom of the conduction band E_0 is both composition and temperature dependent. In the following some of the formulas for calculating the band gap of the composition $Hg_{1-x}Cd_xTe$ are given:

$$E_g = -0.3424 + 1.838x + 0.148x^4 + (0.0629 + 7.68 \cdot 10^{-4}T)\frac{1 - 2.14x}{1 + x} \tag{101}$$

$$E_g = -0.302 + 1.93x - 0.18x^2 + 0.832x^3 + 0.535(1 - 2x) \cdot 10^{-3}T \tag{102}$$

$$E_g = (-0.303 + 1.777x + 0.132x^2) + \frac{(6.3 - 15.47x + 5.92x^2)T^2}{10^4(11 + 67.7x + T)} \tag{103}$$

$$E_g = -0.295 + 1.87x - 0.28x^2 + 0.35x^4 + (6 - 14x - 3x^2) \cdot 10^{-4}T \tag{104}$$

These models for the energy gap were described respectively by Schacham and Finkman [33], Hansen et al. [34], Laurenti et al. [35] and Chu et al. [36]. Figure (3) shows the
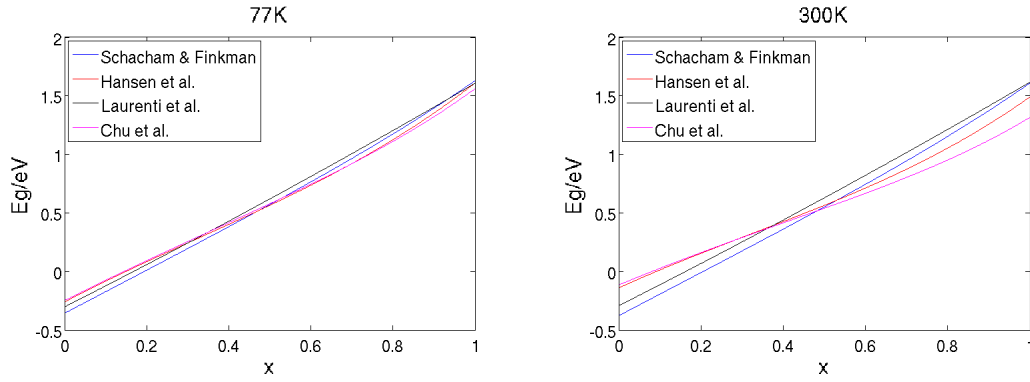
Figure 3: Energy gap models at temperatures 77K(left) and 300K(right)

plots of the models as a function of the composition x at temperatures 77K and 300K. It becomes obvious that the models agree much more at the lower temperatures that at higher. The models are mostly said to be valid up to around 300K, which is the temperature used in the calculations for this thesis. The model chosen for the calculation of the band gap is the one by Schacham and Finkman.

The problem of the band gap is a very prominent one for MCT. $CdTe$ has a large positive band gap, whereas $HgTe$ actually has a negative band gap which results in the light hole band being flipped up to serve as a conduction band and the actual conduction band to substitute the light hole band by lying below the valence band edge. When considering the pseudopotentials, DFT is applied which deals with properties at T=0K. As the band gap of MCT is also dependent on the temperature, the symmetry-related mutual repulsion of the conduction and light hole bands will naturally lead to a change in the effective mass of these two bands as the distance (the band gap) varies. This makes the Scissors operator procedure highly inadequate, whereas these changes are clearly visible in the KPBAND calculations. Also LDA+U is an old fashioned and insufficient approach. More modern trends towards replacing LDA+U include the EECE-method [23],[24] where a fixed parameter $\alpha \in (0, 1)$ determines the amount of exact (Hartree-Fock) exchange to be included. The advantage of EECE is that a single number α is equivalent to having an environmentally dependent U. A single α therefore corresponds to different values of U, depending on the local environment of the atom and orbital where the correction is applied. However, given the circumstances the combination of LDA+U and scissors operator was the best available solution.

Regarding the band gap problem at different temperatures, MCT is one of very few semiconductors where the band gap can increase with temperature. If the alloy fraction x of $CdTe$ exceeds 0.5, MCT has a normal, increasing band gap with temperature, just as for $CdTe$. With a surplus of $HgTe$ in the alloy however, the anomalous temperature dependence of the bang gap sets in.

The formula by Hansen et al. [34] proposes a finite band gap at T=0K. The main failure of DFT is its inability to predict this gap, because it does not allow excited states. On the other hand, the GW approach allows excited states, and will therefore overcome this inadequacy. When it comes to the band gap at temperatures above T=0K, we should also make a distinction between lattice and carrier temperatures. Thorough discussions on the temperature dependence of the band gap including the above mentioned anomaly

of MCT can be found in [37],[38],[39],[40].

3.1.3. Calculation of discrete energy gradients and second derivatives

Derivatives could be calculated numerically by transferring differentials to small intervals, meaning $\partial E_n / \partial k_\alpha \rightarrow \Delta E_n / \Delta k_\alpha$, where $\alpha \in (x, y, z)$, and similarly for the second derivatives. This technique however may lead to large inaccuracies, especially when the curvature of the bands are strong. It is more convenient to use the following analytically derived formulas:

$$\frac{\partial E_n}{\partial k_\alpha} = (\mathcal{C}_n^*)^T \mathcal{H}_\alpha \mathcal{C}_n \tag{105}$$

$$\frac{\partial^2 E_n}{\partial k_\alpha \partial k_\beta} = (\mathcal{C}_n^*)^T \mathcal{H}_{\alpha\beta} \mathcal{C}_n + 2 \sum_{\substack{i \\ E_i \neq E_n}} \frac{\Re\{((\mathcal{C}_n^*)^T \mathcal{H}_\alpha \mathcal{C}_i)((\mathcal{C}_i^*)^T \mathcal{H}_\beta \mathcal{C}_n)\}}{E_n(0) - E_i(0)} \tag{106}$$

These formulas only produce inaccuracies caused by the computers precision instead of the one caused by the discretization of the differentials. The formulas are general in the sense that they do not depend on how the states at the Γ-point are chosen. The derivation of these formulas are given in Appendix A.

The second derivatives are not used in the calculations for this thesis, but they are included in the program for later use.

3.2. Band structure calculations via *ab initio* pseudopotentials

In this section it is described how ABINIT and related programs has been used in order to produce band structures. Further these calculations are converted to a format similar to that of KPBAND program, and finally they are used to calculate scattering rates through the SCRATES program.

First the peudopotentials are generated for a PAW basis through the generator ATOM-PAW.v2.2, and then converted into a format compatible with ABINIT using the converter Atompaw2Abinit.v3.2.0. Then a coarse and a fine mesh is calculated separately by ABINIT. These meshes correspond to those calculated by KPBAND, however in order to save computation time only a $1/48$ wedge (see Figure 4) of the mesh is actually calculated, and through symmetry operations the complete cube is filled in. These operations together with the coarse and fine mesh coupling and conversion to KPBAND energy files are all done by the written program ABINITCONVERTER.f90. At last gradient and second derivative files are calculated by discrete differentiation by the written program ABINITDIFFERENTIATOR.f90. As there still is no way to extract and use the eigenvectors from ABINIT, the ones produced by KPBAND are used instead.

3.2.1. Generating pseudopotentials

The starting pseudopotentials for the materials in question are generated by the ATOM-PAW.v2.2 pseudopotential generator, specifically for solving the Kohn-Sham Schrödinger

equation for a PAW basis. As this program is completely unrelated to the ABINIT-project, so the pseudopotentials have to be converted into a format which can be read by ABINIT. This is done by the Atompaw2Abinit.v3.2.0 pseudopotential converter.

The intended approach was to first produce calculations for $Hg_{0.72}Cd_{0.28}Te$ through a VCA where $Hg_{0.72}Cd_{0.28}$ would be replaced with a fictitious MC (Mercury-Cadmium) atom, and then to use the exact same calculations as would be done for $CdTe$ or $HgTe$. However, as both ABINIT and ATOMPAW.v2.2 had a few issues with producing VCA calculations, this had to be laid on ice. The second best solution was chosen, which was to perform modified $HgTe$ calculations instead of MCT calculations.

As stated in section (2.3.2.) the LSDA Exchange-correlation-functional (XC) was used instead of the GGA XC-functional due to lack of time to perform the proper analyses. For generating smooth pseudo wavefunctions the P. Blöchl scheme [41] was used for all of the materials. This scheme uses the cutoff-function

$$k(r) = [\sin(\pi r/r_{PAW})/(\pi r/r_{PAW})]^2. \tag{107}$$

Further the projectors are orthogonalized by the Gram-Schmidt procedure. Finally the Troullier-Martins scheme is used to produce the pseudopotentials [12] which was more closely described in section 2.3.1.

3.2.2. Initializing and running ABINIT calculations

Running ABINIT and giving it the correct input was the main obstacle in this part of the work. Normally it is only interesting to calculate an image of the band structure which would only be a line from the L-point, through the Γ-point to the X-point, then to the K-point, and back to the Γ-point. Such an image is shown in Figure 8, but the main task is to produce a 3 dimensional mesh (or a coarse and a fine mesh) eclipsing 1BZ for calculating scattering rates, and to use for MC simulations.

Another problem is that *ab initio* programs such as ABINIT and WIEN2K are very heavy programs to run, so reducing the amount of required calculations to make is crucial. For this reason only a wedge covering $1/48$ of the complete mesh is calculated, where the rest is filled in by symmetry operations of this wedge. This greatly reduces the running time from day to hours, and memory requirement to that contained in a normal computer. The wedge mentioned is shown in the 1st octant in Figure 4 and 5.

ABINIT does iterative calculations, meaning the results from one run is used in the next run until some convergence criterion has been met. For this work, and also for most other works with *ab initio* programs, the convergence runs are first made crudely for some points in a more coarse mesh. These points are the so-called special-points and are in this work contained in a $12 \times 12 \times 12$-mesh, where as the meshes extracted for later calculations are $40 \times 40 \times 40$-meshes. The first run is done in order to ease the calculations of the second and final runs for the final coarse and fine meshes.

In extracting the wedge the correct points in reciprocal space have to be given in the input files. This is done by lines through the selected mesh points in the direction parallel to the (100)-direction, close to the x-axis, then moving outwards in y- and z-direction. This procedure is shown in Figure 5 where the red X'es are the actual points in the $40 \times 40 \times 40$-meshes. The way one declares this in ABINIT is to specify the
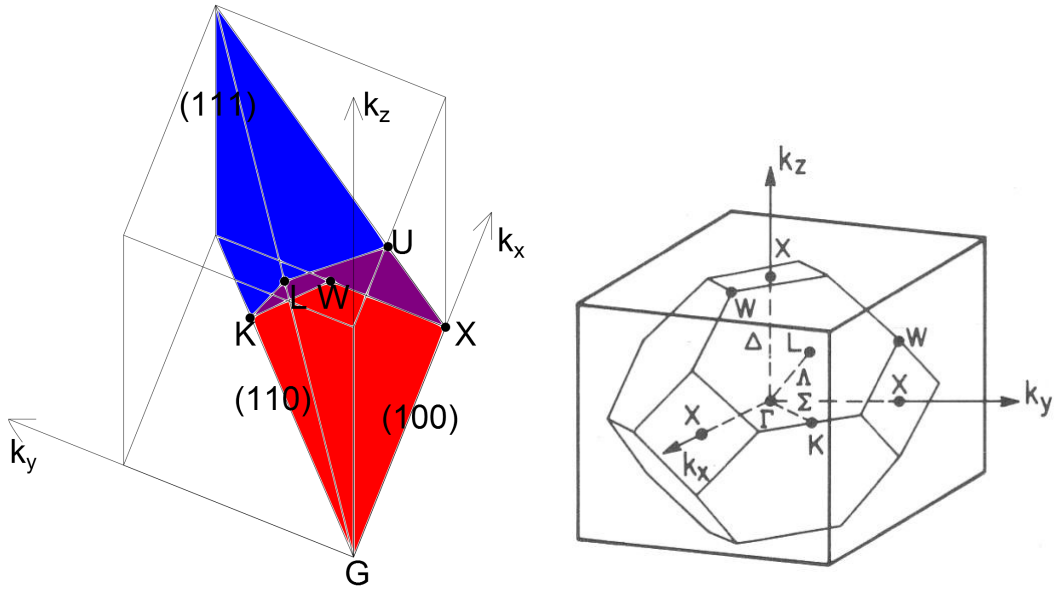
Figure 4: Left side: $1/48$ wedge of the 1BZ. Red is actually inside 1BZ, blue is outside 1BZ, and purple is the 1BZ edge.
Right The complete 1BZ for a general FCC structure such as zincblende.
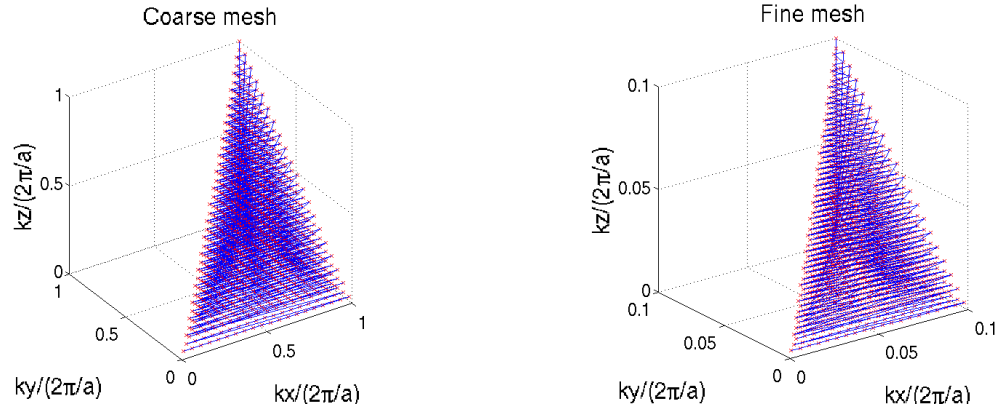
Figure 5: The red X'es are calculated mesh points, and the blue lines show the points order of calculation. The coarse mesh is on the left, and the fine mesh is on the right

endpoints of these lines and the number of points between them. If only the 1D band structure were to be calculated the input file would be short and without a lot of time consuming calculations. In this case this would be very complicated, so the program ABINITGENERATOR.f90 was created in order to generate the input files for both the coarse and the fine mesh calculations. A more thorough overview of the ABINIT input parameters may be found in appendix B.

3.2.3. Converting and generating band structure files

The energies from KPBAND are stored in ".TAB" un-formatted files, where the ten first positions in the files are allocated to information about the calculations (number of mesh points, maximum wave vector, etc. see Appendix D.1.4.). The following algorithm is used for storing the energies:

I=10

```
DO Z=0,MESHNU-1 !(KZ>0 FOR Z.GE.(MESHNU/2))
DO Y=0,MESHNU-1 !(KY>0 FOR Y.GE.(MESHNU/2))
DO X=0,MESHNU-1 !(KX>0 FOR X.GE.(MESHNU/2))
I=I+1
WRITE(10,REC=I) <STORE ENERGY FOR CURRENT K-POINT> ! FOR SOME
FILE WHERE UNIT=10
END DO
END DO
END DO
```

The corresponding calculated values from ABINIT are Z=MESHNU/2,MESHNU-1, Y=Z,MESHNU-1 AND X=Y,MESHNU-1 . Knowing this fact the indices for the 48 analogous points are calculated. In this occasion the values of X, Y and Z are shifted so that the first point is in (1,1,1) instead of (MESHNU/2,MESHNU/2,MESHNU/2). The formulas of the indices are the following six formulas, each containing eight permutations:

For $IZ \pm 1\ IY \pm 1\ IX \pm 1$, where all of the eight possible combinations are used for an arbitrary choice of (X,Y,Z):

$$I = \begin{aligned}(IX * X + MESHNU/2) + (IY * Y + MESHNU/2) * MESHNU \\ + (IZ * Z + MESHNU/2) * MESHNU * MESHNU\end{aligned}$$
(108)

$$I = \begin{aligned}(IX * X + MESHNU/2) + (IY * Z + MESHNU/2) * MESHNU \\ + (IZ * Y + MESHNU/2) * MESHNU * MESHNU\end{aligned}$$
(109)

$$I = \begin{aligned}(IX * Z + MESHNU/2) + (IY * X + MESHNU/2) * MESHNU \\ + (IZ * Y + MESHNU/2) * MESHNU * MESHNU\end{aligned}$$
(110)

$$I = \begin{aligned}(IX * Y + MESHNU/2) + (IY * X + MESHNU/2) * MESHNU \\ + (IZ * Z + MESHNU/2) * MESHNU * MESHNU\end{aligned}$$
(111)

$$I = \begin{aligned}(IX * Y + MESHNU/2) + (IY * Z + MESHNU/2) * MESHNU \\ + (IZ * X + MESHNU/2) * MESHNU * MESHNU\end{aligned}$$
(112)

$$I = \begin{aligned}(IX*Z + MESHNU/2) + (IY*Y + MESHNU/2)*MESHNU\\+(IZ*X + MESHNU/2)*MESHNU*MESHNU\end{aligned}$$

$$(113)$$

This leads to I being an array with a length of 48. The energies are then stored in the following manner:

```
<REA(14)> ! READ ENERGIES FOR THE 14 BANDS (2 AND 2 EQUAL BANDS)
AT CURRENT POSITION X,Y,Z
CALL FILLINDEX(X,Y,X,I)
DO IJ=1,48 ! RUNNING THROUGH ALL THE 48 PERMUTATIONS
DO LI=1,7 ! STORING FOR ALL 7 BANDS
WRITE(10+LI,REC=10+I(IJ)) (REA(2*LI-1)+REA(2*LI))/2.
END DO
END DO
! UNIT.EQ.11 IS THE SPLIT-OFF BAND, UNIT.EQ.12 IS THE LIGHT HOLE BAND,
ETC.
```

3.2.4. Calculating discrete energy gradients and second derivatives

As ABINIT only gives out energy files the gradients and second derivative have to be discretely calculated. This is done by the backwards-forwards approach of discrete differentiation. It is important is it to note that if the mesh point is at the mesh border, then the derivative is in stead taken for the point closer to the origin. The derivation of the differentiation formulas may be found in Appendix A.1.2., and are as follows:

$$\nabla_{\vec{k}}E \rightarrow \frac{\Delta E}{\Delta k_i}\vec{e_i} = \frac{E(\vec{k}+\Delta k_i) - E(\vec{k}-\Delta k_i)}{2\Delta k_i}\vec{e_i} \tag{114}$$

$$\frac{\partial^2 E}{\partial k_i \partial k_j} \rightarrow \begin{cases} \frac{E(\vec{k}+\Delta k_i+\Delta k_j)-E(\vec{k}+\Delta k_i-\Delta k_j)-E(\vec{k}-\Delta k_i+\Delta k_j)+E(\vec{k}-\Delta k_i-\Delta k_j)}{4\Delta k_i \Delta k_j}, i \neq j \\ \frac{E(\vec{k}+\Delta k_i)-2E(\vec{k})+E(\vec{k}-\Delta k_i)}{(\Delta k_i)^2}, i = j \end{cases} \tag{115}$$

where $(i,j) \in (x,y,z)$. Discrete differentiation tends to deviate from the true values where the higher derivatives are large, which is the case close to the Γ-point. An additional problem with the discrete differentiation, is that for smaller differentials, numerical "noise" is attracted.

