2022-04-18

Accurate Band Energies of Metals with Quadratic Integration

Jeremy John Jorgensen
Brigham Young University

Accurate Band Energies of Metals with Quadratic Integration

Jeremy John Jorgensen

A dissertation submitted to the faculty of
Brigham Young University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Gus L. W. Hart, Chair
Branton J. Campbell
Karine Chesnel
Eric W. Hirschmann
Jean-Francois S. Van Huele

Department of Physics and Astronomy

Brigham Young University

ABSTRACT

Accurate Band Energies of Metals with Quadratic Integration

Jeremy John Jorgensen
Department of Physics and Astronomy, BYU
Doctor of Philosophy

Materials play an important role in society. Historically, and even at present, materials have been discovered by trial and error, and many of the most useful materials have been discovered by chance. The high-throughput approach aims to remove (as much as possible) chance and guesswork at the experimental level by filtering out materials candidates that are not predicted to exist. Many successes have been recorded.

In the high-throughput approach to materials discovery, machined-learned models of materials are created from databases of theoretical materials. These databases are the result of millions of density-functional-theory (DFT) simulations. The size and accuracy of the data in the databases (and, consequently, the predictions of machined-learned models) are most affected by the band energy calculation; most of the computation of a DFT simulation is computing the band energy in the self-consistency cycle, and most of the error in the simulation comes from band energy error. The band energy is obtained from a two-part multidimensional numerical integral over the Brillouin or irreducible Brillouin zone.

A quadratic approximation and integration algorithm for computing the band energy in 2D and 3D is described. Analytic and semi-analytic integration of quadratic polynomials over simplices improves the accuracy and efficiency of the calculation. A method is proposed for estimating the error bounds of the quadratic approximation that does not require additional eigenvalues. Error propagation of approximation errors leads to an adaptive refinement approach that is driven by band energy error. Because adaptive meshes have little symmetry, integration is performed within the irreducible Brillouin zone, and a general algorithm for computing the irreducible Brillouin zone is described.

The efficiency of quadratic integration is tested on realistic empirical pseudopotentials. When compared to current integration methods, uniform quadratic integration over the irreducible Brillouin zone sometimes results in fewer k-points for a given accuracy. Adaptive refinement fails to improve integration performance because band energy error bounds are inaccurate, especially at accidental crossings at the Fermi level.

Keywords: Brillouin zone integration, electronic band structure, band theory, density-functional theory, Fermi level, Fermi surface, band energy, total energy, band crossings, high-throughput, materials discovery, materials science

ACKNOWLEDGMENTS

This dissertation is the result of the sustained efforts of many individuals, principal of whom is my research advisor, Gus Hart. His generosity with his time and willingness to meet frequently kept this project moving. It would not have come as far as it did without his support and knowledge. I am grateful for his mentorship in research and life and for the privilege I had of doing research with him.

Another invaluable mentor has been Tom Sederberg. Tom generously gave time and effort over the course of many years to help see the project through. As soon as Tom joined late into the project, progress accelerated. I am grateful for his reliability and endurance with a project that at times appeared unconquerable. I want to acknowledge Derek Thomas who also contributed substantially early on. Many other faculty at BYU have unselfishly contributed: Branton Campbell, Tyler Jarvis, Rodney Forcade, and Brett Hess. Their expertise has been critical in many different areas. Thanks to all of the members on my committee: Branton Campbell, Karine Chesnel, Eric Hirschmann, and Jean Francois Van Huele. Our annual progress reports went a long way towards keeping the project more or less on track.

I started graduate school with Wiley Morgan, Conrad Rosenbrock, and Chandramouli Nyshadham, all of whom have been good examples of successful graduate students. I have had the pleasure to work with many undergraduates over the years: Nate Foulk, John Christensen, Natalie Coy, Hayden Oliver, Spencer Hart, and Matt Burbidge. I appreciate your efforts and am glad for the chance I had to mentor, in part, your research.

Last but foremost is the support I have received away from school-related activities. My family has been a source of immense support throughout my graduate tenure. I am grateful for the encouragement from my parents, John and Karyn, and will forever be indebted to them for a loving upbringing and ongoing support. I am grateful for the kindness and friendship of my siblings, in-laws, nephews, and nieces. Thanks for the gift of your presence in my life.

Contents

Chapter 1

Materials and society

1.1 Materials ancient and modern

Materials are the often overlooked building blocks of all societal comforts. The clothes on our backs, the house over our heads, the paper or screen from which you read are possible because of the unique properties of materials. All technologies, both ancient and modern, are dependent on suitable materials. If left unconstrained, the scope of materials science could touch all aspects of the physical universe itself. Because materials play an important role in many aspects of life, materials with properties heretofore unknown will make possible unimaginable scientific advancements.

Figure 1.1 Ancient Chinese cauldrons called dings were made of bronzes or ceramics and used to cook and store food and for religious offerings. The ding above is from the Los Angeles County Museum of Art (Commons 2019). It is estimated to have been made between 1300-1100 B.C. and is made of cast bronze with a black paste inlay.

In the past, primitive tools made of naturally occurring materials helped ancient people survive in a competitive environment. Flint was used to craft spearheads, axes, knifes, and start fires. Wood, another naturally occurring material, has been made into a many useful and varied objects, such as

furniture and housing. Ceramics were used anciently for dish-ware and enabled a greater variety of foods to be cooked over fire. Materials are so essential to the development of civilization that periods of history have been named after the predominant materials of those ages (stone, copper, bronze, iron, and now silicon). It is hard not to downplay the importance of materials in the evolution of society. Even today, from synthetic fabrics to field-effect transistors in computing and communications circuitry, there is scarcely any aspect of society that materials science has not influenced.

The materials of today use all of the naturally occurring elements of the periodic table, whereas materials from not too long back used a small fraction of the elements available. Advancements in materials used to be a process of taking a known material and improving it by mining it more effectively, bringing down the cost of production, and producing it more consistently. This aspect of materials development continues today but in addition, materials with superior properties than those off-the-shelf are discovered and developed for specific applications. This approach to the development of materials is called materials by design or inverse design. There are often many applications of a material that have not been considered at the time of discovery.

1.2 Materials prevent tragedies

One of the most notable tragedies in history could have been avoided if materials with superior mechanical properties had been implemented. The Titanic was called unsinkable because it was designed to stay afloat if as many as four of its sixteen water-tight compartments were flooded. It is believed that when it collided with an iceberg at speed, five compartments were compromised. Examination of the hull and its rivets has shown that they played a large part in the sinking of the Titanic (Foecke 2008). It is possible the Titanic may have not capsized or at least capsized at a slower rate and made it to a nearby dock at Carpathia, if the poor quality material of the rivets in

the Titanic's hull had been high-quality wrought iron or steel. Another possible cause of sinking is brittle fracture of the steel plates of the ship's hull. Metals with greater mechanical properties could have prevented both failures. There were materials available with suitable mechanical properties but they were not used for financial reasons, so bringing down the cost of materials is an important aspect of materials science and engineering.

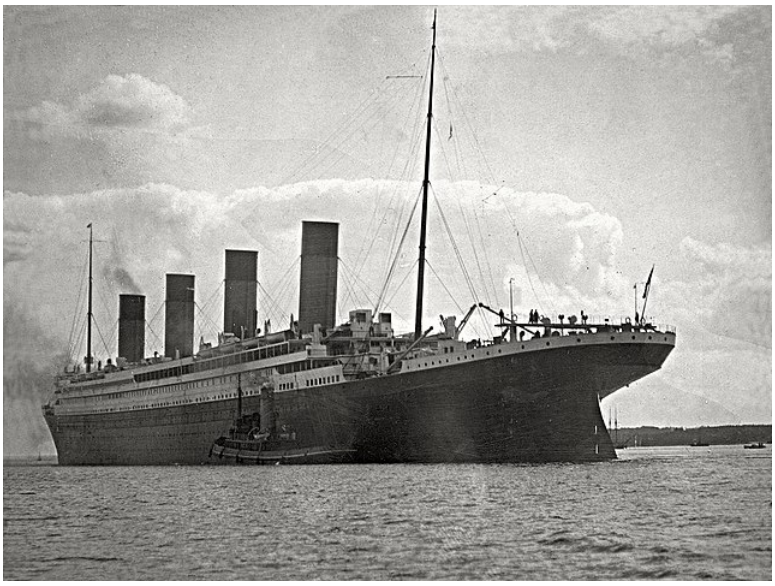


Figure 1.2 The sinking of the titanic is one notable disaster that could have been prevented had it been constructed with materials of superior mechanical properties. This photo of the Titanic was taken in 1912 (Commons 2021).

Corrosion is a problem that may forever affect materials science that comes with heavy costs to the economy, environment, and human lives. It is estimated that the financial burden of corrosion is 3% to 4% of the global gross domestic product (El-Sherik 2017), which today runs about $3 trillion USD. It is estimated that this cost could be reduced by up to $1 trillion from improvements in corrosion-resistant materials.

Corrosion is responsible for many industrial accidents. For example, in 1987 in Los Angeles, 8 employees were killed and 18 injured in an oil refinery explosion that resulted from corrosion

of a vapor line. (OSHA 1988). Accidents caused by corrosion affect the environment and human livelihood. On March 3, 2000 in Hunt, Texas, corrosion caused a leak in a fuel line and spilled 500,000 gallons of gas. The fuel made its way to Lake Tawakoni, which provides the the city of Dallas with $1/3$ its clean water. When traces of a cancerous gasoline additive were found in the lake, the water line was closed for 5 months until the additive was undetectable (Rieken 2016). More recently, two teenagers riding ATVs in Huntersville, North Carolina came across a bubbling liquid in the ground and reported the finding to the local fire department. Their discovery would eventually become, after many investigations and a 1,600 page report, one the largest pipeline spills in US history at 1.2 million gallons of fuel (Martin 2021). The cause of the spill was a crack in the pipeline caused by external corrosion. The leak occurred even though the pressure in the pipeline was well below the operating threshold. The integrity of the entire 5,500-mile Colonial Pipeline is currently being questioned and investigated (Sorg 2021).

1.3 Materials win wars

Materials developments affected the outcome of World War 2. Military equipment at the time required large amounts of rubber: 1,000 pounds for an airplane, 2,000 pounds for a tank, 75 tons for a battleship, and 32 pounds for each soldier's equipment (Museum 2021). When the Japanese seized Indonesia, they cutoff the supply of natural rubber to the United States. Subsequently, the U.S. was forced to mass produce synthetic rubber which was coincidentally discovered shortly before the onset of the war but had not previously been produced in such quantities as those required for military equipment.

Synthetic rubber was also used to insulate radar equipment, which was pivotal to the allies winning the war. The inventor of radar, Robert Watt, commented on the influence of synthetic rubber (Roberts 1989), polythene or polyethylene, "The availability of polythene transformed the design,

Figure 1.3 Shown above is a slab of synthetic rubber from a production facility in California around 1940 (Image: Library of Congress, LC-USW33-028402-C.).

production, installation, and maintenance problems of airborne radar from the almost insoluble to the comfortably manageable (owing to its almost ideal properties of insulation and structural integrity) . . . A whole range of aerial and feeder designs otherwise unattainable was made possible, a whole crop of intolerable air maintenance problems was removed. And so polythene played an indispensable part in the long series of victories in the air, on the sea, and on land, which were made possible by radar."

Teflon and high octane fuels, both developed during the war, also gave the allies an advantage in combat: high octane fuels gave aircraft more power and teflon was essential in the development of the nuclear bomb.

1.4 Current and future needs for materials

The advancement of machinery, communication, robotics, and computing depend on and are limited by available materials. Vacuum tubes in computers were replaced by transistors and integrated circuits once scientists were able to obtain very accurate material purities. Between 1968–2014, the rate at which fuel was burned in commercial aircraft dropped 45% (Kharina & Rutherford 2015). This was due in part to running the motor at higher temperatures, which was made possible by the discovery of superalloys with higher melting points. Traditional incandescent lights are being replaced at a fast rate by LED lights, which consume 85% less electricity and last 25 to 100 times longer than traditional incandescent lights (Popovich 2019). LED lights are possible because of the discovery of semi-conducting materials with specific band gaps that emit light in the visible spectrum. The tremendous benefit to mankind of LED lighting was recognized when those who took part in the discovery of the blue light-emitting diode, without which white light could not be created, won the 2014 Nobel Prize in Physics. There are countless examples of materials improving the technologies and the quality of life of people, especially those of meager financial means.

Many materials have yet to be discovered that could solve problems currently faced by society regarding clean energy, corrosion, and computing. Often there are materials that could solve problems (such as high-efficiency solar panels) but are impractical because they are too expensive to manufacture, difficult to work with, or are too rare to be produced at large scale and meet demand. Many practical aspects of a material's lifecycle from production to disposal are concerns of materials science and engineering.

Fusion reactors that produce more energy than the energy required to start and maintain the fusion reaction have been a pipe dream for scientists and conservationists for a very long time. The latest attempt at mimicking the thermonuclear processes of stars comes from one of the most ambitious and expensive collaborative science experiments in the history of humankind. The International Thermonuclear Experimental Reactor (ITER) located in the south of France hopes

to return 10 times the energy required to initiate the thermal reaction, despite many failures and not one success at a net positive energy return (Rincon 2014). One prominent attempt is the Mirror Fusion Test Facility at Lawrence Livermore National Laboratory that was mothballed a day after it was constructed and whose facility still remains as a ghost and testament to the indomitability of nuclear fusion (Booth 1987). With its abandonment came the eventual collapse of all mirror fusion research. Yet optimism can still be found and new facilities are under construction such as the SPARC tokamak at MIT (Creely et al. 2020).

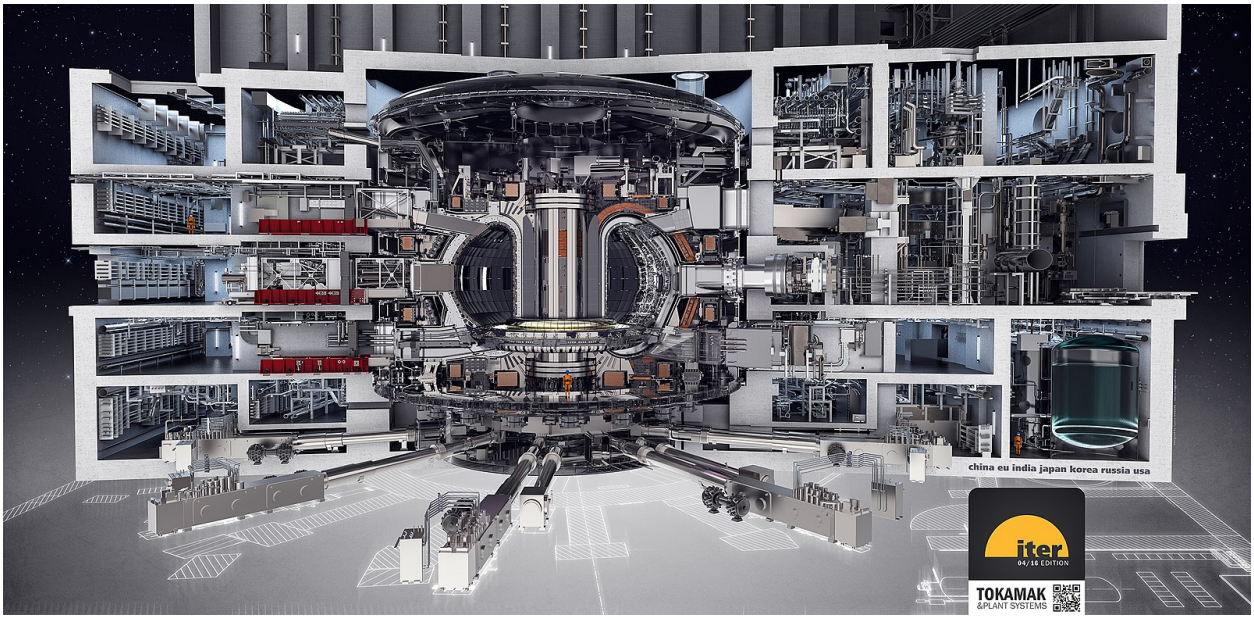


Figure 1.4 The ITER fusion reactor is a massive, collaborative scientific venture that seeks to make fusion energy a reality. Progress in materials science brings technological dreams like thermonuclear fusion closer to reality by accelerating development. The diagram of the tokamak (Wikipedia contributors 2021) above illustrates its size and complexity (the people dressed in orange give an idea of the scale).

Materials that are capable of withstanding bombardment of neutrons and extreme temperatures are necessary for magnetic confinement fusion (the ITER reactor will reach 100 million degrees). One isotope used in fusion reactions is extremely rare—the world supply of tritium is estimated to be 20 kg (Gibney 2021). Other hydrogen isotopes are readily available and could help fusion account

for a substantial amount of our energy demands. Once the fusion reaction starts, the inner region of the tokamak becomes radioactive, and humans cannot enter to make adjustments to experiments for many months (18 months for the ITER reactor). Materials that can withstand neutron bombardment without becoming radioactive would make fusion experiments by magnetic confinement easier to perform. During fusion, radioactive particles would damage diagnostic and observational sensors if they were not protected by a concrete barrier. This makes it difficult to observe all that occurs during the reaction. Sensors resilient against radiation would improve the accuracy of observations and potentially speed up development.

Batteries are becoming more popular than ever with the electrification of all things related to transportation. Most batteries in electric transportation and consumer electronics are lithium-ion. These batteries are often made of heavy, rare, and toxic metals, such as cobalt, that are difficult to dispose of properly and are not found in abundance (University 2020). Progress has been made in looking into materials that could replace these metals while not compromising aspects of the performance of the battery, including lifecycle, charge capacity, and rates of charging and discharging. One such material that could replace the cobalt cathode of lithium-ion batteries is a synthetic organic material made of phosphaviologens within carbon nanotubes (Bridges et al. 2020). This particular material delivered similar performance of traditional heavy metal cathodes while also being abundant and easily recycled because its elements are naturally interconverted with the earth in the carbon cycle. Organic batteries will have a greater role in sustainable energy storage in the future.

Hence, we see that materials and engineering applications of materials will continue to solve outstanding problems that society faces, and there is a great need to increase the rate at which materials are discovered.

Chapter 2

The evolution of materials discovery

2.1 Scientific paradigms

The term "scientific paradigm" (or "disciplinary matrix") was coined by Thomas Kuhn, a philosopher of science, in his book *The Structure of Scientific Revolutions* in 1962. Kuhn proposed that the evolution of science wasn't as rational or orderly as it was believed to be. Periods of science where theories are not questioned or challenged, where there are incremental improvements, and where the methods for performing science are agreed upon are termed by Kuhn "normal science". In normal science, the community holds in common certain ideas as truths, ways of thinking, and ways of performing science. Normal science occurs within the context of a scientific paradigm. The value of a paradigm comes from its ability to solve scientific puzzles and its explanatory and predictive power. Examples of scientific paradigms are Newtonian or quantum mechanics.

Discrepancies between reality and paradigm are found as more scientific puzzles are solved. At first, these are explained away or ignored. Eventually, there are too many anomalies or the anomalies are too severe to ignore, and a process of revision occurs. This revolutionary process or paradigm shift is a destructive and creative process. According to Kuhn, this process is often arational and influenced by social and political forces. Kuhn described paradigm shifts as "the greatest and most original bursts of creativity." (Horgan 2012) A paradigm shift is a time of disagreement. Often scientific progress is slowed during the revolution because individuals are expending effort disputing science instead of doing science and solving scientific puzzles. Some fields, many related to social or political science, never settle into normal science because the community never reaches a consensus. This may be related to the nature of the problems they are trying to solve. If agreement is reached, normal science again ensues. The accepted ideas and methods of science for the next paradigm are influenced by authority figures and majority opinion. The new paradigm usually maintains the predictive power of the previous while giving explanations for the anomalies the previous paradigm failed to explain or predict. The classification of organisms in biology or taxonomy is a theory of science undergoing dispute or a paradigm shift, and it may never reach the state of normal science

if the community fails to achieve consensus.

Some paradigm shifts occur without motivation from anomalies that the paradigm fails to predict. Revolutionary discoveries may occur within normal science that spur a new scientific paradigm. The discovery of the double-helical structure of DNA is one such example.

It was put forth by Jim Gray in *The Fourth Paradigm* that there have been four general, overarching paradigms that apply to all of science (Hey et al. 2009). It is important to note that scientific paradigms build upon the previous paradigms and do not replace them; importantly, the methods of performing science in one paradigm continue into the succeeding paradigms. For example, the first general paradigm was experimental exploration of the universe through trial and error. This method of science continues today and may forever be the method of separating scientific fact from fiction.

Theoretical science, the second paradigm, began with such figures as Newton or Kepler. In this paradigm, reality, the universe, physical phenomenon,... are modeled with mathematical formulas and tested against empirical measurements. A body falling under the force of gravity may be modeled mathematically with an algebraic curve. A mathematical relationship between the mass of a celestial body and its force on another may be expressed using symbols that represent the distances between the bodies and their masses. The path traced by planets were predicted and verified by Kepler to trace ellipses, another type of quadratic algebraic curve. In the second paradigm, mathematics became an additional tool used by scientists to understand and predict physical phenomenon.

In the second paradigm, progress in understanding a physical phenomenon begins with a simple approximation that is exactly solvable (has an analytic or closed-form solution). But simple models often do not accurately reflect reality except in specific situations or in particular limits. More accurate but complicated models usually cannot be solved exactly. For example, a closed-form solution to the motion of two orbiting point masses can be found, but no such solution exists for the motion of three orbiting bodies. This is the well-known three-body problem. Problems

in mathematics with no initial physical corollary often are not exactly solvable—the roots of polynomials of degree 1–4 have closed-form solutions but higher degree monomials do not. And so we see that there are many theoretical problems in many branches of science or other disciplines that do not have closed-form solutions.

The invention and development of computers brought about the third scientific paradigm, numerical methods or computational science, which can address a host of problems that could not be solved otherwise. Numerical methods find approximate solutions to problems that do not have analytic solutions, and usually these approximations become more accurate with increasing computational resources and time. Numerical methods come with proofs that they work so there can be confidence in the solutions they give even though the problems they solve do not have known solutions. For the examples mentioned above, the roots of higher-order polynomials may be approximated with numerical root-finding methods, such as Newton's method, (Galántai 2000) and the motion of three bodies may be determined from numeric integration methods (for example, Verlet or Runge-Kutta).

We are currently on the cusp of the fourth scientific paradigm in which large amounts of data control and direct more aspects of the scientific process: the questions to ask, the experiments to perform, the data to collect, the best models for the data, and the interpretation of the data. Several scientific fields, among them are astronomy and particle physics, have already had to make adjustments to day-to-day science to account for more data than they are capable of storing or processing. Currently, progress is being made towards data-intensive science with investments in modalities of data capture, curation, and analysis. In the future, data sharing and permanence—enduring, open-access, collaborative digital libraries of scientific data—will be integral to individual and interdisciplinary science (Bell et al. 2009). The fourth paradigm is in its infancy and can be found in approaches to science like machine learning, but it may eventually give way to another scientific paradigm in which human involvement is very limited. This could happen if general

artificial intelligence or superintelligence is achieved (Bostrom Accessed Oct, 2021).

The scientific paradigms in materials science started with physical experiments on materials through trial and error or guesswork. Afterwards, empirical measurements and mathematics were used to describe and predict the properties of materials. Simulations of materials on computers later became a tool for understanding materials and processes of materials too complex to model by pen and paper. Today materials are better understood, studied, and predicted with data-driven materials science. These paradigms are shown diagrammatically in Fig. 2.1.

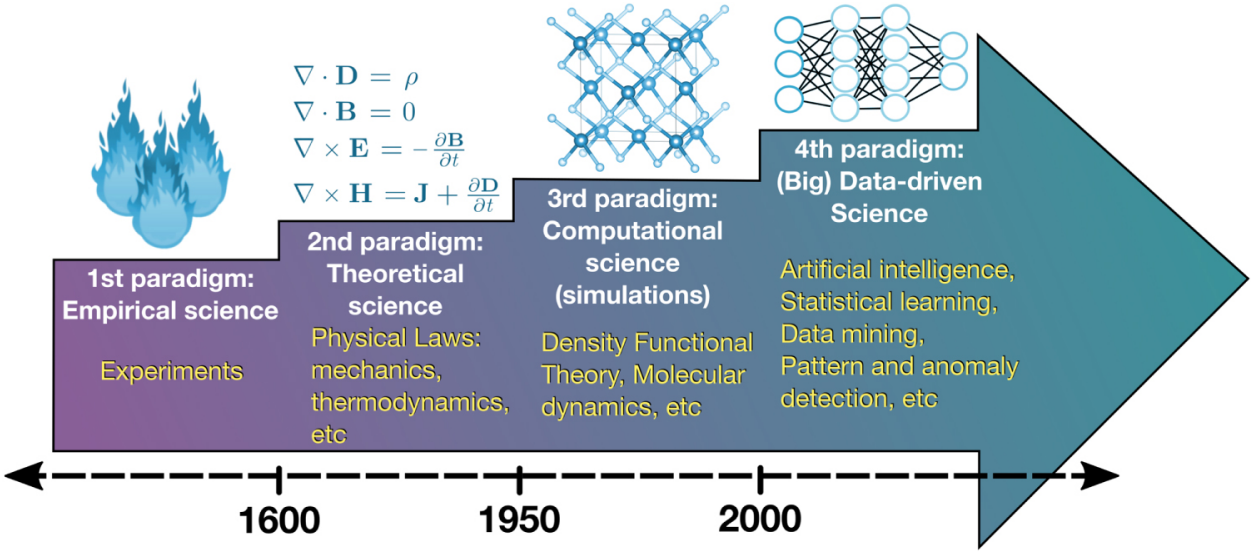


Figure 2.1 Shown in the diagram (Schleder et al. 2019) are the four scientific paradigms and their influence on materials science. One paradigm doesn't replace another; subsequent paradigms build upon previous ones.

2.2 The role of chance in materials discovery

Chance has played a role in some of the most influential discoveries. It will continue to play a role in discovery because it isn't possible to know everything there is to know. It is surprising how many of the most common, everyday materials have been discovered serendipitously (not sought after

and discovered by accident) or pseudo-serendipitously (sought after but discovered accidentally). Although many materials discoveries happen by chance, there is almost always someone involved with the discovery with expertise that recognizes the novelty and potential of the discovery.

Rubber, given its name when it was found that it could rub off pencil markings, is a tree emulsion that natives of South America used to form into a ball to play games. Many tried to find useful applications for it, but their attempts were frustrated because rubber becomes brittle at low temperatures and loses its shape at high temperatures. Charles Goodyear attempted for many years to stabilize rubber, going into excessive debt in the process. During one such attempt, rubber and a sulfur solution accidentally mixed on a hot stove. Charles noted that the rubber did not melt. He later left a piece outside in the cold over night and found out the next day that it was still flexible. He eventually canonized the process, produced a patent, and called his creation vulcanized rubber. Variations of this rubber are used to this day in car tires.

Synthetic rubber was pseudo-serendipitously discovered when an unexpected liquid from a chemical separation process of acetylene polymerized on its own when left out over the weekend at a Du Pont lab. This synthetic rubber would become Neoprene, which is used commonly for shoe soles, industrial hoses, window gaskets, and the bonding agent in double-sided belts (Roberts 1989).

At 18 years old, William Perkin decided that during the holiday break he would attempt to create a synthetic anti-malarial drug because the only known remedy came from the bark of a tree it the East Indies. He eventually tried adding carbon and hydrogen to the organic compound aniline but obtained instead a black mass. While cleaning out the flask, he noticed the water turned purple and thought it might be able to color fabrics. He sent a sample to a local dye company where his idea proved correct. With the financial support of his parents, he abandoned his studies to create the first artificial dye factory. The synthetic dye was so cost effective that practically everyone could afford purple fabrics, which until then were limited to the very wealthy because, to produce just one gram of natural dye, thousands of mollusks had to be harvested (Roberts 1989).

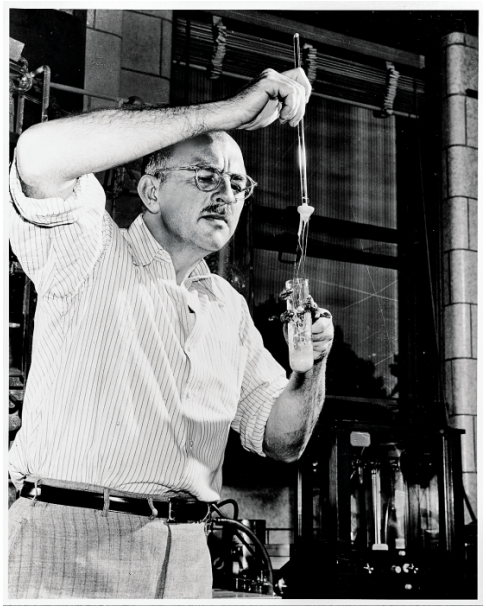


Figure 2.2 Shown is a re-enactment (Tullo 2016) of the cold drawing to create nylon at a DuPont Laboratory by Julian Hill.

Carothers and Hill, two scientists at a DuPont lab, were attempting to create the largest polymers possible using a molecular still. After one run, Hill noticed that the polymer solution was thick and sticky. He created a game where others in the lab were given a small amount of the polymer and tasked with stretching it as far as possible. This process is the same as cold drawing. Hill noticed that the fibers created when the polymer was stretched became stronger. When Hill and Carothers later studied the strings, they found that stretching the polymer aligned its molecules and created crosslinks between the molecules. After further study and experimentation, these polymer fibers would become the first synthetic fabric, nylon (Gaughan 2010).

Another DuPont scientist, Roy Plunkett, was hired to study chlorofluorocarbon (CFC) refrigerants. He prepared tetrafluoroethylene, a component gas that would later be needed to create CFC gas, and left the container of gas in dry ice over night. The next morning, he at first thought the gas had leaked out when no gas came out of the container. He sawed the container in half and found that the gas had polymerized into a substance waxy and white, which went against the best knowledge

Figure 2.3 Shown is a re-enactment (Poffenberger 2021) of the discovery of Teflon at a DuPont Laboratory by Roy Plunkett and colleagues.

of the behavior of tetrafluoroethylene at the time. Plunkett continued to study the waxy polymer and found it to have excellent properties: it was resistant to heat and corrosion, and very slippery. Its first application was containing a very corrosive, intermediary gas in the development of radioactive materials for the nuclear bomb in WWII (Gaughan 2010).

The previous materials mentioned are just a few examples of the many materials discovery by accident. The list of materials discovered by accident is quite long and includes shatterproof glass, super glue, silly putty, play dough, penicillin, stainless steel, cellophane, saccharin, dynamite, anesthesia, scotch guard, chewing gum, velcro, and silk (Roberts 1989).

Materials scientists would like chance to play a smaller role in the discovery of materials. In other words, they would like to predict materials that exist before they are discovered instead of coming across new materials by accident. There are so many elements and known crystal structures that testing each possible combination would take longer than the age of the universe, and these are only the crystalline materials! It simply isn't practical to search the space of all possible materials by trial and error.

2.3 Challenges to materials modeling

Modeling materials is a challenging task because of the many varieties of materials. Materials come in many different phases (solid, liquid, gas, ...). They can have an organized structure (crystals) or a disorganized structure (glass). The atoms in solids can be held together by different types or competing bonds: metallic (copper and aluminum), ionic (table salt), covalent (diamond), molecular (paraffin wax), and others. The electrons may be localized (insulators), delocalized (conductors), or in between (semi-conductors). There are many length and time scales at which a material may be modeled, from microscopic atomic models to the macroscopic continuum models.

As an example of the complexity of materials modeling, it took a book with over 400 pages to catalog just the 1D continuum models (Steinmann & Runesson 2021), very specific and low-dimensional models. Entire scientific disciplines are devoted to studying materials at one scale. Phenomena of materials that occur at one scale can affect the properties of the material at another (nanoporous materials). Some properties of materials are dynamic and time-dependent. A model may build in phenomena from different time and length scales (multiscale modeling). There are many properties of materials that can be modeled (mechanical, thermal, electrical, chemical, ...), and often different models are employed to study each property separately. One could spend an entire career studying the mechanical properties of a metal without ever considering its electrical properties. The properties of the material can vary by the direction of measurement (non-isomorphic). The number of possible materials is uncountable, as are the categories of materials.

Often tradeoffs have to be considered when approaching a materials model because, at present, it simply isn't possible or practical to use the most accurate quantum mechanical models for any macroscopic amount of a material for any observable time scale. It is common belief that the more physical phenomena a model can capture, the greater the predictive power of the model (Steinmann & Runesson 2021). Because phenomena for smaller scales affect those at larger scales, the most accurate models typically are those smallest in scale as possible (quantum), but they are also so

computationally expensive as to render them impractical today. Density-functional theory and codes built on this theory have become very practical as a compromise between speed and accuracy. Density-functional theory will be explored in greater detail in subsequent chapters.

2.4 Examples of the successful prediction of materials

Not all materials are discovered by chance, and it is hoped that in the future scientists will be able to predict the existence and properties of the most exotic materials before discovery. Anticipating the composition of promising materials is important because there are simply too many possibilities to realistically attempt to synthesize them all, and knowledge of available materials will allow engineers to make informed decisions for applications of materials.

Historically, one of the greatest tools for describing the properties of materials has been the periodic table. Invented by Dmitri Mendeleev in part from inspiration obtain from a dream and in part by grouping elements by their properties, it has facilitated the discovery of many base elements from which all materials are made. Mendeleev himself (who did not know about protons, neutrons or electrons) accurately predicted several elements that were missing from the table, including germanium, gallium, and scandium.

A recent research review by Saal et al. highlighted many of the materials that were predicted with machine-learning based approaches to materials prediction (Saal et al. 2020). The robust and general nature of materials prediction by machine learning is evident in the variety of materials successfully predicted: small molecules, inorganic solids, half-heusler alloys, high-entropy alloys, perovskites, glasses, polymers, and metallic glasses. Despite these successes, it is pointed out by the authors that only a small fraction of predictions of materials are verified experimentally, indicating a potential need for high-throughput experimental validation of predictions. The reproducibility crisis currently plaguing the social sciences could affect the science of materials prediction as well.

As the companion of the Islamic Prophet Muhammad, Abu Bakr, said: "Without knowledge action is useless and knowledge without action is futile."

Members of the Materials Simulation Group at Brigham Young University successfully predicted many new Ni-based, ternary superalloys from a combinatorial search over a subspace of materials space (Nyshadham et al. 2017). Many of the alloys predicted from the search—which involved about 2,000 first-principles calculations—were new or did not have phase diagrams reported in the literature. At least two of the predicted alloys were later verified experimentally, and work continues on these alloys with efforts aimed towards stabilization of the super alloy phase (Tirado et al. 2018).

Machine learning models benefit from knowledge of materials that exist and that do not exist. Unfortunately, there is an absence of literature on experiments that have failed to synthesize materials. This is just one area where the publication of null results is in the interest of scientific discovery.