3.3. Calculation of the scattering rates

The program SCRATES has been slightly revised in order to calculate scattering rates numerically, also based on the 8×8 and 14×14 $\vec{k}\cdot\hat{p}$-models. The switch between them is simply done by changing three parameters, $MS = \{6, 8, 14\}$, $MSH = \{3, 4, 7\}$ and $MS = \{12, 16, 28\}$ to the model size for which the calculations in KPBAND were

made. This program does calculations based on the same mesh in $\vec{k}$-space as KPBAND (and ABINIT) which were discussed in section 3.1. SCRATES uses the numerical values for the energy bands and energy gradients. If the overlap factors are to be calculated based on the eigenvectors, also these are used.

3.3.1. Calculating transition probability

In order to obtain the scattering rate for an initial state $|n, \vec{k}>$, the transition probability from this state to anywhere in a final band n' has the be calculated. This is done by summing eq. (4) over all possible $\vec{k}'$. As the space of $\vec{k}'$ is very dense, the summation is approximated to an integral, where V is the crystal volume.

$$P_{nn'}^m(\vec{k}) = \frac{2\pi}{\hbar} \frac{V}{(2\pi)^3} \int d^3k' |M_{nn'}^m(\vec{k}, \vec{k}')|^2 \delta(E_n(\vec{k}) + \Delta E^m - E_{n'}(\vec{k}')) \qquad (116)$$

As mentioned in the beginning of section 3.3 the part of 1BZ considered is divided into cubes, with sides $2b$. The energy in the scattering mechanism has to be conserved, and from this, a shell of possible vectors $\vec{k}'$ has to be integrated over. As the final state energies most likely are not contained in the cube centers, but somewhere else else in the cubes. The final state energy in a given cube i is Taylor expanded from the cube center to a plane inside the cube, a so called constant energy plane.

$$E_{n'}(\vec{k}') = E_{n'}(\vec{k}') + \vec{k}'' \cdot |\nabla_{\vec{k}_i} E_{n'}(\vec{k}_i)| \qquad (117)$$

$$\vec{k}' = \vec{k}_i + \vec{k}'' \qquad (118)$$

$\vec{k}_i$ is the vector to the cube center i, and $\vec{k}''$ is the vector from this center to a point in the final energy plane. The vector $\vec{k}''$ is further divided into the normal vector to the plane $\vec{k}_\|$ and a vector inside this plane $\vec{k}_\perp$. This leaves the following expression for the final state vector:

$$\vec{k}' = \vec{k}_i + \vec{k}_\| + \vec{k}_\perp \qquad (119)$$

The vectors $\vec{k}_i$ are picked out by checking if the largest possible energy in a cube is greater then the final energy, and the minimum energy there is lower than the final energy. As all energy bands are assumed to be continuous in $\vec{k}$-space, the final energy shell has to pass through this cube.

The length of the vector $w = |\vec{k}_\||$ is calculated under the assumption that the energy in a cube increases linearly along the energy gradient. It is then calculated as follows:

$$w = |\vec{k}_\|| = \frac{E_{n'}(\vec{k}') - E_{n'}(\vec{k}_i)}{|\nabla_{\vec{k}} E_{n'}(\vec{k}_i)|} \qquad (120)$$

The vectors $\vec{k}_\perp$ will not have to be calculated explicitly as it will be all over the of the final energy plane $S(\vec{k}_\|)$ in the cube which will be calculated. In the calculations of the transition probability in each cube ΔE^m is approximated to be constant, which is
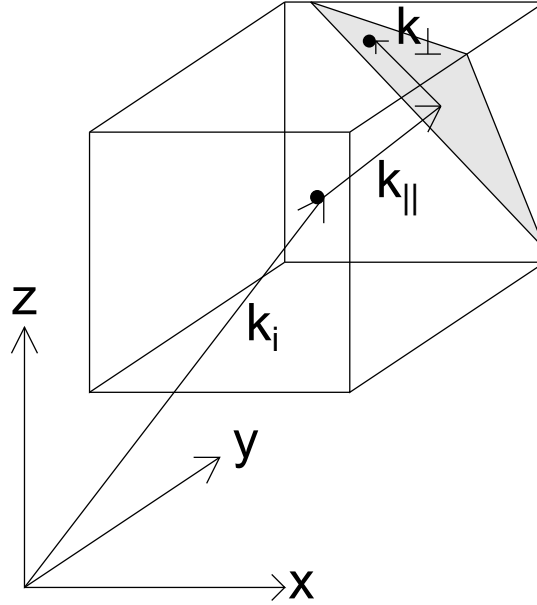
Figure 6: Division of final wave vector

a good approximation if the $\vec{k}$-mesh is chosen fine enough. All of the approximations and techniques mentioned in this section lead to the following formula for calculating eq. (115) numerically:

$$
\begin{aligned}
P_{nn'}^m(\vec{k}) &= \frac{2\pi}{\hbar} \frac{V}{(2\pi)^3} \int d^3k' |M_{nn'}^m(\vec{k}, \vec{k}')|^2 \delta(E_n(\vec{k}) + \Delta E^m - E_{n'}(\vec{k}')) \\
&\approx \sum_i \int dk_{||} \delta(E_n(\vec{k}) + \Delta E^m - E_{n'}(\vec{k}_i) - k_{||}|\nabla_{\vec{k}} E_{n'}(\vec{k}_i)|) \\
&\quad \cdot \int_{S(\vec{k}_{||})} d_\perp^k |M_{nn'}^m(\vec{k}, \vec{k}')|^2
\end{aligned}
\tag{121}
$$

The squared matrix element integral is approximated by the average over the final state plane in the cube.

$$
< |M_{nn'}^m(\vec{k}, \vec{k}')|^2 >_{S(k_{||})} = \frac{1}{S(k_{||})} \int_S d^2k_\perp |M_{nn'}^m(\vec{k}, \vec{k}')|^2
\tag{122}
$$

From this the transition probability is calculated using the state density formalism

$$
P_{nn'}^m(\vec{k}) = \frac{2\pi}{\hbar} \sum_i D(E_n(\vec{k}) + \Delta E^m, \vec{k}_i) < |M_{nn'}^m(\vec{k}, \vec{k}')|^2 >_{S(k_{||})}
\tag{123}
$$

where the density in a given cube is

$$D(E_n(\vec{k}) + \Delta E^m, \vec{k}_i) = \frac{V}{(2\pi)^3} \frac{S(k_{||})}{|\nabla_{\vec{k}} E_{n'}(\vec{k}_i)|} \tag{124}$$

3.3.2. Constant energy plane

The plane area of final energy in a cube is calculated following the Gilat and Raubenheimer method [42]. The normal vector for the plane $\vec{n} = [l_1, l_2, l_3]$ is derived using the energy gradient in the center of the final state cube. $\vec{n}$ is a unity vector, and is transformed in order to satisfy the following:

$$l_1 \geq l_2 \geq l_3 \geq 0 \tag{125}$$

Because of the symmetry in the cubes, the transformation does not in any way change the size of the cross-section, as long as the ratios between the components of $\nabla_{\vec{k}} E_{n'}(\vec{k}_i)$ and $\vec{n}$ are the same. Since all coordinates of $\vec{n}$ are chosen positive there is a maximum of four corners in each cube which may be cut away by the cross-section. The corners at points (b, b, b), $(b, b, 0)$ and $(b, 0, b)$ are the same in all cases, but based in the projection of the plane the points $(b, 0, 0)$ and $(0, b, b)$ are the options for being the forth corner. In the two cases where the fourth point also is cut away, there are two different formulas for calculating the resulting plane area decide. To which formula to use, it has to be figured out which of the corners that have been cut away. This is done by checking how far the cross-section is removed from the cube centroid. If it is closer than the its parallel plane through the given corner, the corner is cut away. These are the formulas for the lengths from the cross-section at cube center to the corners (in opposite order of mentioning in this section):

$$\begin{aligned} w_1 &= b|l_1 - l_2 - l_3| \\ w_2 &= b(l_1 - l_2 + l_3) \\ w_3 &= b(l_1 + l_2 - l_3) \\ w_4 &= b(l_1 + l_2 + l_3) \end{aligned} \tag{126}$$

The following are the formulas for the cross-section with the conditions for using them:

$$S(w) = \begin{cases} \frac{4b^2}{l_1}, & \text{when } 0 < w < w_1 \text{ and } l_1 \geq l_2 + l_3 \\ \frac{2b^2(l_1 l_2 + l_1 l_3 + l_2 l_3) - (w^2 + b^2)}{l_1 l_2 l_3}, & \text{when } 0 < w < w_1 \text{ and } l_1 < l_2 + l_3 \end{cases} \tag{127}$$

$$S(w) = \frac{b^2(3l_2 l_3 + l_1 l_2 + l_1 l_3) + wb(l_1 - l_2 - l_3) - (1/2)(w^2 + b^2)}{l_1 l_2 l_3}, \quad \text{when } w_1 < w < w_2 \tag{128}$$

$$S(w) = 2\frac{b^2(l_1 + l_2) - wb}{l_1 l_2}, \quad \text{when } w_2 < w < w_3 \tag{129}$$

$$S(w) = \frac{(b^2(l_1 + l_2 + l_3) - w)^2}{2l_1 l_2 l_3}, \quad \text{when } w_3 < w < w_4 \tag{130}$$

Eq.(126) are areas of a parallelogram and a hexagon respectively, eq. (127) is the area

of a pentagon, eq. (128) is the area of a trapezoid, and eq. (129) is the area of a triangle. The formulas were made by finding the intersections between the plane and the cube edges and then calculating the areas thereof.

3.3.3. Calculation of the average squared matrix elements

The average squared matrix elements need to be calculated in order to determine the scattering rates. All the squared matrix elements used by Brudevoll et al. [13] have previously been implemented in SCRATES, except for the alloy scattering squared matrix element which is given by Ridley [15] and in sections 2.1.3 and appendix C of this thesis. As mentioned in section 2.1.2 the following scattering mechanisms are available to be calculated:

- Polar optical phonon scattering (absorption and emission)

- Acoustic deformation potential phonon scattering (absorption and emission)

- Non-polar optical phonon scattering (absorption and emission)

- Ionized impurity scattering

- Alloy scattering

The formulas for their squared matrix elements without overlap factors are again given in Appendix C, where $\vec{q} = \vec{k}' - \vec{k}$. The phonon scattering mechanisms all either absorb or emit a phonon. All of these are proportional to the number of existing phonons, $N(\vec{q})$, where $\vec{q}$ is the phonon's wave vector:

$$N(\vec{q}) = \frac{1}{exp(\frac{E(\vec{q})}{k_B T}) - 1} \tag{131}$$

$E(\vec{q})$ is the energy of the phonon which is totally absorbed or emitted. For an acoustic phonon it is assumed that the sound velocity is the same for all modes, although it is not strictly the case. This leads to the energy of the acoustic phonons being $E(\vec{q}) = \hbar v_s q$, where v_s is the average sound velocity. The optical phonons which are transmitted however are insensitive to the magnitude of q. They are all assumed to have the same a frequency ω_0, meaning they all have and energy $E(\vec{q}) = \hbar \omega_0$. Ionized impurity an alloy scatterings have no energy transaction.

The method for calculating the squared matrix element was extracted from Fischetti and Laux. There it was assumed that the squared matrix element varied so slowly in a cube, that is was a good approximation to say that the scattering occurred from an initial state cube centroid, to a final state cube centroid. However, if the case was that the scattering could occur from one cube back to itself, it would lead to $\vec{q} = \vec{k}' - \vec{k} = 0$. When looking at the formulas for the squared matrix elements for polar optical phonon scattering and ionized impurity scattering this would lead to problems as they both have q in their denominators. Fortunately, the large energy transfer for optical phonons often lead to $q = 0$ being a rare case. If the $\vec{k}$-mesh is chosen too coarse, the same cube back scattering could be possible. The calculations of ionized impurity scattering is

also mostly saved from being calculated to infinity. The reason for this is the factor β in the denominator, so this problem will not occur.

The calculation of the squared matrix elements in this thesis however is based on extrapolations from an initial vector $\vec{k}$ to a final vector $\vec{k}'$. The initial vector is still a cube centroid vector. The final is found by extrapolation from a proper final state cube as in Fischetti and Laux by the energy gradient to the proper constant energy plane, so that the energy in the scattering process is conserved. That means, using the approximation $\vec{k}' = \vec{k}_i + \vec{k}_{\parallel}$ instead of $\vec{k}' = \vec{k}_i$. This is similar to the method used to find the scattering area, only that $\vec{k}_{\perp} = 0$ is chosen to calculate $|M(\vec{q})|$. This method produces more accurate calculations of the scattering, and where the chance of obtaining $q = 0$ for polar optical phonons is further reduced. Still $|M(\vec{q})|$ may vary too much for it to be properly approximated to be constant in these scattering planes, so great inaccuracies may occur. This could be solved by performing an averaging of $|M(\vec{q})|$ for different values of $\vec{k}_{\perp}$ in the cross-section. Unfortunately time would not allow for this to be done.

As the linear extrapolations from the cube centroids were added, the only cases where the approximation of $|M(\vec{q})|$ are significant inaccurate occur when $\vec{q} \to 0$ and $\vec{k} \to 0$. The solution to this is to introduce analytical solutions in this area, or to make a finer mesh near $\vec{k} = 0$ so that the inaccuracies are acceptable. Only the second option is relevant for this thesis, as the first would have to be implemented separately in the MC program. This finer mesh has to be chosen to be constructed before calculating the band structures (see Appendix D.1.2).

3.3.4. Calculation of overlap factors

The calculation of overlap factors should for most scattering mechanisms be included. SCRATES can calculate this in three ways:

- Unity overlapping factors

- Analytical calculations

- Calculations based on the band eigenvectors.

The first method is trivial and is only applicable for conduction band electrons in lower energy regime, and if overlap factors are not a part of the scattering mechanism (here only the non-polar optical phonon scattering). The analytical method was derived by Wiley [43], and was derived from the 4×4 model. This model is only applicable for unstrained 6×6 model, so this one is not used for this thesis. However, the formulas, where $\theta = \angle(\vec{k}, \vec{k}')$ are:

$$|G_{nn'}(\vec{k}, \vec{k}')|^2_{intraband} = \frac{1}{4}(1 + 3\cos^2\theta) \tag{132}$$

$$|G_{nn'}(\vec{k}, \vec{k}')|^2_{interband} = \frac{3}{4}\sin^2\theta \tag{133}$$

The third method is to calculate the overlap factors from the band eigenvectors for the different model sizes as follows:

$$|G_{nn'}(\vec{k}, \vec{k'})|^2 = \frac{1}{2} \sum_{\mu=1}^{2} \sum_{\mu'=1}^{2} |\mathcal{F}^{\dagger}_{n'\mu'\vec{k'}} \mathcal{F}_{n\mu\vec{k}}|^2 \qquad (134)$$

This formula is only valid for normal processes as the factor $exp(i\vec{G}\vec{r})$ is not included. Also the eigenvalue solver in KPBAND is not able to distinguish between the spin up and the spin down state in a given band, so the two eigenvectors from a given band are mixed. Therefore, the calculation of the overlapping factors is averaged between the two spin configurations.

4. Results and discussion

In this chapter the results from the band structure- and scattering rate calculations are presented. The first section gives the discussion of the band structure calculations for temperatures 77K and 300K using the 14×14-model in the revised KPBAND program and the *ab initio* pseudopotential method in the ABINIT program. As stated in section 3.1.1. the energies are stored in a cubic mesh grater than and totally eclipsing the 1BZ. Calculations for the 6×6- and 8×8-models are still possible in the KPBAND program, but as the 14×14-model is far superior to the other, they are deemed irrelevant for this thesis. As mentioned in section 2.2.4. cubic harmonic corrections are post-imposed on a few of the bands in order to acquire a more realistic description of these band. These corrections are only applied on light- and heavy holes and 1st conduction band, due to the fact that only these are relevant for this thesis and in the future MC program.

The second section covers the scattering rates of electrons and holes. Both intra- and interband scatterings are calculated for the light- and heavy hole bands, and only intraband scatterings are calculated for the 1st conduction band. As the ionized impurity scattering rate calculation had some numerical difficulties for lower energies, this scattering mechanism is currently laid on ice for future revisers of this program. The same is also the case for the electron scattering of the polar octical phonon scatterings. However the alloy scattering calculation has been included in SCRATES, and has been calculated in this work. All considered scattering mechanisms are based on both the KPBAND and ABINIT band structures. At the end the total scattering rate is calculated by the Matthiessen rule(eqs. (3) and (8)) with a discussion of the importance of the scattering mechanisms. When concerning the MC program also the inverse of the total scattering rate, the free flight time is important, so a discussion of this is included at the end of this chapter.

4.1. Energy band structures

As has been said earlier, the calculations of the scattering rates(eq. (8)) and carrier transport(eqs. (1) and (2)) in a semiconductor depends on its energy band structure. The energy, carrier states, energy gradients and second derivatives are all calculated by the program KPBAND. ABINIT does only calculate the energy, so the energy gradient and second derivatives are calculated by discrete differentiation. For scattering rates which include the overlapping integral, the eigenstates are extracted from the corresponding KPBAND calculations. Although it is not ideal, it is allowed because the $\vec{k} \cdot \hat{p}$- and the *ab initio* pseudopotential method should produce the same band structures, and therefor ultimately the same carrier states.