Chapter 3

Materials databases in the high-throughput approach

The high-throughput approach to materials prediction is well described by Curtarolo et al. (2013):

> [High-throughput computational materials design] is simple yet powerful: create a large database containing the calculated thermodynamic and electronic properties of existing and hypothetical materials, and then intelligently interrogate the database in the search of materials with the desired properties. Clearly, the entire construct should be validated by reality, namely the existing materials must be predicted correctly and the hypothetical ones should eventually be made. Such a reality check feeds back to the theory to construct better databases and increase predictive power.

The high-throughput approach to materials prediction is a process in which as little effort as possible is spent on the investigation of a material until it becomes apparent that the material is likely to exist or have promising features. A blind search for a material with specific properties would likely start with a uniform sampling of a materials subspace, possibly informed by previous success or knowledge. A machine-learned model would be built on these simulations and be used to inform subsequent DFT simulations. Eventually, a collection of the most promising materials candidates from DFT simulations would be investigated experimentally. In this way, materials are investigated at first with the computationally inexpensive machine-learned interatomic potential, then simulated with computationally expensive DFT, and finally experimentally. This chapter looks at the generation of materials databases from DFT simulations and the pros and cons of the high-throughput approach for materials prediction.

3.1 Failing quickly: the high-throughput approach

Failure is the most likely outcome of any endeavor, and nowhere is this more often the case than for attempts to discover new materials. Experimental synthesis of materials is a painstaking and

arduous endeavor that can often feel like guesswork due to the complexity of systems of materials, the non-intuitive nature of submicroscopic interactions over a variety of time and length scales. For example, Edison is reported to have failed 1,000 times before discovering a filament that would keep a light bulb lit for 14.5 hours (Matulka & Wood 2013). The high-throughput approach is a way to fail quickly with less effort.

The materials of primary interest are metals. It is well known that density-functional-theory (DFT) simulation of metals are computationally expensive, but this expense is small in comparison to the experimental methods of synthesis of metals: arc melting (Abu et al. 2014), induction melting (Frenzel et al. 2004), Bridgman solidification (Zuo et al. 2015), powder metallurgy (Torralba et al. 2019), spark plasma sintering (Shen et al. 2002), and mechanical alloying (Benjamin 1976), to name a few. There is a significant advantage, as far as time and effort go, to simulating materials with DFT before attempting to synthesize materials experimentally.

To further reduce the computational effort required to filter out the best candidates of materials for experimental validation, machine-learned models that are even more computationally efficient (about a thousand times faster than DFT) are trained on databases of DFT data (Hart et al. 2021). These surrogate models can be used to make informed decisions about what systems of materials that, when simulated with DFT and added to the database, would make the model more accurate. In summary, machine-learned models inform decisions of what materials to simulate with more expensive DFT, and DFT informs decisions of what materials merit attempts at experimental synthesis. This process is summarized in Fig. 3.1.

3.2 Querying databases of materials

Given a database of materials, the next step is to search through the database for materials often with extreme properties—for example, materials with extreme electrical conductivities or strength.

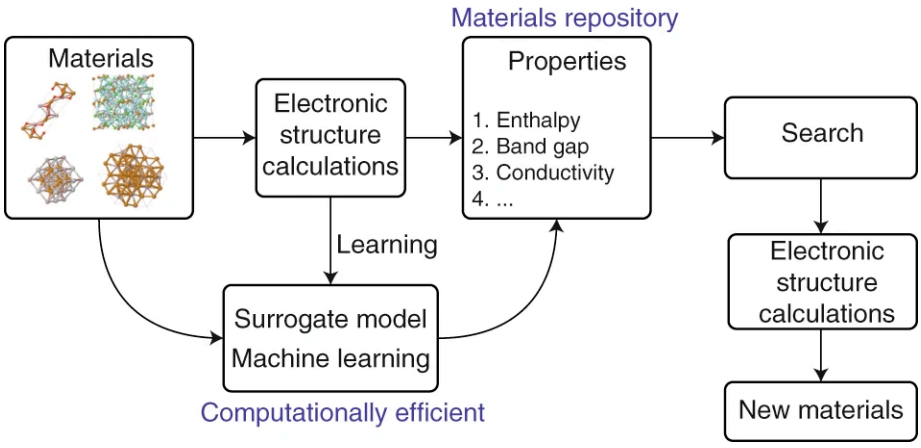


Figure 3.1 Above is a summary of the high-throughput approach to materials discovery (Nyshadham 2019). Instead of running inefficient electronic structure calculations of candidate materials (the upper path) to get properties of the materials, efficient surrogate models (the lower path) trained on electronic structure calculations (density functional theory calculations) can be used to obtain the same properties more efficiently. Once a sufficient number of materials have been computed, an intelligent search is performed to identify materials to be simulated with electronic structure methods. The end result is a list of materials that are very probable to exist to experimentally investigate.

The manner in which a database is searched is by means of a "descriptor."

A descriptor is a variable that relates the microscopic properties of a material (band energy, band gap, density of states, etc.) to a desired macroscopic property (critical temperature, thermal conductivity, magnetic susceptibility, etc.) (Curtarolo et al. 2013). A descriptor can be composed of many microscopic properties, which means there are practically endless possibilities for a valid descriptor. One such descriptor in the literature is the minimum ratio of effective masses of electrons to holes or holes to electrons for characterizing non-proportionality in scintillators (Setyawan et al. 2009). One advantage of machine-learned models is they can automatically select the optimal descriptor for a macroscopic property based on correlations in the materials database (Imbalzano et al. 2018).

In the high-throughput approach, the search for optimal materials is not limited to the materials contained in the database. A suitable descriptor may recommend simulations to be performed to augment the database and thereby improve the accuracy and efficiency of the machine-learned model. In other words, the algorithm may identify optimal materials for a given property from a certain descriptor that may not be in the database. Algorithms that implement this type of extrinsic search for materials include support vector machines (Sundararaghavan & Zabaras 2005) and neural networks (Sha & Edwards 2007). The automatic selection of a descriptor allows a less-informed scientist to perform the search, although including greater physical insight into the factors that affect a given material property is advantageous.

3.3 Generation of databases of materials

Reproducibility is a hallmark of good science that it is no less important in the high-throughput approach to materials discovery. To this end, it is important to understand all the parameters that go into the DFT simulations that are used in the generation of the database, knowledge of which

permits the scientist to know, to some extent, the limitations of the data and what can be reasonably predicted. If it is known that a property of a material depends on a subatomic interaction that was not taken into account in the DFT simulation of the material, it is unreasonable to build a machined-learned model or to perform a search over the database for materials with the property. For example, one would not want to search for optimal magnetic properties in a database of DFT calculations that did not use appropriate exchange-correlation functionals or ones that depend on the magnetic field or paramagnetic current density (Grayce & Harris 1994).

DFT codes have become very large and complicated after hundreds of scientist-years of development, and have grown in proficiency at addressing many different physical, material, chemical, and biological problems (Pribram-Jones et al. 2015). Consequently, every DFT simulation takes hundreds of parameters, and it is these parameters that control the level of theory and the precision of the simulation. To simplify databases, scripts that run DFT calculations will often use the same input parameters for all the simulations in a materials database (Calderon et al. 2015; Jain et al. 2013). These parameters are stored and are accessible along with the results of the simulations.

3.4 Database generation with fixed parameters

Often, very accurate DFT calculations require tailoring the parameters of the calculation to a single, specific material. The ability to select the algorithmic parameters ideal for a given system is lost when many systems of materials are simulated using the same input parameters. The properties of materials often vary in extremes, and the set of parameters that are ideal for one system may not be ideal for another, which may lead to either inefficiencies in the computation or inaccuracy in the results.

Ideally, a DFT calculation is as accurate as possible and requires as little computational effort as possible. DFT calculations are generally accepted to be no more accurate than 0.1 meV in the

calculation of formation enthalpies; increasing the basis-set cutoffs and decreasing convergence tolerances will make the calculation more precise, but these efforts will not be fruitful because the accuracy is limited by theory and not the implementation of DFT. Often, the effects of algorithmic parameters on the precision of various quantities is not well understood, but efforts are being made to understand the relationship between the input parameters and the accuracy of predictions of properties of materials (Herbst et al. 2021).

An example of the effect of input parameters on the accuracy of DFT computations came from a study on smearing and tetrahedron methods in DFT (Jorgensen & Hart 2021). It is commonly believed that smearing fixes the slow convergence issues in the self-consistent cycle for metals, but this simply is not the case; smearing may reduce the number of self-consistent cycles, but it often decreases the accuracy of the DFT calculation, which subsequently decreases the accuracy of machined-learned models trained on the data. Smearing is hazardous because what is considered a large amount of smearing is dependent on the system. It was found in the study that smaller smearing parameters are optimal for high-throughput DFT calculations so that the likelihood of using too much smearing is minimized.

Often, many of the default values of input parameters are used for high-throughput DFT calculations, but there is no guarantee that the default values will be ideal for simulations of diverse materials. One outcome of the smearing parameter study was a reduction of the default smearing value in the DFT software package VASP. In hindsight, it is apparent that the value of the default smearing parameter used in the generation of the DFT database AFLOW is too large at 0.1 eV per atom for metals (Calderon et al. 2015). According to the results in the smearing study, the default smearing value for many DFT codes is too large for general metallic systems.

Just as one material may require a more stringent convergence criterion than another, one material may require a level of theory more advanced than another due to the presence of exotic subatomic interactions. High-through calculations often have to compromise between the level of

theory and computational cost or, in other words, accuracy and efficiency. For example, spin-orbit coupling is not included in DFT calculations in the AFLOW database because of computational expense; molecular dynamics based relaxations are also not performed (Calderon et al. 2015).

Chapter 4

The basics of density functional theory

4.1 Praise for density-functional theory

Every year over the past decade tens of thousands of papers using density-functional theory (Hohenberg & Kohn 1964; Kohn & Sham 1965) (DFT) have been published. Remarkably, 2 of the top 10, and 12 of the top 100 most cited papers relate to DFT (Van Noorden et al. 2014). The ubiquity of DFT stems from its ability to address a diverse portfolio of physical, material, chemical, and biological problems, including inorganic crystal structure (Brandenburg & Grimme 2013; Hautier et al. 2010; Woodley & Catlow 2008), band gaps (Morales-García et al. 2017; Verma & Truhlar 2017; Xiao et al. 2011), charge transport (Cai et al. 2006; Delgado et al. 2010; Lherbier et al. 2008), corrosion inhibitors (Obot et al. 2015; Verma 2018; Xia et al. 2008), heterogeneous catalysis (Bligaard & Nørskov 2007; Nørskov et al. 2011; 2002), molecular properties and spectroscopy (Neese 2009; Stephens et al. 1994; Zhan et al. 2003), chemistry of transition metals (Connolly & Williams 1983; Cramer & Truhlar 2009; Zhao & Truhlar 2006), phase transformations (Connolly & Williams 1983; Gracia et al. 2007; Vishnu & Strachan 2010), surface structures and properties (Meyer & Marx 2003; Nolan et al. 2005; Nørskov et al. 2011), vibrational frequencies (Scott & Radom 1996; Stephens et al. 1994; Wong 1996), molecular dynamics (Car & Parrinello 1985; Hafner 2008; Kresse & Hafner 1994a) and chemical reactions in solutions and enzymes (Field 2002; Hu & Yang 2008; Mulholland 2005). This broad applicability has enabled DFT simulations to (1) gain utility in academic research and industrial sectors (Eyert et al. 2018) where it has become an established tool in automotive, aerospace, energy, chemicals, electronics, oil and gas, metals and alloys, glass and ceramics, and polymer sectors and (2) grow into the most popular electronic structure and quantum mechanical method (Pribram-Jones et al. 2015; van Mourik et al. 2014).

4.2 The many-body Schrödinger equation

Density-functional theory is an approximation to the many-body Schrödinger equation or Schrödinger's equation for systems with many nuclei and electrons. The many-body Schrödinger equation is complicated because one has to consider all possible pair-wise interactions between electrons and nuclei: electron-electron, electron-nucleus, and nucleus-nucleus interactions. Another complication when trying to solve the many-body Schrödinger equation is called the exponential wall (the complexity of the problem grows exponentially with the size of the system), which makes it impractical to solve the many-body Schrödinger equation for large systems.

Density functional theory is one of many approximations to the many-body Schrödinger equation. Approximations vary in complexity and accuracy; density functional-theory is the most practical and lies somewhere in the middle between accuracy and efficiency. The most difficult part of solving the many-body Schrödinger equation is approximating the Coulomb repulsion between electrons.

There are many ways in which the equation for the many-body Schrödinger equation can be simplified. Many scientific disciplines have specialized units, and quantum mechanics is no different. We can simplify the equation by working in atomic units or expressing quantities in units that have sizes similar to what is being studied, which for quantum mechanics are the electron mass, the elementary charge, the speed of light, Bohr's radius, Planck's constant, and so on.

In the case of solids and molecules, one approximation that significantly simplifies the many-body Schrödinger equation is fixing the positions of the atoms (not allowing the atoms to move). This stands to reason because the masses of the atoms are much larger than the masses of the electrons. Also, the science of X-ray crystallography would not exist and X-ray diffraction would not work very well if the atoms in a crystal moved far from their equilibrium positions. The result of this approximation is the kinetic energy of the nuclei is zero, the Coulomb repulsion between nuclei is constant, and the many-body Schrödinger equation no longer depends on nuclear coordinates. A related simplification it called the Bohr-Oppenheimer approximation which allows the electron

wavefunctions and nuclear wavefunctions to be treated separately.

Another approximation used to simplify the many-body Schrödinger equation is called the independent-electrons approximation. Since electron-electron interactions are the most difficult part of solving the many-body Schrödinger equation, ignoring the Coulomb interaction between electrons simplifies the equation significantly. The result of this simplification is that the many-body wavefunction, which is the solution of the many-body Schrödinger equation, can be written as the product of the single-particle wavefunctions, also called orbitals. There is one single-particle wavefunction for each electron. It is common to ignore the spin of the electrons, but spin can easily be included when necessary.

In many cases, completely removing the electron-electron interactions is an oversimplification. One method of adding back these interactions is the mean-field approximation where each electron experiences the "average" potential of all the electrons. In this way, the many-body Schrödinger equation remains independent of the coordinates of the nuclei. However, the potential of the electrons depends on the electron density, which is obtained by solving Schrödinger's equation. In other words, there is a system of equations to solve, and the solutions of one equation must satisfy the constraints of the others. The solution to these equations is obtained, in practice, through the self-consistent field method, also known as the Hartree-Fock method. This is an iterative method that adjusts the one-electron wave functions until the electron density no longer changes (to within a tolerance) or the cumulative energy of the electrons no longer changes.

Due to the anti-symmetry of electrons (the Pauli exclusion principle), the solution to the many-body Schrödinger equation with the fixed nuclei and independent electrons approximations can be written as a Slater determinant, which is a way of enforcing antisymmetry on the one-electron wavefunctions. The Hartree-Fock equations, which give the solution to the many-body Schrödinger equation, are obtained when the single-electron wavefunctions are orthonormal and written as Slater determinants and the energy of the system is minimized with respect to variations in the single-

electron wavefunctions. The only addition in the Hartree-Fock equations is a new potential term called the Hartree exchange potential. The Hartree exchange potential is a consequence of Pauli's exclusion principle and is very difficult to evaluate because it is non-local and requires integration over an addition variable. Whereas the mean-field approximation is a classical treatment (because the average potential was introduced into the many-body Schrödinger equation using equations from classical electrostatics), Hartree-Fock is considered a quantum treatment of a quantum system.

Two more modifications are made to the many-body Schrödinger equation before it takes the form of the equation solved in density-functional theory: simplifying the non-local Hartree exchange potential to a local exchange potential and adding another potential term called the correlation potential. The correlation potential takes into account the fact that electrons are repelled by one another. These two potentials do not have an exact form yet, and it is likely that whoever finds an exact form for these potentials—in other words, whoever is able to perfectly describe all the interactions between submicroscopic particles in materials and molecules—will win a Nobel prize. Considering all these approximations, the many-body Schrödinger equation in atomic units becomes one of the Kohn-Sham equations (Giustino 2014)

$$\left[-\frac{\nabla^2}{2}+V_n(\mathbf{r})+V_H(\mathbf{r})+V_x(\mathbf{r})+V_c(\mathbf{r})\right]\phi_i(\mathbf{r})=\varepsilon_i\phi_i(\mathbf{r}) \tag{4.1}$$

where $-\frac{\nabla^2}{2}$ accounts for the kinetic energy, $V_n(\mathbf{r})$ is the Coulomb potential of the nuclei experienced by the electrons, $V_H(\mathbf{r})$ is the Hartree potential or the average Coulomb potential of the electrons experienced by each of the electrons, $V_x(\mathbf{r})$ is the simpler local-exchange potential, $V_c(\mathbf{r})$ is the correlation potential, $\phi_i(\mathbf{r})$ are the single-electron wavefunctions, and ε_i are the energies of the single-electron wavefunctions. We note that the potentials $V_H(\mathbf{r})$, $V_x(\mathbf{r})$, and $V_c(\mathbf{r})$ come from the Coulomb interactions of electrons with electrons and are separate potentials because of our use of the independent-electrons approximation and replacing the many-body wavefunction with a product of single-electron wavefunctions.

4.3 Self-consistency in density-functional theory

Density functional theory is based primarily on theorems developed by Hohenberg and Kohn (Hohenberg & Kohn 1964). One of these theorems states that the ground state energy of a many-body system can be determined from the ground-state electron density (one limitation of the theorem and DFT is it does not hold for excited states; the energy of an excited state has to be obtained from the many-body wavefunction). The proof of the theorem is inferred from other theorems Hohenberg and Kohn also proved: the ground state electron density uniquely determines the potential of the nuclei, the many-body electron wavefunction is uniquely determined by the potential of the nuclei, and the energy of the system is obtained from the many-body wavefunction. Starting with the ground state electron density n, we obtain the energy of the system: $n \to V_n \to \Psi \to E$ (where Ψ is the many-body electron wavefunction).

The ground state energy is a function of the electron density, and the electron density is a function of the real-space coordinates. The term density *functional* theory comes from the ground state energy, whose solution is often the main objective in a DFT calculation, being a functional or a function whose argument is itself a function.

To obtain the total energy of a system in a DFT calculation, the Kohn-Sham equations have to be solved (Giustino 2014):

$$\left[-\frac{\nabla^2}{2} + V_{\text{tot}}(\mathbf{r})\right]\phi_i(\mathbf{r}) = \varepsilon_i\phi_i(\mathbf{r}) \tag{4.2}$$

$$V_{\text{tot}}(\mathbf{r}) = V_n(\mathbf{r}) + V_H(\mathbf{r}) + V_{xc}(\mathbf{r}) \tag{4.3}$$

$$V_n(\mathbf{r}) = -\sum_I \frac{Z_I}{\mathbf{r} - \mathbf{R}_I} \tag{4.4}$$

$$\nabla^2 V_H(\mathbf{r}) = -4\pi n(\mathbf{r}) \tag{4.5}$$

$$V_{xc}(\mathbf{r}) = \frac{\delta E_{xc}[n]}{\delta n}(\mathbf{r}) \tag{4.6}$$

$$n(\mathbf{r}) = \sum_i |\phi_i(\mathbf{r})|^2 \tag{4.7}$$

V_{tot} is the total potential, $V_{xc}(\mathbf{r})$ is the exchange-correlation potential (a potential term that takes into account everything unknown about the potential), Z_I are the atomic numbers of the nuclei, $\mathbf{R}_I$ are the positions of the nuclei, and E_{xc} is the exchange-correlation energy; all other variables are described in Eq. 4.1. The exchange-correlation potential contains all unknown contributions to the potential, and the hope is this term is small compared to all known contributions to the potential.

An approximation to the exchange-correlation potential with an analytic solution to the exchange energy and a simple numerical solution to the correlation energy comes from the study of the homogeneous electron gas model. This model is more complicated than the free electron model where the external potential is zero because it includes the Coulomb repulsion of electrons and has a constant external potential. Applying the homogeneous electron gas to real systems is a technique that has been around for longer than DFT.

The accuracy and efficiency of DFT depends almost entirely on the approximation of the exchange-correlation potential. As such, many approximations and corrections to approximations have been developed (Sousa et al. 2007). The local spin density approximation (LSDA) introduces spin-dependence to the local density approximation (LDA) (Dirac 1930). The generalized gradient approximation (GGA) methods make the exchange and correlation depend on the electron density and the gradient of the electron density. The GGA approximations fall into two categories: functionals whose parameters come from fittings to empirical data (Becke 1986) and functionals whose parameters are obtained from principles from quantum mechanics, including correct behavior at limits (Perdew 1986). The meta-GGA approximations introduce dependence of the exchange-correlation energies on higher order gradients of the local electron density or on the kinetic energy density (Becke 1993). Hybrid-GGAs combine the exchange-correlation of GGA with a fraction of exact exchange from Hartree-Fock methods (Becke 1988); the fraction included comes from fits to empirical data. Unlike hybrid-GGA, which starts with GGA and adds a fraction of exact exchange, the hybrid-meta GGA methods start with meta-GGA and depends on the local electron density

and its gradients, the kinetic energy density, and the Hartree-Fock exchange (Adamo & Barone 1998). There are also non-local methods where exchange and correlation depend on integrals of the electron density (Vuckovic & Gori-Giorgi 2017).

Given an exchange and correlation functional, we are in a position to calculate the total energy of a system. This is shown as a flow chart in Fig. 4.1. The elements in the flow chart outlined in red are of most interest to this dissertation. The choice of exchange and correlation functional, the number of *k*-points, and the energy cutoffs for plane-wave expansions are important parameters that affect the accuracy and cost of DFT simulations. The algorithm initially guesses the electron density either at random or possibly from electron densities from isolated atoms, among other methods of preconditioning (Kumar et al. 2020). The total potential is calculated once the one-electron charge density is known (including nuclei, Hartree, exchange, and correlation contributions). The one-electron wavefunctions and energies are then calculated, the charge density is recomputed using the one-electron charge densities, and the total energy of all electrons is subsequently computed. The total or band energy $E_{\mathrm{T}}(\mathbf{k}, \varepsilon_f)$ ($\mathbf{k}$ is the reciprocal-space coordinates or *k*-point and ε_f is the Fermi level) will be explained in greater detail in the next chapter. Self-consistency has been achieved when the change in the total energy from two consecutive charge densities varies by less than a given tolerance $\varepsilon_{\mathrm{tol}}$.

The objective of this manuscript is to showcase a method of reducing the cost of each self-consistent iteration by reducing the number of k-points required to compute the total energy $E_{\mathrm{T}}(\mathbf{k}, \varepsilon_f)$. DFT algorithms allow the user to relax the positions of the atoms in addition to relaxing the electrons (computing the ground-state electron density) and since the total energy is computed every time the atoms or electrons are relaxed, the opportunity for improvements in efficiency are significant.

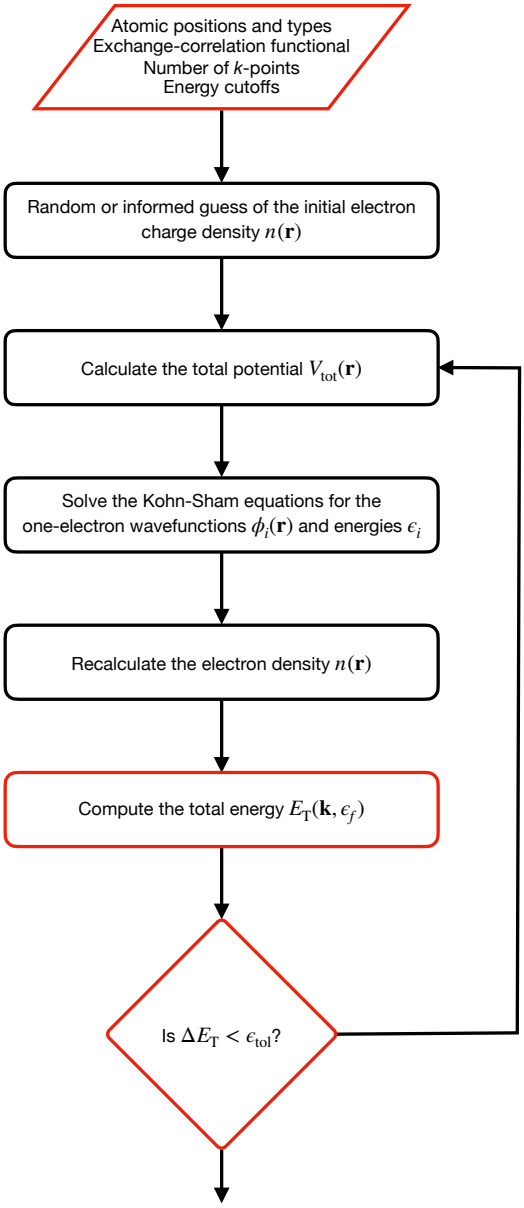


Figure 4.1 A flow chart of the self-consistency cycle performed in density-functional theory codes.

4.4 Time spent solving eigenvalues

Solving for eigenvalues in the self-consistent cycle in DFT codes takes up the majority of the computation time for a metallic system. To verify, a simple test was run of a VASP simulation (a DFT calculation) of a 20-atom alloy CoNiV (15 Co, 1 Ni, 4 V). At each k-point, there is a unique plane-wave expansion of the wavefunction, so the number of terms in the expansion—and hence, the size of the Hamiltonian matrix from which eigenvalues are calculated—is unique but on average, there were 2,400 basis functions. The simulation had 160 symmetrically-unique k-points.

In this back-of-the-envelope calculation, Julia was used to compute the amount of time required to solve for the eigenvalues of a $2,400 \times 2,400$, Hermitian matrix. The functions in the linear algebra library in Julia call functions from the Fortran linear algebra library, so the computation of eigenvalues in Julia and Fortran are very similar.

In Fortran, the eigenvalues are computed using an iterative-matrix diagonalization scheme, so the program doesn't have to compute all the eigenvalues. This is useful because the lowest eigenvalues are the only ones needed. Julia also has an option for calculating only the lowest eigenvalues but it did not result in a reduction in time for the eigenvalue computation. It took about 1 second to calculate the eigenvalues for a $2,400 \times 2,400$, Hermitian matrix in Julia. The amount of time to compute eigenvalues increases by a factor of 12 for a matrix twice as big ($4,800 \times 4,800$) and decreases by a factor of 6 for a matrix half as big ($1,200 \times 1,200$).

When a DFT simulation of a metal is run, the band energy is calculated every time the electrons and ions are relaxed. The simulation of CoNiV had 19 ionic relaxations and, on average, 14 electronic relaxations per ionic relaxation. This means the eigenvalues had to be evaluated at all 160 k-points $19 \times 14 = 266$ times, and the total time spent calculating eigenvalues was over 13 hours. The total time spend on the DFT simulation for CoNiV was 20 hours, so the majority of the DFT simulation of CoNiV was spent on eigenvalue computations.

Chapter 5

How to calculate the band energy

In the last chapter, it was shown that the self-consistent cycle in a density-functional theory calculation will continue until the variation in the total energy is less than a provided tolerance. This chapter looks in greater detail at how to compute the total energy or band energy. Some introductory concepts in solid-state physics will not be explained in detail. A later chapter will explain the concepts of unit cell, Brillouin zone, and irreducible Brillouin zone in detail. Any introductory textbook on solid-state physics or electronic band structure should be a good reference to understanding unfamiliar concepts mentioned in this chapter.

5.1 Integrals over the Brillouin zone

This section covers the general form of integrals of expectation values of single-particle wavefunctions over Brillouin zones whose solutions describe the properties of materials. The goal is to arrive at expressions for the density of states, number of states, and band energy from the general integral form. The information in this section in largely based on the lecture notes by Volker Eyert (Eyert 2012).

Integration over the Brillouin zone generally takes the form

$$G_A(E) = \frac{1}{\Omega}\sum_m \int_\Omega \mathrm{d}^3\mathbf{k}\, \frac{A_m(\mathbf{k})}{E - \varepsilon(\mathbf{k}) + i0^+}, \tag{5.1}$$

where Ω is the volume of the Brillouin zone, m is the energy dispersion index, $A_m(\mathbf{k})$ is the single particle expectation value or generalized spectral density given by

$$A_m(\mathbf{k}) = \langle \psi_m(\mathbf{k}) | A | \psi_m(\mathbf{k}) \rangle \tag{5.2}$$

with $\psi_m(\mathbf{k})$ being the single particle wavefunction, as well as the eigenvectors of the Hamiltonian operator H; $\varepsilon_m(\mathbf{k})$ are the electron energies or the eigenvalues of H; 0^+ is a positive infinitesimal, and $G_A(E)$ is the retarded Green's function of the operator A.

The Dirac identity is commonly employed to split this integral into real and imaginary parts. Dirac's identity is given by

$$\frac{1}{E-\varepsilon_m(\mathbf{k})\pm 0^+} = \frac{P}{E-\varepsilon_m(\mathbf{k})} \mp i\pi\delta(E-\varepsilon_m(\mathbf{k})) \tag{5.3}$$

where P is the Cauchy principal value. Plugging this into the retarded Green's function in Eq. 5.1 gives

$$\begin{aligned} G_A(E) &= \frac{1}{\Omega}\sum_m \int_\Omega \mathrm{d}^3\mathbf{k}\, \frac{A_m(\mathbf{k})}{E-\varepsilon(\mathbf{k})+i0^+} \\ &= \frac{1}{\Omega}\sum_m \int_\Omega \mathrm{d}^3\mathbf{k} A_m(\mathbf{k}) \Big(\frac{P}{E-\varepsilon_m(\mathbf{k})} \\ &\qquad - i\pi\delta(E-\varepsilon_m(\mathbf{k}))\Big) \\ &= \frac{1}{\Omega}\sum_m \int_\Omega \mathrm{d}^3\mathbf{k} A_m(\mathbf{k}) \frac{P}{E-\varepsilon_m(\mathbf{k})} \\ &\qquad - \frac{i\pi}{\Omega}\sum_m \int_\Omega \mathrm{d}^3\mathbf{k} A_m(\mathbf{k})\delta(E-\varepsilon_m(\mathbf{k})) \\ &= R_A(E) - i\pi I_A(E) \end{aligned}$$

where

$$R_A(E) = \frac{1}{\Omega}\sum_m \int_\Omega \mathrm{d}^3\mathbf{k} A_m(\mathbf{k}) \frac{P}{E-\varepsilon_m(\mathbf{k})} \tag{5.4}$$

and

$$I_A(E) = \frac{1}{\Omega}\sum_m \int_\Omega \mathrm{d}^3\mathbf{k} A_m(\mathbf{k})\delta(E-\varepsilon_m(\mathbf{k})). \tag{5.5}$$

The real and imaginary parts are related by Kramers-Kronig's relation

$$R_A(E) = P\int \mathrm{d}E' \frac{1}{E-E'} I_A(E') \tag{5.6}$$

One of the most common Brillouin zone integrals is the one performed calculating the total energy or band energy. In order to calculate the band energy, a separate Brillouin zone integral must be performed to calculate the Fermi level or the highest occupied state. The Fermi level

is determined by integrating the density of states. The density of states can be calculated by substituting $A = \mathscr{I}$, with $\mathscr{I}$ the identity operator, into Eq. 5.5 giving

$$n(E) = I_{\mathscr{I}}(E) = \frac{1}{\Omega}\sum_m \int_\Omega \mathrm{d}^3\mathbf{k}\,\delta(E - \varepsilon_m(\mathbf{k})). \tag{5.7}$$

The total charge Q is given by

$$Q = \int \mathrm{d}E\, n(E)\theta(E - \varepsilon_f) \tag{5.8}$$

where θ is the Heaviside step function, and ε_f is the Fermi energy. In practice, the total charge Q is given and the candidate Fermi level is adjusted by means of a root-finding algorithm until the total charge is obtained to within a sufficient tolerance. Once the fermi energy is calculated, the energy weighted density of states is calculated by substituting $A = H$ into Eq. 5.5 giving

$$n(E)E = I_H(E) = \frac{1}{\Omega}\sum_m \int_\Omega \mathrm{d}^3\mathbf{k}\,\varepsilon_m(\mathbf{k})\delta(E - \varepsilon_m(\mathbf{k})). \tag{5.9}$$

The total band energy results from integrating the energy weighted density of states:

$$E_\mathrm{T} = \int \mathrm{d}E\, n(E)E\,\theta(E - \varepsilon_f) \tag{5.10}$$

It is possible to express the integrals in Eqs. 5.6 and 5.5 in terms of surface integrals with the following substitution:

$$\mathrm{d}^3\mathbf{k} = \mathrm{d}S\mathbf{u}_\perp \cdot \mathrm{d}\mathbf{k}_\perp \tag{5.11}$$

where S is a constant energy surface and $\mathbf{u}_\perp$ is a unit vector perpendicular to a local element of S. The vector $\mathbf{u}_\perp$ takes the form

$$\mathbf{u}_\perp = \frac{\nabla_\mathbf{k}\varepsilon_m(\mathbf{k})}{|\nabla_\mathbf{k}\varepsilon_m(\mathbf{k})|}. \tag{5.12}$$

Plugging this into Eq. 5.11 yields

$$\begin{aligned}\mathrm{d}^3\mathbf{k} &= \mathrm{d}S\mathbf{u}_\perp \cdot \mathrm{d}\mathbf{k}_\perp \\ &= \mathrm{d}S\frac{\nabla_\mathbf{k}\varepsilon_m(\mathbf{k})}{|\nabla_\mathbf{k}\varepsilon_m(\mathbf{k})|} \cdot \mathrm{d}\mathbf{k}_\perp \\ &= \frac{\mathrm{d}\varepsilon_m(\mathbf{k})\mathrm{d}S}{|\nabla_\mathbf{k}\varepsilon_m(\mathbf{k})|}.\end{aligned} \tag{5.13}$$

Plugging Eq. 5.13 into Eq. 5.16 gives a new expression for the density of states in terms of surface integrals

$$\begin{aligned} n(E) &= I_{\mathscr{I}}(E) \\ &= \frac{1}{\Omega}\sum_m \int \mathrm{d}\varepsilon_m(\mathbf{k}) \int_S \frac{\mathrm{d}S}{|\nabla_{\mathbf{k}}\varepsilon_m(\mathbf{k})|}\,\delta(E-\varepsilon_m(\mathbf{k})). \end{aligned} \tag{5.14}$$

An expression for the energy weighted density of states in terms of a surface integral results from plugging Eq. 5.13 into Eq. 5.9

$$\begin{aligned} n(E)E &= I_H(E) \\ &= \frac{1}{\Omega}\sum_m \int \mathrm{d}\varepsilon_m(\mathbf{k}) \int_S \frac{\mathrm{d}S}{|\nabla_{\mathbf{k}}\varepsilon_m(\mathbf{k})|}\,\varepsilon_m(\mathbf{k})\delta(E-\varepsilon_m(\mathbf{k})). \end{aligned} \tag{5.15}$$

Expressions for the density of states, total energy, and other intrinsic material properties in terms of volume and surface integrals are common in the literature. The expression for the total charge and band energy in terms of surface integrals are easier to work with in practice because they are easier to interpret geometrically.

5.2 Mathematical description of the band structure in 2D

The electronic band structure (or simply band structure) of a crystalline material is traditionally thought of as the set of eigenvalues of the one-electron Hamiltonian matrix $H(k_x,k_y)$ over the reciprocal unitcell or irreducible Brillouin zone (IBZ), and often visualized over paths of high symmetry in "spaghetti plots", one of which is shown in Fig. 5.1. Another way of thinking of the band structure S is as an algebraic surface given by the polynomial equation $F(k_x,k_y,\varepsilon)=0$ where $F(k_x,k_y,\varepsilon)$ is the characteristic polynomial of the $r\times r$ Hamiltonian matrix $H(k_x,k_y)$. Topologically, S is comprised of r infinite real sheets that are each single-valued in ε. Denote $S_i = \{(k_x,k_y,\varepsilon_i) \mid \varepsilon_i \text{ is the i}^{\text{th}} \text{ eigenvalue of } H(k_x,k_y)\}$ and, overloading the notation, $S_i(\varepsilon) = \{(k_x,k_y) \mid (k_x,k_y,\varepsilon) \in S_i\}$.

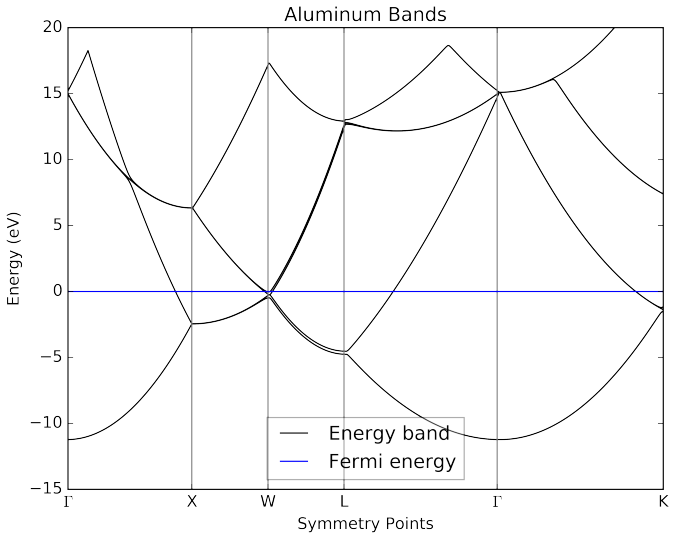


Figure 5.1 A "spaghetti plot" of the electronic band structure of Al over lines of high symmetry throughout the Brillouin zone. Points of high symmetry are labeled across the x-axis.

5.3 Calculating the band energy of periodic materials

In DFT calculations, the self-consistency cycle continues until the total energy of the system between subsequent iterations is less than a provided tolerance. Many physical phenomena contribute to the total energy of a metal, but the one that requires the most computational resources and has the greatest amount of numerical noise and error is the band energy. Calculating accurate band energies leads to more iterations of the self-consistency cycle and longer DFT calculations. In this section we describe how to compute the band energy and the difficulties of the computation for metallic materials in 2D.

The first step in computing the band energy is determining the highest occupied energy state or Fermi level. One method of computing the Fermi level is by integrating the density of states up to a given energy cutoff and adjusting the cutoff until the expected number of states is found, which is a given for any configuration of a material. The electronic density of states can be written as a double

integral (in 2D),

$$n(\varepsilon) = \frac{1}{\Omega}\sum_m \int_\Omega \mathrm{d}^2\mathbf{k}\,\delta(\varepsilon - \varepsilon_m(\mathbf{k})), \tag{5.16}$$

where n is the electronic density of states, ε is energy, Ω is the area of the Brillouin zone, m is the sheet index, $\mathbf{k}$ is the reciprocal lattice vector, and $\varepsilon_m(\mathbf{k})$ is the energy of the mth sheet at position $\mathbf{k}$. The density of states can equivalently be written as a line integral (surface integral in 3D) (Eyert 2012)

$$n(\varepsilon) = \frac{1}{\Omega}\sum_m \int \mathrm{d}\varepsilon_m(\mathbf{k}) \int_{\sigma_m} \frac{\mathrm{d}\sigma_m}{|\nabla_\mathbf{k}\varepsilon_m(\mathbf{k})|}\,\delta(\varepsilon - \varepsilon_m(\mathbf{k})) \tag{5.17}$$

where σ_m is a level energy curve on the mth sheet $\sigma_m \in S_m(\varepsilon) = 0$ (when S_m is the mth sheet of the algebraic surface). The total charge Q within the Fermi surface is given by

$$Q = \int \mathrm{d}E\, n(E)\theta(\varepsilon_f - E) \tag{5.18}$$

where θ is the Heaviside step function. In practice, the total charge, or the number of valence electrons of a system, is used to calculate the Fermi level, and the energy for the Fermi level is correct when it yields a total charge that is the same as the total charge given. The band energy is given by

$$\varepsilon_\mathrm{B} = \int \mathrm{d}\varepsilon\, n(\varepsilon)\varepsilon\,\theta(\varepsilon_f - \varepsilon), \tag{5.19}$$

or equivalently (by taking advantage of the coarea formula, which relates integrals over open sets to level sets)

$$\varepsilon_\mathrm{B} = \sum_m \int_{\varepsilon_m(\mathbf{k})\in\sigma_m} \mathrm{d}^2\mathbf{k}\,\varepsilon_m(\mathbf{k}) \tag{5.20}$$

where σ_m is restricted to energies below the Fermi level. These concepts are illustrated in 2D in Fig. 5.2.

Two numerical integrals have to be solved to calculate the band energy: one gives the Fermi level and the other, which depends on the Fermi level, gives the band energy. The band energy is more difficult to calculate than might be expected because:

1. At every k-point, the time-independent Schrödinger equation is solved. This means solving eigenvalue problems where the order of the matrices is the same as the number of basis functions. In a plane wave basis, the number of basis functions is on the order of hundreds to thousands.

2. The band structure of crystalline materials is multi-valued, there is no inherent order to the eigenvalues, and it is not possible to connect the sheets of neighboring k-points such that all sheets have continuous derivatives. This is usually solved by taking the lowest eigenvalues at every k-point as connected, the second lowest as connected, and so on, but sorting eigenvalues introduces sharp features in the energy sheets along which derivatives are discontinuous, as shown in Fig. 5.2.

3. Because the Fermi level is computed numerically and gives the shape of the Fermi "shadow" or region of integration for the band energy, there is some uncertainty in the region of integration in Eq. 5.20.

4. The region of integration, the projection of the sheets below the Fermi level onto the $k_x \times k_y$ plane or shadows of the sheets, takes on complicated shapes, as shown in Fig. 5.2. When sheets intersect at the Fermi level (sheet crossings), the shadows have sharp corners.

5.4 Methods of sampling the Brillouin zone

Early in the history of density functional theory calculations, people were restricted in the number of k-points they could afford in DFT calculations because of computational limitations of their time. Baldereschi provided a means of selecting a single k-point where the best estimate of the band energy was expected for various crystal structures. Later, generalizations of what came to

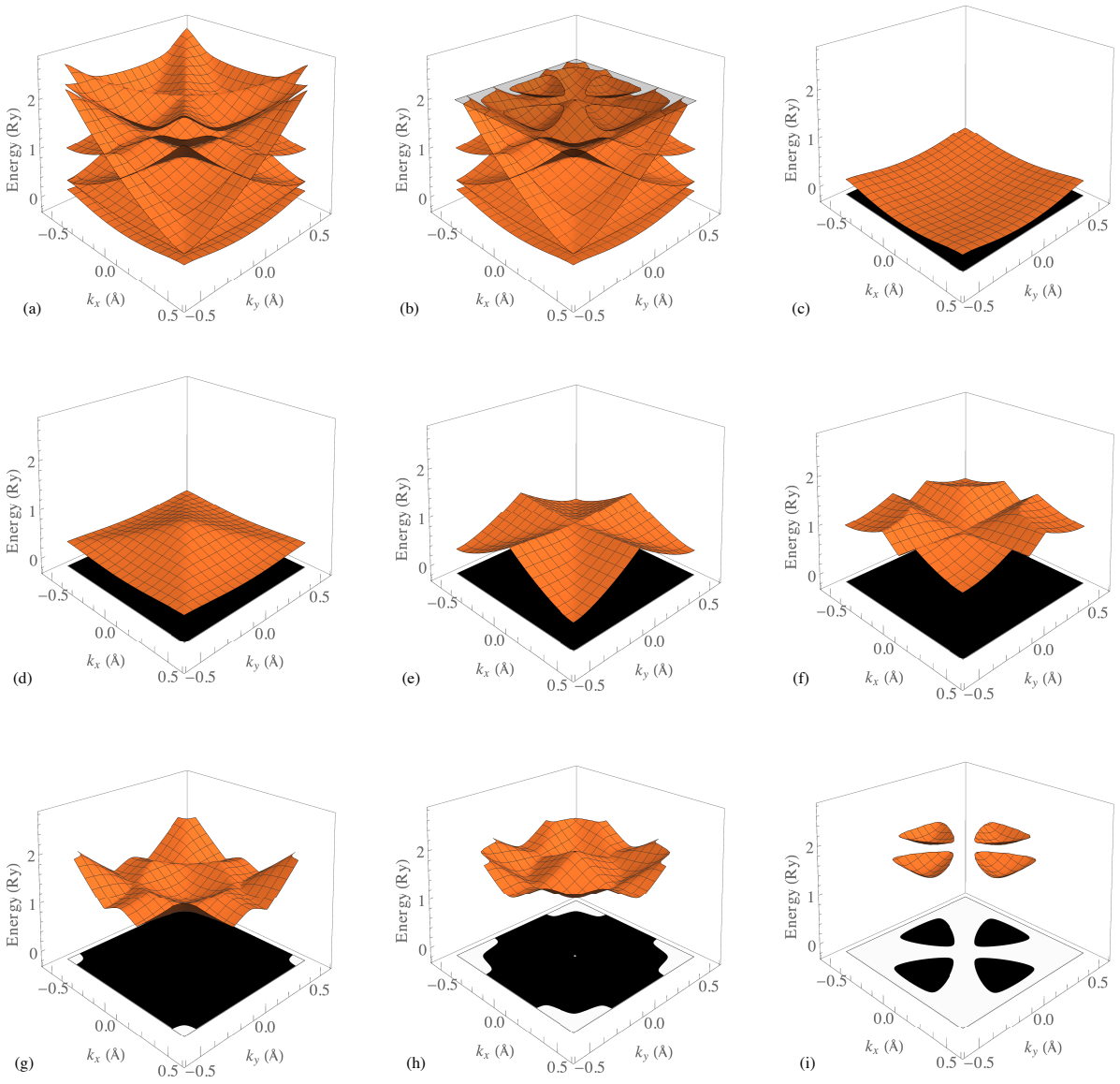


Figure 5.2 An illustration of concepts related to the band energy calculation. In (a) is shown the band structure S of a toy empirical pseudopotential model. In (b) is shown only the the band structure beneath the Fermi level. The Fermi level is the energy coordinate of the gray plane. In (c)-(i) the band structure has been separated into sheets. The shadows of the sheets, in the $(\mathbf{k}_x \times \mathbf{k}_y)$-plane, in (c)-(i) represent the integration domain in the band energy calculation. The sum of the areas of the shadows is the total charge. Creases in the sheets in (e) and (f) result from degenerate eigenvalues. The band structure has discontinuous derivatives along the creases.

be called the *Baldereschi point* came from Chadi and Cohen that included sets of points (Chadi & Cohen 1973) that could be expanded as large as required. Monkhorst and Pack would generalize both Baldereschi and Chadi and Cohen in what would become the standard method of selecting k-points in DFT computations (Monkhorst & Pack 1976). There is nothing special about Monkhorst-Pack grids; they are the same as the regular grids commonly found in finite element methods. Recently, generalizations of Monkhorst-Pack (MP) grids (namely, generalized regular grids or Fourier quadrature points (Froyen 1989; Moreno & Soler 1992; Wisesa et al. 2016)) have show up in the literature. These grids often have greater symmetry and uniformity than MP grids, which leads to a smaller number of reduced points and greater computational efficiency of DFT calculations (Morgan et al. 2018). Special point methods are used in conjunction with the rectangular method to calculate the band energy.

The reason the existence of a Fermi surface makes metals inefficient can be interpreted in two ways: it removes the periodicity of the integral by restricting the integration to complicated subdomains of the Brillouin zone, or it introduces discontinuities in the band structure below the Fermi level. The rectangular integration method requires both smoothness and periodicity for fast convergence. In the restriction of the integration domain interpretation, the rectangular method poorly approximates the shadows, shown in Fig. 5.4, resulting in jagged edges along the border and large integration errors. In the discontinuity interpretation, the Fourier coefficients of a Fourier expansion of the band structure no longer rapidly drop to zero with increasing terms in the expansion.

The rectangular method also calculates the Fermi level inefficiently. The Fermi level is calculated with the rectangular method by sorting the discrete energies from all the sampling points of a regular grid over the Brillouin zone, and identifying the highest occupied energy from the number of k-points and valence electrons. This is shown in Fig. 5.3.

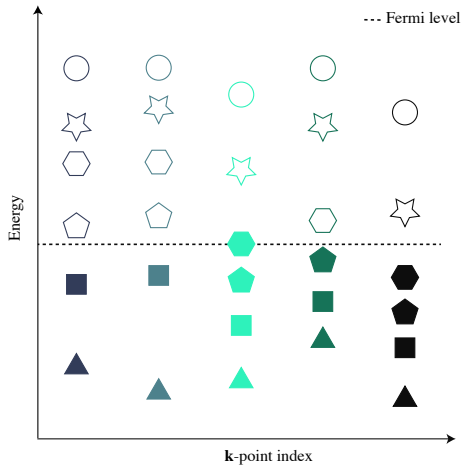


Figure 5.3 An illustration of the Fermi level calculation in metals with the rectangular method. In the figure, all the shapes of a given color represent the spectrum of energies at the same k-point. The triangles are the lowest eigenvalues of each k-point, the squares the second lowest, and so on. Hollow shapes are unoccupied states and shaded are occupied. The Fermi level is calculated by occupying the lowest energy states until there are no more electrons. In this example, there are 5 k-points, 6 valence electrons, and $6/2 \times 5 = 15$ occupied states of the available 60 states that were calculated (at any k-point there are an infinite number of states). The energy of the highest occupied state is the Fermi level.

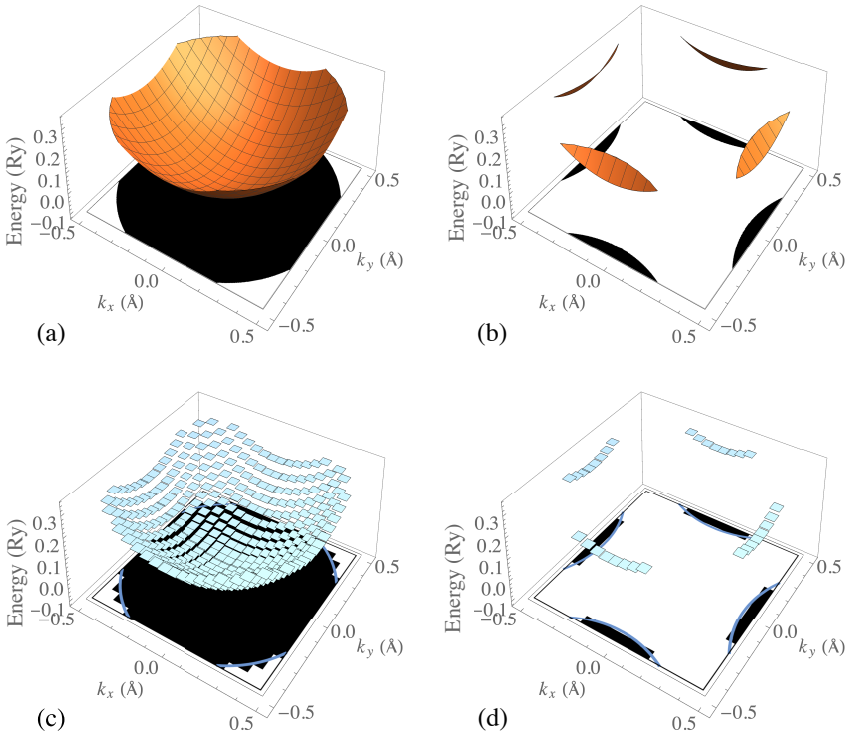


Figure 5.4 The rectangular integration method applied to a 2D free electron model with two electrons. In (a)-(b) are shown the first two sheets along with their shadows. In (c)-(d) is shown the piece-wise constant approximation of the first two sheets, as well as the shadows of the the sheet approximations. The blue curves in (c) and (d) in the $(\mathbf{k}_x \times \mathbf{k}_y)$-plane are the true boundaries of the shadows. As can be seen, the rectangular method both poorly interpolates the two sheets and their shadows.

5.5 Pitfalls of smearing methods for metals

Smearing the band structure at the Fermi level is a way of handling charge sloshing (Marzari 1996; Woods 2018) or speeding up the band energy convergence for metals (Methfessel & Paxton 1989), and is frequently used in DFT calculations of metals. Although smearing may help keep the forces on the atoms from changing abruptly during atomic relaxations (Kratzer & Neugebauer 2019), there is no observed improvement in the band energy convergence or reduction in the number of self-consistency cycles when adding smearing to DFT calculations of metals. One possible reason smearing does not improve the band energy convergence of metals could be due to error in the location where smearing should occur or the Fermi level.

Our tests of smearing on total energy convergence for metals, one of which is shown in Fig. 5.5, show that smearing rarely yields improved total energy convergence. When smearing does show improved convergence, it is often limited to intervals with small numbers of k-points and the smearing type and amount that show improved convergence for one metal often show worse convergence than or the same convergence as no smearing for other metals.

5.6 Higher-order methods of calculating the band energy

A host of higher-order interpolation and extrapolation methods for performing band structure integrations are found in the literature (Gilat 1972). These tend to focus on integrals that are more general than the band energy, such as the density of states and phonon spectra, but are applicable to band energy calculations.

The discrete methods (Blackman 1937; Brust 1965; Gilat & Dolling 1964) aim to increase the density of a grid without increasing the number of eigenvalue problems solved. This is accomplished by performing Taylor expansions about each k-point in a sparse grid, and approximating the energies within the vicinity of the k-point by the Taylor expansion. The linear and quadratic discrete methods

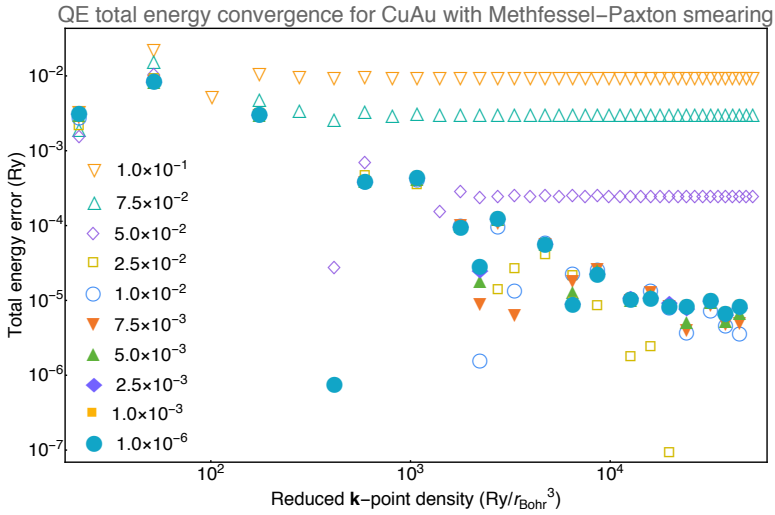


Figure 5.5 The total energy convergence of the metal CuAu with Methfessel-Paxton smearing with varying smearing parameters. The smearing parameters appear in the plot legend in Ry. No amount of smearing consistently shows improvement over no smearing. If the amount of smearing is too much, the total energy converges to the wrong value. Calculations were performed in Quantum Espresso (Giannozzi et al. 2009; 2017).

are named after the number of terms retained in the Taylor expansions.

The aim of the analytic methods (Blöchl et al. 1994; Boon et al. 1986; Gilat & Raubenheimer 1966; Oriade 1980) is not to increase the number of discrete energies, but interpolate the band structure and perform integrations analytically. These include interpolation with linear and quadratic polynomials, and extrapolation with Taylor expansions. Much of the discussion around analytic methods is on how the Brillouin zone, or irreducible Brillouin zone, should be divided into subcells, the degree of the interpolation, how to divide subcells such that the number of resulting subcells is the same as the number of terms in the expansion, and how to make the subcells as spherical as possible. Polynomial interpolation introduces errors at sheet crossings, but extrapolation is reportedly unaffected.

The discrete and analytic methods have been combined in what are called hybrid methods (Cooke & Wood 1972; Methfessel et al. 1983). A Taylor expansion about k-points on a sparse grid is used to effectively produce discrete energies on a dense grid. These energies are then interpolated

and integrated analytically.

Despite the prevalence of higher order interpolation methods in the literature, we are only aware of the tetrahedron method being implemented in DFT codes. Blöchl 's improved linear tetrahedron method is an analytic method that corrects the interpolation to account for errors from the leading term missing in the polynomial expansion (Blöchl et al. 1994). The total energy convergence of metals has been tested using tetrahedron methods and one typical result is shown in Fig. 5.6. The tetrahedron method usually shows slight improvement over the rectangular method in our tests of the error convergence of total energies.

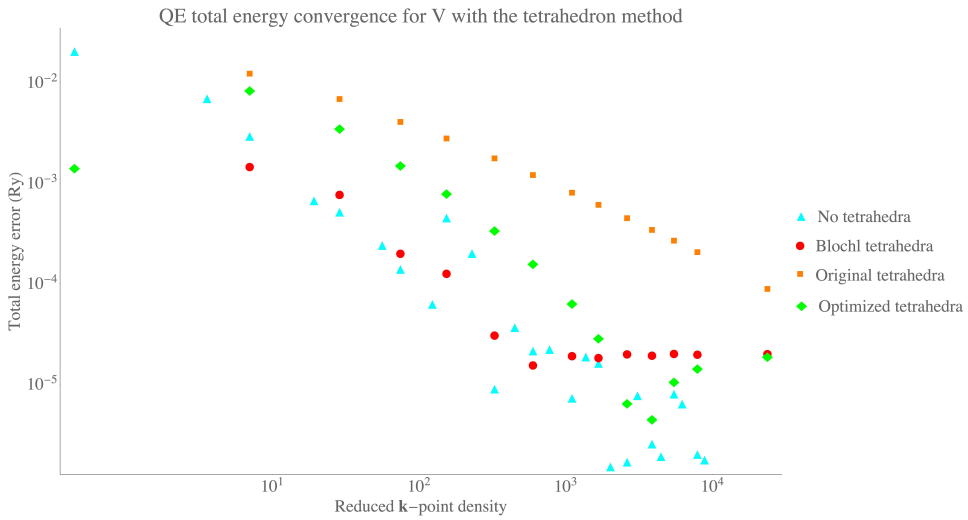


Figure 5.6 The total energy convergence of the metal V with Blöchl 's improved tetrahedron method.

5.7 Connecting the energy sheets

Several papers have looked into connecting the energy sheets in ways that make the individual sheets smooth (André et al. 1972; Chen 1977; Yazyev et al. 2002). Their methods are taking the derivative of the sheets (directly with a Gaussian basis or with $k \cdot p$ perturbation theory in a plane wave basis) or, for a local basis, using the orbital coefficients and overlap matrix to establish a mapping between

energy sheets. One paper on accidental crossings (accidental crossings occur when two sheets intersect away from high symmetry points, lines and planes) argued that crossings away from high symmetry points, lines, and planes are vanishingly improbable (Herring 1937), and that by avoiding locations of high symmetry the problem of connecting of the sheets disappears. Electronic band structures are typically plotted along paths of high symmetry within the IBZ, and contain no information about accidental crossings. For our toy models (empirical pseudopotentials with arbitrary form factors) accidental crossings are prevalent, and avoiding locations of high symmetry does not solve the problems introduced by accidental crossings. One of the simplest models, the free-electron model, has many accidental crossings.

There is interest in connecting the energy sheets as a means of restoring the smoothness of the sheets for reasons of interpolation. When the energies are sorted from least to greatest at each k-point, all crossings become avoided crossings (avoided crossings occur when two sheets come very close together but do not intersect), and the derivatives of the band structure are no longer continuous. If one were to interpolate the energy sheets, there would be large interpolation errors near crossings due to these sharp features being difficult to approximate using polynomial functions of the form $z = f(x,y)$. Difficulties encountered attempting to connect the sheets are shown in Fig. 5.7.

The band structure S is generally an irreducible *algebraic surface*, which means that $F(k_x,k_y,\varepsilon)$ cannot generally be factored into two or more polynomials of lower degree. An algebraic surface S is reducible if its equation, $f(x,y,z) = 0$, can be factored into two equations, $f(x,y,z) = g(x,y,z) \times h(x,y,z)$. Then, S is the union of the two surfaces given by $g(x,y,z) = 0$ and $h(x,y,z) = 0$. The curves along which two surfaces intersect will be curves of self-intersection on S. In other words, a reducible surface *always* has one or more curves of self-intersection. However, as the Riemann surface example shows in Fig. 5.8, an algebraic surface may have a curve of self-intersection but not be reducible. The Reimann surface can be rewritten into the polynomial equation $f(x,y,z) =$

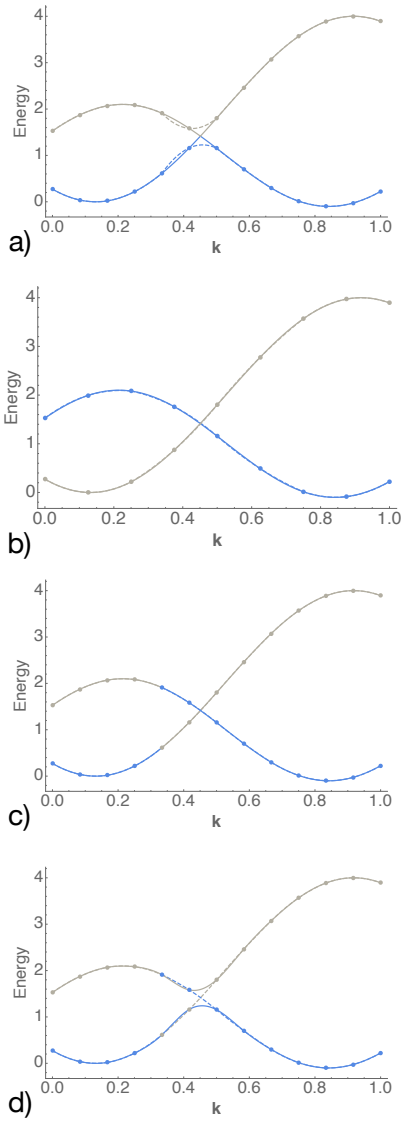


Figure 5.7 Connecting energy sheets. In all plots, the solid lines are the energy sheets and dashed lines are a piecewise quadratic approximation of the energy sheets. The blue and gray curves represent two sheets. In a), two sheets are shown with a crossing. There are large approximation errors where the two sheets intersect. In b) the two sheets have been selected in a way that neither has sharp corners. It may be possible to recognize locations where two sheets cross, and swap the sheets in the interval where they cross. This is shown in c). In d), locating where sheets intersect is complicated by avoided crossings, which, from a coarse sampling, have the appearance of sheet crossings.

$4xz^2 + y^2 - 4z^2 = 0$. This surface has a self-intersection curve along the entire x-axis, but $f(x,y,z)$ cannot be factored. Another example with the electronic structure of Barium is shown in Fig. 5.9.

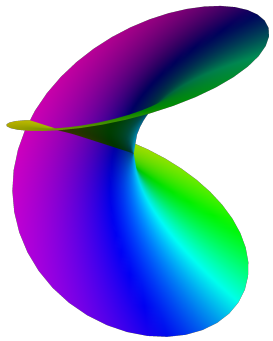


Figure 5.8 The Riemann surface for $f(z) = \sqrt{z}$ where z is a complex number. It is not possible to split the Riemann surface into two sheets that are both continuously differentiable everywhere on the surface. It is an example of an irreducible algebraic surface that has a curve of self-intersection.

5.8 Discontinuities from a finite basis expansion

At each k-point, the wavefunction is represented with a finite number of basis functions. In the case of a plane wave expansion, each reciprocal lattice point represents a term in the basis expansion, and all reciprocal lattice points within a circle (in 2D) centered about the k-point under consideration are kept in the expansion. Two k-points in the Brillouin zone may have different terms in their respective wavefunction expansions due to the fact that the reciprocal lattice points contained in a circle centered about one may not be the same as the lattice points contained in a circle centered about the other. At locations in the Brillouin zone where a term is added or removed from the wavefunction basis, the band structure changes discontinuously. If the radius of the circle is small, there will be fewer terms in the expansion and the size of the discontinuity will be large. As the

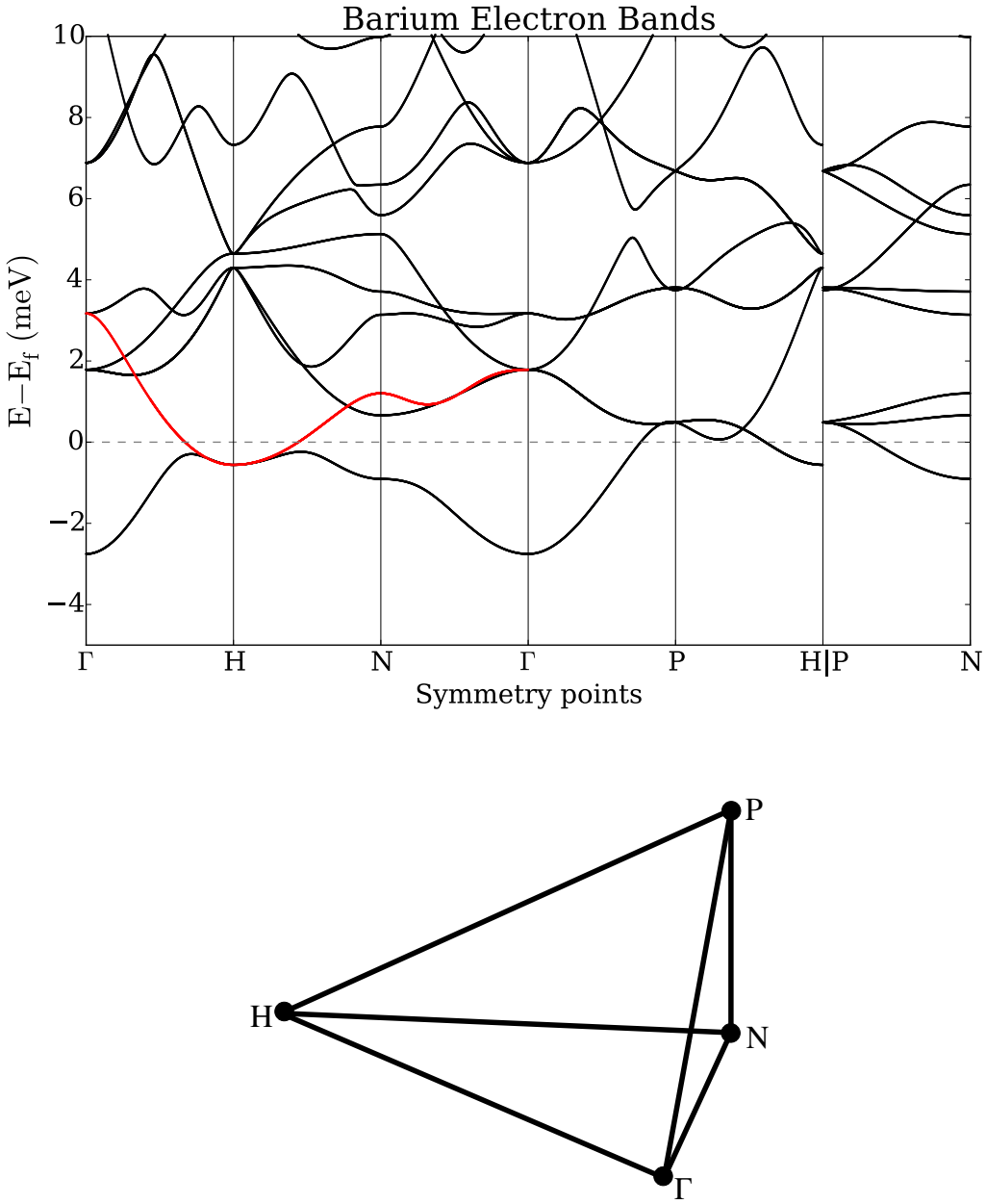


Figure 5.9 The electronic structure of Barium along high symmetry paths. Shown above is the electronic structure and below is the IBZ of Barium. Highlighted in red is a closed path that starts and ends at Γ, but has different energies at the start and finish.

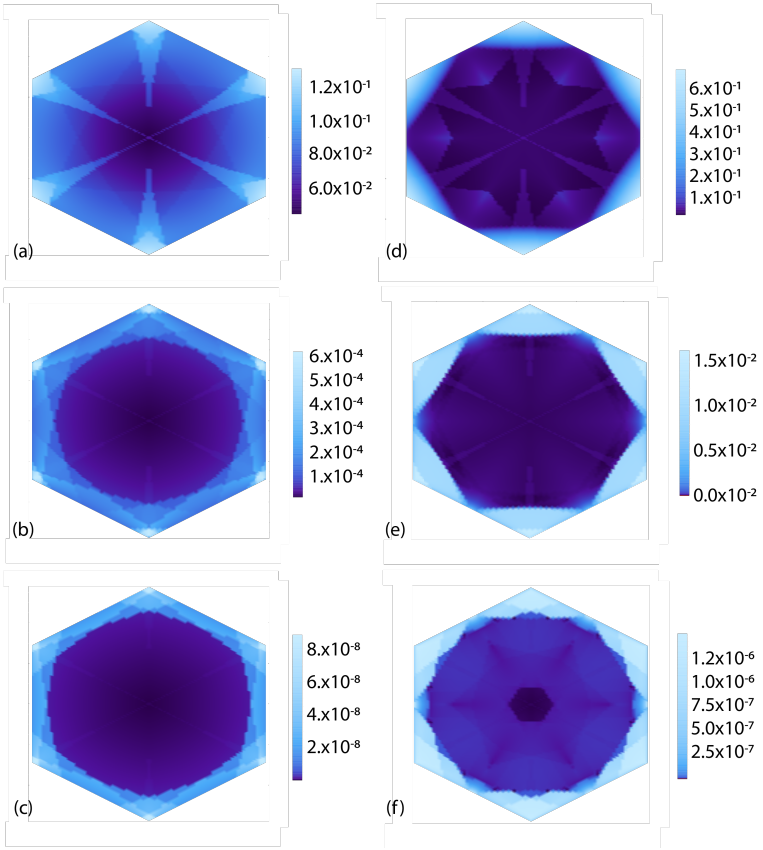


Figure 5.10 A demonstration of discontinuities in the band structure from a finite wavefunction expansion. Shown in the figure is a color map of the error over the Brillouin zone of a toy empirical pseudopotential. In the left column is the error for the first band, and the right column is the error for the sixth band. The number of basis functions in figures (a) and (d), (b) and (e), and (c) and (f) are 19, 61, and 127, respectively. Notice the size of the discontinuities (the scale in the plot legends) drops as the number of terms in the basis expansion increases. The errors in these plots are the difference between a model whose basis is centered about the origin and one centered about each k-point, so the sizes of the discontinuities are exaggerated.

radius of the circle containing all terms in the expansion increases, the size of the discontinuities decrease, which is demonstrated in Fig. 5.10.