The parameters used for the calculations of the band structure are experimental values and models which were picked up from a few different articles (see section 3.1.1 and Appendix B). It should be noted that MCT is a very difficult semiconductor to describe, so there does not exist any well accepted parameter values or parameter models. As an example of this, is the four different models which were proposed for the energy gap in section 3.1.1. All of the models were dependent upon the composition ratio as well as the temperature. Concerning the $\vec{k} \cdot \hat{p}$-model a very common approach is to use parameter values for $CdTe$ as MCT parameters, even if the composition is closer to
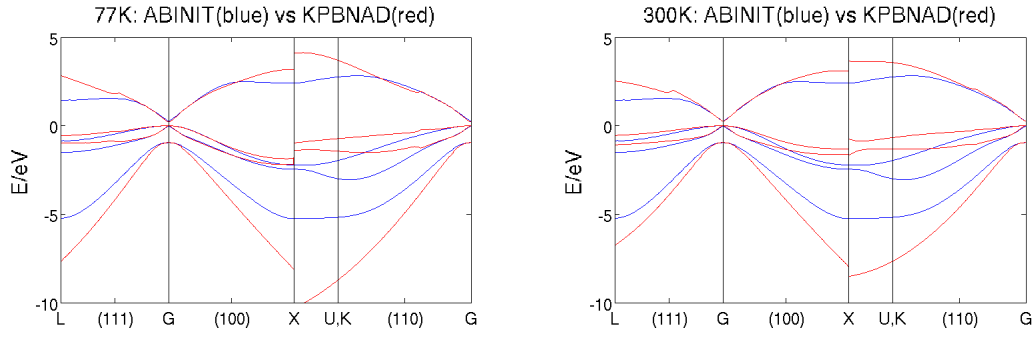
Figure 7: Band structure plots. The blue lines are from the $14 \times 14\ \vec{k} \cdot \hat{\vec{p}}$-model, and the red lines are from the *ab initio* pseudopotential model. 77K plots are on the left and 300K plots are on the right. All bands structures are extracted using the reader made by Tore Sivertsen Bergslid, readtore.f90

$HgTe$. For the *ab initio* calculations however the staring point for MCT is $HgTe$, which is because the composite choice is closer to this than the other.

The parabolic corrections in KPBAND were chosen by trying and failing in order to obtain band structures similar to those of the *ab initio* approach. Figure 7 shows plot of the *ab initio* calculation in red, and the KPBAND calculations for 77K and 300K in blue.

It should further be noted that the band structures resulting from the *ab initio* pseudopotential have been somewhat modified from the raw data given by ABINIT. The first problem is the reference energy for the calculations, which typically may be derived from the potential. This sets a fictitious energy level on the system, so that the entire system needs to be shifted along the energy axis. This is not really a problem for future calculations, but for the intention of pointing this out this fact the raw band structure is shown in Figure 8 with both higher and lower lying bands. A further interesting fact is that some of the bands tend to "split', which is most notable in the split-off and the light hole bands in the (110)-direction. However when using the band structure from ABINIT to do scattering rate calculations the band "splitting" can not be taken into consideration, and the bands have been averaged as can be seen in Figure 8. Also as mentioned in section 2.3.3. the band gaps have been corrected by using the scissors operator.

4.2. Scattering rates

In this section the scattering rates in the (111)-direction are presented. The calculations are based on both the 14×14-$\vec{k} \cdot \hat{\vec{p}}$ model and the *ab initio* pseudopotential method at 77K and 300K. The rates are compared with the results from Bertazzi et al. [8], derelle et al. [14], Skåring [44] and Halvorsen [1]. A very important difference between this work and the works produced by Derelle et al. and Skåring is that they base their calculations on approximated analytical band structures, where as this work is based on more advanced methods for discretely deriving the band structures. This causes the rates to be much less smooth then those for the other two works. The advantage however of the methods used in this work is that the band structures are derived more closely to the first principals, so they should be closer to the true descriptions of the material.
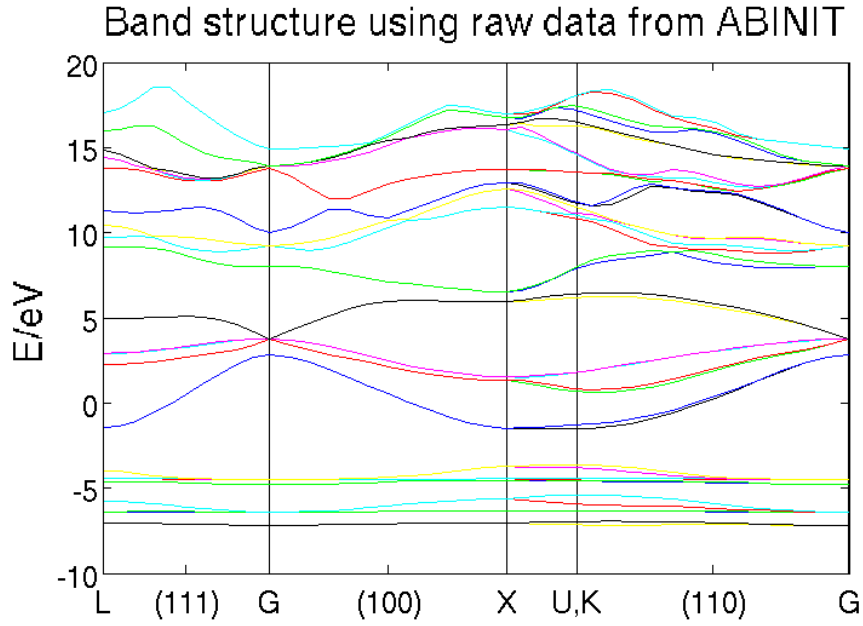
raw.png

Figure 8: Band structures using raw data from ABINIT, with a band gap of 19meV

Further in this section, "LH" denotes light hole, "HH" denotes heavy hole and "CB" denotes the 1st conduction band.

4.2.1. Acoustic deformation potential phonon scattering

The acoustic phonons produce significantly lower scattering rates than the other mechanisms, typically the other rates are more than 5 higher. The acoustic phonons produce rates typically in the range of 10^{11} to 10^{13} for HH $\rightarrow$ HH and LH $\rightarrow$ HH, and rates up to 10^{12} for LH $\rightarrow$ LH and HH $\rightarrow$ LH using both models for calculating the band structures. This is exactly as expected, and it is in good agreement with the results from Halvorsen [1]. The other mentioned works either do not explicitly present these rates, or they do not mention them at all. However these results for the acoustic phonon scattering is in good agreement with current knowledge of MCT. The CB $\rightarrow$ CB scattering rates tend to lie in the range between 10^9 and 10^{11}. Due to the fact that this scattering mechanism is not very prevalent this scattering may be ignored when performing MC simulations.

4.2.2. Alloy scattering

Alloy scattering is as previously stated an addition to SCRATES as a part of this work, which was presented in section 2.1.3. This mechanism is presented by Bertazzi et al. [8] and Derelle et al. [14]. Although the rates in this work typically lie between 10^{12} to 10^{14} for HH $\rightarrow$ HH it seams to be slightly lower than for the two mentioned works. The reason may be that Bertazzi et al. and possibly Derelle et al. presents the total of HH $\rightarrow$ HH and HH $\rightarrow$ LH scatterings in their rates, which in turn would account for the discrepancy between this and their works. The results are roughly the same for 77K and 300K, which makes sense as there are no proportionality between the rate and any phonon occupancy density. The slight difference in the scattering rates for the two

Figure 9: Acoustic deformation potential phonon absorption plots in the (111)-direction. 77K on to and 300K on the bottom. Results form KPBAND on the left and from ABINIT on the right.
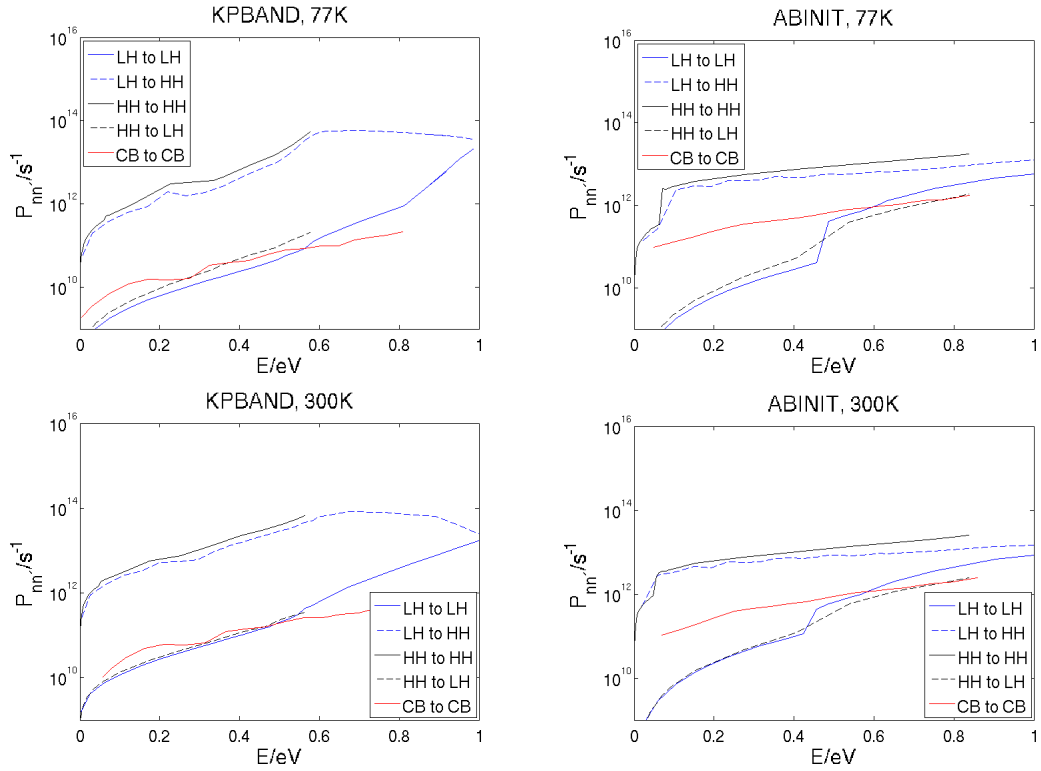
Figure 10: Acoustic deformation potential phonon emission plots in the (111)-direction. 77K on to and 300K on the bottom. Results form KPBAND on the left and from ABINIT on the right.

Figure 11: Alloy scattering plots in the (111)-direction. 77K on to and 300K on the bottom. Results form KPBAND on the left and from ABINIT on the right.

different temperatures is then attributed to the differences in the band structures for 77K and 300K. Here it should be noted that the rates from this work and that from Bertazzi et al. are far from as smooth as that from Derelle et al. As stated in the introduction to this section, the reason for this is that Derelle et al. base the band structures on analytical formulas, where as this work and that of Bertazzi et al. are both based on discretely derived structures more closely to first principal calculations, where smoother rates are much less likely. The results using ABINIT are still somewhat smoother than that using KPBAND, but here it can not be said that either calculation agrees more with those of the previous works. This scattering mechanism is however very prevalent and should definitively be taken into account when performing MC simulations.

4.2.3. Polar optical phonon scattering

Scattering by optical phonons are the most prevalent scattering mechanisms considered in this work. Skåring used the actual particle distribution (in k-space) implemented in the analytical MC program, allowing for screening to be included. The rates in this thesis are based solely on the SCRATES program, so the screening influence was not available. However, following the example of Skåring the rates would decrease to a level comparable to what was expected. It is very important to take this scattering mechanism into account when performing MC simulations. It was originally intended to include electron scattering for this mechanism, but due to some calculation problems this was laid on ice for successors of this work. This problem is also seen for LH $\rightarrow$ LH scattering, especially for the ABINIT calculations where a few spikes appear in the

Figure 12: Polar potential phonon absorption plots in the (111)-direction. 77K on to and 300K on the bottom. Results form KPBAND on the left and from ABINIT on the right.

Figure 13: Polar potential phonon emission plots in the (111)-direction. 77K on to and 300K on the bottom. Results form KPBAND on the left and from ABINIT on the right.
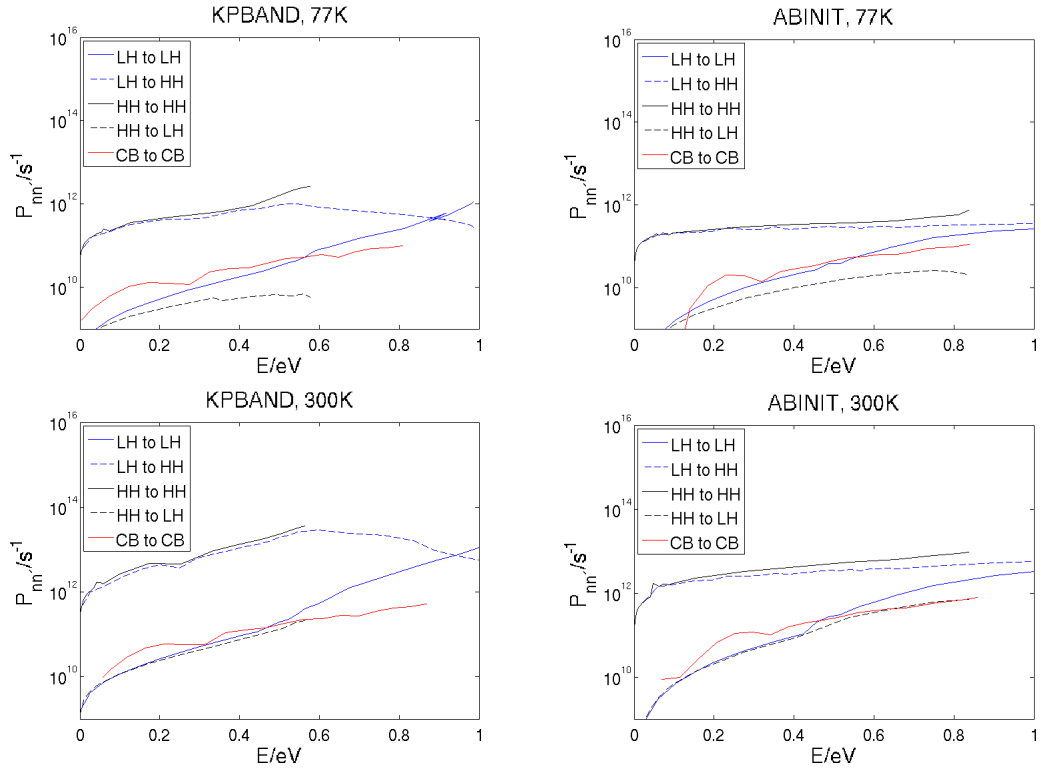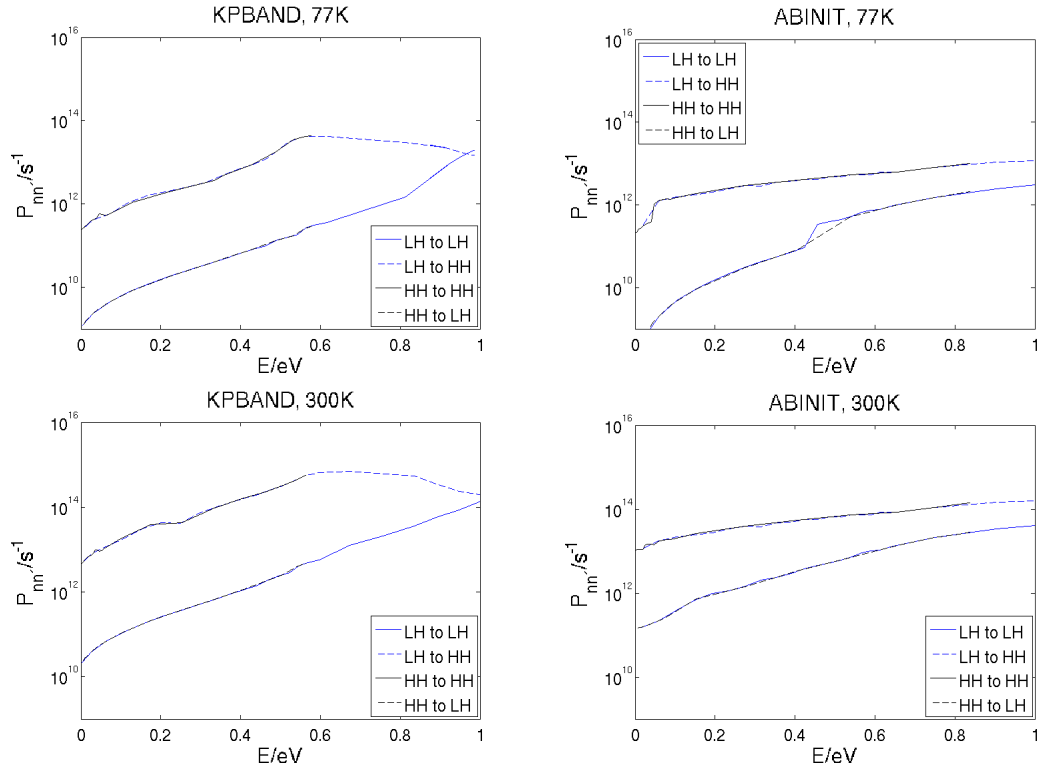
Figure 14: Nonpolar potential phonon absorption plots in the (111)-direction. 77K on to and 300K on the bottom. Results form KPBAND on the left and from ABINIT on the right.

range of 0.4 to 0.7 eV. The reason for the problems with LH and CB is the low effective mass of these carriers. The masses are roughly 0.035 electron mass for this material. Because of the large resulting energy to momentum ratio, the carriers are only scattered by a small phonon $\vec{q}$. This causes the integrand to vary very rapidly over the final cube plane. The solution would be to perform a multipoint averaging of $|M(\vec{q})|$, and this method was also mentioned in section 3.3.3. In the MC program these rates are post-smoothed which further decreases the problem. Such an algorithm has already been implemented by Tore Sivertsen Bergslid.

4.2.4. Nonpolar optical phonon scattering

Nonploar optical phonon scattering, or optical deformation potential scattering is the most prevalent scattering mechanism for MCT, typically ranging from 10^{12} to 10^{14} for HH $\rightarrow$ HH and LH $\rightarrow$ HH scattering at 77K, and about 2 times greater at 300K for emission. This is in good agreement with Bertazzi et al. and Skåring. Compared to the results of Halvorsen for $GaAs$ the rates are a magnitude of 10 higher. Due to the lack of a well known squared optical phonon coupling constant for MCT we had to assume a likely value. Based on results by Bertazzi et al. a value 20 times larger than that of $GaAs$ was chosen. As this scattering mechanism produces the largest scattering rates it is the most important rate to include when performing MC simulations in MCT.
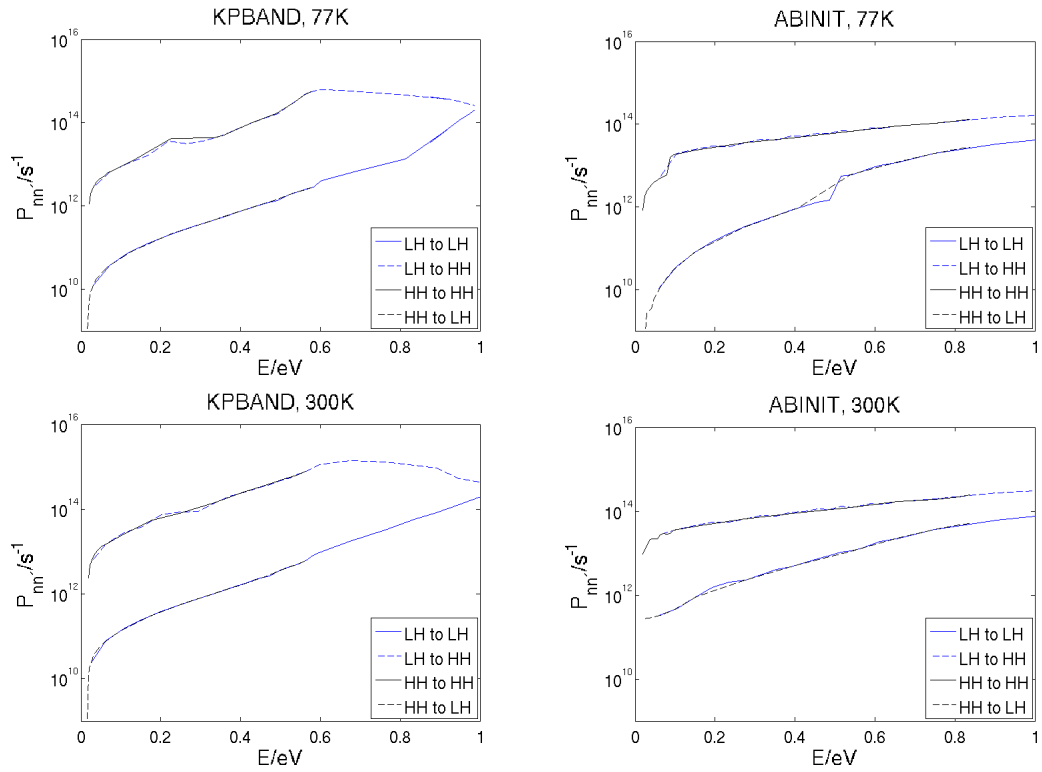
Figure 15: Nonpolar potential phonon emission plots in the (111)-direction. 77K on to and 300K on the bottom. Results form KPBAND on the left and from ABINIT on the right.
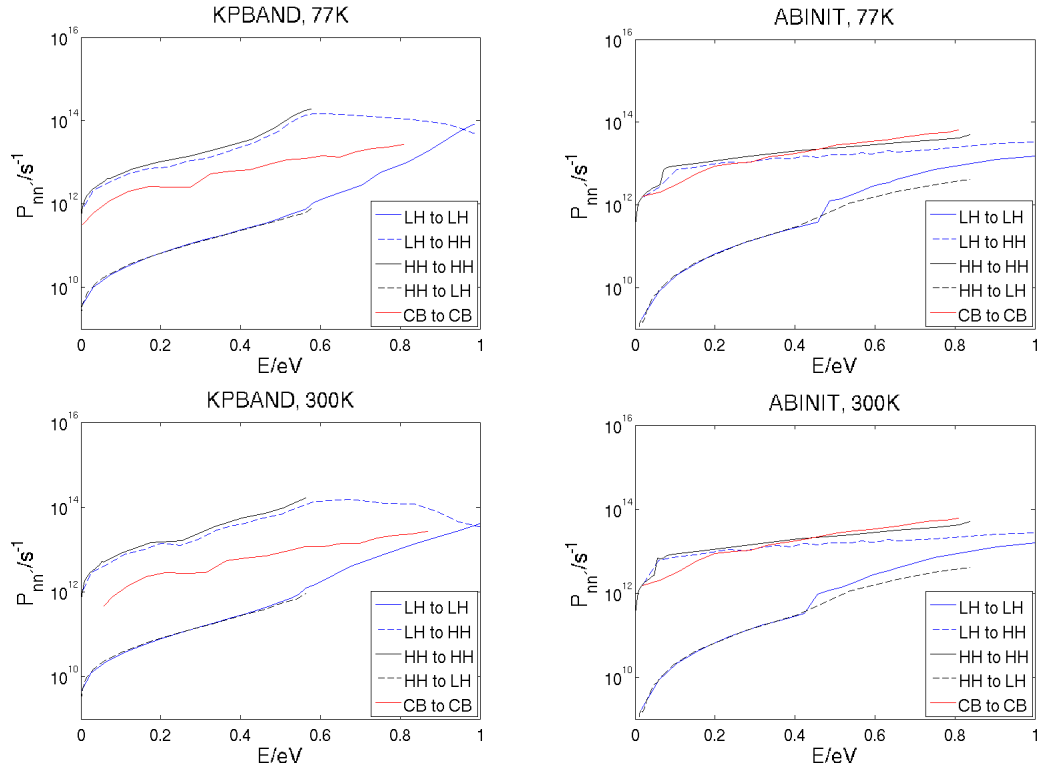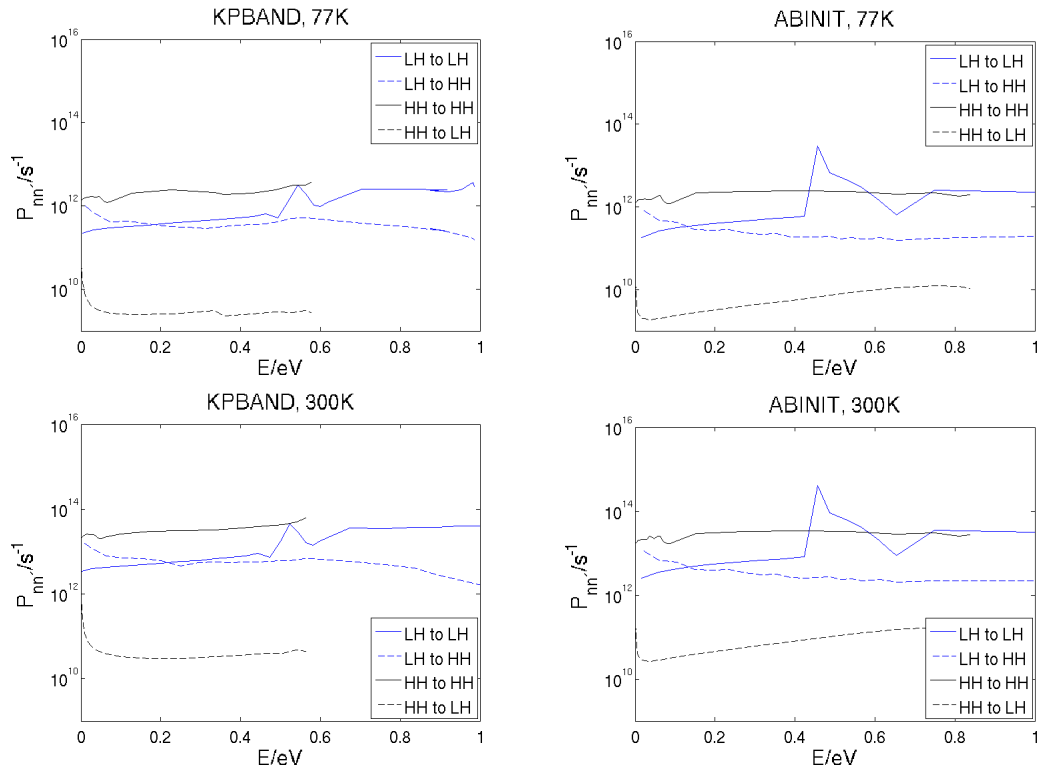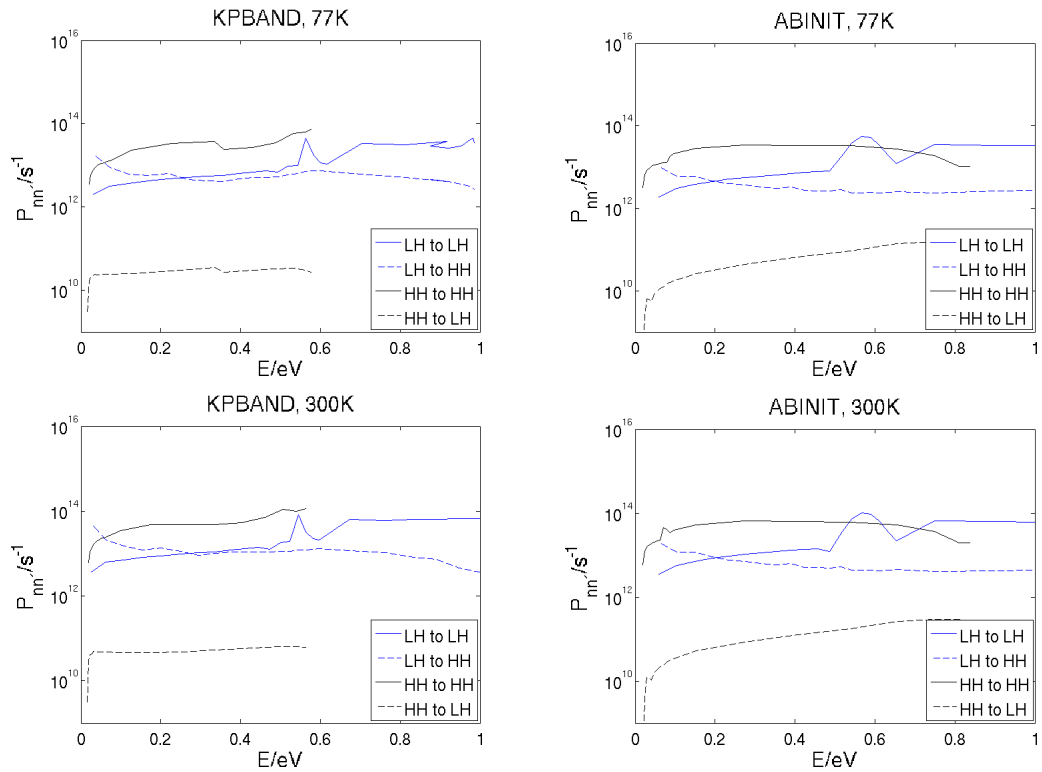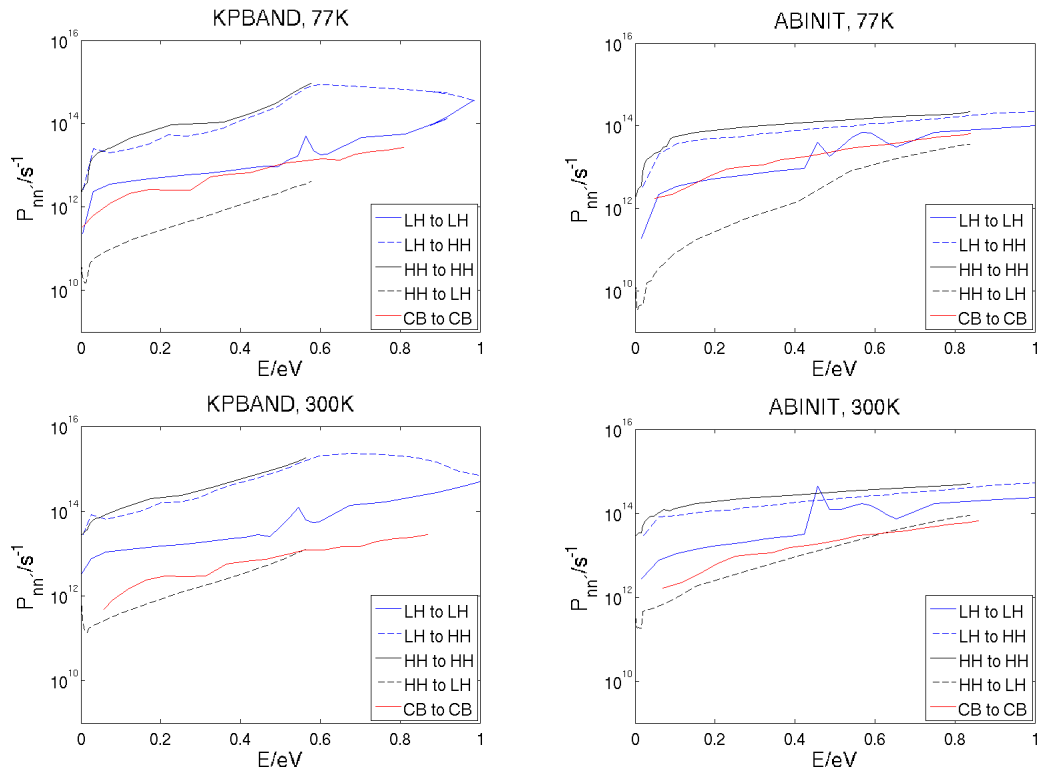

Figure 16: Total scattering rate plots in the (111)-direction. 77K on to and 300K on the bottom. Results form KPBAND on the left and from ABINIT on the right.
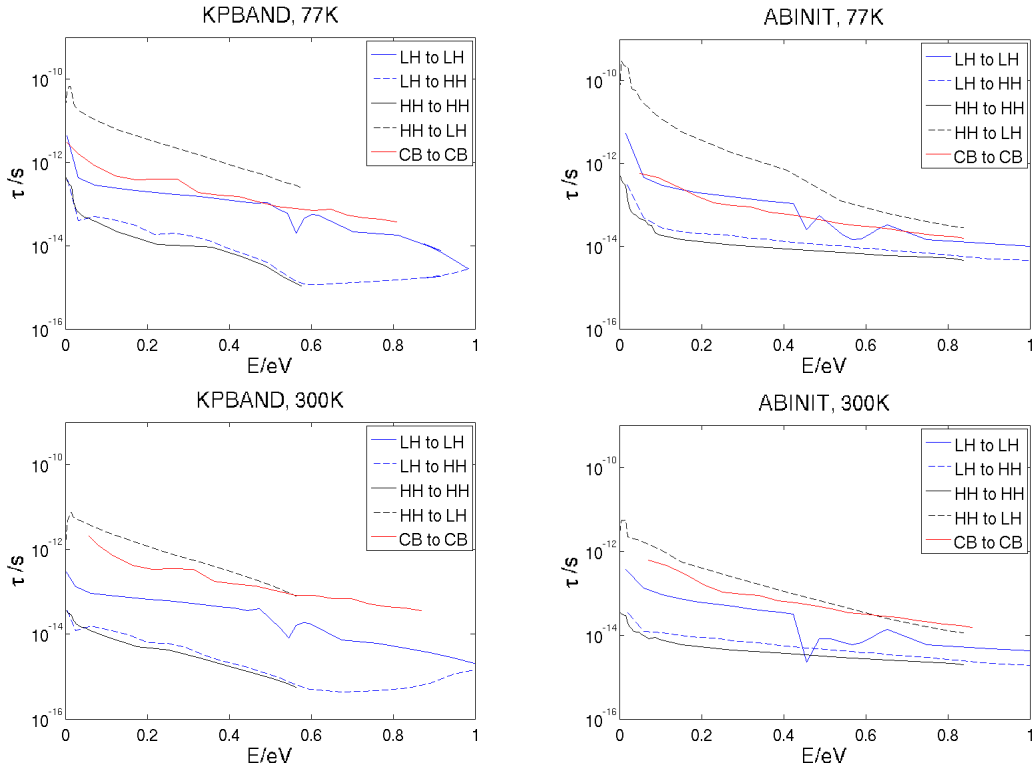
Figure 17: Free flight time plots in the (111)-direction. 77K on to and 300K on the bottom. Results form KPBAND on the left and from ABINIT on the right.

4.2.5. Total scattering rates and free flight time

For holes the polar and nonpolar optical phonons are the strongest scattering mechanisms of those that are included in this thesis. However, it may also be said that alloy scattering is quite strong as well.

For the scattering mechanisms for electrons included in this thesis, alloy and polar optical scattering are the most prominent. When comparing this rate with those resulting from acoustic phonons, the scattering due to the potential variations of the alloy is 100 times stronger.

The free flight time following eqs. (3) and (8) will become the inverse of the total scattering rate.

4.3. Final remarks regarding pseudopotential calculations

We have seen that the concept of pseudopotentials makes it easier to carry out a VCA calculation. Indeed, the highly skilled trickery of the pseudopotentials lends hope that introducing another abstract entity such as the VCA potential may be more appropriate in that environment. However, the pseudopotential method itself is rather less appealing than methods operating directly on the actual crystal potential. That being said, the pseudopotential method can also be formulated in empirical versions, whereas the direct methods are only formulated as self-consistent approaches.

The consequences of this are profound. For example, full-band MC codes are all based on empirical pseudopotentials. One reason for this is the use of MC codes on

technologically important semiconductor alloys, since an empirical pseudopotential formulation makes it easier to mix the pseudopotential into a VCA potential. Furthermore, the rigid pseudoion model for deformation potential scattering comes as a handy benefit once the pseudopotential is established.

Admittedly, some disadvantages appear when the concept of "rigid" is scrutinized. For the sake of accuracy, one would have wished to use self-consistent potentials in the lattice displacement, and not rigid potentials blindly following the atoms.