5.9 A case for adaptive mesh refinement

There are at least two reasons a uniform sampling of the unit cell can be inefficient at calculating the band energy. The first has to do with how DFT is used in practice. Generally, DFT simulations of materials have an accuracy goal around 1 meV/atom for the total energy, regardless of the metal. The Fermi level plays a large role in how difficult it is to obtain this precision. Consider the functions

$$w_1(x) = \begin{cases} e^{\cos 2\pi x} & \text{if } x \in [a, 1-a] \\ 0 & \text{otherwise} \end{cases} \tag{5.21}$$

$$w_2(x) = \begin{cases} 10 + w_1(x) & \text{if } x \in [a, 1-a] \\ 0 & \text{otherwise} \end{cases} \tag{5.22}$$

where $a = 0.1859$. The function $w_1(x)$ is plotted in Fig. 5.11. Both w_1 and w_2 are integrated over the interval $x \in [0,1]$ with the midpoint rectangular method. The integration domain is not restricted to the interval $x \in [a, 1-a]$ where both functions are continuous because, in practice, the location of the discontinuities is not known. The integration error convergence of both functions is shown in Fig. 5.12. An accuracy of 0.01 requires about 100 sample points for w_1, and almost 10 times as many points are needed to achieve this same accuracy for w_2. The only difference between the two is the height of the discontinuity.

The results in Fig. 5.12 make sense because the errors of the rectangles that straddle the discontinuities are proportional to the heights of the discontinuities. In other words, if the Fermi level (height of the discontinuities) is far from zero, the integration errors of rectangles at discontinuities will have large errors compared to rectangles without discontinuities. If the Fermi level is near zero, the errors of rectangles at discontinuities will be similar to those away from discontinuities. For the 2D quadratic integration algorithm, this means that, when the Fermi level is large, there is a greater concentration of integration error in tiles with where the sheets cross the Fermi plane. Localized errors is a scenario that adaptive mesh refinement (AMR) handles well.

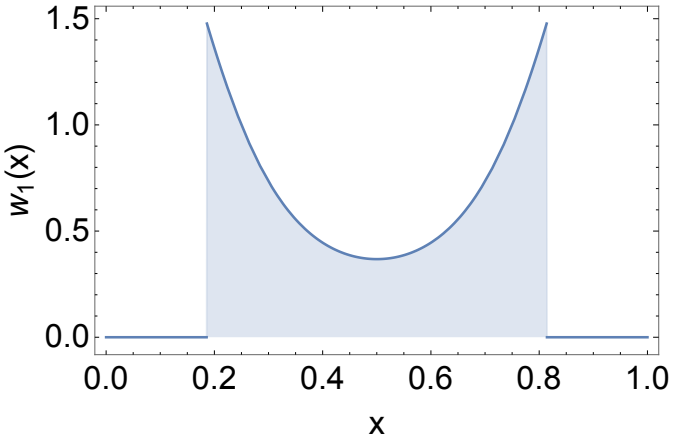


Figure 5.11 A plot of w_1 and the area obtained upon integrating shaded in blue.

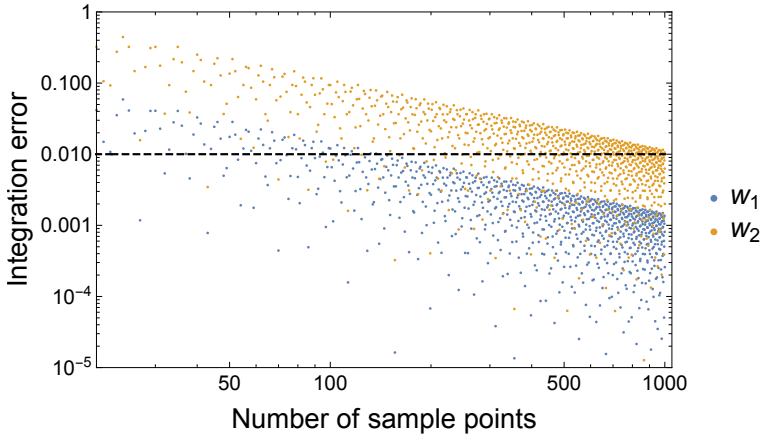


Figure 5.12 The integration error convergence of w_1 and w_2. To guarantee an integration error no greater than 0.01, w_2 requires approximately 10 times the number of sample points.

The second reason uniform mesh refinement is insufficient for band energy calculations is the boundaries of eigenvalue shadows often have fine features that are introduced by non-accidental crossings at or near the Fermi level. As mentioned previously in subsection 5.7, it is not possible to connect the energy sheets such that each entire sheet has greater than C^0 continuity everywhere. Identifying crossings is complicated by the occurrence of avoided crossings. For these reasons, we avoid the problem of connecting sheets by simply sorting the eigenvalues at all k-points. The Fermi curve has sharp corners when two sheets cross at the Fermi level, and the curve along which they cross has non-zero slope in the energy direction. If there is a crossing not at the Fermi level but near it, the energy sheets may have high curvatures, which may cause the borders of the eigenvalue shadows to also have high curvature. Principally, crossings near the Fermi level cause fine features in the eigenvalue shadows, and these fine features are difficult to represent accurately with a coarse sampling of the IBZ. A dense uniform sampling or AMR become necessary to accurately represent these fine features. A few examples of fine features in the eigenvalue shadows are shown in Fig. 5.13.

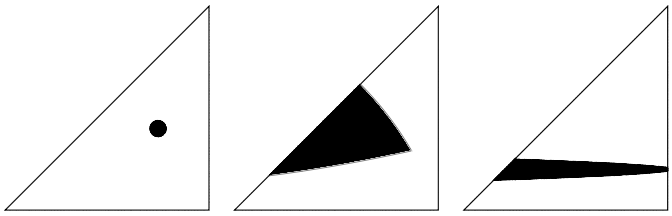


Figure 5.13 Examples of fine features in the eigenvalue shadows. From left to right, these include "islands" when the Fermi level is near a local minimum or maximum of a sheet, sharp corners when there is a crossing at the Fermi level, and thin "strips" where the Fermi level is near where two sheets cross.

Chapter 6

A closer look at smearing and tetrahedron methods

All the material in this chapter comes from a paper that has already been published (Jorgensen & Hart 2021).

For many DFT practitioners, DFT codes are "black boxes". This is understandable because the list of problems DFT can address continues to grow, the codes themselves are increasingly complex, and the ratio of experts to non-experts running DFT is shrinking. Often too little thought goes into the values of input parameters of DFT algorithms, with many opting for default values even though these may lead to inaccurate calculations. Mattsson et al. (Mattsson et al. 2004) have demonstrated the sensitivity of numerically precise DFT results on k-point sampling, basis set cutoffs, and smearing. Mehl (Mehl 2000) draws attention to the fact that many use smearing with little justification or validation and drew attention to cases where smearing leads to inaccurate predictions of a material's properties.

The use case of interest in this paper is computational high-throughput screening of materials (Curtarolo et al. 2012; Jain et al. 2013; Saal et al. 2013). An example of high-throughput screening is identifying stable alloys. To do this, a large number of DFT calculations are performed on a number of elements in varying concentrations and crystal structures. Those with the lowest formation enthalpy at a given concentration are most likely stable and qualify for further investigation. Tens of thousands of DFT calculations are performed without tailoring the parameters of the simulation for each calculation. A set of robust DFT parameters are desired that give numerically accurate results irrespective of the system.

We ran close to 40,000 DFT calculations to identify a smearing method and smearing parameter that would give improved computational performance for a wide variety of materials. We found that smearing reduces the number of self-consistency field (SCF) cycles to a small degree and large amounts of smearing leads to inaccurate DFT calculations. To complicate matters, the optimal smearing is dependent on the system, smearing method, smearing parameter, and k-point density. DFT manuals recommend default smearing parameters that are often too big and lead to inaccurate

calculations.

We also tested the affect of tetrahedron methods and found they play a minor role in improved precision in DFT calculations, with uncorrected tetrahedron methods providing less numerical precision than the non-tetrahedron methods. Among tetrahedron methods, Blöchl performs the best, but is only marginally better than integration without tetrahedra and without smearing.

6.1 Background of smearing and tetrahedron methods

DFT codes calculate the ground state electronic density by means of a self-consistency field cycle. Smearing the electronic structure was proposed to deal with band sloshing (Marzari 1996) and to reduce the number of k-points required by accelerating the convergence of the band energy calculation with respect to k-point density (Methfessel & Paxton 1989). Band sloshing typically occurs when there are many states near the Fermi level that alternately shift above and below the Fermi level in SCF cycles, resulting in the charge moving back and forth, never settling (Tassone et al. 1994; Woods 2018). Tetrahedron methods were also developed to reduce the number of k-points in DFT calculations (Blöchl et al. 1994).

The desired result of smearing or tetrahedron methods is more efficient DFT simulations from reduced SCF iterations or fewer k-points. This work shows that the optimal amount of smearing is usually much lower than what is recommended, and that there is no single, optimal parameter—the effectiveness of smearing is dependent on the metal, k-point density, and smearing method. In general, a smearing value smaller than the recommended value is safer and equally efficient.

DFT calculations of metals take much more time than insulators often because they require much higher k-point densities to achieve the same numerical accuracy. The difficulty with metals stems from their Fermi surfaces. The Fermi surface is a surface separating the occupied regions of the electronic structure from the unoccupied regions. Fig. 6.1 demonstrates the incredible convergence

of an insulator, Si, compared to the considerably worse convergence of a metal, Al.

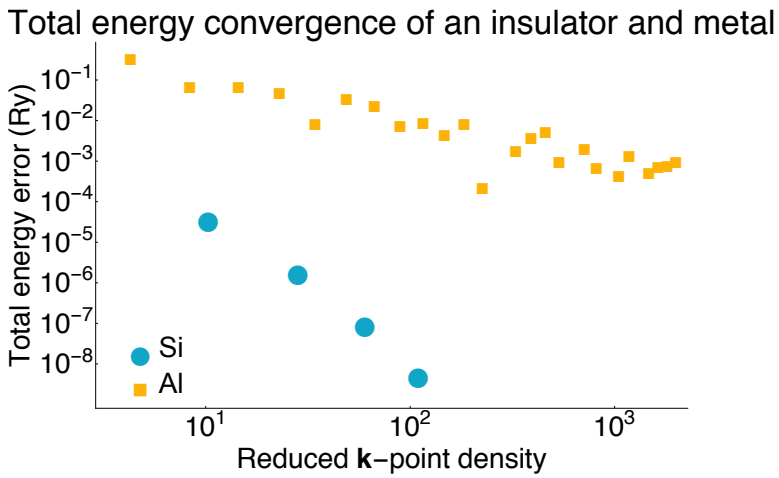


Figure 6.1 A comparison of the convergence of the total energy of a metal, Al, and insulator, Si, with respect to k-point density. The rate of convergence of an insulator is exponential while that of a metal is linear. The differences in convergence rates and the erratic nature of the convergence of metals, leads to metals requiring far more computational resources, especially in circumstances such as high-throughput screening where high numerical accuracy is desired.

The band energy calculation for insulators converges quickly because the occupied bands are smooth and periodic. In Fig. 6.1, we used the rectangular integration method with a Monkorst-Pack grid (Monkhorst-Pack grids are *regular grids* (Morgan et al. 2019)) to calculate the band energy. The expected error convergence from integrating with the rectangular method decreases as $1/N^2$ in 1D where N is the number of points, but there are classes of functions (Weideman 2002) that are smooth and periodic whose convergence rates are much faster, ranging from algebraic, but higher order ($1/N^4$, for example), geometric (r^N, $0 < r < 1$), or exponential (e^{-N}). Insulators lack a Fermi surface and are cases where the rectangular method excels.

Integrals of smooth and periodic functions (functions whose Fourier expansions quickly drop to zero) with the rectangular method converge rapidly because lower order Fourier terms are integrated exactly with rectangles. This is illustrated in Fig. 6.2. The rapid convergence of the Fourier coefficients should be compared to the error convergence with rectangular integration in Fig. 6.3.

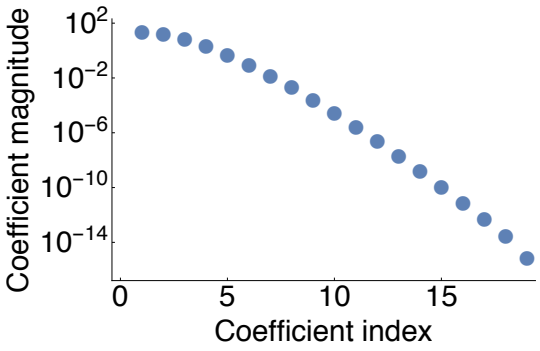


Figure 6.2 The magnitude of the Fourier coefficients in a Fourier expansion of the function $y(x) = e^{2\cos x}$. The magnitude of the 20th term in the expansion is around 10^{-15}, which is proportional to the error in the integral of $y(x)$ over one period with the rectangular method with 20 integration points.

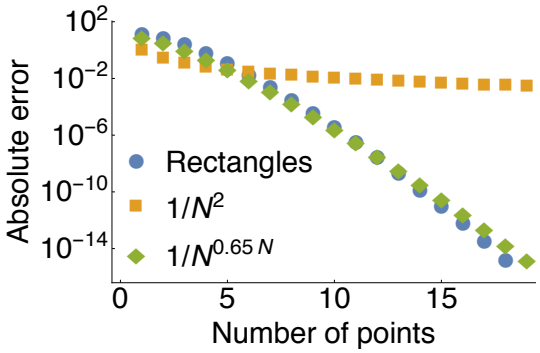


Figure 6.3 A demonstration of the excellent convergence of the rectangular integration method on smooth periodic functions. We integrated the function $e^{2\cos x}$ over one period with varying numbers of integration points N. We show the error convergence with the rectangular method with blue circles, the expected algebraic convergence with orange squares, and a convergence fit with green diamonds.

The smoothness of the energy bands is removed for metals by the introduction of a Fermi level or highest occupied state that does not lie in a band gap (as it does for non-metals). Smearing methods attempt to restore the smoothness of the energy bands and the fast convergence of band energy calculations by smoothing out the discontinuities. See the top of Fig. 6.4

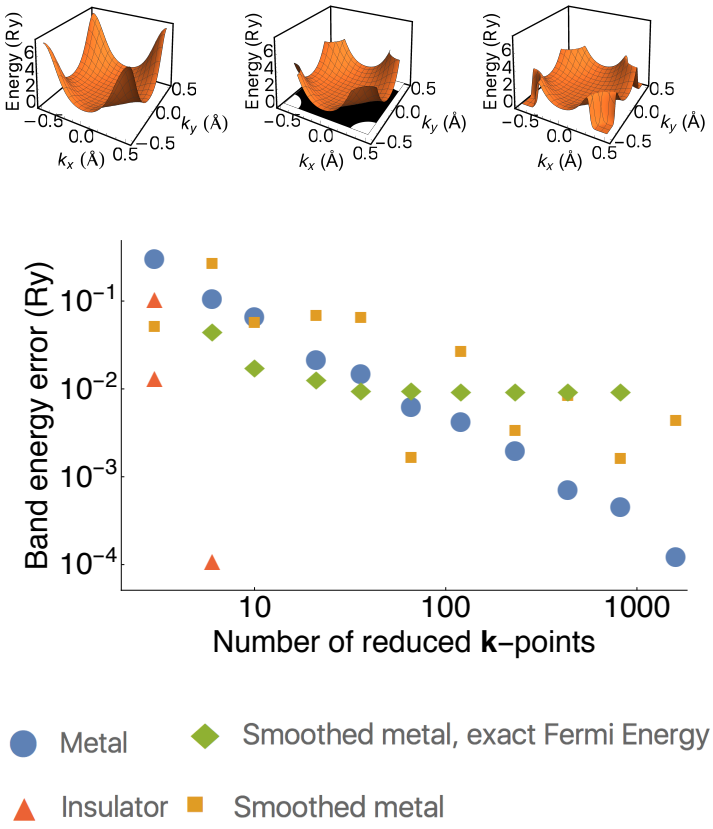


Figure 6.4 A comparison of the band energy convergence of $w_1(\mathbf{k}) = e^{-\cos(2\pi \mathbf{k}_x)-\cos(2\pi \mathbf{k}_y)}$ when treated as an insulator, a metal, a smoothed metal, and a metal with an exact Fermi level that has been smoothed. We show a single sheet of w_1 when treated as (a) an insulator, (b) a metal, and (c) a metal that has been smoothed. Below the band plots is shown their band energy convergences along with the convergence of a metal with an exact Fermi level that has had its discontinuity smoothed. Smoothing shows no improvement in the band energy convergence. Smoothing with an exact Fermi level shows some improvement in band energy error, and eventually the band energy converges to the wrong value at high k-point densities.

The band energy integral can take the following form

$$\sum_n \int_{\mathbf{k}\in\mathbb{U}} \mathrm{d}\mathbf{k}\, E(\mathbf{k},n)\,\theta(E_f - E(\mathbf{k},n)), \tag{6.1}$$

where $\mathbf{k}$ is a point in reciprocal space $\mathbb{R}^3$, n is the band index, $\mathbb{U}$ is the reciprocal unit cell, $E(\mathbf{k},n)$ is the value of the electronic band structure at a given k-point and band, E_f is the Fermi level, and θ is the step function. In this form, the integral is over the entire unit cell, and the integrand is discontinuous, resulting in the poor convergence of metals mentioned above. For insulators, the Fermi level lies in a band gap and the step function has no effect on the integrand.

The approach of Methfessel and Paxton (MP) is to replace the occupation step function with a continuous approximation of it so that the band structure remains smooth and continuous for metals (Methfessel & Paxton 1989). They expand the step function in Hermite polynomials; the expansion integrates polynomials of N-th order exactly. The δ function approximation is

$$\delta(x) = \sum_{n=0}^{N} A_n H_{2n}(x) e^{-x^2}, \tag{6.2}$$

where A_n are expansion coefficients found from the orthogonality of Hermite polynomials, $H_{2n}(x)$ are Hermite polynomials, and e^{-x^2} is a Gaussian weight function. Integrating the delta function they obtained approximations of the step function S:

$$S_0(x) = \frac{1}{2}(1 - \mathrm{erf}(x)) \tag{6.3}$$

$$S_N(x) = S_0(x) + \sum_{n=1}^{N} A_N H_{2n-1}(x) e^{-x^2}. \tag{6.4}$$

These polynomial approximations are shown in Fig. 6.5. The zeroth order approximation of the step function is equivalent to Fermi-Dirac smearing; the higher order terms are corrections. In their paper, MP showed that Fermi-Dirac smearing is only accurate when the integrand is constant near the Fermi level. MP smearing is accurate when the integrand can be represented by a polynomial of degree $2N$ within an interval where the Gaussian weight function in the expansion is appreciably nonzero.

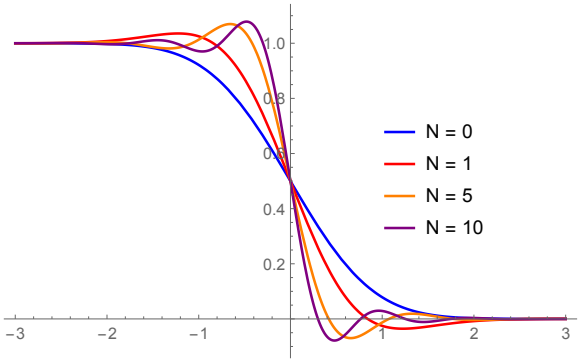


Figure 6.5 Methfessel and Paxton (Methfessel & Paxton 1989) used Hermite polynomial expansions to create smooth approximations to the step function. N is the number of terms in the expansion.

Marzari et al. draw attention to drawbacks to Methfessel-Paxton broadening (Marzari 1996): the thermal distribution loses the property of being monotonic, and the occupation numbers are no longer positive definite (Marzari 1996). As a consequence, the generalized entropy, the steepest descent directions, and the theorems for representing density matrices lose their explicit forms. Marzari's approximation of the delta function is

$$\tilde{\delta}(x) = \frac{2(2-\sqrt{2}x)}{\sqrt{\pi}} e^{-[x-1/\sqrt{2}]^2}, \tag{6.5}$$

where $x = \frac{\mu-\varepsilon}{\sigma}$, μ is the Fermi level, ε is an energy variable, and σ is the electronic temperature or smearing parameter. The benefit of Marzari-Vanderbilt smearing is occupations are positive definite. Lastly, Gaussian smearing is common in DFT codes and takes the form

$$g(\varepsilon) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}((\varepsilon-\mu)/\sigma)^2}, \tag{6.6}$$

where all symbols are the same as those defined in Eq. 6.5.

Tetrahedron methods are an alternative or additional approach to smearing that attempt to improve the poor convergence of metals. Tetrahedron methods split the unitcell or irreducible Brillouin zone into tetrahedra and interpolate the band structure with a linear polynomial within each tetrahedra (Jepson & Anderson 1971; Lehmann & Taut 1972). Integrations are performed

Figure 6.6 Comparison of total energy errors in VASP with and without smearing. The value of the smearing parameter increases down the rows (values in eV are 1.0×10^{-3}, 1.0×10^{-2}, 2.5×10^{-2}, 5.0×10^{-2}, 7.5×10^{-2}, 1.0×10^{-1}, 2.5×10^{-1}, 5.0×10^{-1}, 7.5×10^{-1}, and 1.0×10^{0}). The k-point density gets larger with each column (number of k-points is 3^3, 4^3, ..., 40^3). Smearing in VASP usually does not improve total energy accuracy. Even in Al where smearing results are best, the most likely outcome of smearing is no improvement. The optimal smearing is dependent on the smearing parameter, smearing method, k-point density, and metal.

Figure 6.7 Comparison of force errors in VASP with and without smearing. The value of the smearing parameter increasing down the rows (value in eV are 1.0×10^{-3}, 1.0×10^{-2}, 2.5×10^{-2}, 5.0×10^{-2}, 7.5×10^{-2}, 1.0×10^{-1}, 2.5×10^{-1}, 5.0×10^{-1}, 7.5×10^{-1}, and 1.0×10^{0}). The k-point density gets larger with each column (number of k-points is 3^3, 4^3, ..., 40^3). Smearing improves the accuracies of forces in VASP, but the optimal smearing depends on the smearing parameter, smearing method, k-point density, and metal.

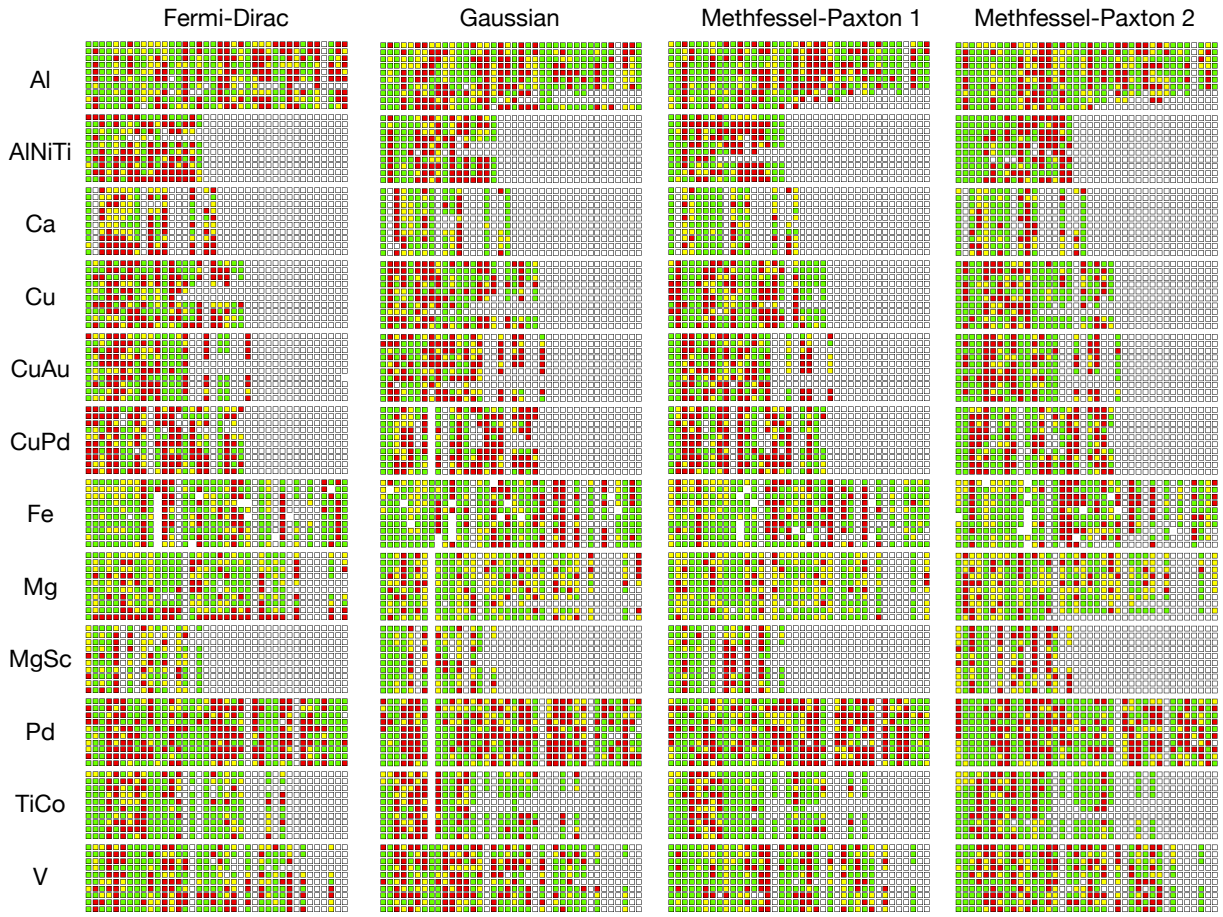


Figure 6.8 Comparison of the number of SCF iterations in VASP with and without smearing. The value of the smearing parameter increasing down the rows (value in eV are 1.0×10^{-3}, 1.0×10^{-2}, 2.5×10^{-2}, 5.0×10^{-2}, 7.5×10^{-2}, 1.0×10^{-1}, 2.5×10^{-1}, 5.0×10^{-1}, 7.5×10^{-1}, and 1.0×10^{0}). The k-point density gets larger with each column (number of k-points is 3^3, 4^3, …, 40^3). Smearing most often decreases the number of SCF iterations in VASP by at least 1 iteration; it less often but frequently increases the number of SCF iterations by 1.

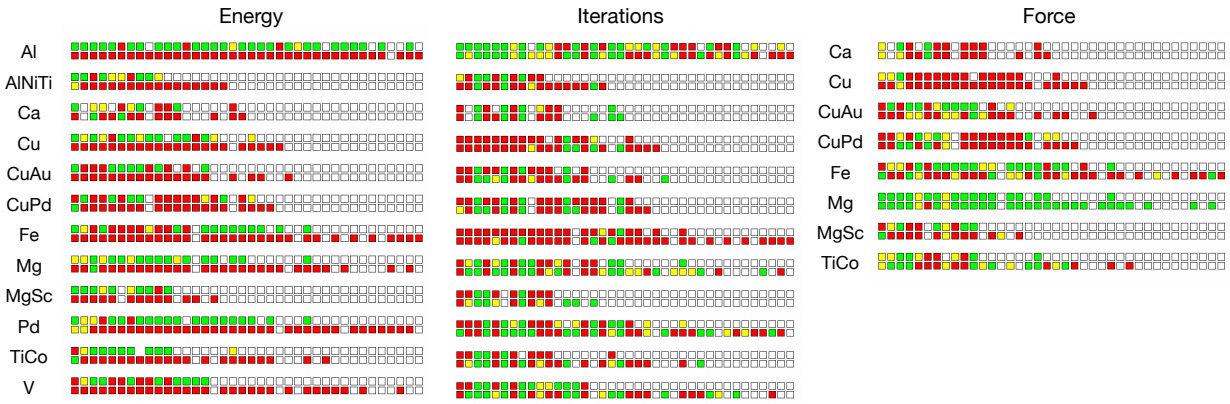


Figure 6.9 The total energies, forces and number of SCF iterations in VASP with tetrahedron methods compared against the same values obtained without smearing or tetrahedra. The tetrahedron methods down the rows are linear and Blöchl's tetrahedra. The *k*-point density gets larger with each column (number of *k*-points is 3^3, 4^3, ..., 40^3). Blöchl's tetrahedron method most often improves the total energy accuracy. Tetrahedron methods may improve or reduce the number of SCF iterations and force accuracies, giving random and mixed results for both. Only for Mg did tetrahedra consistently improved the forces.

analytically within each tetrahedra, and the Fermi surface is approximated by a polygon. Blöchl et al. improved upon the linear tetrahedron method by accounting for errors from missing terms in the polynomial expansion (Blöchl et al. 1994). Kawamura et al. generalized Blöchl's corrections to more general Brillouin zone integrals, including phonon frequency and response functions (Kawamura et al. 2014).

6.2 Testing smearing and tetrahedron methods

In order to test if smearing and tetrahedron methods reduce the number of SCF cycles, improve forces on atoms, or reduce the *k*-point density required for a given accuracy, we ran close to 40,000 total DFT calculations in Quantum Espresso (QE) (Giannozzi et al. 2009; 2017) and VASP (Kresse & Furthmüller 1996; Kresse & Hafner 1993). VASP was chosen because of its popularity and QE because it is open source. In all calculations, the structural degrees of freedom were fixed.

In QE we tested 12 metallic systems (Al, AlNiTi, Ca, Cu, CuAu, CuPd, Fe, Mg, MgSc, Pd, TiCo, and V), and compared the total energy convergence, stress convergence, and number of SCF cycles for different smearing and tetrahedron methods. We used QE version 6.3 for all calculations. The smearing methods we tested were Fermi-Dirac, Gaussian, Marzari-Vanderbilt, and Methfessel-Paxton smearing. Each smearing method was tested with 10 different smearing parameters, ranging from 10^{-10} Ry to 10^{-1} Ry. The tetrahedron methods tested in QE included linear, Blöchl, and Kawamura. The pseudopotentials were obtained from http://www.quantum-espresso.org/pseudopotentials. All pseudopotentials implemented the generalized gradient approximation (Perdew et al. 1996), had nonlinear core corrections, and were scalar relativistic.

In VASP we tested the same 12 metallic systems tested in Quantum Espresso. We used a pre-release version of VASP version 6.0 for all calculations. The smearing methods we compared were Fermi-Dirac, Gaussian, and 1st and 2nd order Methfessel-Paxton. Each smearing method was tested with 11 different smearing values, ranging from 10^{-5} eV to 1 eV. The tetrahedron methods included linear and Blöchl's tetrahedron methods, and the amount of smearing for tetrahedra tests in VASP was $\sigma = 1 \times 10^{-5}$ eV with Gaussian smearing. All pseudopotentials for the tests in VASP used the projector-augmented wave method (Blöchl 1994; Kresse & Hafner 1994b; Kresse & Joubert 1999), the generalized-gradient approximation (Perdew et al. 1996), and augmentation charge corrections.

Our results for smearing in VASP are shown in Figs. 6.6–6.8. Our results for tetrahedron methods in VASP are in Fig. 6.9. We compare the convergence of the energy components of the total energy in VASP for a few metals in Fig. 6.16. See the supplementary information of the published article for addition plots.

For both VASP and QE, the energy cutoffs were 2× the largest ENMAX or wfc_cutoff in the system's pseudopotential file, respectively. Forces or stresses were not compared for 4 of the systems (Al, AlNiTi, Pd, and V) because forces were zero by symmetry.

To improve readability and because the results for VASP and QE are very similar, we only show results in the main text for VASP. Results for QE and k-point convergence plots for all calculations performed in this study can be found in the supplementary information of the published article.

6.3 Analysis of smearing and tetrahedron methods

Our goal in running these tests was to find a smearing method and smearing parameter robust enough to accurately simulate many different metals, and identify input parameters that would make it possible to treat DFT codes as a black box. Another objective was to see if smearing or tetrahedra would increase the efficiency or accuracy of DFT simulations of metals. The quantity *error ratio* is introduced to make it easier to compare total energies and forces with and without smearing. It is defined as

$$\varepsilon_r = \frac{\log(\frac{\varepsilon_s}{\varepsilon_n})}{\log(10)}, \tag{6.7}$$

where ε_s is the error with smearing and ε_n is the error without smearing, both of which are measured at the same k-point density. As an example, if the error ratio is -1 for the total energy error for a VASP or QE simulation, smearing is $10\times$ more accurate than the same simulation without smearing. If the error ratio were 1, no smearing would be $10\times$ more accurate than smearing. The error in the total energy for QE and VASP is simply the difference in the calculated energy and an energy "answer" obtained without smearing at a large k-point density ($100 \times 100 \times 100$ Monkhorst-Pack grid). At the time of writing, one of the routines in VASP has a hard exit if the smearing parameter is less than 10^{-5} eV that prevents investigating smearing parameters of smaller values without adjusting internal parameters and recompiling. For this reason, the value of the smearing parameter for tests "without smearing" is 10^{-5} eV. In QE, the value of the smearing parameter is 1×10^{-10} Ry. The error for forces in VASP is

$$\varepsilon^f = \sum_{\text{atoms}} ||\mathbf{F}_s - \mathbf{F}_a||, \tag{6.8}$$

or 2-norm where $\mathbf{F}_s$ is the force on one of the atoms in the atomic basis, $\mathbf{F}_a$ is the force "answer", which we approximate by the force on the same atom without smearing at a very large k-point density. QE returns the stress tensor instead of the forces on atoms. The error for stresses is the difference in the determinants of the stress tensors. For the number of iterations, we do not use the error ratio but take the difference in the number of electronic iterations with and without smearing.

We include "traffic light" plots to make it easier to see how smearing and no smearing compare for all the plots in the appendix. In traffic light plots for forces, stresses, and energies, there is a green box where smearing has 25% less error than no smearing, a red box were smearing has 25% more error, and a yellow box for in between the two. For the number of SCF iterations, there is a green box where smearing resulted in at least one fewer iteration than no smearing, a red box where smearing resulted in at least one more iteration, and a yellow box between. This information is also included in the legends of the figures. DFT codes have an accuracy up to around 1 meV/atom. We ignore simulations that resulted in total energy errors less than 0.1 meV/atom, and mark them with a white box. A missing box is placed where a DFT simulation failed. Traffic light plots are shown in Figs. 6.6–6.9.