Today, phonon software packages that allow electrons to adapt self-consistently to the phonon induced position change of the atoms are indeed available. They are available as free-standing packages that can be used together with the $ab\ initio$ codes. But embracing such advances leads us immediately back to the formulation problem of the VCA. Solving the band structure problem with a self-consistent VCA approach seems to be the way to go. And indeed, there are electron structure codes that can make self-consistent VCA. For example, the $ab\ initio$ Augmented Plane Wave Code WIEN2K can deal with averaged atoms containing fractional electrons, but only to a certain extent. The averaged atoms must be similar, typically nearest horizontal neighbours in the periodic table. This is the common situation with doping atoms. But if, for example, we consider averaging Hg and Cd atoms in MCT, we immediately realize that Hg and Cd are nearest vertical neighbours in the periodic table. Such unfortunate circumstances dominate the technologically important alloys. These alloy atoms are chemically compatible because of similarities in the outer shells responsible for the binding, but the atoms are quite different in their inner shells. Capability of the ABINIT code to generate self-consistent alloy pseudopotentials for a VCA description of the MCT bands was tried out, but did not give satisfactory results. The problem was briefly put forward to the ABINIT discussion forum, where the preliminary conclusion was that ABINIT is not yet capable to generate alloy PAW pseudopotentials $ab\ initio$, and one would therefore have to resort to norm-conserving or empirical pseudopotentials.

As regards carrier-phonon scattering, one effect of the alloy that VCA misses is that two sets of phonons are generated, corresponding to the $HgTe$ and the $CdTe$ sublattice motion respectively. For instance, polar optical scattering will then appear with two different phonon frequencies [45].

In order to resolve the phonon eigenfrequencies of the $HgTe$ and $CdTe$ sublattices one could make large supercells with different plausible configurations of the Hg and Te atoms, all with the same alloy fraction x. Taking an average over the results from different supercells would give a representation of the phonons. In the scattering problem, the resulting existence of two sets of phonons could be straightforwardly accommodated [45].For the band electronic states, one could now choose between a VCA description and a supercell formulation. The use of a supercell results in a smaller Brillouin zone with more bands. In order to apply the MC program efficiently and also for the averaging procedure, all supercell band structures should be mapped onto the ordinary Brillouin zone used for VCA and ordinary zincblende structure semiconductors. This also goes for the phonon problem. As of yet, there is no automatic way of doing such a "back-folding" of Brillouin zones even in a well-established portfolio such as the WIEN2K code, but a back-folding program could be made, although it is a non-trivial task.

Doing the supercell approach for the electronic states certainly relieves us of the prob-

lem regarding combined VCA and *ab initio* formulation of alloys like MCT, but the size of the supercell creates another problem. The inherent band gap issue of semiconductors ideally requires the use of the GW procedure, but supercells and GW calculations constitute a poor match, due to the computationally intensive GW procedure.

However, simplified approaches can come to the rescue here. In particular, simple correcting procedures for the band states, like the modified Becke-Johnson approach [24],[25], are available in some codes and will work also for larger cells. One must be aware though, that the latter approach is not an energy functional in the sense of the DFT, so it should not be used for finding the equilibrium volume (lattice constant) of the material.

5. Conclusions

In the process of writing this thesis the following achievements have been made:

- The program KPBAND for calculating kp band structures has been expanded to be applicable over the whole Brillouin zone

- Band structures for the specific case of $Hg_{1-x}Cd_xTe$ have been calculated for temperatures 77K and 300K

- Similar band structures have been extracted from the $ab\ initio$ program ABINIT, which have been converted into KPBAND like format

- For both methods phonon and alloy scattering rates have been calculated using the revised program SCRATES

- Alloy scattering rates has been implemented into SCRATES, and the program has been updated to take the new files from KPBAND and ABINIT

- Several MATLAB programs for plotting band structures and scattering rates have been written

KPBAND was revised by expanding the $\vec{k} \cdot \hat{p}$-Hamiltonian matrix from a 6×6 matrix valid only for valence bands, to an 8×8 and a 14×14 matrix also including conduction bands. Generalizations of the algorithms were made for easy swapping between the different matrix sizes. Further cubic harmonic corrections were made in order to correct the bands for light hole, heavy hole and 1st conduction electron.

The $ab\ initio$ program ABINIT has been used to calculate MCT-like band structures. The calculations were made from $ab\ initio$ pseudopotentials using a projector augmented wavefunctions(PAW) basis. In order to obtain and make use of the calculations a few complementary programs were made to initialize, convert and differentiate the results. The resulting band structures (are shown in Figure 7 and) are in good agreement with the results obtained by Bertazzi among others. Also the band structure plots show that the results from KPBAND and ABINIT are very similar close to the Γ-point.

Unfortunately severe ripples in the calculation ionized impurity scattering arose. This owes to the rapid $\vec{q}$-dependent variation of the matrix element over the energy conserving surface intersecting a final cube. This particular algorithm thus proves too inaccurate, so an averaging of $|M(\vec{q})|^2$ and corrective post-calculation smoothing as described in section 3.3.3. should be made. Such smoothing procedures were as previously stated developed in Sivertsen Bergslids Master thesis, and could be directly applied here. This ripple problem also appeared for polar optical phonons as well, but here only the electron scattering was affected.

In future works the calculation of ionized impurity and polar optical phonon scattering rates should be improved in order to be fully reintroduced in SCRATES. Also the Auger and carrier-carrier scattering rates which were calculated by Falch using KPBAND should be calculated based on results from ABINIT. Another step forward would be to include the actual eigenstates from ABINIT when considering the overlap integrals.

The band gap problem causes no trouble for KPBAND, but for ABINIT and $ab\ initio$ methods in general this is a critial issue. The two methods used for solving this in ABINIT were the LDA+U and the Scissors operator. (The first is a pre-solution rigid interaction and the second is a post-solution operator.) Both methods are rather outdated and should be replaced by eg. Exact Exchange Correlated Electrons (EECE) and modified Becke-Johnson (mBJ) potentials, but these would have to be implemented into ABINIT.

$Ab\ initio$ methods are relatively much used for extracting data relating to mechanical, optical and spectroscopic properties, they are however barely touched upon in calculations on carrier transport. However, one early attempt of this has been made by Yoder et al. [46] to analyse Si. When analysing electronic properties of an alloy such as MCT a Virtual Crystal Approximation (VCA) is preferable, which to only a very limited degree is possible $ab\ initio$ in WIEN2K. The seemingly easiest method is to apply pseudopotentials. This is a relatively simple task when applying them empirically, but it is rather difficult $ab\ initio$. Although brave attempts were made to create a true $ab\ initio$ VCA in ABINIT, we had to settle for an MCT-like $HgTe$. Alternatively multiple supercells could be made with back-folding into an average MCT electronic structure. This is a method which as of today has not been implemented. However, if such an approach were to be taken, pseudopotentials would not automatically be the preferable $ab\ initio$ method to choose.

A. Derivatives of energy

In order to solve the semi-classical equations of motion, eqs. (1) and (2), and calculating the scattering rates, both the gradient and the second derivative of the energies need to calculated. Even though the only the gradient is relevant for this thesis, the second derivative is calculated as well by KPBAND, as it is meant to be a module of a larger MC-program used to solve these equations. In order to avoid discretization errors in the derivatives, analytical formulas are used instead.

The energies $E_n(\vec{k})$ are found by solving the energy eigenvalue problem for every point in the $\vec{k}$-mesh(es):

$$\mathcal{H}\mathcal{C}_n(\vec{k}) = E_n(\vec{k})\mathcal{C}_n(\vec{k}) \tag{135}$$

$\mathcal{H}$ is the complex hermitian Hamilton matrix derived for the 6×6, 8×8 or 14×14 $\vec{k} \cdot \hat{p}$-models. $\mathcal{C}_n(\vec{k})$ is the resulting complex coefficient eigenvector solution for the eigenvalue problem above. The normalization of the eigenvector is unity. All energies from the eigenvalue problem are real.

Notations and further assumptions in this section are the following:

- α and β represent components (x, y, z)

- $\mathcal{H}_\alpha = (\partial \mathcal{H})/(\partial k_\alpha)$ and $\mathcal{H}_{\alpha\beta} = (\partial^2 \mathcal{H})/(\partial k_\alpha \partial k_\beta)$

- The derivative of an eigenvector is a linear combinations of all of them: $(\partial \mathcal{C}_n)/(\partial k_\alpha) = \sum_i d_{ni}^\alpha \mathcal{C}_i$

- From the orthonormality of the eigenvectors and previous line provide $d_{ni}^\alpha = -d_{in}^{\alpha *}$, $\Re\{d_{nn}^\alpha\}$

A.1. Energy gradient

A.1.1. Energy gradient using KPBAND

Starting from the energy eigenvalue problem for band n

$$\mathcal{H}\mathcal{C}_n = E_n \mathcal{C}_n \tag{136}$$

the components α of the energy gradient are derived by the derivative of the equation by the corresponding wave vector component k_α. This produces the following expression:

$$\mathcal{H}_\alpha \mathcal{C}_n + \mathcal{H}\frac{\partial \mathcal{C}_n}{\partial k_\alpha} = \frac{\partial E_n}{\partial k_\alpha}\mathcal{C}_n + \mathcal{E}_n \frac{\partial \mathcal{C}_n}{\partial k_\alpha} \tag{137}$$

Due to the assumption that the derivatives of the eigenvectors are linear combinations of all of them, and that they are all orthonormal, the second term on each sides of the equation above cancels each other out when the equation is normalized by the transposed complex conjugate of the eigenvector. The following formula for the components of the gradient is the result of this operation:

$$\frac{\partial E_n}{\partial k_\alpha} = \mathcal{C}_n^\dagger \mathcal{H}_\alpha \mathcal{C}_n \tag{138}$$

A.1.2. Energy gradient using ABINIT

As mentioned previously in this chapter, the gradient (and the second derivatives) are derived by discrete differentiation. This is done by replacing the differentials by defined steps in reciprocal space Δk_α both forwards and backwards in a direction where all steps are of one step in the k-meshes:

$$\frac{\partial E_n}{\partial k_\alpha} \rightarrow \frac{\Delta E_n}{\Delta k_\alpha} = \frac{E(\vec{k}+k_\alpha) - E(\vec{k}-k_\alpha)}{2k_\alpha} \tag{139}$$

However, if this approach leads to $E(\vec{k}+k_\alpha)$ or $E(\vec{k}-k_\alpha)$ the either of the meshes then a non-existent mesh point is required. The solution is to replace this derivative with that of a nearby lying point, so $\vec{k} \rightarrow \vec{k} \pm \Delta k_\alpha$. As an example, the gradient for a point on the positive x-border, away from both y- and z-borders:

$$\nabla_{\vec{k}} E(\vec{k}) \rightarrow \frac{\Delta E_n}{\Delta k_\alpha} = \frac{E((\vec{k}-\Delta k_x)+\Delta k_x) - E((\vec{k}-\Delta k_x)-\Delta k_x)}{2\Delta k_x}\vec{e_x} + \frac{E(\vec{k}+\Delta k_y)-E(\vec{k}-\Delta k_y)}{2\Delta k_y}\vec{e_y} + \frac{E(\vec{k}+\Delta k_z)-E(\vec{k}-\Delta k_z)}{2\Delta k_z}\vec{e_z} \tag{140}$$

In other words, only the x-component of the gradient takes this shift in the k-mesh.

A.2. Energy second derivatives of energy

A.2.1. Energy second derivatives using KPBAND

Components of the second derivatives also differentiate on the vector component k_β. Firstly eq. (136) is differentiated by k_β, giving the following expression:

$$\mathcal{H}_{\alpha\beta}\mathcal{C}_n + \mathcal{H}_\alpha\frac{\partial \mathcal{C}_n}{\partial k_\beta} + \mathcal{H}_\beta\frac{\partial \mathcal{C}_n}{\partial k_\alpha} + \mathcal{H}\frac{\partial^2 \mathcal{C}_n}{\partial k_\alpha \partial k_\beta}$$
$$= \frac{\partial^2 E_n}{\partial k_\alpha \partial k_\beta}\mathcal{C}_n + \frac{\partial E_n}{\partial k_\alpha}\frac{\partial \mathcal{C}_n}{\partial k_\beta} + \frac{\partial E_n}{\partial k_\beta}\frac{\partial \mathcal{C}_n}{\partial k_\alpha} + E_n\frac{\partial^2 \mathcal{C}_n}{\partial k_\alpha \partial k_\beta} \tag{141}$$

Normalizing this expression with $\mathcal{F}_n^\dagger$, and using the the assumptions at the start of this section, the following expression is derived:

$$\frac{\partial^2 E_n}{\partial k_\alpha \partial k_\beta} = \mathcal{C}_n^\dagger \mathcal{H}_{\alpha\beta}\mathcal{C}_n + \sum_{i \neq n} d_{ni}^\beta \mathcal{C}_n^\dagger \mathcal{H}_\alpha \mathcal{C}_i + \sum_{i \neq n} d_{ni}^\alpha \mathcal{C}_n^\dagger \mathcal{H}_\beta \mathcal{C}_i \tag{142}$$

The coefficients d_{ni}^α and d_{ni}^β are determined by using the linear combination approach on eq. (136) and applying the Hamiltonian matrices on the eigenvectors.

$$\mathcal{H}_\alpha \mathcal{C}_n + \sum_i d_{ni}^\alpha E_i \mathcal{C}_i = \frac{\partial E_n}{\partial k_\alpha}\mathcal{C}_n + E_n \sum_i d_{ni}^\alpha \mathcal{C}_i \tag{143}$$

Next this expression is normalizing by an eigenvector $\mathcal{C}_j^\dagger$ and then rearranged.

$$d_{ni}^\alpha = \frac{\mathcal{C}_i^\dagger \mathcal{H}_\alpha \mathcal{C}_n}{E_n - E_i}, i \neq n \tag{144}$$

Using the property $d_{ni}^\alpha = -d_{in}^{\alpha\,*}$ the following formula for the second derivative results:

$$\frac{\partial^2 E_n}{\partial k_\alpha \partial k_\beta} = \mathcal{F}_n^\dagger \mathcal{H}_{\alpha\beta} \mathcal{F}_n + 2 \sum_{\substack{i \\ E_i \neq E_n}} \frac{\Re\{(\mathcal{C}_i^\dagger \mathcal{H}_\alpha \mathcal{C}_n)(\mathcal{C}_n^\dagger \mathcal{H}_\alpha \mathcal{C}_i)\}}{E_n - E_i} \tag{145}$$

A.2.2. Energy second derivatives using ABINIT

The ABINIT second derivatives are derived in the same manner as the different components of the gradients. As previously in this section, the derivatives are taken with respect to the k-vector components k_α and k_β, where both $\alpha = \beta$ and $\alpha \neq \beta$ are possible.

$$\frac{\partial^2 E_n}{\partial k_\beta \partial k_\alpha} \rightarrow \frac{\partial}{\partial k_\beta} \left(\frac{E(\vec{k}+\Delta k_\alpha)-E(\vec{k}-\Delta k_\alpha)}{2\Delta k_\alpha} \right)$$
$$\rightarrow \frac{E(\vec{k}+\Delta k_\alpha+\Delta k_\beta)-E(\vec{k}+\Delta k_\alpha-\Delta k_\beta)-E(\vec{k}-\Delta k_\alpha+\Delta k_\beta)+E(\vec{k}-\Delta k_\alpha-\Delta k_\beta)}{4\Delta k_\alpha \Delta k_\beta} \tag{146}$$

In the cases where $\alpha = \beta$ a simpler expression results:

$$\frac{\partial^2 E_n}{\partial k_\alpha^2} \rightarrow \frac{E(\vec{k}+2\Delta k_\alpha)-2E(\vec{k})+E(\vec{k}-2\Delta k_\alpha)}{4\Delta k_\alpha^2} \tag{147}$$

In these cases double steps in the k-mesh is to be done. As this is not necessary, and also increases the deviation associated with the discretion, the double steps are replaced with single steps. In other words $2\Delta k_\alpha \rightarrow \Delta k_\alpha$. Then the final formulas for the second derivatives are derived:

$$\frac{\partial^2 E_n}{\partial k_\alpha^2} \rightarrow \begin{cases} \frac{E(\vec{k}+\Delta k_\alpha)-2E(\vec{k})+E(\vec{k}-\Delta k_\alpha)}{\Delta k_\alpha^2}, & \text{when } \alpha = \beta \\ \frac{E(\vec{k}+\Delta k_\alpha+\Delta k_\beta)-E(\vec{k}+\Delta k_\alpha-\Delta k_\beta)-E(\vec{k}-\Delta k_\alpha+\Delta k_\beta)+E(\vec{k}-\Delta k_\alpha-\Delta k_\beta)}{4\Delta k_\alpha \Delta k_\beta}, & \text{whenever else} \end{cases} \tag{148}$$

B. Physical constants and material parameters

In this part of the appendix physical constants and material parameters are listed. They are listed with symbols, values, variable names in the programs and the explanations for the material parameters.

B.1. Physical constant

The following commonly known physical constants are used:

Physical constants		
Symbol	Value	Variable
$\hbar$	$1.05459 \cdot 10^{-34} Js$	HBAR
ϵ_0	$8.8542 \cdot 10^{-12} C(Nm^2)^{-1}$	EPS0
e	$1.60219 \cdot 10^{-19} C$	EC
k_B	$1.3807 \cdot 10^{-23} JK^{-1}$	KB

B.2. Material parameters used in KPBAND

As mentioned earlier in the thesis, the $\vec{k} \cdot \hat{p}$-method related parameters are picked up from a few different writings. Some of them have been modified (reference marked by a star) by a fitting process. These writings are made by Chang et al. [2], Mroczkowski et al. [47], Bertazzi et al. [7], Capper [48], Schacham and Finkman [33], and Mayer and Rössler [49].