To determine which set of parameters would work best in general, we took the mean and mean deviation of the energy error ratios, force or stress error ratios, and iteration differences for all 12 metals for a given smearing method and parameter or tetrahedron method. These are shown in Fig. 6.10. It is apparent from these plots that the smaller the smearing parameter, the more accurate the energies and forces. Smearing results in a reduction in the number of SCF iterations that is *independent* of the value of the smearing parameter. The average reduction is 4–5 iterations regardless of the method of smearing or the smearing parameter. Smearing in VASP with a smearing parameter around 1×10^{-1} eV to 1×10^{-2} eV appears to improve forces by a small amount but have large mean deviations that often result in less accurate forces. The larger the smearing parameter, the more likely one will get large error in the energy or forces. Blöchl's tetrahedron method in

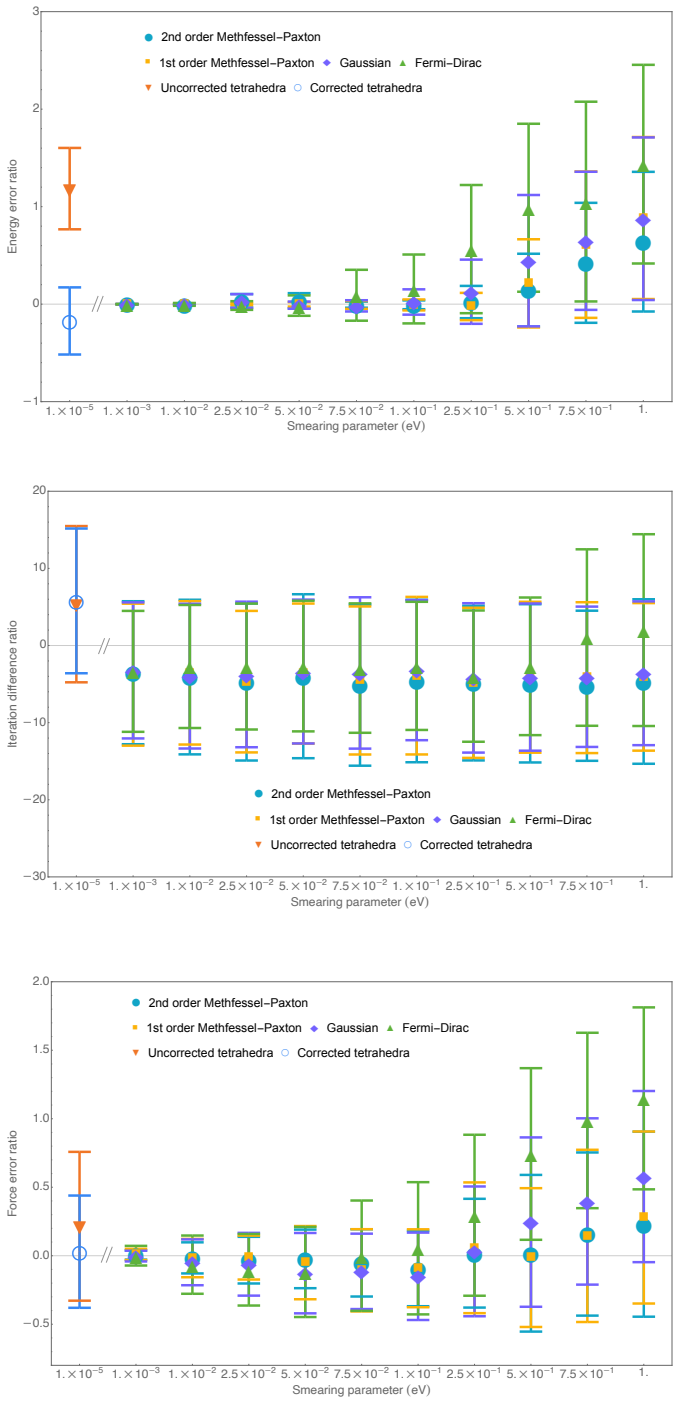


Figure 6.10 Mean and mean deviation of energy ratios, iteration differences, and force ratios in VASP. Tetrahedron methods were only tested with the smallest smearing parameter.

VASP resulted in more accurate total energies but a large mean deviation indicates it sometimes leads to less accurate energies.

Fig. 6.15 shows a typical result of smearing on Ca as computed in VASP. Many more smearing tests can be found in the supplementary information of the corresponding published article. A major feature of the smearing tests is the leveling off of the convergence for large amounts of smearing. This is expected because the integral of the smeared band structure is different from the unsmeared, and the difference of the two integrals is the same as the error where the error convergence levels off. Smearing is expected to show improvement over no smearing for k-point densities just before the density where the convergence levels off. The lack of improvement could be related to uncertainty in the Fermi level or the energy value where smearing occurs (see Eq. 6.1). We demonstrate these ideas in Fig. 6.4. The DFT runs with practically no smearing ($\sigma = 1 \times 10^{-5}$ eV in VASP) often show the best error convergence.

There are very few cases where Methfessel-Paxton smearing improves total energies: Al with σ values of 1 eV and 0.75 eV are two examples. These same smearing parameters show very poor performance—leveling off of the error—for many other systems with Methfessel-Paxton smearing. There is no single, optimal value of the smearing parameter for all metals; the optimal smearing depends on the metal. One takeaway from the data is the smaller the smearing, the more accurate the total energies and forces. The size of the Fermi level plays a role in how much smearing occurs for a given smearing parameter, and greater caution is needed when selecting a single smearing parameter to study systems with widely varying Fermi levels or total energies. One cannot recommend a "rule of thumb" as there is not one. This is demonstrated in Fig. 6.11.

The level of smearing is not the same for the different smearing methods (Gaussian, Fermi Dirac, etc.). It is generally believed that Methfessel-Paxton smearing is superior to other smearing methods. However, Methfessel-Paxton smearing appears to perform better because it results in less smearing for the same smearing parameter than other smearing methods. For example, Fermi-Dirac

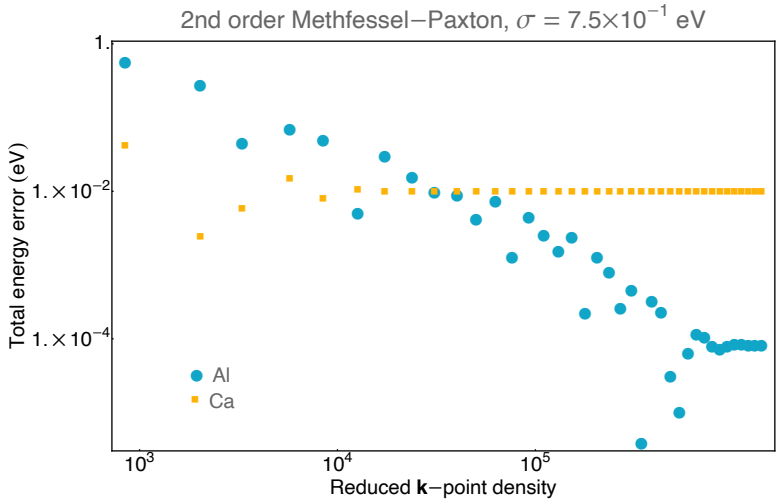


Figure 6.11 The total energy error convergence for Al and Ca with second order Methfessel-Paxton smearing with a smearing parameter value of 0.75 eV. For Al, this is one of the few cases where Methfessel-Paxton smearing performs better than no smearing. The same smearing method and parameter performs very poorly for another metallic system, Ca, and demonstrates that the optimal smearing is dependent on the system.

and Methfessel-Paxton smearing perform the same in Fig. 6.12.

Elastic constants converge slowly with respect to k-point density and require extreme k-point densities to converge without smearing. Smearing will cause the elastic constants to converge with a reasonable number of k-points but like other tests in this paper, what they converge to differs from the elastic constants without smearing (see Fig. 6.13). Smearing may help the elastic constants agree with experimental values (Louail et al. 2004). In a comprehensive study, it seems likely that smearing would improve agreement with experiment for many systems but make some systems further from agreement. Broad testing of the effect of smearing on elastic constants is not in the scope of this work—the reader is encouraged to perform their own tests.

In the plots in Fig. 6.16, the convergence of the component energies of the total energy for Al, Ca, and Cu with 1st order Methfessel-Paxton smearing in VASP is observed. The atomic energy contribution to the total energy is ignored due to its lack of dependence on smearing or k-point density. The errors in the component energies all decrease as the amount of smearing decreases. One

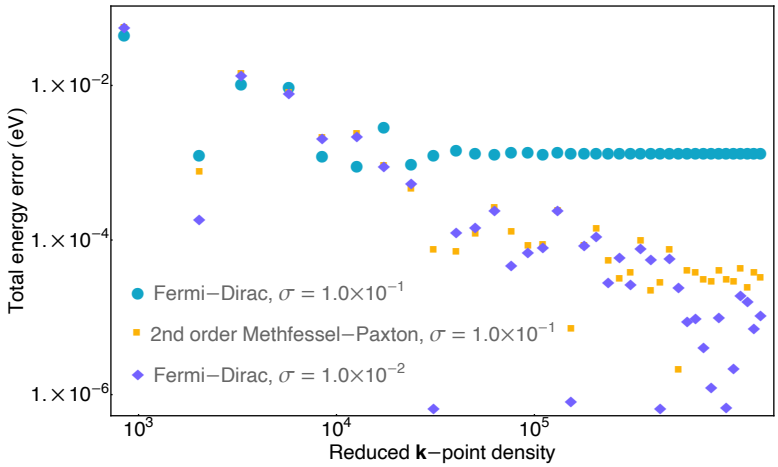


Figure 6.12 The total energy error convergence for Ca. For the same value of the smearing parameter, Fermi-Dirac smearing results in more smearing than second order Methfessel-Paxton, which is apparent from the leveling off the convergence. It would appear that Methfessel-Paxton smearing performs better than Fermi-Dirac, but it is possible to select a smaller smearing parameter for Fermi-Dirac where the two smearing methods have comparable amounts of smearing and performance. In the limit the smearing parameter goes to zero, the performance of the two is identical.

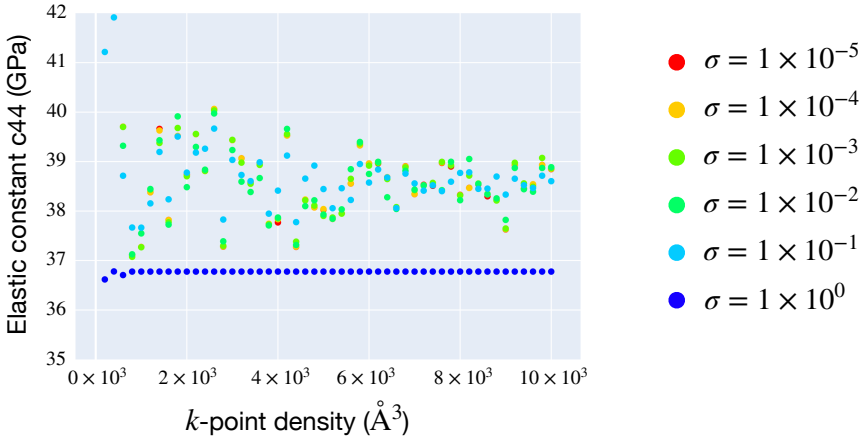


Figure 6.13 Convergence of the elastic constant with respect to k-point density for varying smearing parameters in VASP. The system is HCP Zr and the smearing method is Gaussian. A large smearing parameter causes the elastic constant to converge quicker but converges to a value different than it would with no smearing. In some cases, the value of the elastic constant with smearing agrees better with experiment.

exception is Cu with 2nd order Methfessel-Paxton smearing where there is improved convergence with $\sigma = 2.5 \times 10^{-1}$ than $\sigma = 1.0 \times 10^{-5}$, but, as previously discussed, this is an exception; this same smearing results in worse performance in other metals. Notice that some of the errors in the component energies are correlated.

For VASP (Fig. 6.17), we look at the performance of tetrahedron method methods on Al, Ca, Cu, and CuAu. The performance of the uncorrected tetrahedron method also shows very consistent, poor convergence, but also consistent improvement in total energy error with higher k-point densities. The corrected tetrahedron method shows the same or better convergence of the total energy than calculations without tetrahedra and practically no smearing.

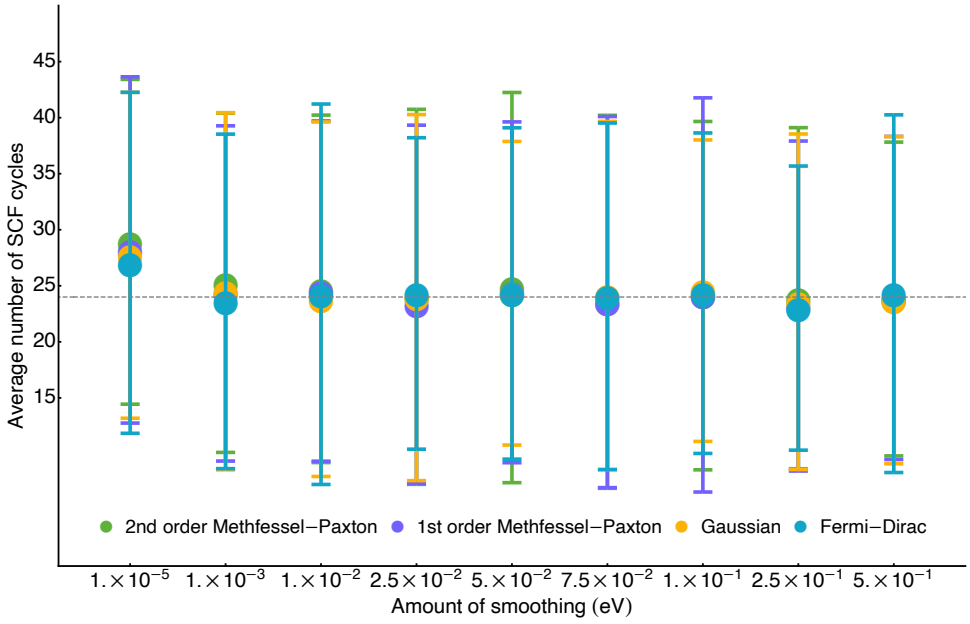


Figure 6.14 The number of SCF cycles averaged over all 12 metals in VASP. The VASP calculations with practically no smearing require slightly more iterations. The horizontal black line is a guide for the eye.

Finally, we look at the effect of smearing and tetrahedron methods on the number of SCF iterations. Smearing was developed to fix issues in the SCF cycle related to band sloshing, which should result in fewer SCF iterations. In QE, smearing and tetrahedron methods had little effect on

the total number of SCF cycles needed to reach convergence. At higher k-point densities, the number of SCF iterations for all smearing and tetrahedron methods convergences to the same value to within a few iterations. In VASP, the number of SCF iterations to reach self-consistency is sporadic and random at all k-point densities and for all methods and amounts of smearing. Fig. 6.14 shows the average of the number of iterations in the SCF cycle for all systems of a given smearing method and value. The average of the standard deviations is represented with error bars. The smallest smearing value has an average number of SCF iterations that is 4 or 5 more iterations than all the others. All other smearing parameters, regardless of the method or amount, have approximately the same average and average standard deviation. Smearing has little effect on the number of SCF iterations in QE and reduces the number of iterations in VASP by around 5 iterations.

6.4 Summary

We ran about 40,000 DFT calculations on a suite of twelve metals to test the efficiency of smearing and tetrahedron methods. We used four smearing methods, three tetrahedron methods, and ten different smearing parameters in Quantum Espresso, and twelve metals, four smearing methods, two tetrahedron methods and eleven different smearing parameters in VASP. Smearing has significant, negative effects on the total energy when the smearing parameter is large, and little systematic, positive effect when small. Smearing has little effect on the number of SCF iterations in QE and decreases the number of SCF iterations in VASP by about 5 iterations on average. The observed reduction is *independent* of the smearing parameter.

Tetrahedron methods have no effect on the number of SCF cycles in QE and significantly increase the number of SCF cycles in VASP. In QE, Blöchl's tetrahedra slightly improved total energies and stresses. In VASP, Blöchl's tetrahedra improved total energies on average. Small amounts of smearing showed slight improvement in the forces and stresses in VASP and QE.

Due to the risk of selecting a parameter that results in too much smearing, and the minimal effect of smearing on the number of SCF cycles, we recommend using a very small smearing parameter in DFT calculations, especially for high-throughput or machine learning applications. We recommend using Blöchl's tetrahedron method in QE due to minor improvements to total energies and stresses. Blöchl's tetrahedron method in VASP improved total energies but significantly increased the number of SCF cycles. We emphasize that, although smearing may lead to minor reductions in the number of SCF cycles and slightly more accurate forces for certain systems, smearing leads to inaccurate DFT calculations when the smearing parameter is large, and there does not exist an optimal smearing parameter; there is no reliable "rule of thumb" to follow when choosing smearing parameters. The optimal smearing parameter is dependent on the system, smearing method, smearing parameter, and k-point density. As little smearing as possible is the safest option when treating DFT as a black box.

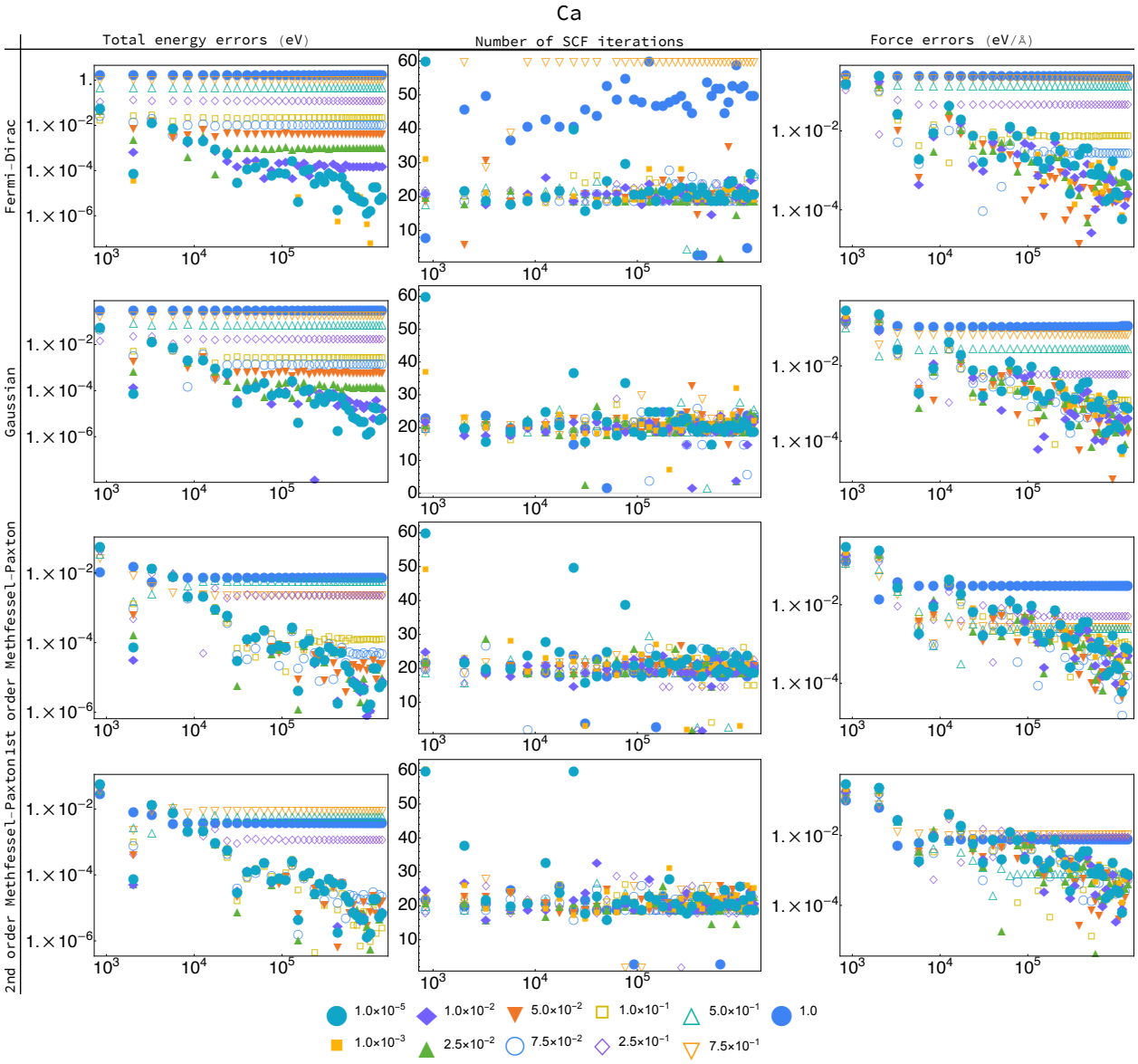


Figure 6.15 The total energy convergence, number of SCF cycles, and force convergence for Ca in VASP. For all plots, the x-axis is the reduced k-point density in units of cubic Angstroms. The legend at the bottom gives the amount of smearing in electron volts.

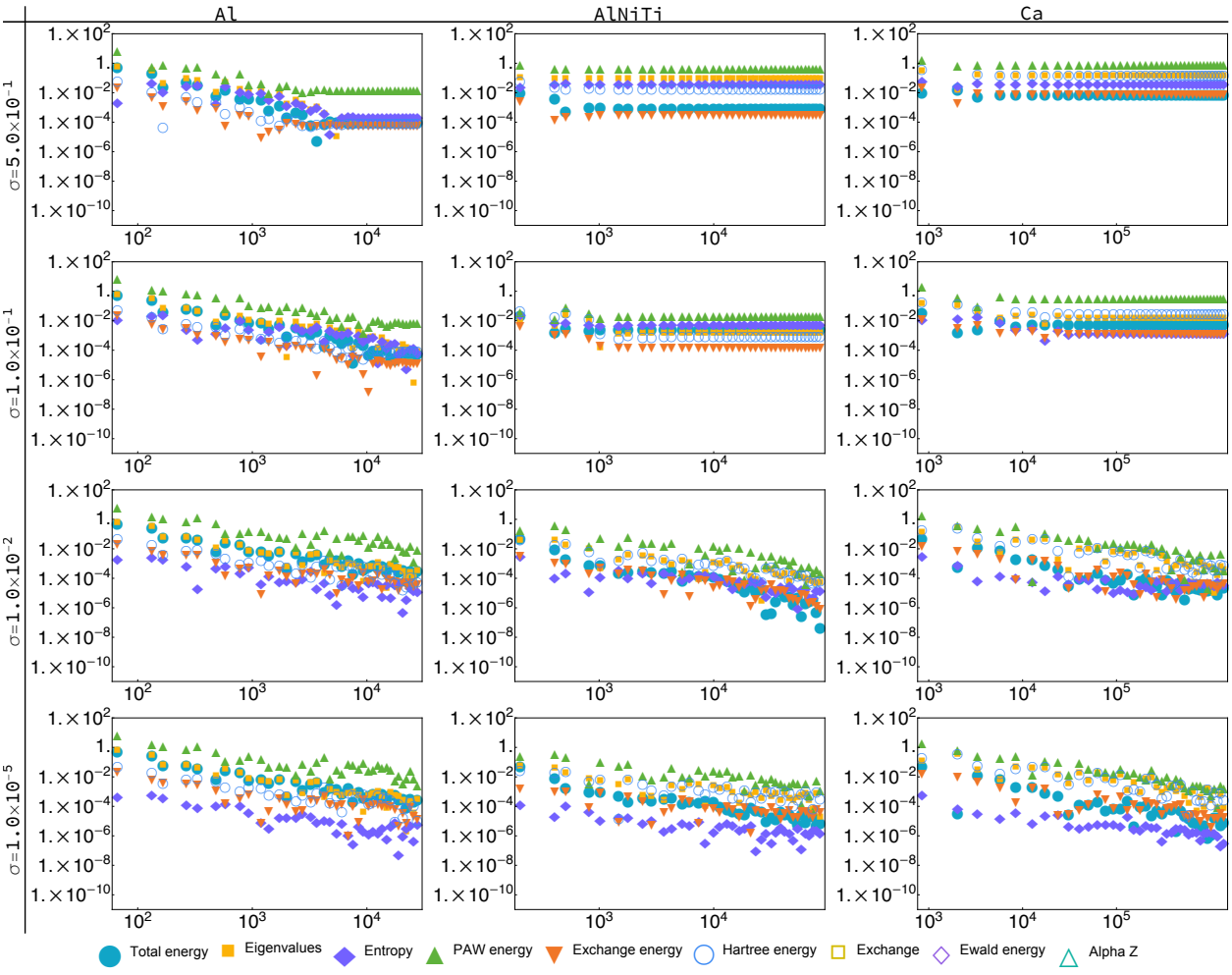


Figure 6.16 The convergence of components of the total energy in VASP for the metals Al, AlNiTi, and Ca with 1st order Methfessel-Paxton smearing. The atomic energy contribution to the total energy is left out due to its lack of dependence on the amount smearing or the k-point density. For all plots, the x-axis is the reduced k-point density in units of cubic Angstroms.

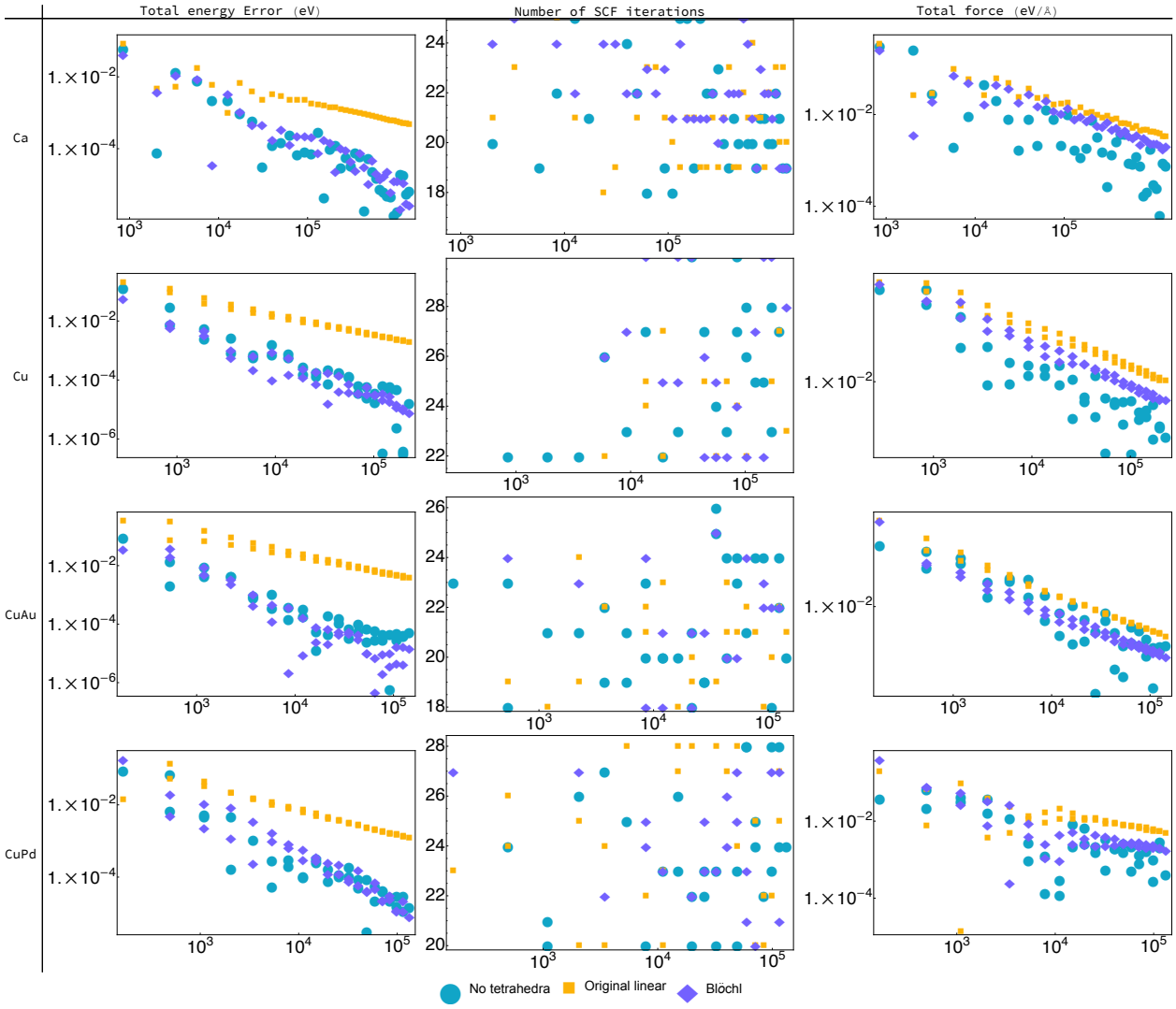


Figure 6.17 The total energy convergence, number of SCF cycles, and force convergence of Ca, Cu, CuAu, and CuPd with tetrahedron methods in VASP. For all plots, the x-axis is the reduced k-point density in units of cubic Angstroms.

Chapter 7

Using symmetry to calculate the band energy more efficiently

The content in this chapter also comes from a paper that has been accepted for publication.

Computing intrinsic properties of materials using density-functional theory requires numerical integration (Cancès et al. 2020). For example, the energy of the electrons (band energy) and the number of electrons in a given energy state (density of states) are properties of materials that are obtained by numerical integration. The domain of integration for these integrals is a Voronoi cell called the first Brillouin zone (referred to throughout this chapter as simply Brillouin zone or BZ). A Voronoi cell is the region of space closer to one point in a mesh than to any other point in the mesh. In terms of geometry, the BZ is a convex polyhedron that often has a complicated shape. The integrand for the band energy or density of states is a periodic, algebraic surface called the electronic band structure. An example BZ for a 3D lattice is shown in Fig. 7.1, and the electronic band structure for a 2D toy model is shown in Fig. 7.2.

Figure 7.1 The Brillouin zone for a body-centered tetragonal lattice. The Brillouin zone is the integration domain of integrals that give properties of materials. It is also a convex polygon that often has a complicated shape. This plot, and many others in this chapter, were created with SymmetryReduceBZ.

The electronic band structure is computationally expensive to evaluate because each evaluation means solving an eigenvalue problem of a Hermitian matrix of order from hundreds to thousands. The number of evaluations is reduced by up to a factor of 48 by using the symmetry of the material, which allows one to reuse eigenvalues. In other words, if two points are symmetrically equivalent (for example, a rotation by 90° maps one point to the other), the eigenvalues are identical at both

points.

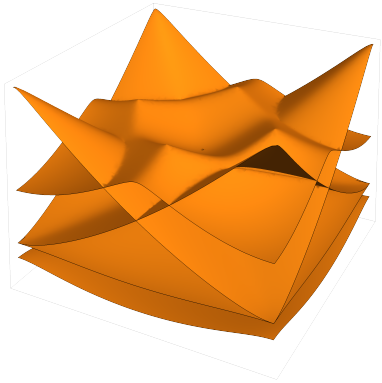


Figure 7.2 The electronic band structure or algebraic surface of a 2D toy model of a material. The band structure is often the integrand for integrals that give properties of materials. A band structure in 2D was chosen due to the difficulty of visualizing in more than 3D. The Brillouin zone in this case is simply a square.

Typically, the band energy is computed using the rectangular method despite its erratic and low-order error convergence (Jorgensen & Hart 2021). The greater the symmetry of the integration grid, the greater the reduction in the number of eigenvalue computations. Calculating grids with the greatest possible symmetry has been an active area of research (Morgan et al. 2018). The majority of the band energy computation is spent solving eigenvalue problems at points on a uniform grid over the BZ (Wende et al. 2019), so the savings from symmetry can be significant (as mentioned, up to 48 times more efficient).

However, uniform grids are not efficient because integration errors are not spread uniformly throughout the BZ but concentrated in small regions. This is demonstrated in Fig. 7.3 for the band energy calculation but is also the case for the calculation of other properties of materials. In the figure, most of the error is concentrated around the discontinuities (integration errors are proportional to the height of the discontinuities), and a preferential sampling close to the discontinuities leads to improved integration error convergence.

Adaptive meshes are computationally more efficient but break symmetry (very few of the points in the mesh are equivalent to other points in the mesh by symmetry). Whereas symmetry may make

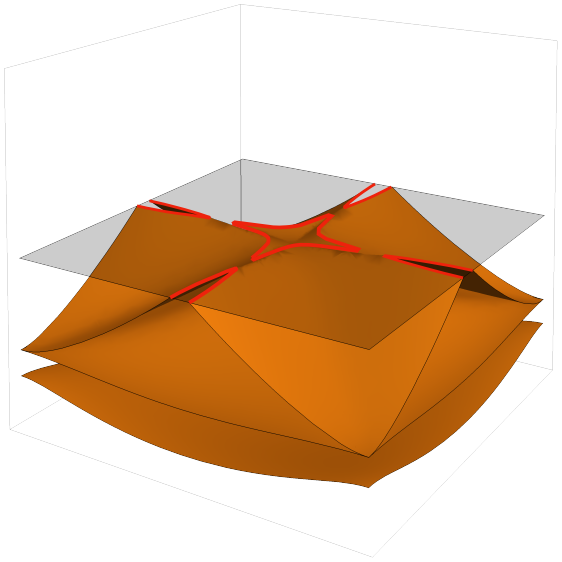


Figure 7.3 The band energy is the volume beneath the sheets of the algebraic surface below a cutting plane (shown in gray in the plot). The sheets are discontinuous where they intersect the cutting plane, and these discontinuities (highlighted in red) are the primary source of error in the calculation of the band energy.

uniform grids up to 48 times more efficient, symmetry does very little to improve the efficiency of adaptive meshes.

This apparent drawback of adaptive meshes is avoided by integrating solely within a part of the BZ called an *irreducible* Brillouin zone or IBZ. An IBZ is a closed polytope Q within a BZ such that any lattice symmetry g that moves any point of the interior of Q must move Q to a new polytope gQ, which only overlaps with Q on the boundary, if at all. By integrating within an IBZ, one has the freedom to refine regions where integration errors are large without taking an efficiency hit from breaking symmetry. Like the BZ, an IBZ is a convex polyhedron that often has a complicated shape (see Fig. 7.4).

Adaptive integration schemes split the domain of integration into subelements of various geometries. Because symmetry-breaking is no longer a concern when working in an IBZ, one has freedom to choose the shape of integration subelements (for example, simplices or hexahedra).

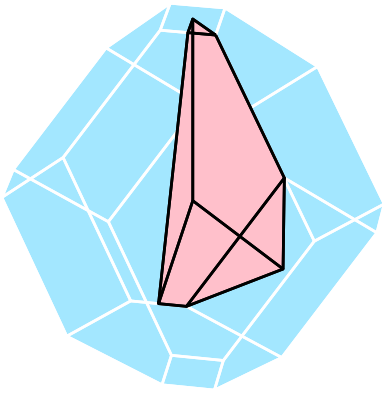


Figure 7.4 The Brillouin zone (in blue) and irreducible Brillouin zone (in pink) for a body-centered tetragonal lattice. The irreducible Brillouin zone is a convex polyhedron that often has a complicated shape.

Another benefit of integrating within an IBZ is avoiding many of the intersections of the sheets of the algebraic surface, which typically occur at locations of high symmetry. Intersections of sheets in regions of high symmetry, called non-accidental crossings, occur at the boundary of an IBZ and have no effect on the accuracy of the interpolation of the band structure within an IBZ. *Accidental* crossings, which occur *within* an IBZ, are still problematic and affect the accuracy of interpolation significantly.

There are algorithms for calculating the BZ (Finney 1979), a topic covered in nearly all solid-state physics textbooks. Despite many papers on calculating points and lines of high symmetry in the BZ (Munro et al. 2020), we are aware of only one other algorithm (Otero-de-la Roza & Luaña 2011) for calculating an IBZ, but it has little explanation and no results for verification.

In what follows, we present an algorithm that uses point symmetries to efficiently reduce the Brillouin zone to an irreducible Brillouin zone. The only inputs required are the lattice vectors, the atomic basis, and the crystal symmetries. A proof of the algorithm is provided in the appendix, and an implementation of the algorithm is available as open-source software. In summary, with a representation of an IBZ, properties of materials may be calculated with more efficient, higher-order,

adaptive integration schemes because low-symmetry integration points do not affect the efficiency of the calculation.

7.1 Calculating Brillouin zones

In many calculations, materials are treated as ideal crystals with atoms arranged periodically. The atoms of crystals lie on a lattice, an example of which is given in 2D in Fig. 7.5.

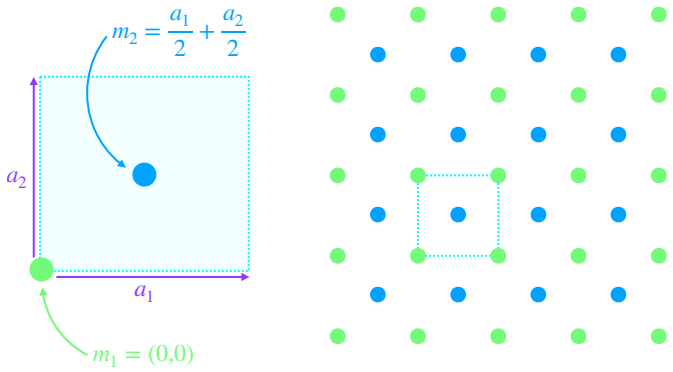


Figure 7.5 An example of a 2D crystal. Integer multiples of the lattice vectors a_1 and a_2 generate each point in the lattice. The positions of the atoms m_1 and m_2 with respect to the unit cell (the cyan square) are given in terms of the lattice vectors. The different colors of the dots indicate different atom types.

The arrangement of the atoms in an infinite crystal can be characterized by its lattice vectors and atomic basis

$$x = A(\mathbf{n} + \mathbf{m}_\alpha) \tag{7.1}$$

where x is the position of an atom in the crystal, A is a matrix whose columns are the lattice vectors (in 2D, a_1 and a_2), $\mathbf{n}$ is a vector of integers, and $\mathbf{m}_\alpha$ is the offset of the αth atom in the unit cell in lattice coordinates. The reciprocal lattice, obtained by taking the Fourier transform of the real-space lattice, satisfies the relation $AB = I$ (there may be a factor of 2π, depending on the convention) where I is the identity matrix and B is a matrix whose columns are the reciprocal lattice vectors

(b_1 and b_2 in Fig. 7.6). A lattice may be thought of as a tessellation of $\mathbb{R}^n$ where the tile of the tessellation is called the unit cell. For a single lattice, there are an infinite number of possible tiles that can tesselate $\mathbb{R}^n$. The BZ is a unit cell comprising the region of reciprocal space closest to the origin than to any other reciprocal lattice point. These concepts are illustrated in Fig. 7.6.

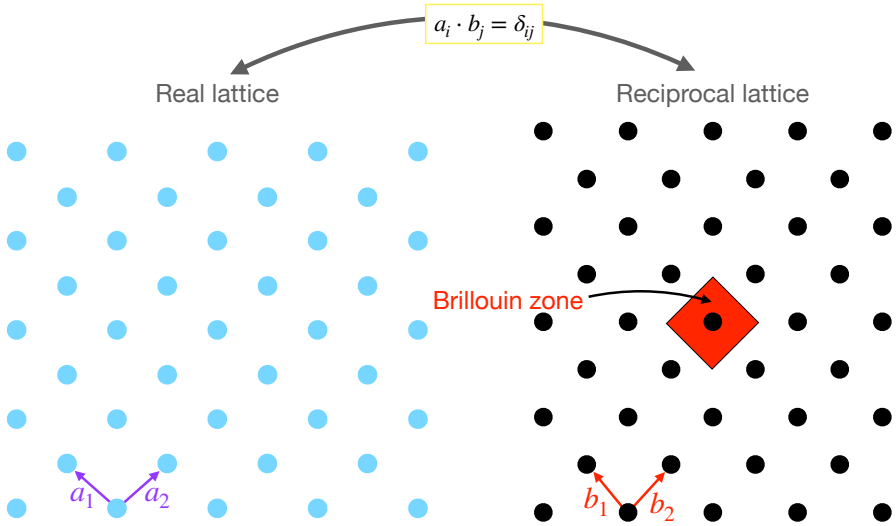


Figure 7.6 A Fourier transform maps the real-space lattice to the reciprocal lattice. The real and reciprocal lattice vectors satisfy the relation $a_i \cdot b_j = \delta_{ij}$ where δ_{ij} is the Kronecker delta. The Brillouin zone is one of infinite options for the unit cell of the reciprocal lattice and is the region of space closest to the origin of reciprocal space than any other point in the reciprocal lattice.

Constructing the BZ is a topic almost always treated in introductory solid state physics textbooks. The algorithm is typically described as follows:

- Calculate a few lattice points [1] near the origin.
- Order the lattice points by distance from the origin.
- For each lattice point near the origin:
  - Determine the straight line segment that connects the origin to the lattice point.

[1] This is explained in greater detail later on.

- Calculate the perpendicular bisector of the line segment (this is a straight line that intersects the line segment at its midpoint and is perpendicular to the line segment).
- Calculate intersections of the perpendicular bisector with any previously calculated perpendicular bisectors.
- Discard any intersection that is on the side of any perpendicular bisector that is opposite the origin.
- Break out of the loop if a convex hull made from the intersections has the same size as $\det(B)$.

In this approach, one has to compute intersections of perpendicular bisectors and discard intersections that are opposite the origin of any bisector. This becomes more complicated in 3D where the bisectors are planes instead of lines, and intersections of bisecting planes are lines instead of points. The algorithm is illustrated in Fig. 7.7.

The algorithm is simplified significantly by taking advantage of half-spaces. A half-space is a tuple $(\mathbf{n}, d)$ where $\mathbf{n}$ is a unit vector that is normal to the perpendicular bisector (in this case), and d is the distance from the origin to the bisector in the direction of $\mathbf{n}$. This is illustrated in Fig. 7.8.

With half-spaces, one does not need to calculate intersections of bisectors nor keep track of intersections. The algorithm for calculating the BZ becomes:

- Calculate a few lattice points near the origin.
- For each lattice point near the origin:
  - Calculate the half-space for the point (the distance is the norm of the point divided by 2, and the normal vector is in the same direction as the vector that points from the origin to the lattice point).
  - Calculate the intersection of the half-space with any previously calculated half-spaces.

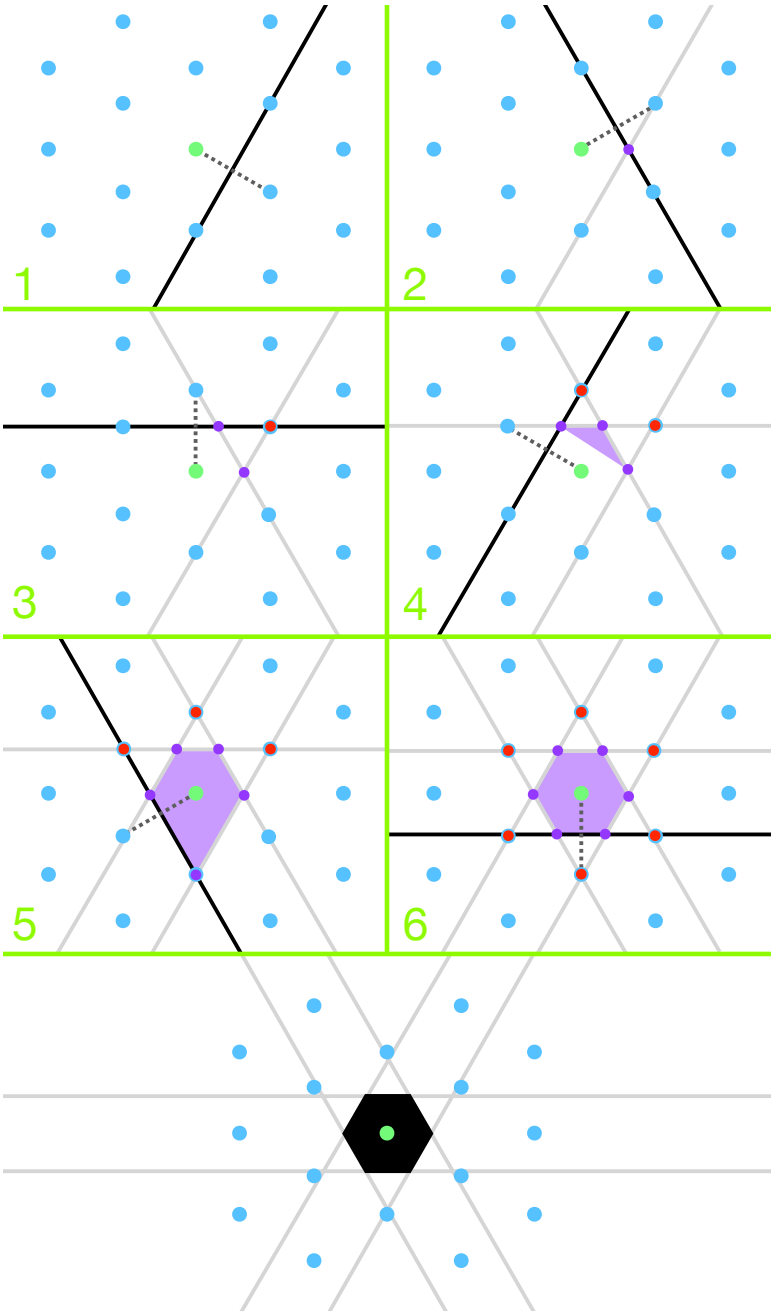


Figure 7.7 An illustration of the typical algorithm for computing the Brillouin zone. The reciprocal lattice points are blue dots, the origin is the green circle, line segments connecting the origin to lattice points are dashed lines, the perpendicular bisectors are the solid black or gray lines, intersections of bisectors that are kept are violet dots, and intersections that are discarded are red dots. After each iteration, a candidate BZ is constructed (if possible) from intersections of bisectors, shown as violet polygons in the figure.

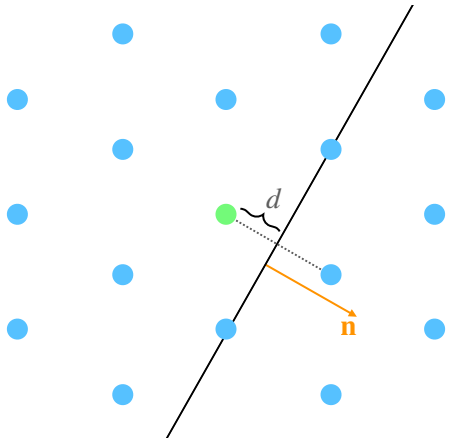


Figure 7.8 A half-space is a tuple $(\mathbf{n}, d)$ where $\mathbf{n}$ is a unit vector normal to the perpendicular bisector (the orange arrow in the figure) and d is the distance from the origin (green circle) to the bisector (solid black line) in the direction of $\mathbf{n}$.

- Break out of the loop if the size of the intersection of all the half-spaces is the same as $\det(B)$.

This approach to calculating the BZ is illustrated in Fig. 7.9. A more detailed outline of the algorithm that gives the BZ is provided below.

Algorithm 1 Construct the BZ

procedure CONSTRUCT_BZ(B) ▷ B is a matrix with reciprocal lattice vectors as columns
  $V \leftarrow$ Reciprocal lattice points near the origin.
  $M \leftarrow$ Initialize the set of half-spaces.
  for $v \in V$ **do**
    $H_v \leftarrow \{x \in \mathbb{R}^n | x \cdot \frac{v}{|v|} \leq \frac{|v|}{2}\}$ ▷The half-space bisecting the line segment from O to v
    $M \leftarrow M \cup \{H_v\}$ ▷ Insert H_v into M.
  end for
  return M
end procedure

The unit cell for a crystal that is as small as possible and contains only one lattice point is called

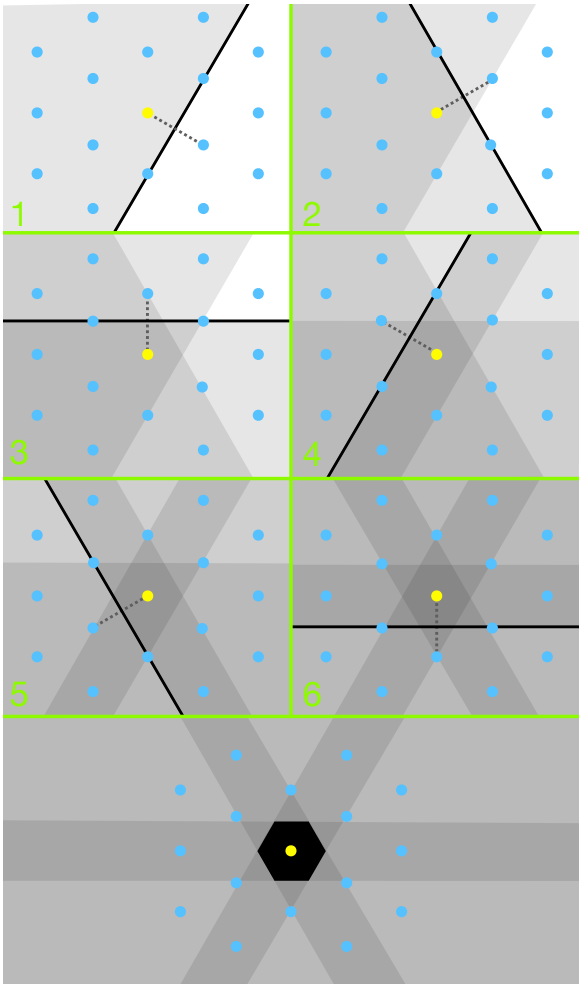


Figure 7.9 A simpler algorithm for computing the Brillouin zone involving half-spaces. The origin is the yellow circle, the reciprocal lattice points near the origin are blue dots, and half-spaces are the gray-shaded regions. The intersection of half-spaces gives the Brillouin zone.

a *primitive* unit cell. Fig. 7.6 is an example of a primitive unit cell. The same example with a non-primitive unit cell is shown in Fig. 7.10. It is beneficial to make the unit cell primitive because the integral in the calculation of properties of materials is over fewer sheets of the algebraic surface, and the algebraic surface has fewer self-intersections. Several codes are available for making unit cells primitive (Morgan et al. 2021; Ong et al. 2013). In the case of the band energy calculation, self-intersections introduce very fine features that are difficult to capture without large numbers of integration points. This is shown in Fig. 7.11

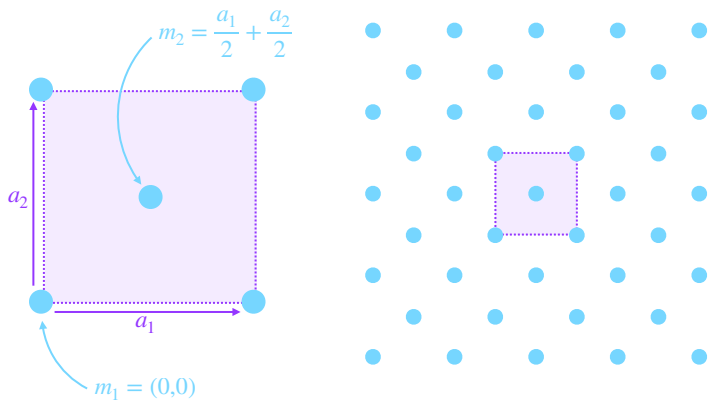


Figure 7.10 A non-primitive unit cell for a crystal with square symmetry.

In Algorithm 1, it was left unspecified how many reciprocal lattice points are needed; one might not include enough to determine the correct BZ or include far too many. (Usually, the number of lattice points is small, and this is not an issue except when the reciprocal unit cell is very skew. See Fig. 7.12 for an example in 2D.) In 2D and 3D, as long as the reciprocal lattice vectors are Minkowski reduced [2] the BZ is guaranteed to lie within the set of unit cells that share a vertex at the origin [3] This puts constraints on the number of reciprocal lattice points that have to be included in the calculation of the BZ. To be specific, the lattice points that are sufficient to determine the BZ in

[2] Minkowski reduction makes the lattice vectors as short as possible. See 108 for details.

[3] See the appendix of 63.

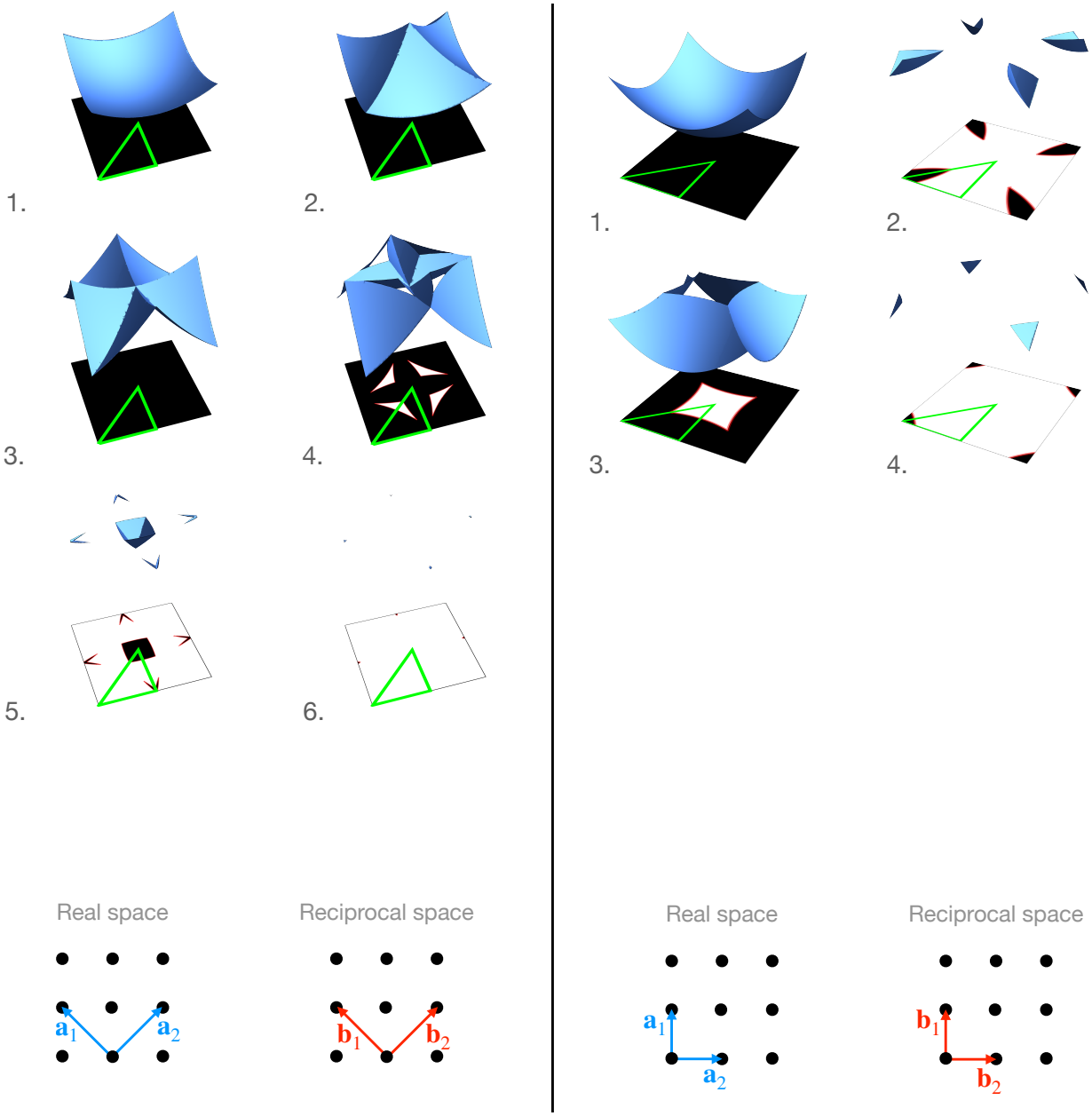


Figure 7.11 The sheets of the algebraic surface on the left side of the figure are for a non-primitive basis (lattice vectors) for a 2D free-electron model (often used to model metallic materials), while the sheets on the right are for a primitive basis. Below each sheet is the shadow of the sheet or the domain of integration for the band energy calculation as well as the IBZ outlined in green. Calculating the band energy with a non-primitive is much more difficult because the integration is over more sheets, and the sheets of the algebraic surface have more self-intersections that create difficult-to-integrate fine features. Sharp corners in the shadows of the sheets are very difficult to approximate when they are interior to the IBZ, which are present for the non-primitive basis on the left.

2D are

$$x = B \begin{pmatrix} i \\ j \end{pmatrix}, \qquad i, j \in \{-2, -1, 0, 1, 2\}, \tag{7.2}$$

where B is a matrix with the reciprocal lattice vectors as columns. In 3D, there is one more iterator over the same range, that is,

$$x = B \begin{pmatrix} i \\ j \\ k \end{pmatrix}, \qquad i, j, k \in \{-2, -1, 0, 1, 2\}. \tag{7.3}$$

Minkowski reduction makes the algorithm robust because it specifies the number of reciprocal lattice points needed.

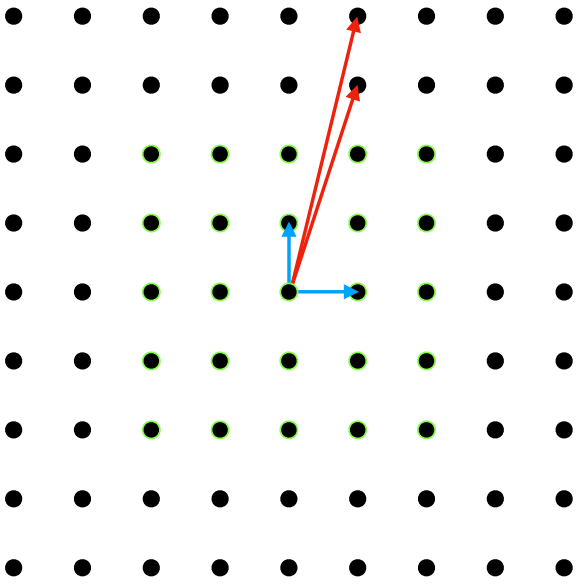


Figure 7.12 The lattice vectors in red and blue both produce the same lattice, shown in black. In general, the lattice points outlined in green are the only lattice points needed to compute the BZ, and often only a subset of these points are necessary. For the lattice vectors that have not been Minkowski reduced (shown in red), far more lattice points (large integer values of i and j in Eq. 7.2) have to be considered to generate all the lattice points outlined in green.

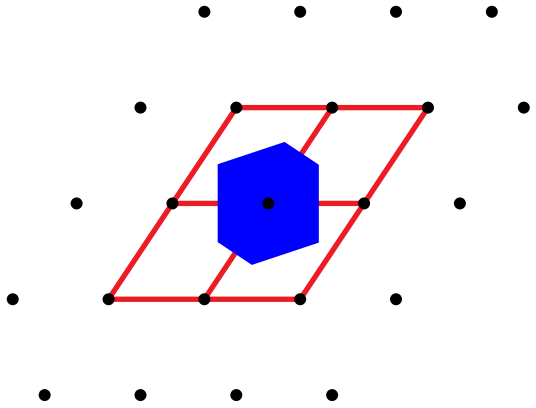


Figure 7.13 The Brillouin zone of a 2D lattice lies within the conventional unit cells that have a vertex at the origin when the lattice basis has been Minkowski reduced [4]. Reciprocal lattice points are shown as black points, the unit cells are outlined in red, and the Brillouin zone is shaded in blue.

7.2 Calculating irreducible Brillouin zones

The following is the algorithm for generating an irreducible Brillouin zone. Although the basic concepts of the construction for the BZ are well known and discussed in most textbooks, they do not provide a practical representation for the BZ or a complete description of the geometry of the BZ (vertices, edges, faces, and volume). To the best of our knowledge, prior to our algorithm, a rigorous and general algorithm for calculating a point-symmetry reduced Brillouin zone (an irreducible Brillouin zone) and characterizing its geometric features has not been developed.

The point symmetries of the real-space crystal are used to reduce the BZ to an IBZ as follows. We define G to be the point group of the crystal's space group (the space group includes rotational and translational symmetries), $I \in G$ is the identity operator, P is the BZ convex polyhedron, and V is the set of vertices of P. An outline of the algorithm that reduces the BZ to an IBZ is given below (for $x, y \in \mathbb{R}^n$ the symbol $d(x, y)$ denotes the distance between x and y). See the Appendix for more details.

Reducing the BZ to an IBZ is shown in Fig. 7.14. When the algorithm finishes, the resulting IBZ is the intersection of the half-spaces in N (see Algorithm 2), which is returned as a convex hull object. A convex hull object is a convenient way to store vertices, edges, and faces of the IBZ.

We have written the IBZ algorithm in Julia (Bezanson et al. 2017) with dependencies on the Julia Polyhedra library (Legat et al. 2019) and the C qhull package (Barber et al. 1996a). The code includes functions for visualizing the BZ and IBZ (many of the figures in this paper were produced from it) and can be called from Python.

Algorithm 2 Construct an IBZ from P (the BZ)

procedure CONSTRUCT_IBZ(M,G) ▷ M is a set of half-spaces whose intersection is P, and G is the (finite) symmetry group of P.
  $V \leftarrow$ The set of vertices of P.
  $F \leftarrow G \setminus \{I\}$ ▷ Initialize the set of symmetries.
  $N \leftarrow \text{copy}(M)$ ▷ Initialize the set of half-spaces.
  for $v \in V$ **do**
    for $g \in F$ **do**
      $v' \leftarrow gv$ ▷ Calculate the transformed point.
      if $v' \neq v$ **then**
        ▷The half-space bisecting the line segment from v to v'
        $H_{v,g} \leftarrow \{x \in \mathbb{R}^n | d(x,v) \leq d(x,gv)\}$
        $N \leftarrow N \cup \{H_{v,g}\}$ ▷Insert $H_{v,g}$ into N.
        $F \leftarrow F \setminus \{g\}$ ▷ Remove g from F.
      end if
    end for
  end for
  return N
end procedure

7.3 Testing the implementation

We know beforehand the relationship (Aroyo et al. 2014) between the IBZ volume Vol_{IBZ}, the BZ volume Vol_{BZ}, and the size of the point group n_p:

$$\text{Vol}_{\text{IBZ}} = \frac{\text{Vol}_{\text{BZ}}}{n_p}. \tag{7.4}$$

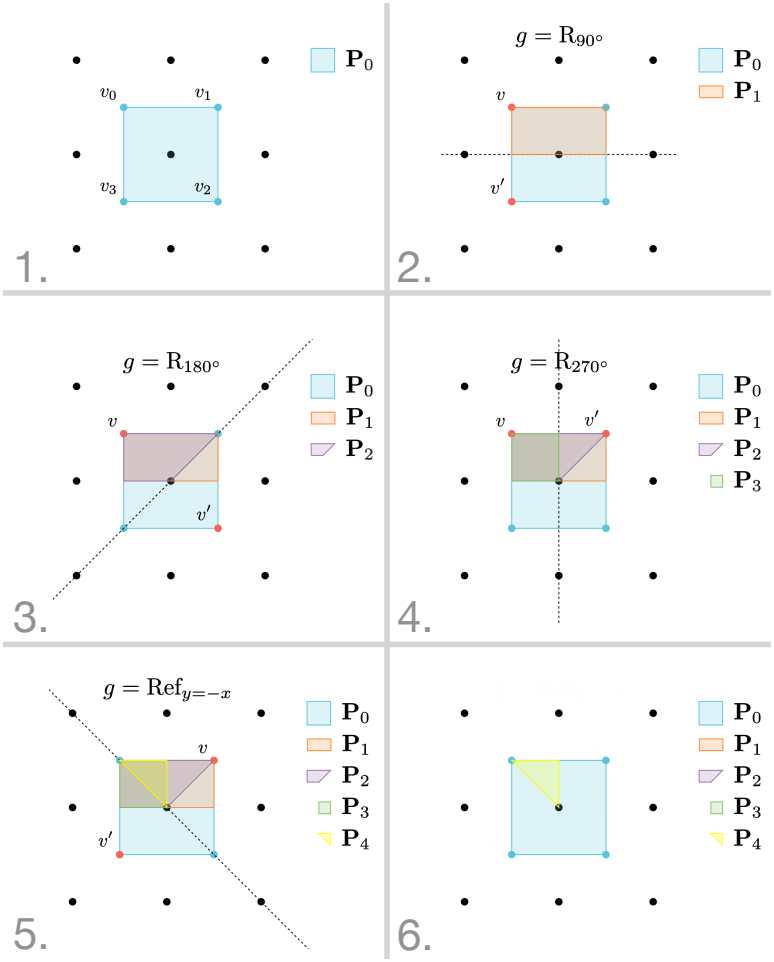


Figure 7.14 An illustration of the IBZ calculation. We begin with the BZ labeled P_0. Each operator in the point group successively reduces the size the BZ. Moving down the figure, we show the reduction of the BZ resulting from selected operators from the point group and end with the IBZ labeled P_4.

This relation is verified for each IBZ calculation. For testing, we *unfold the IBZ*. The IBZ is unfolded by applying each operator $g \in G$ to each of the vertices of the IBZ. If the algorithm is working correctly, the unfolded vertices are the same as the vertices of the BZ. Together these two calculations guarantee the IBZ is correct. The second step is necessary because it is possible to get the correct volume reduction but have the wrong shape. The IBZ is not unique, and the IBZ obtained from the algorithm depends on the order of the BZ vertices V and the order of the point operators G. In some cases, faces (3D) or edges (2D) of the IBZ may be translationally or rotationally equivalent (see the appendix for a discussion). Plots of the results of the IBZ algorithm for each of the 14 Bravais lattices are shown in Fig. 7.15 (We only show one BZ and IBZ even though some Bravais lattices have multiple BZs (Setyawan & Curtarolo 2010); we would also obtain many more IBZs for atomic bases that break the symmetry of the lattice).

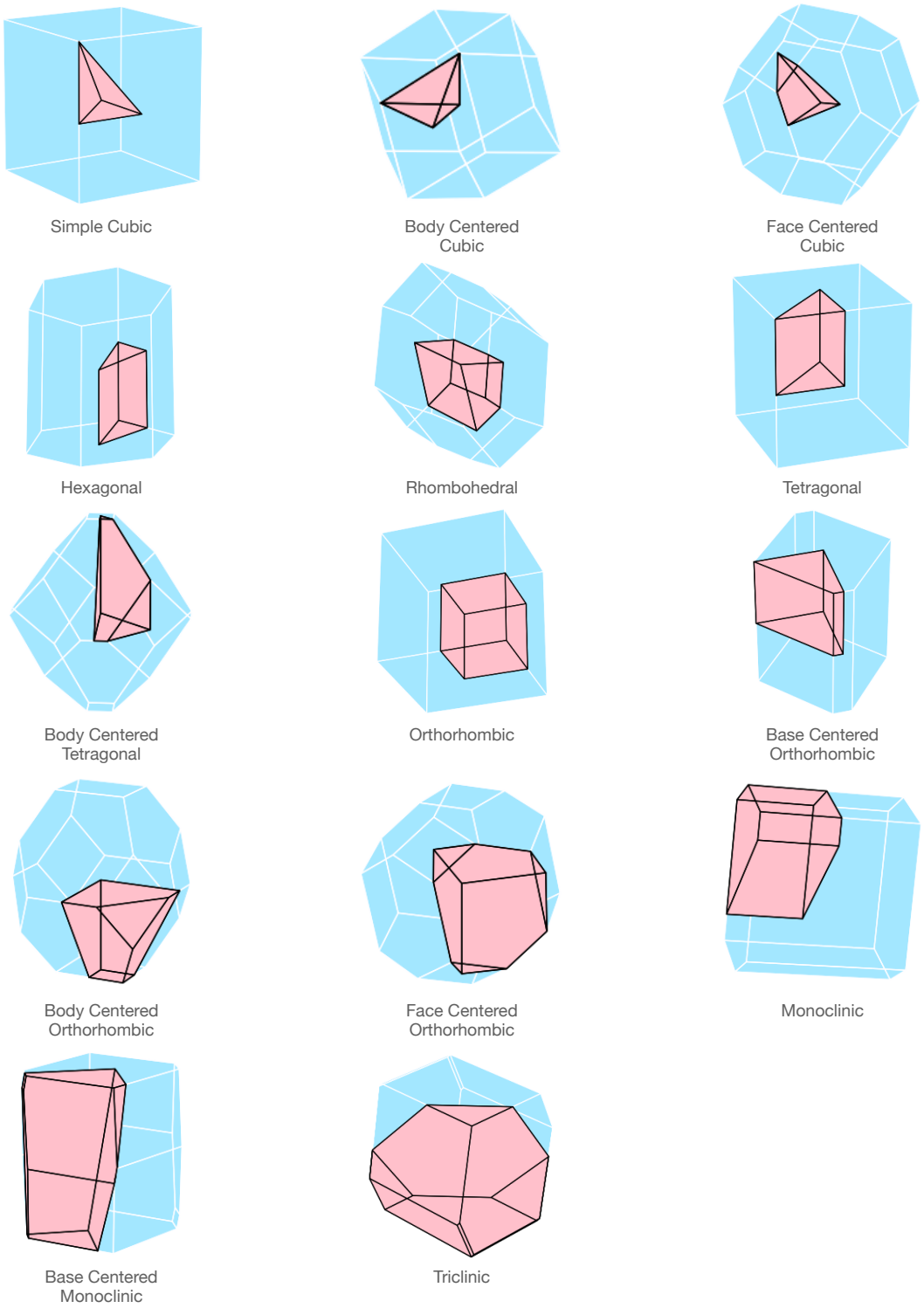


Figure 7.15 One BZ and IBZ for each of the 3D Bravais lattices.

Chapter 8

An adaptive quadratic integration algorithm

8.1 Quadratic approximation

8.1.1 Preliminary setup

The quadratic algorithm begins by calculating the irreducible Brillouin zone and then sampling it uniformly. The density of the initial grid over the IBZ is an adjustable parameter of the algorithm. If high accuracy is desired and the initial mesh is coarse, the first few iterations of adaptive refinement add sample points all over the IBZ and is more or less equivalent to starting the algorithm with a greater number of points. The IBZ is a convex polytope. Creating a closed, uniform mesh over a polytope is not as straight forward as it is for a simplex, so the IBZ is first split into simplices with the Delaunay method (Barber et al. 1996b) and then each simplex is sampled uniformly. The Delaunay method creates simplices that are close to equilateral, which makes the initial mesh more uniform. The simplices may be various sizes so to make the initial mesh as uniform as possible, the number of sample points in each simplex is determined by the size of each simplex. Once there is a uniform mesh, the mesh is split into simplices with the Delaunay method. This process is illustrated in Fig. 8.1.