Material parameters (for the $\vec{k} \cdot \hat{p}$-method) for $Hg_{0.72}Cd_{0.28}Te$			
Symbol	Value	Variable	Explanation
a[48]	6.4844Å	a0	Lattice constant
$m*$[7]	-	-	Effective masses used for the Luttinger parameters
$\gamma_1^{14}(77K)$	0.623	A	1st Luttinger parameter for 14×14 (value of variable)
$\gamma_2^{14}(77K)$	-0.504	B	2nd Luttinger parameter for 14×14 (value of variable)
$\gamma_3^{14}(77K)$	0.336	D	3rd Luttinger parameter for 14×14 (value of variable)
$\gamma_1^{14}(300K)$	0.980	A	1st Luttinger parameter for 14×14 (value of variable)
$\gamma_2^{14}(300K)$	-0.650	B	2nd Luttinger parameter for 14×14 (value of variable)
$\gamma_3^{14}(300K)$	0.700	D	3rd Luttinger parameter for 14×14 (value of variable)
P_0 [47]	$8.0 eV$Å	P0	Valence to 1st conduction band momentum
P_1 [49]	$i6.5 eV$Å	P1	2nd conduction to 1st conduction band momentum
Q [49]	$7.7 eV$Å	Q	Valence to 2nd conduction band momentum
$E_0(77K)$[33]	$0.211 eV$	E0	Minimum energy of 1st conduction band
$E_0(300K)$[33]	$0.265 eV$	E0	Minimum energy of 1st conduction band
E_1[7]	$3.77 eV$	E1	Minimum energy of 2nd conduction band
Δ_0[7]	$0.95 eV$	DELTA0	Valence band spin orbit splitting
Δ_1[49]	$0.25 eV$	DELTA1	2nd conduction band spin orbit splitting
Δ' [49]	$0.0 eV$	DELTAQ	Valence to 2nd conduction band spin orbit splitting

B.3. Material parameters used in ABINIT

The ABINIT related parameters are mainly obtained from experience with using the program itself, so that the results are in agreement with the knowledge of the material.

Material parameters (for the *ab initio* pseudopotential method) for $Hg_{0.72}Cd_{0.28}Te$			
Symbol	Value	Variable	Explanation
$a[48]$	12.2575	acell	Lattice constant (in bohr)
	12.2575		
	12.2575		
$\vec{a}$	0.0 0.5 0.5	rprim	Primitive lattice structure(of a FCC/zincblende)
	0.5 0.0 0.5		
	0.5 0.5 0.0		
—	2	ntypat	Number of atom types involved
—	2	natom	Number of atoms in a primitive unit cell
—	1	typat	Atom type placement
	2		
—	0.00 0.00 0.00	xred	Atom placement in the primitive
	0.25 0.25 0.25		unit cell of a zincblende
Z	80.	znucl	Nuclei charges of the involved atoms
	52.		
—	64	nband	Bands to be calculated
—	2	nspinor	Spinors considered
E_{cut}	$10.Ha$	ecut	Energy cutoff for basis wavevectors
$E_{CutSmear}$	$0.5Ha$	ecutsm	Energy cutoff smearing
—	$50.Ha$	pawecutdg	PAW double grid energy cutoff
$NGrid1$	12	ngkkpt1	Special points grid
	12		
	12		
$E_{lim,1}$	$10^{-6}Ha$	toldfe1	Special points energy convergence criterion
$NGrid2$	40	-	Grid for further calculations
	40		(only a $1/48^{th}$ *is wedge*
	40		
			(only a $1/48^{th}$ *is wedge extracted*)
$E_{lim,2}$	$10^{-12}Ha$	toldfe2	Special points energy convergence criterion
—	1	usepawu	Statement for including LDA+U
—	2	lpawu	Statement for including U on Hg's d orbitals
	-1		and not on Te
U	8. 0. eV	upawu	Strength of the U potential in eV
	0.		
	eV		

B.4. Material parameters used in SCRATES

Also the scattering parameters have been picked up from different writings. Mostly from Capper [48], but also one from Lund [45], one from Boris et al. [50], and one from Derelle et al. [14].

Material parameters (for the scattering rates) for $Hg_{0.72}Cd_{0.28}Te$			
Symbol	Value	Variable	Explanation
ϵ [48]	11.47	EPSINF	High frequence relative dielectric constant
ϵ_s [48]	16.58	EPSZER	Static relative dielectric constant
$\hbar\omega_0$ [50]	$0.086eV$	OPTEN	Optical phonon energy
E_1 [45]/[51]	$7.00eV/5.60eV$	D	Acoustic deformation potential (for $GaAs$) (electrons/holes)
v_s [48]	$4570ms^{-1}$	S	Average sound velocity
ρ [48]	$7.406gcm^{-3}$	RHO	Mass density
(DK^2)	$3.16 \cdot 10^3 (eV)^2\text{Å}^{-2}$	DKSQR	Squared optical phonon coupling constant
N_i	-	NIMP	Ionized impurity concentration (not used)
DV [14]	1.5eV/0.3eV	DV	Alloy scattering potential (electrons/holes)
x	0.28	COMP	Cd to Te ratio
N_c	$1.47 \cdot 10^{28}m^{-3}$	NCAT	Cation density

C. Formulas for squared matrix elements

Here all of the squared matrix elements are represented. All of the scattering mechanisms should be calculated using the eigenvectors as overlap factors, except for the non-polar optical phonon scattering, where unity overlapping factors should be used. The phonon scatterings have a curly bracket where the upper part represents absorption and the lower part represents emission. These formulas are taken from Brudevoll et al. [13], except that of the alloy scattering which is taken from Ridley et al. [15].

- Polar optical phonon scattering:

$$|M(q)|^2 = \left\{ \begin{array}{c} (\vec{q}) \\ N(\vec{q}) + 1 \end{array} \right\} \frac{e^2 \hbar \omega_0}{2\epsilon_0} \left(\frac{1}{\epsilon - \frac{1}{\epsilon_s}} \right) \frac{1}{q^2} \tag{149}$$

- Non-polar optical phonon scattering:

$$|M(q)|^2 = \left\{ \begin{array}{c} (\vec{q}) \\ N(\vec{q}) + 1 \end{array} \right\} \frac{(DK)^2 \hbar}{4\rho\omega_0} \tag{150}$$

- Acoustic deformation potential phonon scattering:

$$|M(q)|^2 = \left\{ \begin{array}{c} (\vec{q}) \\ N(\vec{q}) + 1 \end{array} \right\} \frac{E_1^2 \hbar}{2\rho v_s} q \tag{151}$$

- Ionized impurity scattering:

$$|M(q)|^2 = \frac{N_i e^4}{\epsilon_0^2 \epsilon_s^2} \frac{1}{(q^2 + \beta^2)^2}, \text{ where } \beta = \left(\frac{N_i e^2}{k_B T \epsilon_0 \epsilon_s} \right)^{1/2} \tag{152}$$

- Alloy scattering:

$$|M(q)|^2 = (DV)^2 \frac{x(1-x)}{N_c} \tag{153}$$

$N(\vec{q})$ is the phonon occupation number, whos formula is as follows:

$$N(\vec{q}) = \frac{1}{exp(\frac{\hbar\omega(\vec{q})}{k_B T}) - 1} \tag{154}$$

For acoustic phonons $\omega(\vec{q})$ is approximated to $\omega(\vec{q}) = v_s q$, and for optical phonons $\omega(\vec{q}) = \omega_0$ is used.

D. Program user guides

D.1. User guide for KPBAND

KPBAND is a program for calculating the energy band structure of a zincblende semi-conductor around the Γ-point. The program does also handle semiconductors under homogeneous strain, however this is not done in this thesis. The solutions from this program is based on the $\vec{k} \cdot \hat{p}$-method energy eigenvalue problem made up by the 6×6, 8×8 or 14×14 $\vec{k} \cdot \hat{p}$-Hamiltonian-matrix. However it is worth noting that the 6×6 Hamiltonian is not sufficient for narrow band gap materials such as $Hg_{0.72}Cd_{0.28}Te$, as it is for $GaAs$. The program is able to perform the following calculations(whichever the user desires to calculate):

- Energies

- Energy gradients

- Second derivatives of the energies

- Eigenvectors

The program calculates values on a cubic mesh in $\vec{k}$-space, with the option of doing calculations also at a finer mesh near the Γ-point. In order to make the program more user friendly it has been made with a module halvorinit.f90 where all model modifications are made and all material parameters are set. The calculation settings are set in KPINI.f90, stored in an initialization file KPBAND.INI and used by KP-BAND2mkl.f90. The following sections describe how to use the complete KPBAND program.

D.1.1. Modifying halvorinit.f90

In the module halvorinit.f90 model size is chosen and material parameters are changed. The model size is chosen by setting manually these values in lines (58-60) of the module: MS, MSH and MSD. A model of size $N \times N$, where $N = \{6, 8, 14\}$ is chosen, then one sets:

$$N: \ MS = N, \ MSH = N/2, \ MSD = 2N \tag{155}$$

Both the initialization program KPINI.f90 and the KPBAND2mkl.f90 use this module, and they will both terminate if the parameters MS, MSH and MSD are not chosen consistently. The material parameter values in Appendix B are also set here. It must be noted that the the Luttinger parameters for a specific material differ based on which model size is used. It is not necessary to remove the eg. the 14×14-model parameters when using eg. the 6×6 model as these only becomes dummy parameters, as long as the parameters MS, MSH and MSD are chosen accordingly.

In previous works the material parameters were hard coded into KPBAND2mkl.f90 (or KPBAND2mkl.f which it was called then), but due to the need for more frequent changes in the parameters of the material (resulting from changes in composition and

temperature) the setting of them was moved to this module. For future works on the KPBAND program the setting of these material input should be set using the KPINI.f90 program. These settings could be stored in .INI files, eg. MCT-28-77K.INI for $Hg_{0.72}Cd_{0.28}Te$ at 77K. These files would then have to be chosen by KPINI.f90 and loaded by KP-BAND2mkl.f90.

D.1.2. Using KPINI.f90

In this program the user makes the following choices for the calculations:

- Setting strain tensor

- Construction of the $\vec{k}$-mesh

- Setting which bands to calculate

- Setting which calculations to save

- Setting names for the output files

By running this program the initialization file KPBAND.INI is produced/created, and this file has to exist in order to run KPBAND2mkl.f90. If KPBAND.INI exists prior to a running of KPINI.f90, the old values in KPBAND.INI are loaded during the run, so one does not necessarily have to make all choices again even if the model size has been changed. If KPBAND.INI does not exist prior to the run all choices have to be made. The user interface is a 'no nonsense' menu system. The user enters and exits the program through the main menu. The following is the main menu of the program:

```
**** KPINI MAIN MENU *****
1.  Set strain tensor.
2.  Describe mesh(es).
3.  Set which bands to calculate.
4.  Set what to calculate.
5.  Define filenames.
6.  Show all data.
7.  Save and quit

ENTER SELECTION :
```

The user selects an option in the menu by typing a number 1-7, where 1-5 are sub menus, 6 displays all selected options, and 7 saves and quits the program. After options in the sub menus are chosen, the user is returned to the main menu.

Selection 1: SET STRAIN TENSOR
In this sub menu the user specify the homogeneous strain tensor. The strain is a second order symmetric tensor, so only six of the elements need to be specified. Below is shown

the tensor elements set from all zero to again all zero. Again only the 6×6-model may use non-zero strain tensor.

```
**** SET STRAIN TENSOR ****

Strain:  .0000000 .0000000 .0000000
         .0000000 .0000000 .0000000
         .0000000 .0000000 .0000000

SPECIFY NEW STRAIN (EXX,EXY,EXZ,EYY,EYZ,EZZ):
<0,0,0,0,0,0><Enter>
```

The new elements are now stored.

Selection 2: DESCRIBE MESH(ES)

In this sub menu the user sets the values in $\vec{k}$-space for which KPBAND2mkl.f90 is to do the calculations. The user may choose the number of cubes in each dimension and the fraction of 1BZ $(2\pi/a_0)$ is to be considered. Further it may be chosen to add a finer mesh around the Γ-point. In this case the user then makes the same choices for both meshes. The following is the use of this sub menu starting with only producing the coarse mesh:

```
**** DESCRIBE MESH(ES) ****
Coarse mesh:
  1.  Number of meshpoints :    30
  2.  Fraction of 2PI/a0    :  0.250

Fine mesh:
  3.  No fine mesh is generated.

ENTER SELECTION (0=EXIT)
<3><Enter>



**** DESCRIBE MESH(ES) ****
Coarse mesh:
  1.  Number of meshpoints :    30
  2.  Fraction of 2PI/a0    :  0.250

Fine mesh:
  3.  Fine mesh is generated.
```

```
   4.   Number of meshpoints :    36
   5.   Fraction of 2PI/a0    :  0.060


ENTER SELECTION (0=EXIT)
<1><Enter>



  Enter number of meshpoints :
<40><Enter>




**** DESCRIBE MESH(ES) ****
Coarse mesh:
   1.   Number of meshpoints :    40
   2.   Fraction of 2PI/a0    :  0.250

Fine mesh:
   3.   Fine mesh is generated.
   4.   Number of meshpoints :    36
   5.   Fraction of 2PI/a0    :  0.060

ENTER SELECTION (0=EXIT)
<2><Enter>

Enter fraction :
<1.00><Enter>

**** DESCRIBE MESH(ES) ****
Coarse mesh:
   1.   Number of meshpoints :    40
   2.   Fraction of 2PI/a0    :  1.000

Fine mesh:
   3.   Fine mesh is generated.
   4.   Number of meshpoints :    36
   5.   Fraction of 2PI/a0    :  0.060

ENTER SELECTION (0=EXIT)
<4><Enter>

Enter number of fine meshpoints :
<40><Enter>

**** DESCRIBE MESH(ES) ****
Coarse mesh:
```

```
  1.   Number of meshpoints :    40
  2.   Fraction of 2PI/a0    :  1.000

Fine mesh:
  3.   Fine mesh is generated.
  4.   Number of meshpoints :    40
  5.   Fraction of 2PI/a0    :  0.060

ENTER SELECTION (0=EXIT)
<5><Enter>

Enter fraction :
<0.10><Enter>


**** DESCRIBE MESH(ES) ****
Coarse mesh:
  1.   Number of meshpoints :    40
  2.   Fraction of 2PI/a0    :  1.000

Fine mesh:
  3.   Fine mesh is generated.
  4.   Number of meshpoints :    40
  5.   Fraction of 2PI/a0    :  0.100

ENTER SELECTION (0=EXIT)
<0><Enter>
```

If '3' were selected once more, the fine mesh would be removed. As '0' now has been selected, the selections are stored and the user has been returned to the main menu.

Selection 3: SET WHICH BANDS TO CALCULATE
Here it is to be selected for which bands the calculations are to be stored. When running this sub menu the answerer to this question has to be given for all of the bands. When the choices have been made the user is returned to the main menu. This sub menu vary based on which model size has been chosen. The following is the use of it based on the choice of the 14×14-model. If one can handle this, one can handle the others:

```
**** SET WHICH BANDS TO CALCULATE ****


+                                           6
+                                          4
+                                          3
+                                          1
currently results are stored for bands :
```

```
Should results for band 1 be stored ?  (Y/N)
<Y><Enter>
Should results for band 2 be stored ?  (Y/N)
<Y><Enter>
Should results for band 3 be stored ?  (Y/N)
<Y><Enter>
Should results for band 4 be stored ?  (Y/N)
<Y><Enter>
Should results for band 5 be stored ?  (Y/N)
<N><Enter>
Should results for band 6 be stored ?  (Y/N)
<N><Enter>
Should results for band 7 be stored ?  (Y/N)
<N><Enter>
```

Selection 4: SET WHAT TO CALCULATE

In this sub menu the user chooses which values to calculate and store. After the choices are made the user is returned to the main menu:

```
**** SET WHICH BANDS TO CALCULATE ****


+                                                       ENERGIES
+                         EIGENVECTORS
+Current settings :

Should energies be calculated ?  (Y/N)
<Y><Enter>
Should gradients be calculated ?   (Y/N)
<Y><Enter>
Should second derivatives be calculated ?  (Y/N)
<N><Enter>
Should eigenvectors be calculated ?  (Y/N)
<Y><Enter>
```

Selection 5: DEFINE FILENAMES

The user may choose the names for the output files. First the user has to select the band for which file names are to be redefined. The procedure for all bands is the same, so only the changing for one of the bands are shown.

```
**** DEFINE FILENAMES ****
Filenames
         Energies   Gradients   Sec.deriv.   Eigenvectors
1.  Band:  SOENE.TAB   SOGRA.TAB   SODER.TAB      SOEIG.TAB
2.  Band:  LHENE.TAB   LHGRA.TAB   LHDER.TAB      LHEIG.TAB
```

```
3.  Band:   HHENE.TAB   HHGRA.TAB   HHDER.TAB      HHEIG.TAB
4.  Band:   CBENE.TAB   CBGRA.TAB   CBDER.TAB      CBEIG.TAB
5.  Band:   CSENE.TAB   CSGRA.TAB   CSDER.TAB      CSEIG.TAB
6.  Band:   CLENE.TAB   CLGRA.TAB   CLDER.TAB      CLEIG.TAB
7.  Band:   CHENE.TAB   CHGRA.TAB   CHDER.TAB      CHEIG.TAB


WHICH DO TOU WANT TO CHANGE (0=EXIT)
<3><Enter>


New filenames for band 3
Enter name of energy file [HHENE.TAB] :
HENE.TAB
Enter name of gradient file [HHGRA.TAB] :
HGRA.TAB
Enter name of sec.  derivative file [HHDER.TAB] :
HDER.TAB
Enter name of eigenvector file [HHEIG.TAB] :
HEIG.TAB


**** DEFINE FILENAMES ****
Filenames
            Energies    Gradients   Sec.deriv.   Eigenvectors
1.  Band:   SOENE.TAB   SOGRA.TAB   SODER.TAB    SOEIG.TAB
2.  Band:   LHENE.TAB   LHGRA.TAB   LHDER.TAB    LHEIG.TAB
3.  Band:   HENE.TAB    HGRA.TAB    HDER.TAB     HEIG.TAB
4.  Band:   CBENE.TAB   CBGRA.TAB   CBDER.TAB    CBEIG.TAB
5.  Band:   CSENE.TAB   CSGRA.TAB   CSDER.TAB    CSEIG.TAB
6.  Band:   CLENE.TAB   CLGRA.TAB   CLDER.TAB    CLEIG.TAB
7.  Band:   CHENE.TAB   CHGRA.TAB   CHDER.TAB    CHEIG.TAB


WHICH DO TOU WANT TO CHANGE (0=EXIT)
<0><Enter>
```

Selection 6: SHOW ALL DATA

```
**** SHOW ALL DATA ****

Strain:  .0000000 .0000000 .0000000
         .0000000 .0000000 .0000000
         .0000000 .0000000 .0000000


Coarse mesh:
  1.   Number of meshpoints :     40
  2.   Fraction of 2PI/a0    :  1.000
```

```
Fine mesh:
  3.  Fine mesh is generated.
  4.  Number of meshpoints :     40
  5.  Fraction of 2PI/a0    :   0.100


+                                           4
+                                           3
+                                           2
+                                           1
+Results are stored for bands :

+                                                     ENERGIES
+                                         GRADIENS
+                           EIGENVECTORS
+The program calculates :

Filenames
            Energies    Gradients   Sec.deriv.    Eigenvectors
1.  Band:   SOENE.TAB   SOGRA.TAB   SODER.TAB     SOEIG.TAB
2.  Band:   LHENE.TAB   LHGRA.TAB   LHDER.TAB     LHEIG.TAB
3.  Band:   HENE.TAB    HGRA.TAB    HDER.TAB      HEIG.TAB
4.  Band:   CBENE.TAB   CBGRA.TAB   CBDER.TAB     CBEIG.TAB
5.  Band:   CSENE.TAB   CSGRA.TAB   CSDER.TAB     CSEIG.TAB
6.  Band:   CLENE.TAB   CLGRA.TAB   CLDER.TAB     CLEIG.TAB
7.  Band:   CHENE.TAB   CHGRA.TAB   CHDER.TAB     CHEIG.TAB


PRESS ENTER TO RETURN...
```

   Selection 7: SAVE AND QUIT

The settings are not written to 'KPBAND.INI' before the user chooses to exit.

```
**** SAVE AND QUIT ****



Data written to file 'KPBAND.INI'
```

D.1.3. Running KPBAND2mkl.f90

When the program 'KPINI.f90' has been run, and the initialization file 'KPBAND.INI' has been produced, then the program 'KPBAND2mkl.f90' may be run. The program only requires that the initialization file exists, and that the module 'halvorinit.f90' contains proper model size parameters (MS, MSH, MSD). Previously running this program

could require too much space on the computer, but now this is rarely a problem. The running commando for this program is very extensive, but it is found in the introduction in 'KPBAND2mkl.f90'. The terminal display the following information during the run, given that the previously chosen setting are used:

```
*************** KPBAND ***************
Program for calculating band structure and
related quantities for MCT

Strain:   .0000000 .0000000 .0000000
          .0000000 .0000000 .0000000
          .0000000 .0000000 .0000000

Coarse mesh:
  Number of meshpoints :    40
  Fraction of 2PI/a0    :  1.000

Fine mesh:
  Fine mesh is generated.
  Number of meshpoints :    40
  Fraction of 2PI/a0    :  0.100

Results are stored for bands :
                  1
                  2
                  3
                  4


The program calculates :
ENERGIES
GRADIENTS
EIGENVECTORS
Output files are:
Output files for band 1 are:
SOENE.TAB
SOGRA.TAB
SOEIG.TAB
Output files for band 2 are:
LHENE.TAB
LHGRA.TAB
LHEIG.TAB
Output files for band 3 are:
HENE.TAB
HGRA.TAB
HEIG.TAB
Output files for band 4 are:
```

```
CBENE.TAB
CBGRA.TAB
CBEIG.TAB

COARSE MESH:
COMPUTING ENERGY LIMIT...
COMPLETED.
COMPUTING MESH...
COMPLETED.
FINE MESH:
COMPUTING ENERGY LIMIT...
COMPLETED.
COMPUTING MESH...
COMPLETED.
```

D.1.4. File formats and use of readenergy.f90

In the construction of the output files, their structures are so that the least possible information is stored. Instead the programs using these files are expected to know their structures. Because the $\vec{k}$-mesh is uniform, only the mesh-descriptions and the calculated values for each point in the coarse are stored. The mesh descriptions are stored in the first 10 locations, and then the values in the coarse mesh are stored, making a total of N locations in the output file. If a fine mesh was chosen to be calculated, 10 new locations after the values in the coarse mesh will the fine mesh descriptions be stored, followed by the values in the fine mesh. The following mesh information are stored:

1/N+1: Number of mesh-points in each dimension.

2/N+2: Maximum wave vector along an axis

3/N+3: Value at zero wave vector

4/N+4: Logical parameter telling whether a fine mesh is calculated at location 4. Always false at location N+4.

5/N+5: 1BZ edge, $2\pi/a_0$

In order to get a picture of the band structure calculated, a program 'readenergy.f90' have been developed in order to extract the energies in the [111]-direction. This program simply takes in an energy output file from 'KPBAND2mkl.f90', and returns $k_{[111]}$ versus energy.

D.2. User guide for SCRATES

SCRATES is a program made to calculate scattering rates for holes in semiconductors numerically, in this case for $HgCdTe$. It is based on the extrapolation algorithm using a

uniform cubic mesh of energies in $\vec{k}$-space, their corresponding energy gradients, and in some cases also their eigenvectors. These values are obtained by running the KPBAND program described in Appendix D.1, where the mentioned values for initial and final band are stored.

The program is set up to calculate rates for the following scattering mechanisms:

- Polar optical phonon scattering (absorption and emission)

- Acoustic deformation potential phonon scattering (absorption and emission)

- Non-polar optical phonon scattering (absorption and emission)

- Ionized impurity scattering

- Alloy scattering

The overlap factors have three possible ways of being calculated:

- Overlap factors amount to unity.

- Analytical expressions given by Wiley

- Overlap factors calculated from the eigenvectors of the $N \times N$(for any of the possible model sizes) $\vec{k} \cdot \hat{p}$ Hamiltonian

The scattering rates calculated by SCRATES are stored as the output files from KP-BAND, as a uniform cubic mesh in $\vec{k}$-space. If a fine mesh is contained in the initial state files, then also the output file for the scattering rates will contain the same fine mesh. The set up for SCRATES is the same as for KPBAND, a module rateinit.f90 to change material and model size parameters, an initialization program SCRINI.f90 producing an initialization file SCRATES.INI to chose settings, and the calculation program SCRATES.f90 to make the calculations. The following subsections contain the user guide for using SCRATES:

D.2.1. Modifying rateinit.f90

In the module rateinit.f90 the scattering rate parameters for the material is chosen manually, and as for KPBAND these parameters were originally hard coded into SCRATES.f90 (or SCRATES.f), and the setting of them was moved for the same reason. As can be seen in Appendix B.4. there are two parameters where the value for electrons and holes are different. This means that when calculating electron scattering rates the parameter line for holes must be commented out, and opposite for holes. And as for halvorinit.f90 the parameter setting in ratesinit.f90 should for future works be set in SCRINI.f90, storing to files like MCT-28-e.INI for electron scattering in $Hg_{0.72}Cd_{0.28}Te$.

The correct model size parameters have to be chosen in line (42) of 'rateinit.f90', and they ave to be the same as were used in KPBAND for producing the input files. A mechanism for checking for the parameter size has of yet not been developed, and a non-consistency could lead to problems when running SCRATES.f90.

D.2.2. Using SCRINI.f90

This program sets the settings for the calculations to be made in SCRATES by producing or redefining an initialization file SCRATES.INI. This program is run much the same way as KPINI.f90, through a main menu set-up where all non-sense terminal input are rejected. If the SCRATES.INI does not exist prior to running, the following messages appears:

```
NO INITIALIZATION FILE IS FOUND
FILE SCRATES.INI HAS BEEN CREATED
```

The file "SCRATES.INI" has now been created. The following is the example of the use of this program where "SCRATES.INI" already does exist, starting from the main menu:

```
....SCRINI MAIN MENU.....

YOU MAY :
1.   CHANGE SCATTERING MACHANISM
2.   CHANGE TEMPERATURE
3.   CHANGE INITIAL AND FINAL BANDS
4.   CHANGE FILENAMES
5.   ALTER NAMES AND NUMBER OF BANDS
6.   SHOW ALL DATA
7.   CHANGE HOW TO CALCULATE OVERLAP IN
8.   SAVE AND EXIT

ENTER SELECTION :
```

A selection is made by pressing a number <1>-<8> and pressing <Enter>. The user is then led to one of the sub menus, or has chosen to display or save the current setting. The following are the use of this program by selecting choices 1-8:

Selection 1: CHANGE SCATTERING MACHANISM
This selection is the sub menu for choosing which scattering mechanism to use.

```
....CHANGE SCATTERING MACHANISM....

THE CURRENT SCATTERING MECHANISM IS : POLAR OPTICAL, ABSORBTION

AVAILABLE SCATTERING MECHANISMS ARE:
1.   POLAR OPTICAL, ABSORBTION
2.   POLAR OPTICAL, EMISSION
3.   NONPOLAR OPTICAL, ABSORBTION
```

```
4.  NONPOLAR OPTICAL, EMISSION
5.  ACOUSTIC DEFORMATION POTENTIAL, ABSORBTION
6.  ACOUSTIC DEFORMATION POTENTIAL, EMISSION
7.  IONIZED IMPURITY SCATTERING
8.  ALLOY SCATTERING

ENTER SELECTION (0=UNCHANGED):
<1><ENTER>
```

Selection 2: CHANGE TEMPERATURE

Here the crystal temperature is set, and here the user has to set a temperature.

```
....CHANGE TEMPERATURE....

THE CURRENT TEMPERATURE IS 77.00 K

ENTER NEW TEMPERATURE [K] :
<100><Enter>
```

Selection 3: CHANGE INITIAL AND FINAL BANDS

Here it is chosen which bands are to be initial and final bands. The bands which are to be options, are defined in selection 5, and their file names are defined in selection 4. The bands to be selected must have been previously defined in selections 4 and 5.

```
....CHANGE INITIAL AND FINAL BANDS....

THE CURRENT SETTING IS :
1.  INITIAL BAND : CB BAND
2.  FINAL BAND   : CB BAND

WHICH DO YOU WANT TO CHANGE (0=NONE) ?
<1><ENTER>

POSSIBLE INITIAL BANDS ARE :
1.  LH BAND
2.  HH BAND
3.  CB BAND

MAKE A SELECTION (0=NONE)
<1><ENTER>

....CHANGE INITIAL AND FINAL BANDS....

THE CURRENT SETTING IS :
```

```
1.   INITIAL BAND : LH BAND
2.   FINAL BAND   : CB BAND

WHICH DO YOU WANT TO CHANGE (0=NONE) ?
<2><ENTER>


POSSIBLE INITIAL BANDS ARE :
1.   LH BAND
2.   HH BAND
3.   CB BAND


MAKE A SELECTION (0=NONE)
<2><ENTER>


....CHANGE INITIAL AND FINAL BANDS....


THE CURRENT SETTING IS :
1.   INITIAL BAND : LH BAND
2.   FINAL BAND   : HH BAND


WHICH DO YOU WANT TO CHANGE (0=NONE) ?
<0><ENTER>
```


Selection 4: CHANGE FILENAMES

Here the names of the optional input files (produced by KPBAND2mkl.f90) and the
output file are defined. The optional bands are again chosen in selection 5, and which
to use in selection 3. The eigenvector file is only needed if the overlap factors are to
be calculated by them (choice 1 in selection 7). If the eigenvectors are not needed, a
dummy file name may be entered.

```
....CHANGE FILENAMES....

THE FILENAMES ARE:


                          ENERGIES   GRADIENTS   EIGENVECTORS

1.               LH BAND : LHENE.TAB   LHGRA.TAB   LHEIG.TAB
2.               HH BAND : HHENE.TAB   HHGRA.TAB   HHEIG.TAB
3.               CB BAND : CBENE.TAB   CBGRA.TAB   CBEIG.TAB


4.   OUTPUT FILE : POA_CC.TAB


WHICH DO TOU WANT TO CHANGE (0=NONE)
<1><Enter>
NEW NAMES FOR LH BAND.
```

```
ENTER ENERGY FILE NAME :
<LHENE.TAB><ENTER>
ENTER GRADIENT FILE NAME :
<LHGRA.TAB><ENTER>
ENTER EIGENVECTOR FILE NAME :
<LHEIG.TAB><ENTER>

....CHANGE FILENAMES....

THE FILENAMES ARE:

                         ENERGIES    GRADIENTS   EIGENVECTORS

1.              LH BAND : LHENE.TAB   LHGRA.TAB   LHEIG.TAB
2.              HH BAND : HHENE.TAB   HHGRA.TAB   HHEIG.TAB
3.              CB BAND : CBENE.TAB   CBGRA.TAB   CBEIG.TAB

4.  OUTPUT FILE : POA_CC.TAB

WHICH DO TOU WANT TO CHANGE (0=NONE)
<4><Enter>
ENTER NEW OUTPUT FILE NAME :
<POA_LH.TAB><ENTER>

....CHANGE FILENAMES....

THE FILENAMES ARE:

                         ENERGIES    GRADIENTS   EIGENVECTORS

1.              LH BAND : LHENE.TAB   LHGRA.TAB   LHEIG.TAB
2.              HH BAND : HHENE.TAB   HHGRA.TAB   HHEIG.TAB
3.              CB BAND : CBENE.TAB   CBGRA.TAB   CBEIG.TAB

4.  OUTPUT FILE : POA_LH.TAB

WHICH DO TOU WANT TO CHANGE (0=NONE)
<0><Enter>
```

Selection 5: ALTER NAMES AND NUMBER OF BANDS

Here names and number of optional bands are altered, and again their file names are defined in selection 4, and chosen for calculation in selection 3. In this selection there are four possibilities, which are to rename bands, add a new band, discard a band and to list the optional bands. This is the chronological order of the choices in this section, but the example of their use, is to be in the order 4th, 2nd, 4th, 1st, 4th, 3rd and 4th in order

to best describe the choices of this selection.

```
....ALTER NAMES AND NUMBER OF BANDS....

YOU MAY :
1.   RENAME BAND
2.   ADD BAND
3.   DISCARD BAND
4.   LIST BANDS

ENTER SELECTION (0=NONE):
<4><ENTER>

THE BANDS DEFINED ARE:
1.   LH
2.   HH
3.   CB

....ALTER NAMES AND NUMBER OF BANDS....

YOU MAY :
1.   RENAME BAND
2.   ADD BAND
3.   DISCARD BAND
4.   LIST BANDS

ENTER SELECTION (0=NONE):
<2><ENTER>
ENTER NAME OF NEW BAND:
SOFF

....ALTER NAMES AND NUMBER OF BANDS....

YOU MAY :
1.   RENAME BAND
2.   ADD BAND
3.   DISCARD BAND
4.   LIST BANDS

ENTER SELECTION (0=NONE):
<4><ENTER>

THE BANDS DEFINED ARE:
1.   LH
2.   HH
3.   CB
```

```
4.  SOFF

....ALTER NAMES AND NUMBER OF BANDS....

YOU MAY :
1.  RENAME BAND
2.  ADD BAND
3.  DISCARD BAND
4.  LIST BANDS

ENTER SELECTION (0=NONE):
<1><ENTER>

YOU MAY RENAME :
1.  LH
2.  HH
3.  CB
4.  SOFF
ENTER SELECTION (0=NONE):
<4><ENTER>
ENTER NEW NAME :
<SO><ENTER>

....ALTER NAMES AND NUMBER OF BANDS....

YOU MAY :
1.  RENAME BAND
2.  ADD BAND
3.  DISCARD BAND
4.  LIST BANDS

ENTER SELECTION (0=NONE):
<4><ENTER>

THE BANDS DEFINED ARE:
1.  LH
2.  HH
3.  CB
4.  SO

....ALTER NAMES AND NUMBER OF BANDS....

YOU MAY :
1.  RENAME BAND
2.  ADD BAND
3.  DISCARD BAND
```

```
4.  LIST BANDS

ENTER SELECTION (0=NONE):
<3><ENTER>

YOU MAY DISCARD :
1.  LH
2.  HH
3.  CB
4.  SO
ENTER SELECTION (0=NONE):
<4><ENTER>

....ALTER NAMES AND NUMBER OF BANDS....

YOU MAY :
1.  RENAME BAND
2.  ADD BAND
3.  DISCARD BAND
4.  LIST BANDS

ENTER SELECTION (0=NONE):
<4><ENTER>

THE BANDS DEFINED ARE:
1.  LH
2.  HH
3.  CB

....ALTER NAMES AND NUMBER OF BANDS....

YOU MAY :
1.  RENAME BAND
2.  ADD BAND
3.  DISCARD BAND
4.  LIST BANDS

ENTER SELECTION (0=NONE):
<0><ENTER>
```

Selection 7: <u>SPECIFY HOW TO CALCULATE OVERLAP IN</u>

Here the method for calculating the overlap integral is chosen, and the menus assume the 14×14-model. However only choice 1. is known to be valid for the expanded models (NB! Choice 1 for non-polar optical phonon scattering). However they may be used for eg. $GaAs$ calculations with the 6×6-model.

```
....SPECIFY HOW TO CALCULATE OVERLAP INTEGRAL....

THESE ARE THE ALTERNATIVES :
1.  BY EIGENVECTORS OF NxN KP-HAMILTONIAN
2.  BY ANALYTICAL EXPRESSIONS AFTER Wiley
3.  BY ASSUMING UNITY OVERLAP INTEGRAL

ENTER SELECTION :
<1><ENTER>
```

Selection 6: SHOW ALL DATA

Here the user is shown the currently selected options. The following will appear on the screen of the computer:

```
....SHOW ALL DATA....

THE CURRENT SCATTERING MECANISM IS : POLAR OPTICAL, ABSORPTION
THE CURRENT TEMPERATURE MECANISM IS : 100.00 K
INITIAL BAND : THE LH BAND
FINAL BAND : THE HH BAND
THE FILENAMES ARE:
                       ENERGIES   GRADIENTS  EIGENVECTORS
           LH BAND : LHENE.TAB   LHGRA.TAB   LHEIG.TAB
           HH BAND : HHENE.TAB   HHGRA.TAB   HHEIG.TAB
           CB BAND : CBENE.TAB   CBGRA.TAB   CBEIG.TAB
THE OVERLAP INTEGRAL IS CALCULATED BY EIGENVECTORS OF THE NxN
KP-HAMILTONIAN
OUTPUT FILE : POA_LH.TAB

PRESS ENTER TO CONTINUE
<ENTER>
```