Each simplex has a sample point at each corner, which is sufficient for a linear interpolation of the band structure within the simplex. In general, interpolation with a higher order polynomial is more accurate but also requires more sample points. In the calculation of the band energy where an eigenvalue problem is solved at each sample point, the performance of the algorithm is affected most by the number of sample points. So it is desirable to keep the number of sample points in the mesh to as few as possible and have a higher order approximation. A higher order interpolation of the band structure is obtained by "recycling" sample points *outside* of the simplex. A few points outside of the simplex—but also near the simplex and surrounding the simplex—are selected and a constrained least-squares fit is used to determine the coefficients of a quadratic polynomial over the simplex. The number of points used to compute quadratic coefficients is another parameter of the

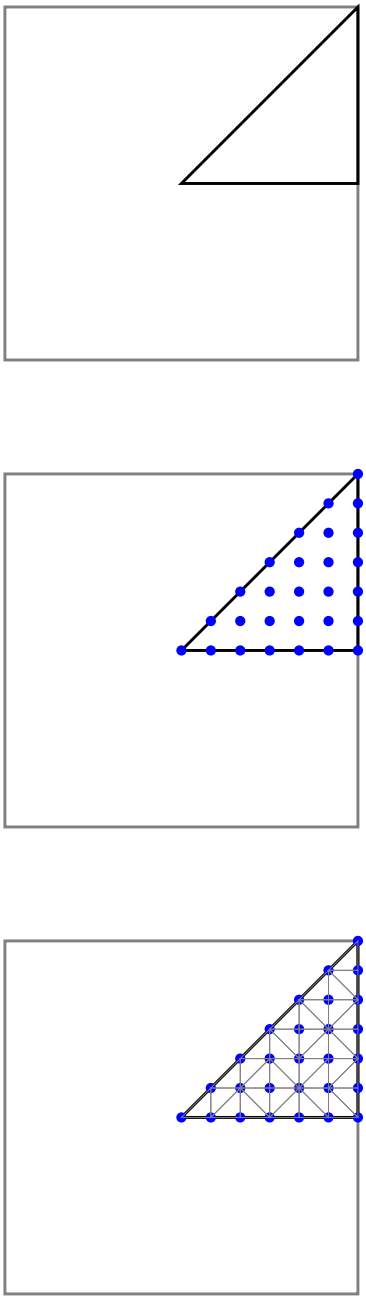


Figure 8.1 The quadratic integration algorithm starts by computing the irreducible Brillouin zone and then sampling it uniformly. The points of the mesh are then split into simplices. In the example above, the square is the Brillouin zone and the triangle is the irreducible Brillouin zone. Points of the mesh are shown in blue.

algorithm. For the results in this dissertation, the minimum number of points is 16 in 2D and 60 in 3D. Of interest is also the error in the interpolating polynomial so that new points may be added where errors are largest. To this end, errors in the quadratic coefficients are also computed. In other words, the quadratic coefficients are intervals instead of single values. The band structure for a 2D toy EPM is shown in Fig. 5.2, and a piece-wise quadratic approximation of the same band structure is shown in Fig. 8.2.

8.1.2 An interval Quadratic Approximation in 3D

This section describes how to calculate the least-squares quadratic coefficients and interval quadratic coefficients in 3D. Similar results are obtain in 2D.

Take a tetrahedron T with vertices

$$V_{2000},\ V_{0200},\ V_{0020},\ V_{0002}. \tag{8.1}$$

Denote $S = (s,t,u,v)$ and $S_i = (s_i,t_i,u_i,v_i)$ (note that this S_i is different than the ith sheet of the algebraic surface introduced in Ch. 5), and define a Barycentric coordinate system

$$P(S) = s*V_{2000} + t*V_{0200} + u*V_{0020} + v*V_{0002} \tag{8.2}$$

Let $f_j(s)$ be a function defining the j^{th} eigenvalues and let

$$q_1 = f_j(1,0,0,0),\ \ q_2 = f_j(0,1,0,0),\ \ q_3 = f_j(0,0,1,0),\ \ q_4 = f_j(0,0,0,1) \tag{8.3}$$

Choose $n > 6$ points (more than 6 points are needed to have an overdetermined system with more equations than unknowns since a quadratic in 3D has 10 coefficients and 4 coefficients are known) with barycentric coordinates $S_i,\ i = 1,\ldots,n$. We seek a least-squares solution for the coefficients $c_1,\ldots,c_6$ of the quadratic function

$$q(S) = q_1s^2 + q_2t^2 + q_3u^2 + q_4v^2 + 2(c_1st + c_2su + c_3sv + c_4tu + c_5tv + c_6uv) \tag{8.4}$$

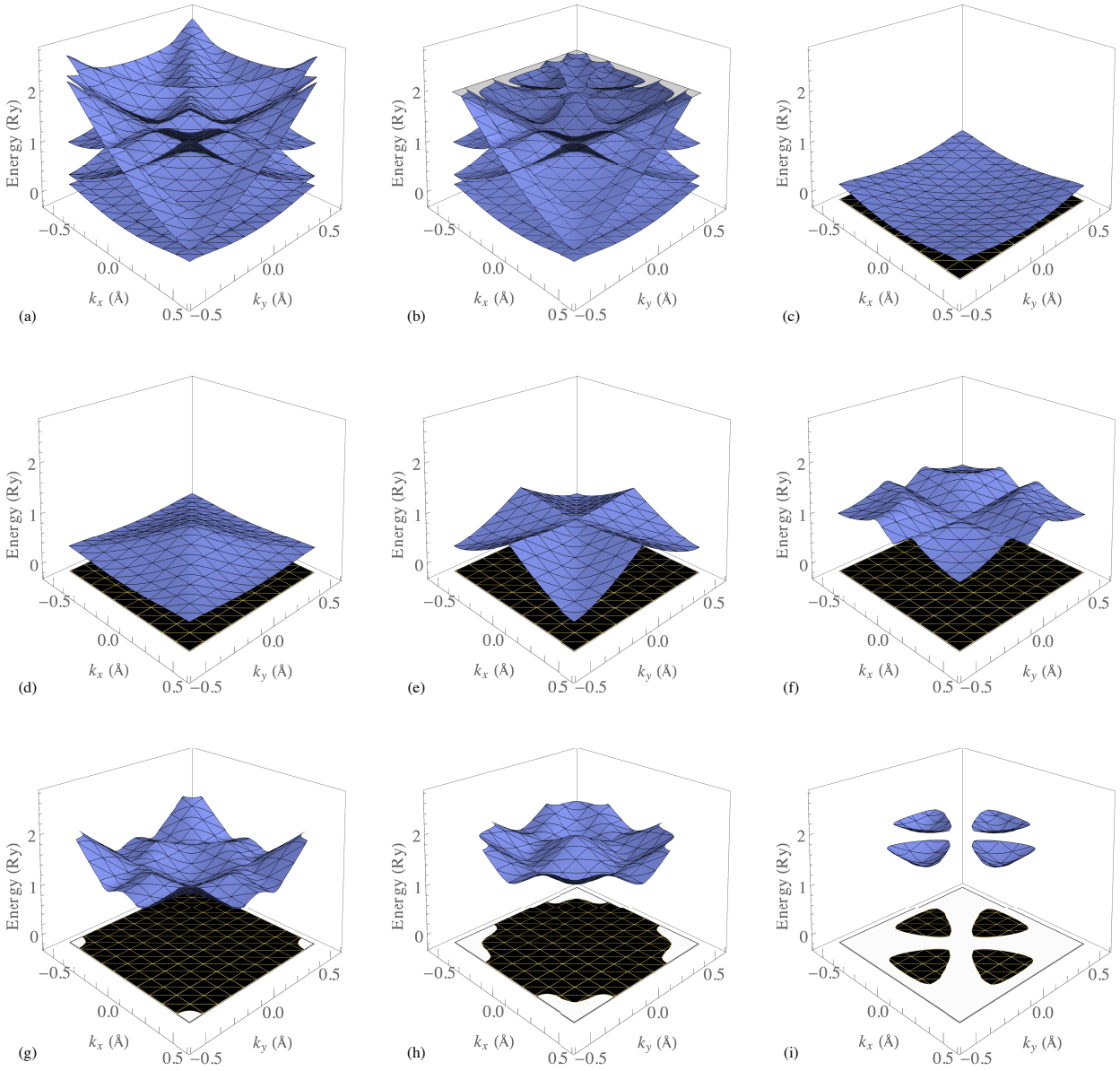


Figure 8.2 The piece-wise quadratic approximation of the band structure of a 2D EPM. For each tile and sheet, the band structure is approximated with a quadratic polynomial. The borders of the shadows of the sheets below the Fermi plane are approximated by quadratic curves. The borders of the shadows are highlighted in yellow in the $(k_x \times k_y)$-plane. Compare to Fig. 5.2.

for which $q(S_i) = f_j(S_i),\ i = 1,\ldots,n$.

The coefficients $c_1,\ldots,c_6$ can be solved using least squares

$$MC = Z \tag{8.5}$$

where

$$M = 2\begin{bmatrix} s_1t_1 & s_1u_1 & s_1v_1 & t_1u_1 & t_1v_1 & u_1v_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ s_nt_n & s_nu_n & s_nv_n & t_nu_n & t_nv_n & u_nv_n \end{bmatrix},\ C = \begin{bmatrix} c_1 \\ \vdots \\ c_6 \end{bmatrix} \tag{8.6}$$

$$Z = \begin{bmatrix} f_j(S_1) \\ \vdots \\ f_j(S_n) \end{bmatrix} - \begin{bmatrix} s_1^2 & t_1^2 & u_1^2 & v_1^2 \\ \vdots & \vdots & \vdots & \vdots \\ s_n^2 & t_n^2 & u_n^2 & v_n^2 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} \tag{8.7}$$

and $n > 6$. Weighted least squares where the weight is inversely proportional to the distance or squared distance from the triangle may improve the fit.

8.1.3 Coefficient error and interval width

Denote the least-squares solution for C by $\bar{C} = [\bar{c}_1,\ldots,\bar{c}_6]^\top$, and

$$\bar{q}(S) = q_1s^2 + q_2t^2 + q_3u^2 + q_4v^2 + 2(\bar{c}_1st + \bar{c}_2su + \bar{c}_3sv + \bar{c}_4tu + \bar{c}_5tv + \bar{c}_6uv). \tag{8.8}$$

For each sample point $f(S_i)$, perturb the coefficients $\hat{c}^i_j = \bar{c}_j + \varepsilon^i_j$ such that

$$f(S_i) = \hat{q}(S_i) = q_1s^2 + q_2t^2 + q_3u^2 + q_4v^2 + 2(\hat{c}^i_1st + \hat{c}^i_2su + \hat{c}^i_3sv + \hat{c}^i_4tu + \hat{c}^i_5tv + \hat{c}^i_6uv). \tag{8.9}$$

Eq. 8.9 is underdetermined. We seek a solution for the $\hat{c}^i_j$ that minimizes $(\varepsilon^i_1)^2 + \cdots + (\varepsilon^i_6)^2$. For samples $i = 1,\ldots,n$, subtract Eq. 8.9 and Eq. 8.8 and define

$$\delta_i = f(S_i) - \bar{q}(S_i) = 2(\varepsilon^i_1s_it_i + \varepsilon^i_2s_iu_i + \varepsilon^i_3s_iv_i + \varepsilon^i_4t_iu_i + \varepsilon^i_5t_iv_i + \varepsilon^i_6u_iv_i) \tag{8.10}$$

or

$$\delta_i = A_i \varepsilon^i \tag{8.11}$$

where

$$A_i = 2[s_i t_i, s_i u_i, s_i v_i, t_i u_i, t_i v_i, u_i v_i] \text{ and } \varepsilon^i = [\varepsilon_1^i, \varepsilon_2^i, \varepsilon_3^i, \varepsilon_4^i, \varepsilon_5^i, \varepsilon_6^i]^\top$$

Letting $D_i = s_i^2 t_i^2 + s_i^2 u_i^2 + s_i^2 v_i^2 + t_i^2 u_i^2 + t_i^2 v_i^2 + u_i^2 v_i^2$, the least-squares solution for Eq. 8.11 is

$$\varepsilon^i = A_i^\top (A_i A_i^\top)^{-1} \delta_i = \frac{\delta_i}{4D_i} A_i^\top$$

The interval coefficient c_j can be defined using the minimum and maximum values of $\hat{c}_j^i$, $i = 1, \ldots, n$.

8.1.4 Extending the IBZ mesh

The points used to determine the interval coefficients ought to be near the triangle and surround the triangle. However, it is not possible to find points that surround the triangles of the mesh that are near the border of the IBZ because there are no points available outside of the IBZ. For example, the lower-left triangle only has points to its right and none to its left if Fig. 8.1. This is overcome by *completing the orbits* of the points of the mesh within the IBZ to obtain symmetrically equivalent points outside of the IBZ and within the BZ. However, the IBZ shares borders with the BZ, so points outsize of the IBZ and BZ are needed. Points outside of the BZ are obtained by translating the points of the mesh by integer multiples of the reciprocal lattice vectors. The minimum number of translations that guarantees all points near but outside of the BZ are

$$x = B \begin{pmatrix} i \\ j \end{pmatrix} \qquad i, j \in \{-2, -1, 0, 1, 2\} \tag{8.12}$$

in 2D and

$$x = B \begin{pmatrix} i \\ j \\ k \end{pmatrix} \qquad i, j, k \in \{-2, -1, 0, 1, 2\} \tag{8.13}$$

in 3D where B is the reciprocal lattice vectors. Completing the orbits and translating points of the mesh can significantly increases the size of the mesh. In 2D, the *extended* mesh is up to 192 times larger than the initial mesh (12 rotations $\times$ 16 translations) and in 3D, the extended mesh is up to 3,072 times larger (48 rotations $\times$ 64 translations). The number of translations mentioned are for reciprocal lattice vectors that are Minkowski reduced (see the appendix of (Hart et al. 2019) for a proof). To keep the extended mesh as small as possible, only points on the boundary of the IBZ and nearest neighbors of points that are on the boundary are considered for translation and rotation. After rotation, translation, or rotation and translation, only points closer than a distance d to the boundary of the IBZ are kept where d is the furthest distance from the boundary to nearest neighbors. The dramatic reduction in the size of the extended mesh that comes from these conditions is shown in Fig. 8.3.

8.1.5 Selecting points for least squares

Extending the mesh made it so all triangles in the mesh have points that surround them. Now, one has to choose which of the points to include in the calculation of least-squares and interval coefficients. We aim to approximate a sheet of the band structure *within* a triangle of a triangulation of the IBZ. It follows that we want to use points that are as close to the triangle as possible because the further away from the triangle, the more the values obtained will deviate from those within the triangle. To make it easier to identify nearest neighbors, we also triangulate the extended mesh, which is shown if Fig. 8.4. It is also appealing to have the points surround the triangle so that the approximation is accurate all throughout the triangle. If more points are used on one side of the triangle, the approximation is more accurate on that side of the triangle. The interval coefficients will not contain potential larger deviations from points on the other side of the triangle. For this reason, the angular variable is split into bins from 0 to 2π and the arbitrary choice of two points per bin is chosen. In each bin, points that are nearest the triangle are selected first. We iterate through

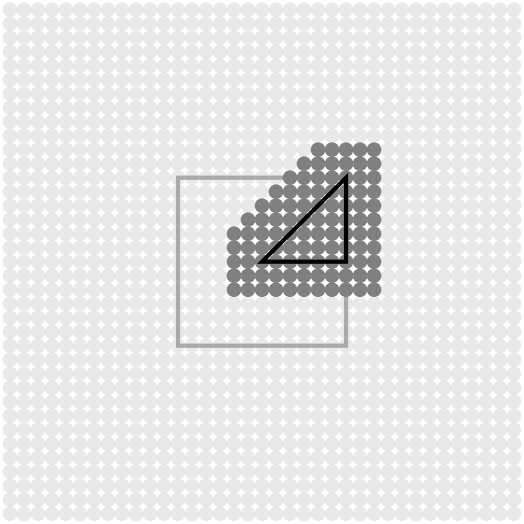


Figure 8.3 The extended mesh includes the points of the mesh and a collection of points outside but near the IBZ. These points are symmetrically equivalent to points within the IBZ so no new eigenvalues need to be computed. The points in light gray are the rotations and translations of all points within the IBZ, and the points in dark gray are the rotations and translations of points near the boundary of the IBZ that are close to the IBZ after the transformation. In this case, the number of dark gray points is about 13 times fewer than the number of light gray points. The reduction is greater in 3D or when there are more translations and rotations to consider.

all bins and select the point that is closest to the triangle in each bin (if there are points) until a minimum number of points is obtained. An example of the neighbors selected for varying numbers of points is shown in Fig. 8.5.

It is difficult to visualize the band structure, the approximation of the band structure with the quadratics with least-squares coefficients, and the bounds of the quadratics in higher dimensions. For this reason, a plot is provided of a toy band structure in 1D (univariate model) in Fig. 8.6.

8.2 Band energy and Fermi volume of quadratics

The approach we present for numerically calculating the band energy would be classified as a quadratic analytic method in the literature. Our work is similar to that of Wiesenekker, Velde, and Baerends(Wiesenekker & Baerends 1991; Wiesenekker et al. 1988) who were interested in solving more general integrals that gave properties of solids in 2D and 3D, such as the density of states. The integrals we consider are the simplest class of integrals they considered.

In 2D, Wiesenekker et al. split the irreducible wedge of the Brillouin zone into triangles. In each triangle and for each sheet of the band structure, they interpolate the sheet with a quadratic polynomial, and analytically calculate surface integrals of the polynomial approximation. They were unable to determine all the cases and corresponding analytic expressions of volume integrals, which are the types of integrals we are interested in solving (the number of states and the band energy). As a work around, they numerically integrated the analytical expressions for the surface integrals using Gauss-Legendre integration. Convergence of the numerical integral was improved by avoiding singular points in the density of states by splitting the integration domain into subintervals, and handling infinite derivatives at the endpoints of the intervals with a simple coordinate transformation.

Similar to Wiesenekker et al. we split the irreducible Brillouin zone into triangles (in 2D) and interpolate with quadratic polynomials. A Bézier representation for the polynomial surface over a

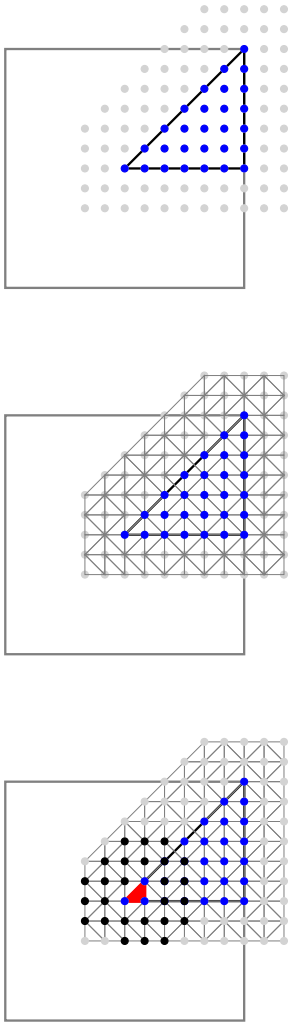


Figure 8.4 In order to calculate the interval coefficients in each triangle, the mesh is extended by symmetry to points outside of the IBZ. The extended mesh is also triangulated to make it easier to identify points in close proximity. In the last frame at the bottom of the figure, the neighboring points for the red triangle are shown in black. These points are used in the calculation of the interval coefficients.

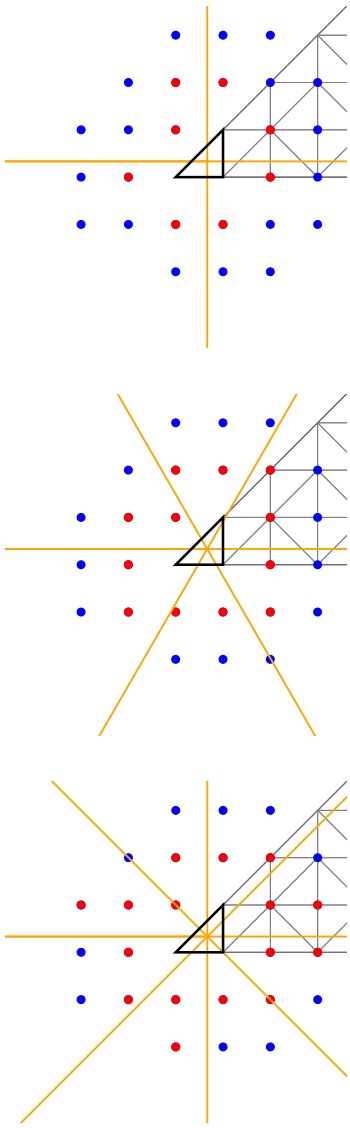


Figure 8.5 The neighbors selected for the calculation of interval coefficients for varying numbers of desired points. All points up to second nearest neighbors are blue dots, chosen points for the calculations are shown as red dots, and angular bins are demarcated by orange lines. Angles are measured with respect to the center of the triangle.The number of angular bins is the number of desired points divided by two.

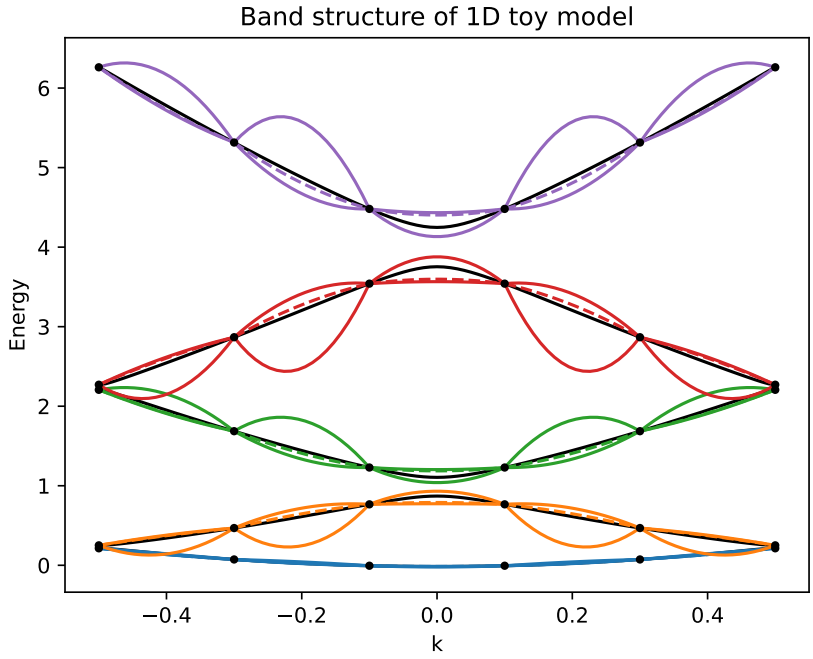


Figure 8.6 The band structure for a 1D toy empirical pseudopotential model is shown as black lines. The approximation of the band structure with quadratic polynomials with least-squares coefficients are the multi-colored, dashed lines, and the bounding quadratics (obtained from the interval coefficients) are the solid, multi-colored lines. The coefficients are obtained through constrained least squares, so the bounding and least squares quadratics agree at the eigenvalues, which are the black dots in the plot. The deviation of the bounding curves from the least-squares curves is exaggerated.

triangle will contribute to the robustness of our algorithm at some point by avoiding the need to solve systems of equations by taking matrix inverses. In contrast to the method of Wiesenekker et al.—who calculated the number of states and band energy with integrals over energy—we calculate the number of states and band energy by integrating over k-space. We obtain, with a little manipulation of the the triangles, analytic expressions for the number of states and band energy in 2D and resort to semi-analytic integration in 3D.

8.2.1 Quadratic polynomials over triangular domains

Concepts from computational geometry (Prautzsch et al. 2013) can make the calculation of the quadratic coefficients more robust by avoiding the need to compute the inverse of a matrix. A Bézier representation for the quadratics not been implemented for the results presented in this dissertation but is an improvement that will be made to the algorithm to improve robustness. A quadratic polynomial surface has a unique Bézier representation,

$$b(s,t,u) = b_{200}s^2 + b_{110}2st + b_{020}t^2 + b_{101}2su + b_{011}2tu + b_{002}u^2 \tag{8.14}$$

where $b(s,t,u)$ is a quadratic Bézier patch, b_{ijk} are Bézier points and s, t, and u are Barycentric coordinates with $s+t+u=1$. An example quadratic Bézier patch and its Bézier points are shown in Fig. 8.7. The Bézier surfaces are explicit, which means that the x- and y-coordinates lie in a lattice. If $b_{ijk} = (x_{ijk}, y_{ijk}, z_{ijk})$ and $\bar{b}_{ijk} = (x_{ijk}, y_{ijk})$ then $\bar{b}_{200}$, $\bar{b}_{020}$, and $\bar{b}_{002}$ are the vertices of the tile, and

$$\begin{aligned}
\bar{b}_{110} &= \frac{\bar{b}_{200} + \bar{b}_{020}}{2} \\
\bar{b}_{011} &= \frac{\bar{b}_{020} + \bar{b}_{002}}{2} \\
\bar{b}_{101} &= \frac{\bar{b}_{200} + \bar{b}_{002}}{2}
\end{aligned} \tag{8.15}$$

Polynomial curves have a unique Bézier representation,

$$\mathbf{b}(t) = \sum_{i=0}^{n} \mathbf{b}_i B_i^n(t), \tag{8.16}$$

where $\mathbf{b}(t)$ is the Bézier curve, t is a parametric variable whose range is $[0,1]$, $\mathbf{b}_i$ are the Bézier points, and B_i^n is the ith Bernstein polynomial of degree n. The Bernstein polynomials are given by

$$B_i^n(u) = \binom{n}{i} u^i (1-u)^{n-i} \tag{8.17}$$

The edges of the quadratic surface in Fig. 8.7 are Bézier curves. The Bézier points of these curves are the same as those of the polynomial surface. We can find all the Bézier points of the polynomial surface by finding the Bézier points of the the Bézier curves around the three edges of the triangle. These concepts are explained in greater detail in Appendix B.

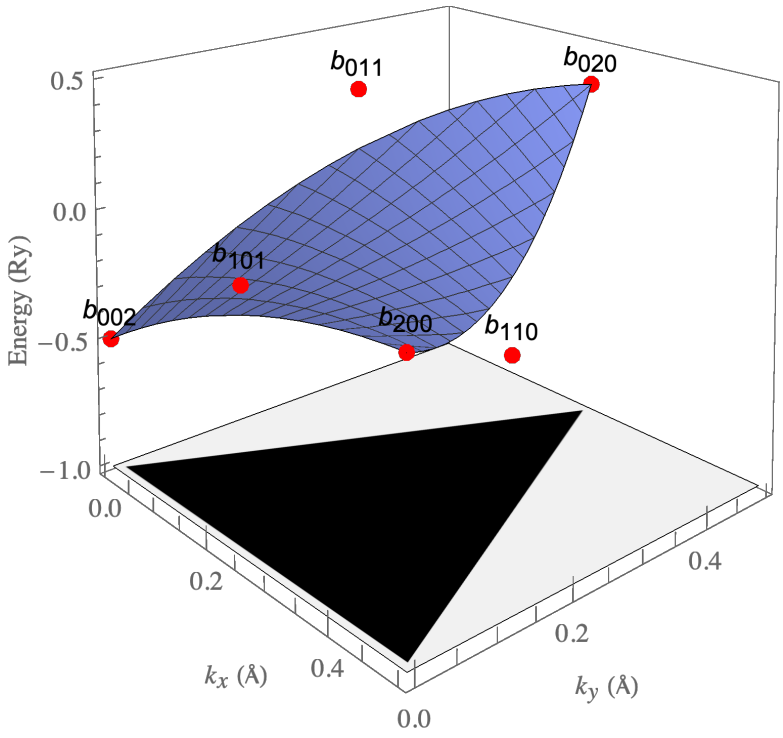


Figure 8.7 A quadratic Bézier patch over a triangular domain. The red points are the Bézier points of the blue Bézier patch. An edge of the Bézier patch is a Bézier curve whose Bézier points are the same as those of the Bézier patch along the same edge.

8.2.2 Analytic areas and volumes in 2D

In order to compute the Fermi level and band energy for a piece-wise quadratic polynomial, we need to be able to compute the area of shadows of quadratic surfaces and the volume below quadratic surfaces. For quadratic surfaces entirely below the Fermi level, the area of the shadow is the area of the triangle and volume beneath the quadratic surface is the average of the polynomial coefficients times the area of the triangle. For quadratic surfaces that are above and below the Fermi level, the shadow of the quadratic is a region bounded by the edges of the triangle and level curves of the quadratic surface. The band energy is the volume beneath the quadratic surface within the shadow. We refer to the area of the shadow as the Fermi area. The Fermi area and band energy are shown in Fig. 8.8 for a single quadratic surface that intersects the Fermi plane (a plane at the Fermi level).

It is possible to calculate analytically the Fermi area and band energy for a quadratic surface within a triangle, but the number of edge cases complicates the computation significantly. For example, the quadratic surface may be a rectangular parabola with a double point within the triangle (see the lower middle frame of Fig. 8.9), up to three edges of the quadratic shadow may be curves of intersection of the quadratic surface with the Fermi plane, the curve of intersection may be tangent to the edge of the triangle, and so on. To eliminate all edge cases, whenever an edge case arises, the triangle is split into sub-triangles such that all sub-triangles do not contain an edge case. In addition, to further simplify integration, the triangle is split whenever the curves of intersection intersect the edges of the triangle at more than two locations. The Delauney method is used to simplify the division of the triangle in sub-triangles. A triangle is continues to be split until all edge cases are eliminated. To split a triangle, points are added at the midpoints of the triangle's edges, at the double point of a hyperbola (if it is inside the triangle) and where the curve of intersection intersects the edges of the triangle. Example of cases where a triangle is divided into sub-triangles is shown in Fig. 8.9.

It is possible to find an analytic form for the curve of intersection for a quadratic polynomial

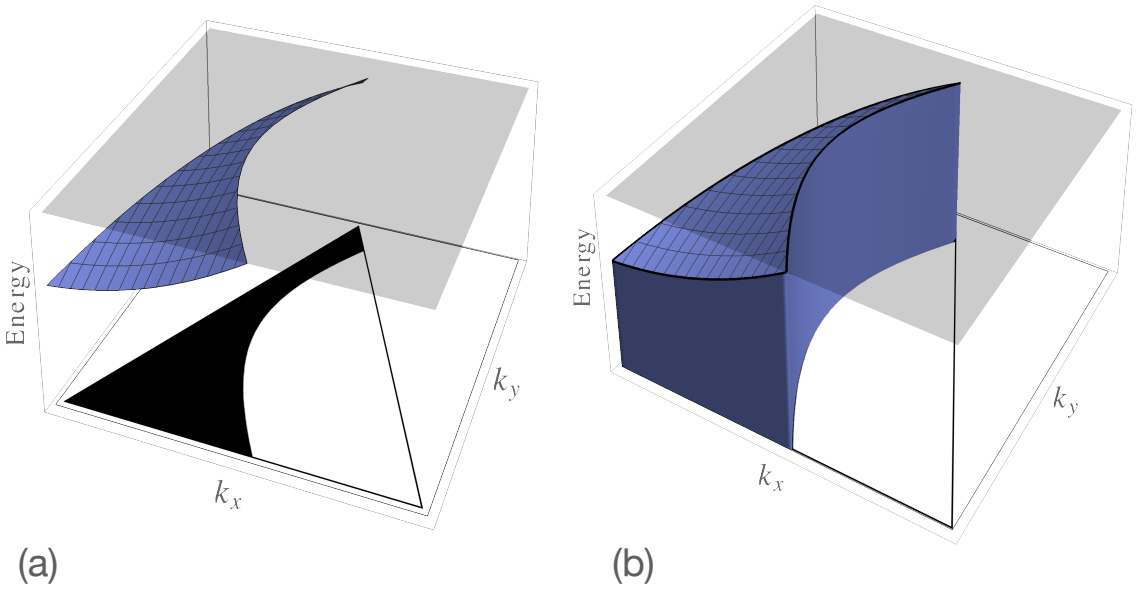


Figure 8.8 An illustration of the Fermi area and band energy for a 2D quadratic polynomial. The outline of the IBZ is the black outlined triangle in the $(k_x \times k_y)$-plane and the Fermi plane—a plane at a height of the Fermi level—is the transparent gray plane. The portion of the quadratic below the Fermi plane and within the triangle is the blue surface. The projection of the quadratic surface below the Fermi plane onto the $(k_x \times k_y)$-plane gives the black region in (a), which is referred to as the Fermi area. The band energy is the volume below the quadratic shown in (b).

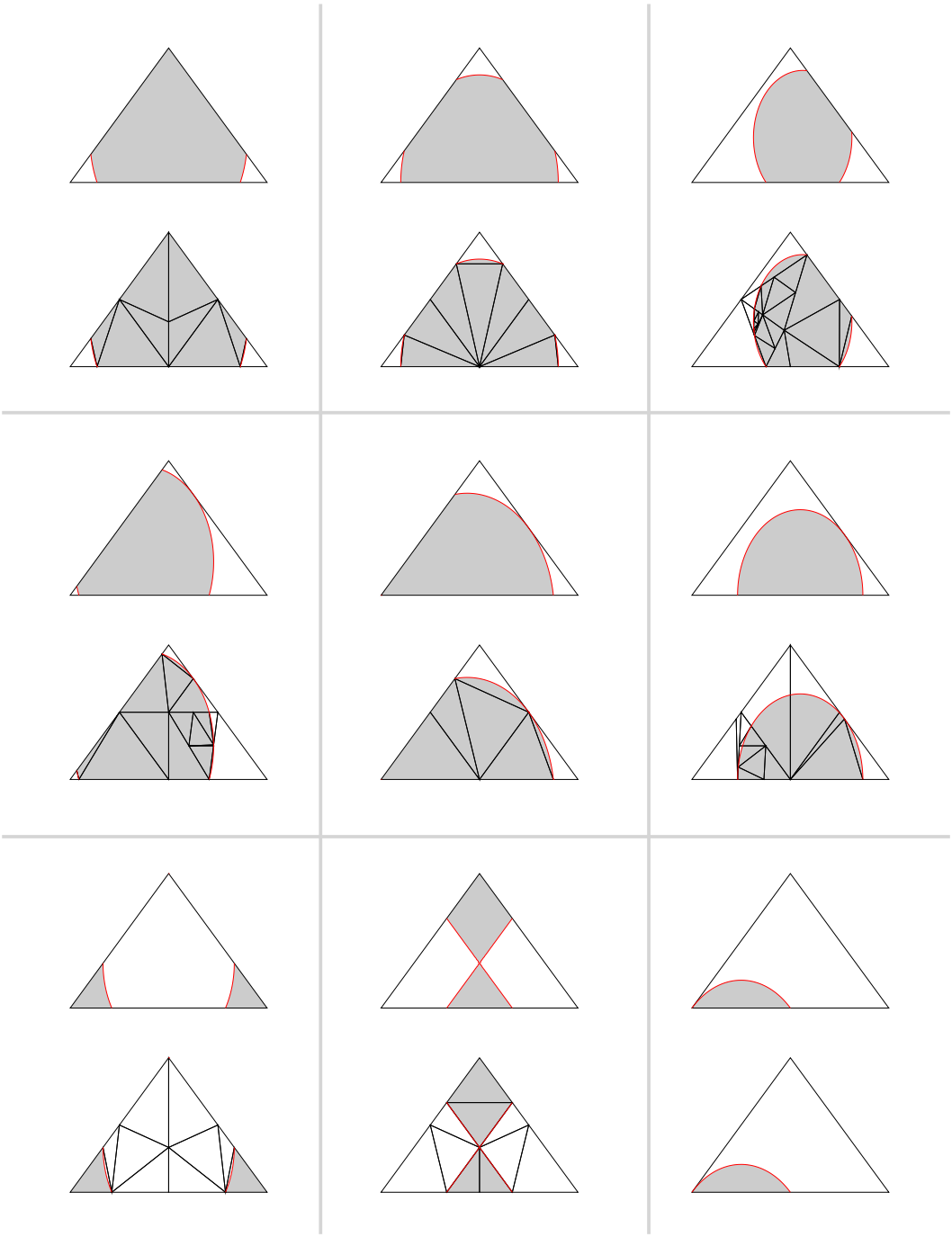


Figure 8.9 Examples of when a triangle is split into sub-triangles to avoid edge cases or simplify integration. The triangles are outlined in black, the shadows are the gray areas, and the curves of intersection are the red curves. One example in the bottom-left shows a case where a triangle does not need to be split.

surface (note that this is not possible for higher order polynomials). The curve of intersection is a quadratic, rational Bézier curve. All one has to do is calculate the middle control point, which is located at the intersection of lines tangent to curve at the beginning and ending Bézier points (where the curve of intersection intersects the triangle). The curve of intersection, tangent lines, and control points are shown in Fig. 8.10. How to calculate the middle control point and weight is explained in Appendix B.