Selection 8: SAVE AND EXIT

By making this selection, the settings chosen are stored to SCRATES.INI. However before the settings are saved, and the program is terminated terminated, the chosen settings are checked. If the chosen settings are not sufficient, some of the following warnings may occur:

```
SCRINI WARNING : SCATTERING MECHANISM ISN'T SET
SCRINI WARNING : INITIAL BAND IS NOT DEFINED
SCRINI WARNING : FINAL BAND IS NOT DEFINED
SCRINI WARNING : ENERGY FILE FOR INITIAL BAND IS NOT DEFINED
SCRINI WARNING : GRADIENT FILE FOR INITIAL BAND IS NOT DEFINED
```

```
SCRINI WARNING : EIGENVECTOR FILE FOR INITIAL BAND IS NOT DEFINED
SCRINI WARNING : ENERGY FILE FOR FINAL BAND IS NOT DEFINED
SCRINI WARNING : GRADIENT FILE FOR FINAL BAND IS NOT DEFINED
SCRINI WARNING : EIGENVECTOR FILE FOR FINAL BAND IS NOT DEFINED
SCRINI WARNING : ENERGY FILES FOR INITIAL AND FINAL BAND ARE
NOT DEFINED
SCRINI WARNING : GRADIENTS FILES FOR INITIAL AND FINAL BAND
ARE NOT DEFINED
SCRINI WARNING : EIGENVECTORS FILES FOR INITIAL AND FINAL BAND
ARE NOT DEFINED
SCRINI WARNING : NO OUTPUT FILE IS DEFINED
```

The settings will however be saved, and the program terminated, regardless if whether any of the above mentioned warning occur. Then the following two sentences (with the warnings between them) will occur on the screen:

```
...SAVE AND EXIT...

DATA SAVED ON FILE SCRATES.INI
```

D.2.3. Running SCRATES.f90

When the program SCRINI.f90 has been run, the initialization file SCRATES.INI has been produced. In order to run SCRATES.f90 the files SCRATES.INI, rateinit.f90 and the requested input files must be pressent. The terminal display the following information during the run, given that the previously chosen setting are used:

```
****************** SCRATES ******************
Program for numerical calculation of scattering
rates with discretized band structure.

SCATTERING MECHANISM IS : POLAR OPTICAL, ABSORPTION
TRANSITION FROM : LH
            TO : HH
AT TEMPERATURE :     100.000000000000      K
OVERLAP INTEGRAL IS CALCULATED FROM EIGENVECTORS
INPUT FILES FOR INITIAL STATES ARE:
LHENE.TAB
LHGRA.TAB
LHEIG.TAB
INPUT FILES FOR FINAL STATES ARE:
HHENE.TAB
HHGRA.TAB
```

```
HHEIG.TAB
OUTPUT FILE IS : POA_LH.TAB
READING DATA FROM FINAL STATES.
COARSE MESH :
ENERGIES...
GRADIENTS...
EIGENVECTORS...
FINE MESH :
ENERGIES...
GRADIENTS...
EIGENVECTORS...
CREATING MIN/MAX TABLES FOR FINAL STATES...
COARSE MESH...
FINE MESH...
COMPUTING RATES:
COARSE MESH...

FINE MESH...
```

D.2.4. File formats and use of readrates.f90

The output files from SCRATES.f90 have the same format as the files from KPBANS2mkl.f90, where only the 1st, 2nd and 4th variables are stored. The readrates.f90 program reads the output files, and creates the values for the incident $\vec{k}$ along the [111]-direction. The only difference is that the rate is plotted against the energy of the incident state.

D.3. User guide for the ABINIT related programs

ABINIT is a program for performing *ab initio* calculations, which for this work has been used to perform *ab initio* pseudopotential band structure calculations. In stead of merely producing a 1D band structure through the specific symmetry points, a 3D mesh of the band structure was extracted. In order to minimize the requirement for computer power only the unique 1/48 wedge of the full 1BZ was extracted, and the complete cube was filled in via symmetry operations.

For running the ABINIT program the input files have to be created. As a 3D wedge of the 1BZ is to be calculated the input files had to be created by a program ABINITGEN-ERATOR.f90 written for this work. Further the conversion of the files to KPBAND format and the symmetry operations were made by a program ABINITCONVERTER.f90 written for this work, where also the coarse and the fine mesh are combined. The gradients and second derivatives are then calculated through discrete differentiation by a program ABINITDIFFERENTIATEOR.f90 written for this work. The settings for running the programs ABINITCONVERTER.f90 and ABINITDIFFERENTIATEOR.f90 are chosen through the program ABINITINI.f90, which stores the chosen settings in the initialization file ABINIT.INI. Also the latter three mentioned programs use the module

halvorinit.f90, because the resulting files from these programs have to have the same framework as that from the KPBAND program.

D.3.1. Using the ABINITGENERATOR.f90

In this program it is described how the input files for ABINIT are generater through the written program ABINITGENERATOR.f90. The current setup of this program generates the input for performing PAW calculations for a 1/48 wedge of a coarse cubic full 1BZ mesh and a fine cubic (10% of 1BZ), each of a $40 \times 40 \times 40$ mesh. The coarse input files are named MCT1.files and MCT1.in, and the fine input files are named MCT2.files and MCT2.in. The input files presuppose the pseudopotential files 80hg.lda.atompaw and 52te.lda.atompaw. All of these settings may of course be changed by the user at its own risk. The following is displayed in the terminal when running the program:

```
COARSE MESH RUNNING FILE "MCT1.files" CREATED

FINE MESH RUNNING FILE "MCT2.files" CREATED

COARSE MESH INPUT FILE "MCT1.in" CREATED

FINE MESH INPUT FILE "MCT2.in" CREATED
```

D.3.2. Using the ABINITINI.f90

In this program the user makes the following choices for the conversion calculations:

- Naming output files from ABINIT

- Naming output files from these programs

By running this program the initialization file ABINIT.INI is produced/created, and this file has to exist in order to run both the ABINITCONVERTER.f90 and ABINITD-IFFERENTIATEOR.f90 programs. If ABINIT.INI exists prior to the running of ABINI-TINI.f90, the old values in ABINIT.INI are loaded during the run, so one does not necessarily have to make all choices again. The user interface is a "no nonsense" menu system. The user enters and exits the program through the main menu. The following is the main menu of the program, where the user makes a choice by typing a number 1-4:

```
**** ABINITINI MAIN MENU *****
1.   RENAME ABINIT OUTPUT FILE.
2.   RENAME OUTPUT FILES.
3.   SHOW SETTINGS.
4.   SAVE AND QUIT.
ENTER SELECTION :
```

Selection 1: <u>RENAME ABINIT OUTPUT FILE</u>

In this choice the user gives the filenames of the coarse and the fine meshes produced by ABINIT.

```
Enter name of coarse ABINIT output files:
<ABINITC><Enter>
Enter name of fine ABINIT output files:
<ABINITF><Enter>


**** SHOW SETTINGS ****
Filenames
           EnergiesA    Energies    Gradients    Sec.  deriv.
1.  Band:  SOENEA.TAB   SOENE.TAB   SOGRA.TAB    SODER.TAB
2.  Band:  LHENEA.TAB   LHENE.TAB   LHGRA.TAB    LHDER.TAB
3.  Band:  HHENEA.TAB   HHENE.TAB   HHGRA.TAB    HHDER.TAB
4.  Band:  CBENEA.TAB   CBENE.TAB   CBGRA.TAB    CBDER.TAB
5.  Band:  CSENEA.TAB   CSENE.TAB   CSGRA.TAB    CSDER.TAB
6.  Band:  CLENEA.TAB   CLENE.TAB   CLGRA.TAB    CLDER.TAB
7.  Band:  CHENEA.TAB   CHENE.TAB   CHGRA.TAB    CHDER.TAB
COARSE ABINIT OUTPUT FILE: ABINITC
FINE ABINIT OUTPUT FILE: ABINITF


PRESS ENTER TO RETURN...
```

Selection 2: <u>RENAME OUTPUT FILE</u> In this choice the user chooses the names of the files produced by the programs ABINITCONVERTER.f90 and ABINITDIFFER-ENTIATEOR.f90. First the user chooses which band, whose names are to be changed. The band is chosen by numbers 1-7, number referring to the bands as in the choice "SHOW SETTINGS". The names are then changed in the order in the following example for changing the light hole band:

```
OF WHICH BAND SHOULD FILENAMES BE CHANGED?(NONE=0):
<2><ENTER>
New filenames for band 2
Enter name of converted ABINIT energy file [LHENEA.TAB] :
<LHENEA.TAB><ENTER>
Enter name of final energy file [LHENE.TAB] :
<LHENE.TAB><ENTER>
Enter name of final gradient file [LHGRA.TAB] :
<LHGRA.TAB><ENTER>
Enter name of final sec.  derivative file [HHDER.TAB] :
HDER.TAB
```

```
**** SHOW SETTINGS ****
Filenames
            EnergiesA    Energies    Gradients   Sec.  deriv.
1.  Band:   SOENEA.TAB   SOENE.TAB   SOGRA.TAB   SODER.TAB
2.  Band:   LHENEA.TAB   LHENE.TAB   LHGRA.TAB   LHDER.TAB
3.  Band:   HHENEA.TAB   HHENE.TAB   HHGRA.TAB   HHDER.TAB
4.  Band:   CBENEA.TAB   CBENE.TAB   CBGRA.TAB   CBDER.TAB
5.  Band:   CSENEA.TAB   CSENE.TAB   CSGRA.TAB   CSDER.TAB
6.  Band:   CLENEA.TAB   CLENE.TAB   CLGRA.TAB   CLDER.TAB
7.  Band:   CHENEA.TAB   CHENE.TAB   CHGRA.TAB   CHDER.TAB
COARSE ABINIT OUTPUT FILE: ABINITC
FINE ABINIT OUTPUT FILE: ABINITF

PRESS ENTER TO RETURN...
```

Selection 3: SHOW SETTINGS

```
**** SHOW SETTINGS ****
Filenames
            EnergiesA    Energies    Gradients   Sec.  deriv.
1.  Band:   SOENEA.TAB   SOENE.TAB   SOGRA.TAB   SODER.TAB
2.  Band:   LHENEA.TAB   LHENE.TAB   LHGRA.TAB   LHDER.TAB
3.  Band:   HHENEA.TAB   HHENE.TAB   HHGRA.TAB   HHDER.TAB
4.  Band:   CBENEA.TAB   CBENE.TAB   CBGRA.TAB   CBDER.TAB
5.  Band:   CSENEA.TAB   CSENE.TAB   CSGRA.TAB   CSDER.TAB
6.  Band:   CLENEA.TAB   CLENE.TAB   CLGRA.TAB   CLDER.TAB
7.  Band:   CHENEA.TAB   CHENE.TAB   CHGRA.TAB   CHDER.TAB
COARSE ABINIT OUTPUT FILE: ABINITC
FINE ABINIT OUTPUT FILE: ABINITF

PRESS ENTER TO RETURN...
```

Selection 4: SAVE AND QUIT
By this choice all settings are store in ABINIT.INI and the program is terminated.

```
**** SAVE AND QUIT ****

DATA WRITTEN TO FILE ABINIT.INI.
```

D.3.3. Running **ABINITCONVERTER.f90**

After the programs ABINIT and ABINI.f90 have been run, and the initialization file ABINIT.INI has been produced, the program ABINITCONVERTER.f90 may be run. This program converts the energy files from the 1/48 wedge produced by ABINIT using the settings from the ABINITGENERATOR.f90 program. The program combines the coarse and the fine meshes and divides up the bands to energy band files like those produced by KPBAND2mkl.f90. The terminal displays the following information during the run:

```
STARTING CONVERSION FROM ABINIT COARSE MESH FILE

CONVERSION OF ABINIT COARSE MESH COMPLETED

STARTING CONVERSION FROM ABINIT FINE MESH FILE

CONVERSION OF ABINIT COARSE MESH COMPLETED

ABINIT CALCULATIONS CONVERTED TO KPBAND FORMAT. FILES STORED
```

D.3.4. Running **ABINITDIFFERENTIATOR.f90**

After the program ABINITCONVERTER.f90 have been run, the program ABINITD-IFFERENTIATOR.f90 may be run. This program generates the gradients and second derivatives by discrete differentiation. The files produced are of the exact same format as those from the KPBAND2mkl.f90 program. The terminal displays the following information during the run:

```
*******ABINITDIFFERENTIATOR*******
PROGRAM FOR DIFFERENTIATING BANDSTRUCTURES FROM ABINIT

PERFORMING COARSE MESH DIFFERENTIATION

PERFORMING FINE MESH DIFFERENTIATION
```

D.3.5. File formats and use of readers

The files produced are the exact same as those produced by the KPBAND program, only that eigenvector file are non-existent. These have to be extracted from KPBAND calculations. The results from this part of this work may be read and used exactly as the results from KPBAND.

References

[1] E. Halvorsen, *Diploma thesis*, 1991.

[2] Y. Chang, C. H. Grein, S. Sivananthan, S. Guha, S. Velicu, M. E. Flatté and V. Nathan, *AFRL-ML-WP-TP-2007-517*, 2007.

[3] K. V. Falch, *Master thesis*, 2013.

[4] Program ABINIT, *http://www.abinit.org/*.

[5] M. V. Fischetti and S. E. Laux, *Phys. Rev. B 38, p.9721*, 1988.

[6] J. M. Hinkley and J. Singh, *Appl. Phys. Lett. 53, p.785*, 1988.

[7] M. Penna, A. Marnetto, F. Bertazzi, E. Bellotti and M. Goano, *Journal of Electronic Materials*, 2009.

[8] F. Bertazzi, M. Moresco, M. Penna, M. Goano and E. Bellotti, *Journal of Electronic Materials DOI: 10.1007/s11664-010-1198-0* , 2010.

[9] E. P. Pokatilov, V. A. Fonoberov, V. M. Fomin and J. T. Devreese, *Phys. Rev. B 64, p.245328*, 2001.

[10] P. Pfeffer and W. Zawadzki, *Phys. Rev. B 53 p.12813*, 1995.

[11] B. Karlsen, *Specialized Project thesis*, 2012.

[12] N.Troullier and J. L. Martins, *Phys. Rev. B 43, p.1993*, 1991.

[13] T. Brudevoll, T. A. Fjeldly, J. Baek and M. Schur, *J. Appl. Phys. 67 p.7373*, 1990.

[14] S. Derelle, S. Bernhardt, R. Haïdar, J. Primot, J. Deschamps, J. Rothman, *IEEE Transactions on Electron Devices, 56 p. 569* , 2009.

[15] B. K. Ridley, *Quantum Processes in Semiconductors fourth ed.*, 1999.

[16] H.R. Trebin, U. Rössler and R. Ranvaud, *Phys. Rev. B20, p.686*, 1979.

[17] H. Mayer and U. Rössler, *Phys. Rev. B44, p.9048*, 1991.

[18] R. M. Martin, *Electronic Structure*, 2008.

[19] J. P. Perdew and K. Burke, *Int. J. Quantum Chemistry, Vol. 57, p.309*, 1994.

[20] V. I. Anisimov and O. Gunnarsson, *Phys. Rev. B42, p.7570*, 1991.

[21] Program WIEN2K, *http://www.wien2k.at/*.

[22] Program ATOMPAW, *http://users.wfu.edu/natalie/papers/pwpaw/man.html*.

[23] F. Tran, P. Blaha, K. Schwarz and P. Novak, *Phys. Rev. B75, p.115131*, 2006.

[24] F. Tran and P. Blaha, *Phys. Rev. B83, p.235118*, 2011.

[25] J. A. Camargo-Martínez and R. Baquero, *Phys. Rev. B86, p.195106*, 2012.

[26] W. C. Herring, *Phys. Rev. 57, p.1169*, 1940.

[27] W. C. Herring and A. G. Hill, *Phys. Rev. 58, p.132*, 1940.

[28] F. Herman, *Rev. Mod. Phys. 30, p.102*, 1958.

[29] F. Herman, *Phys. Rev. 95, p.847*, 1954.

[30] F. Bassani, *Rev. Mod. Phys. 108, p.263*, 1958.

[31] M. Torrent, *ABINIT forum: http://forum.abinit.org/viewtopic.php?f=7t=2157*, May 27, 2013.

[32] F. Long, W. E. Hagston, P. Harrison and T. Stirner, *J. Appl. Phys. 82 p.3414*, 1997.

[33] S. E. Schacham and E. Finkman, *J. Appl Phys. 57 p.2001*, 1985.

[34] G. L. Hansen, J. L. Schmidt and T. N. Casselman, *J. Appl. Phys. 53 p.7099*, 1982.

[35] J. P. Laurenti, L. Camassel, A. Bouhemadou, B. Toulouse, R. Legros and A. Lusson, *J. Appl. Phys. 67 p.6454*, 1990.

[36] B. Li, J. H. Chu, Y. Chang, Y. S. Gui and D. Y. Tang, *Infrared Phys. Technol. 37 p.525*, 1996.

[37] P. B. Allen and M. Cardona, *Phys. Rev. B23, p.1495*, 1981.

[38] M. Cardona, *Solid State Communications 133 p.3*, 2004.

[39] D. Olguín, M. Cardona and A Cantarero, *Solid State Communications 122 p.575*, 2002.

[40] S. Krishnamurthy, A.-B. Chen, A. Sher and M van Schilfgaarde, *Journal of Electronic Materials vol. 24 p.1121*, 1995.

[41] P. E. Blöchl, *Phys. Rev. B50, p.17953*, 1994.

[42] G. Gilat, and L. J. Raubenheimer, *Phys. Rev. 144 p.390*, 1966.

[43] J. D. Wiley, *Phys. Rev. B4 p.2485*, 1971.

[44] Ø. Skåring, *Master Thesis*, 2010.

[45] B. Lund, *Doctoral Thesis*, 1992.

[46] P. D. Yoder, V. D. Natoli, and R. M. Martin, *J. Appl. Phys. vol. 73 p.4378*, 1993.

[47] J. A. Mroczkowski and D. A. Nelson, *J. Appl. Phys. 54 p.2041*, 1983.

[48] P. Capper, *Narrow Gap Cadmium-based Compounds*, 1994.

[49] H. Mayer and U. Rössler, *Solid State Communications 87 p.81*, 1993.

[50] B. Gelmont, B. Lund, Ki-Sang Kim, G. U. Jensen, M. Shur and T. A. Fjeldly, *J. Appl. Phys. 71 p.4977*, 1992.

[51] G. U. Jensen, *Doctoral Thesis*, 1989.