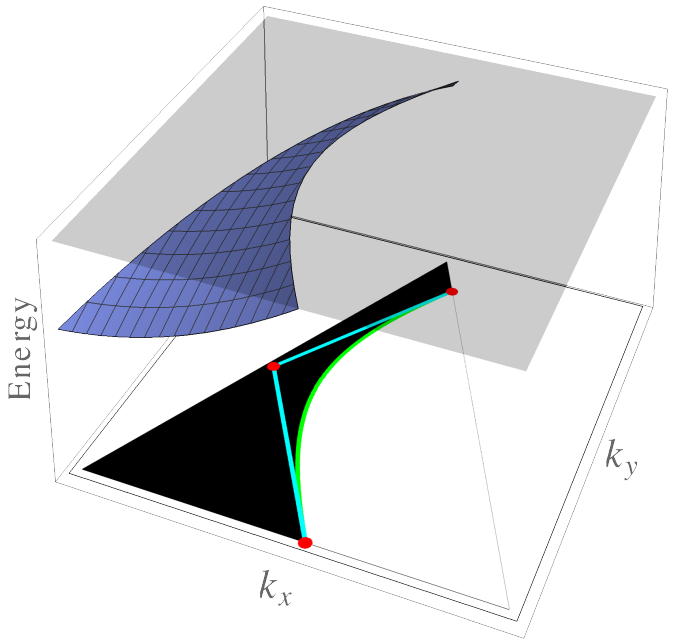


Figure 8.10 The curve of intersection where a quadratic surface intersects the Fermi plane is a quadratic curve that can be represented exactly with a rational quadratic Bézier curve. In this plot, the curve on intersection has been projected onto the $(k_x \times k_y)$-plane. A quadratic Bézier curve is described by its three control points. Two of the control points are located where the curve of intersection intersects the triangle. The middle control point is located at the intersection of lines tangent to the curve (shown in cyan) at the other two control points.

We further simplify the calculation of areas and volumes by expressing all integrals in terms of the control point triangle. The control point triangle is the triangle formed by the control points of the Bézier curve, or the red points in Fig. 8.10. We calculate the area within a control point triangle

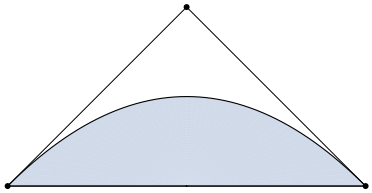


Figure 8.11 The canonical, control-point triangle. The control points are the black dots located at $(-1,0)$, $(1,0)$, $(0,1)$. The Bézier curve is the black curve inside the triangle (not including the straight lines along the border of the triangle). The control point triangle is the triangle formed by drawing lines that connect the control points. The analytic expressions for area and volume in Eqs. 8.18 and 8.20 give the area and volume of the blue shaded region.

for a canonical triangle with corners located at $(-1,0)$, $(1,0)$, $(0,1)$. The area A is dependent on only the weight of the middle rational Bézier point w and takes the form

$$A(w) = Re(w(w + (2\arctan(\sqrt{\frac{-1+w}{-1-w}}))/(\sqrt{-1-w}\sqrt{-1+w})))/(-1+w^2)) \tag{8.18}$$

This expression has a singularity at $w = 1$. When calculating areas, w is compared to 1 and if it is sufficiently close, a power expansion of Eq. 8.18 about $w = 1$ is used instead of the analytic expression for A. Enough terms are included in the expansion that the error from excluded terms is less than machine precision. The first three terms in the power expansion of Eq. 8.18 are

$$A(w) = 2/3 + 4/15(-1+w) - 6/35(-1+w)^2 + \ldots \tag{8.19}$$

The volume beneath the quadratic within the shadow in the control point triangle also has an analytic form. The volume depends on the the weight of the middle control point as well as the six coefficients of the quadratic $\mathbf{c} = (c_0, c_1, c_2, c_3, c_4, c_5)$:

$$\begin{aligned} V(w,\mathbf{c}) = Re((&w(\beta w(-32c_1 + 33c_2 - 32c_3 + 46c_4 + 33c_5 - \\ &2(-26c_0 + 18c_1 + 13c_2 + 18c_3 + 12c_4 + 13c_5)w^2 + 8\gamma w^4) + \\ &6(5c_2 + 6c_4 + 5c_5 + 4(c_0 - 5(c_1 + c_3) + c_4)w^2 + 16c_0 w^4)\arctan(\beta)))/(8\gamma\beta(-1+w^2)^3)), \end{aligned} \tag{8.20}$$

where $\beta = \sqrt{\frac{-w-1}{w-1}}$ and $\gamma = c_0 + c_1 + c_2 + c_3 + c_4 + c_5$. This expression also has a singularity at $w = 1$, and the first 3 terms in a power series expansion about $w = 1$ are

$$V(w, \mathbf{c}) = 6/7 + (2(-11c_0 - 5(c_1 + c_3) + c_4))/(35\gamma) + 4/105(5 + (3c_0 + 5(c_1 + c_3) - c_4)/\gamma)(w-1) + \ldots \tag{8.21}$$

In practice, we keep the first ten terms in the expansions of A and V whenever w is within 0.01 of 1.

8.2.3 Volumes and hypervolumes of quadratics in 3D

As many edge cases as there are for areas and volumes in 2D, there are even more in 3D. A different approach is taken in 3D that takes advantage of the 2D analytic code. In 3D, the IBZ is split into tetrahedra instead of triangles. Our approach to calculating volumes and hypervolumes is to take slices of the tetrahedron parallel to the face with the largest area along a line from the midpoint of the face to the opposite corner. For each slice, the area or volume of the quadratic may be calculated analytically with the 2D code. A value for the area or volume is obtained for each slice and then these values are integrated numerically with Gauss-Legendre quadrature. The slices are taken parallel to the face with the largest area to minimize the number of slices (or maximize the density of slices per unit length). Examples of slices are shown in Fig. 8.12 and the an example of a volume calculation is show in Fig. 8.13.

The convergence rate of the numerical integral performed to calculate Fermi volume and band energy in 3D is improved by splitting the interval of integration at points where the slices are tangent to the surface of the shadow. These are called silhouette points. In the calculation of the Fermi volume (for example), the curve formed by the Fermi areas of each slice has an infinite derivative when the slice is tangent to the surface of the shadow. An example silhouette point is shown in Fig. 8.14. The improvement from using Gaussian Quadrature and splitting the interval at silhouette

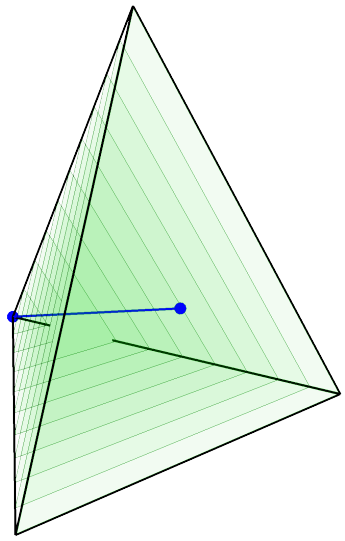


Figure 8.12 When calculating volumes or hypervolumes in 3D, a tetrahedron is sliced parallel to the face with the largest area. The slices are distributed along a line that travels from the midpoint of the face to the opposite corner, and the distribution of slices follows from a numerical quadrature scheme. The slices shown are for Simpson's method, which has the slices evenly spaced.

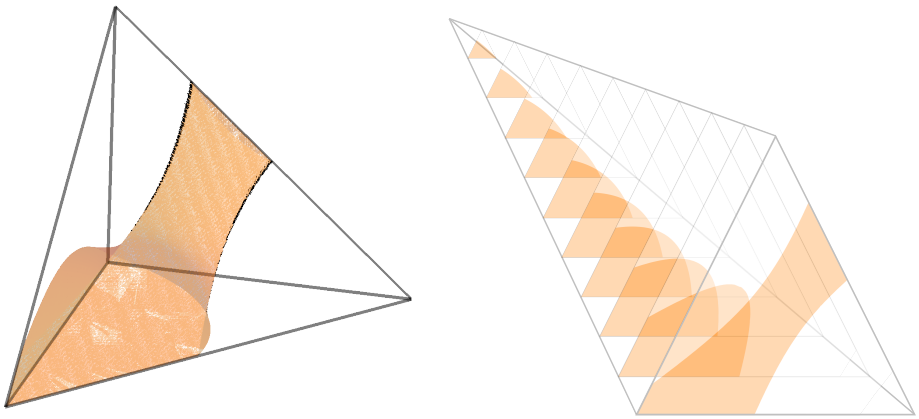


Figure 8.13 In 3D, the Fermi area becomes a Fermi volume and the band energy becomes a hypervolume. On the left is the Fermi volume or the volume bounded by the tetrahedron and level surfaces of a quadratic or the shadow of the quadratic polynomial within the tetrahedron. The band energy is the integral of the quadratic over the Fermi volume. It is possible to numerically calculate the Fermi volume by taking slices of the tetrahedron parallel to one of the faces, calculating the Fermi area of each slice, and then integrating the Fermi areas numerically with numerical quadrature, which in our case is Gauss-Legendre quadrature. On the right are the slices of the tetrahedron with the shadows of the quadratic shown in orange.

points is shown in Fig. 8.15, where it is compared to Simpson's method. In this same figure, one observes corners in the curve of $\frac{dA}{dt}$. These corners occur when the quadratic surface intersects an edge of the tetrahedron and negatively affect the integration accuracy. An improvement that has not yet been implemented would split the interval at values of t where the derivative is discontinuous.

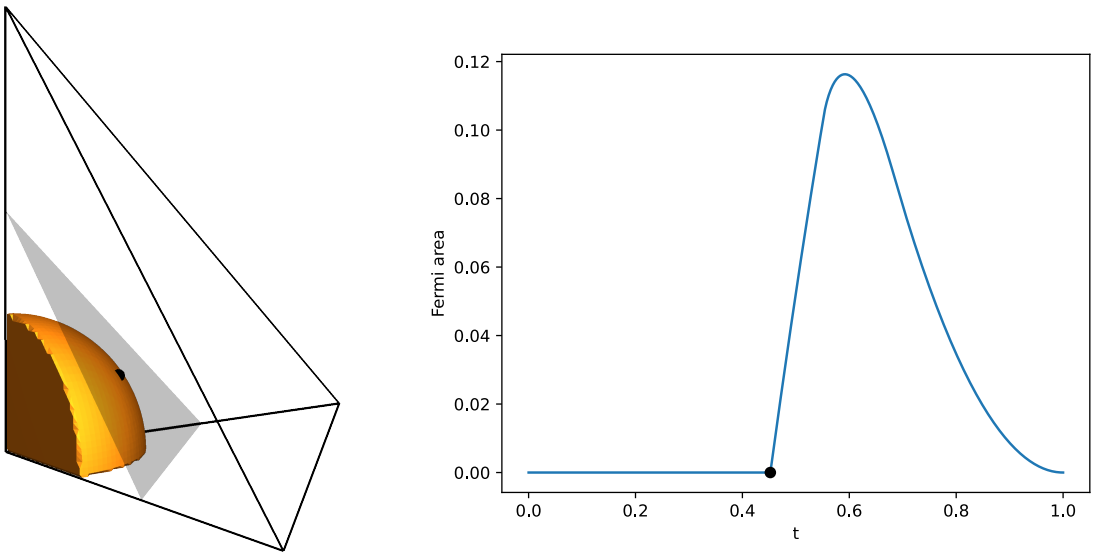


Figure 8.14 Trouble points in the calculation of Fermi volume and band energy are located where the surface of the shadow is tangent to slices of the tetrahedron. One such point is shown on the left. At this point, the curve formed by Fermi areas on each slice has a discontinuous derivative at the silhouette point, shown as a black dot. The accuracy of the integral is improved by splitting the interval at the silhouette point. Splitting reduces the number of slices required and improves the efficiency of the algorithm.

Fig. 8.16 shows a tetrahedron sliced into planes parallel to $\mathbf{P}_{2000}\mathbf{P}_{0200}\mathbf{P}_{0020}$. A plane $\mathbf{P}(v)$ in that family of planes has vertices $(1-v)*(\mathbf{P}_{2000},\mathbf{P}_{0200},\mathbf{P}_{0020})+v*(\mathbf{P}_{0002},\mathbf{P}_{0002},\mathbf{P}_{0002})$. The area or volume of the quadratic of each plane can be calculated in closed form. Denote the closed-form integration of plane $\mathbf{P}(v)$ by $f(v)$. We want to find values of v for which $f(v)$ is not C^1.

Fig. 8.17 shows the labels for the coefficients of the quadratic function.

$f(v)$ is not C^1 for a value of v at which a plane $\mathbf{P}(v)$ is tangent to the quadratic surface. There are

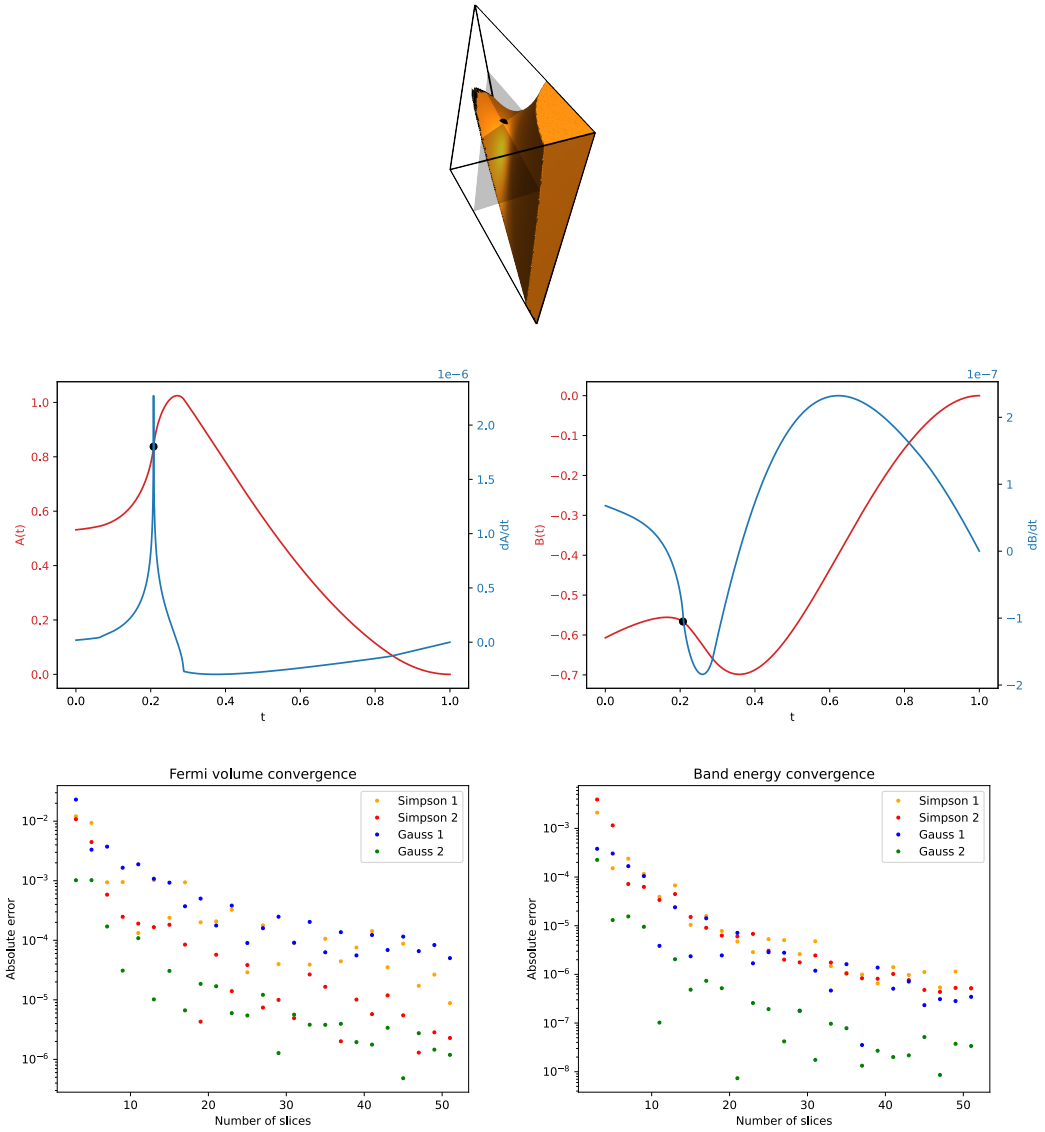


Figure 8.15 The Fermi volume or band energy of a tetrahedron is computed by numerically integrating $A(t)$ and $B(t)$, respectfully, where $A(t)$ is the Fermi area of a triangular slice of the tetrahedron and $B(t)$ is the band energy of the slice. For each slice (each value of t), A and B are computed analytically using the 2D integration code described in the previous section. $A(t)$ and $B(t)$ are plotted along with their derivative, where one observes the derivative of $A(t)$ is singular at the silhouette point. The convergence plots at the bottom of the figure show Gaussian integration when splitting the integration interval performs best. The number in the legend indicates the number of intervals integrated over.

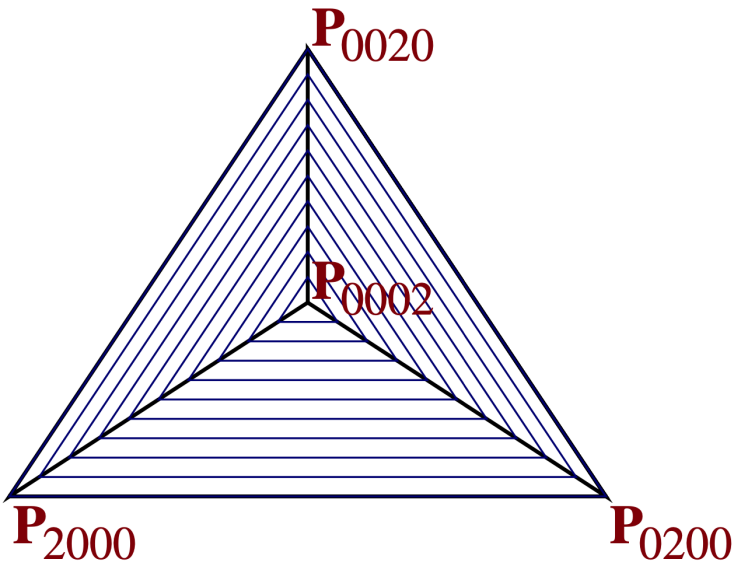


Figure 8.16 Slices through a tetrahedron

always two such planes, although we only care about real values for which $V \in [0,1]$. The desired values of v are computed as follows.

Denote the Barycentric coordinates by s, t, u, v. Denote

$$d_1 = s*c_{2000} + t*c_{1100} + u*c_{1010} + v*c_{1001}$$

$$d_2 = s*c_{1100} + t*c_{0200} + u*c_{0220} + v*c_{0101}$$

$$d_3 = s*c_{1010} + t*c_{0110} + u*c_{0020} + v*c_{0011}$$

$$d_4 = s*c_{1001} + t*c_{0101} + u*c_{0011} + v*c_{0002}.$$

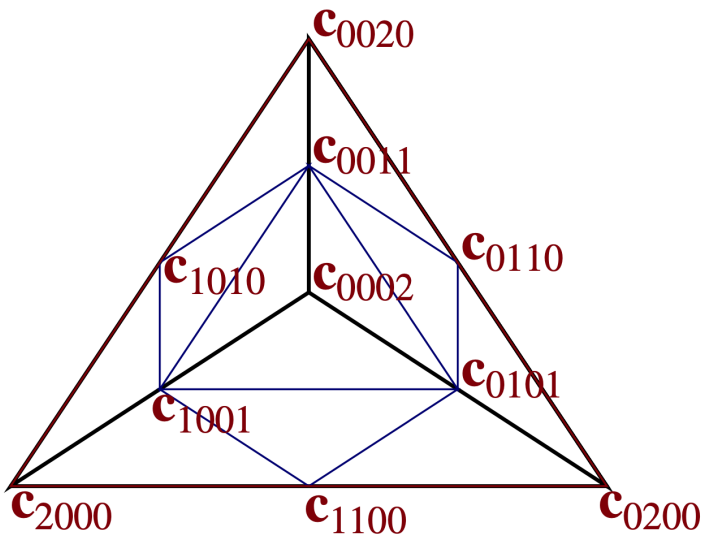


Figure 8.17 The coefficients of a quadratic function

Then the desired values of v satisfy

$$s+t+u+v=1$$

$$d_1 == d_2$$

$$d_1 == d_3$$

$$s*d_1+t*d_2+u*d_3+v*d_4=0$$

8.3 Approximating band energy error contributions

8.3.1 Error cancellation of sheet crossings

At each k-point we solve a matrix equation of the form

$$H(\mathbf{k})\psi(\mathbf{k})=\varepsilon(\mathbf{k})\psi(\mathbf{k}) \tag{8.22}$$

where $\varepsilon(\mathbf{k})$ are the eigenvalues and $\psi(\mathbf{k})$ are the eigenstates. The $\varepsilon(\mathbf{k})$ over the Brillouin zone constitute the band structure or algebraic surface. A single sheet S_i is continuous (Bouckaert et al. 1936), except where there are degenerate eigenvalues where two or more sheets cross. The sheets are separated by sorting the eigenvalues from lowest to highest, and assuming all of the lowest energies belong the the first sheet, the second lowest to the second sheet, and so on. The sorting of the eigenvalues causes continuity of the sheets to drop to C^0 (discontinuous derivatives) at sheet crossings.

Intuitively, we might expect interpolation errors at sheet crossings to cancel: the interpolation of the sheet above overestimates the sheet at the crossing, and the interpolation of the sheet below underestimates the sheet at the crossing. But band energy errors from sheet crossings are no worse than band energy errors of sheets that do not cross, provided the crossing does not occur near the Fermi level. This is due to the commutative property of integration and summation. Take as an example a 2D IBZ that has been split into triangular subelements. In one triangle T, there are two sheets S_1, and S_2 away from the Fermi level, and we want to calculate the sum of the volumes V beneath each sheet:

$$V = \sum_{i=1}^{2} \int_{\mathbf{k}\in T} S_i(\mathbf{k})\,\mathrm{d}\mathbf{k}. \tag{8.23}$$

V is calculated by interpolating each sheet with a quadratic polynomial, $p_i(\mathbf{k})$, and integrating the polynomials over the triangle

$$\begin{aligned} V &\approx \sum_{i=1}^{2} \int_{\mathbf{k}\in T} p_i(\mathbf{k})\,\mathrm{d}\mathbf{k} \\ &= \int_{\mathbf{k}\in T} \sum_{i=1}^{2} p_i(\mathbf{k})\,\mathrm{d}\mathbf{k}, \end{aligned} \tag{8.24}$$

where the commutative property is used to switch the order of integration and summation. Even though S_1, and S_2 are only C^0 continuous, the sum of S_1 and S_2 is C^∞. Let $S_\sigma = S_1 + S_2$. It can be shown that the the quadratic interpolation of S_σ is $p_\sigma = p_1 + p_2$. This means the approximate volume obtained by interpolating the C^0 continuous sheets, integrating the polynomials p_1 and p_2

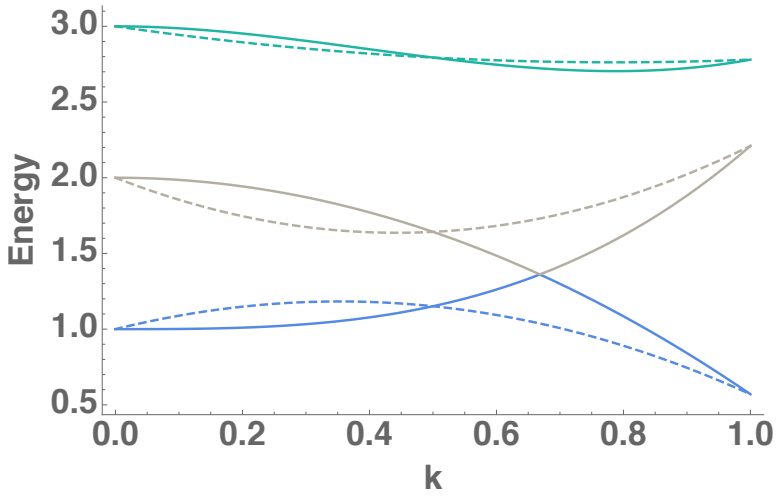


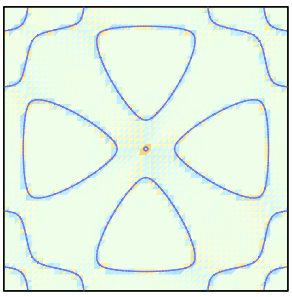
Figure 8.18 A 1D demonstration of interpolation errors at sheet crossings beneath the Fermi level cancelling. The solid blue and gray lines are sheets that cross, and the green line is the sum of the two sheets. The dashed lines are the quadratic interpolations of the sheets of the same color. Interpolating and integrating the blue and gray curves and then summing the integrals is equivalent to interpolating and integrating the green curve.

separately, and adding the two integrals is identical to interpolating the C^∞ continuous sheet S_σ and then integrating. This is illustrated in Fig. 8.18. For this reason, integration errors from sheet crossings beneath the Fermi level are no more severe than integration errors from sheets that do no cross. An example of integration error cancellation is shown in Fig. 8.19.

8.3.2 Errors of completely occupied sheets

Interval coefficients provide a means of estimating the error in the quadratic approximation of the band structure. A calculation of the band energy with a perfect representation of the band structure would give no error. However, a perfect approximation would take an infinite number of eigenvalues, which is impossible and impractical. Zero error in the band energy is not necessary as the theory of density functional theory is currently limited in accuracy to about one meV. Once the band energy error is less than one meV, any additional eigenvalue evaluations that go towards a more accurate representation of the electronic band structure are wasted.

(a)



(b)

Figure 8.19 Heat maps of the integration error. The darker blue color represents positive error, and the orange negative error. In (a) is shown the error for each sheet. In (b) is shown the sum of the errors over all the the sheets along with the Fermi curve in dark blue. Errors from sheet crossings cancel, and the largest errors appear in the tiles along the Fermi curve. In particular, the dark blue and orange crosses in (a) do not appear in (b).

Low error in the band structure representation is only useful (for our purposes) because it decreases band energy error. As described in the previous section, it is possible for there to be very large errors in the approximation of the band structure and small errors in the band energy [1]. In particular, when two or more sheets of the band structure cross and the crossing is below the Fermi level, there are large approximation errors near the crossing, but errors in the band energy from each sheet cancel. If the placement of k-points was driven by approximation error (making the representation of the band structure more accurate), large numbers of k-points would be placed at sheet crossings below the Fermi level. These k-points are wasted because they do not significantly improve the band energy accuracy and are not the primary source of band energy error. Solving for eigenvalues is the largest cost in running a DFT calculation (see Sec. 4.4), so these k-points would increase the computational cost of running a DFT calculation without significantly improving the numerical accuracy of the calculation. Sheet crossings that occur at the Fermi level tell a different story and lead to large approximation and band energy errors.

In 1D, the approximate band energy error is the difference in area between the upper bounding, quadratic curve and lower bounding curve when both curves are below the calculated Fermi level and is the colored regions in Fig. 8.20. The true band energy error is the area between the dashed, black curves and solid, black curves. As mentioned, band energy errors cancel at sheet crossings below the Fermi level. Band energy errors for sheets below the Fermi level without crossings may also cancel if errors have both positive and negative signs, as is the case in Fig. 8.20. One can tell because the dashed, black curve is sometimes above and sometimes below the solid, black curve. However, there is no error cancellation for the approximated band energy error—they all have the same sign. Error cancellation is accounted for in the estimated of band energy error by summing the sheets and then interpolating the *sigma* sheet, which is shown in Fig. 8.21. The approximate error

[1] Band energy is the sum of the volumes of the band structure below the Fermi level within the Brillouin zone times two (spin degeneracy). In this dissertation, band energy also refers to the contribution to the band energy of a subelement of the Brillouin from one or more sheets of the band structure.

of the sigma sheet is much smaller than the sum of the band energy errors of the individual sheets.

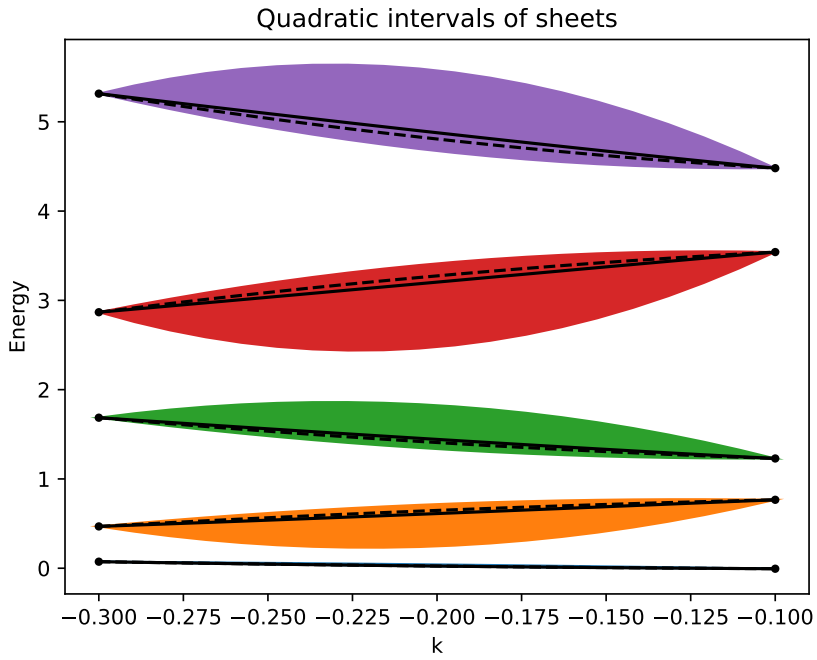


Figure 8.20 Part of the band structure between two k-points of the toy band structure from Fig. 8.6. The true sheets of the band structure between the two k-points are the solid black lines. The least-squares quadratics for each sheet are the dashed, black lines and the bounding quadratic curves obtained from the interval coefficients are the upper and lower bounding curves of the colored areas. The true band energy error is the sum of the areas between the dashed and solid black curves. The approximate band energy error is the sum of the colored areas. Whereas true band energy errors may cancel (to a degree) because errors may have positive and negative signs, the approximate band energy errors do not cancel.

8.3.3 Calculating the Fermi level interval

The band energy calculation is a two step calculation: the Fermi level for a given representation of the band structure is first computed and then the volume of the band structure below the Fermi level is computed. Error in the approximation of the band structure contributes to error in the Fermi

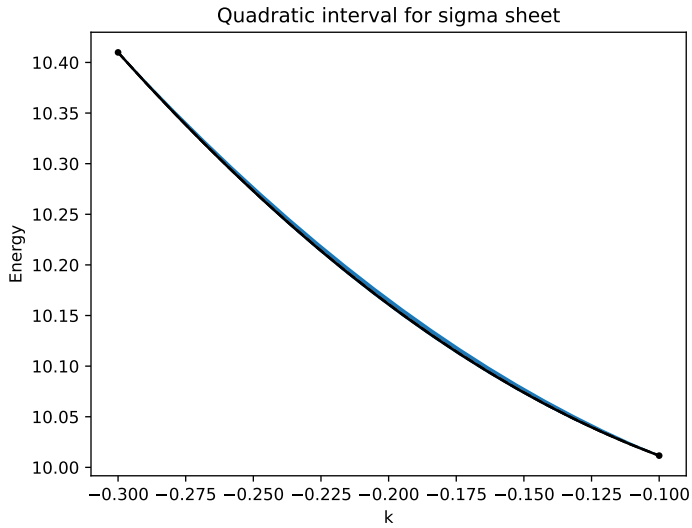


Figure 8.21 The sigma sheet between two k-points obtained by summing the sheets in Fig. 8.20. The estimated band energy error (the shaded blue region) for sheets entirely below the Fermi level error is much smaller and more accurate when using the sigma sheet, rather than the individual sheets.

level, so Fermi level error also contributes to band energy error. Once again, we are interested in lowering the Fermi level error only when it results in less band energy error. The Fermi level determines the locations where the band structure is discontinuous (or the domain of integration). In the rectangular method, the errors away from the Fermi surface are proportional to the width of the rectangles (in 1D; in 2D, they are proportional to the area of the base of the "rectangle", ...) and the errors of rectangles on the Fermi surface are proportional to the height of the rectangles. Errors in the Fermi level change the Fermi surface, which changes the domain of integration for the band energy integral (or the location of discontinuities). This implies, with sufficient error, rectangles outside the *true* Fermi surface may be included or rectangles inside the true Fermi surface may be excluded. The combination of the two effects—larger errors and uncertainty in the integration domain—make locations around the Fermi surface the locations of largest band energy error. As more k-points are added near the Fermi surface, errors may decrease until they are comparable to

errors away from the Fermi surface.

The Fermi level calculation is a root-finding process where the value of the estimated Fermi level is adjusted until the sum of the shadows of the sheets of the band structure has a predetermined value called the Fermi area, which is proportional to the number of electrons and the size of the Brillouin zone. The root-finding process ends once the calculated Fermi area is "close enough" to the true Fermi area. This introduces a numerical tolerance for the Fermi area into the quadratic integration algorithm. It may be possible to determine a large tolerance for the Fermi area that minimizes the number of root-finding iterations. For example, numerical derivatives may be used to estimate the error in the band energy that results from error in the Fermi area:

$$\frac{\mathrm{d}B}{\mathrm{d}A}\Delta A < \Delta B \tag{8.25}$$

where $\frac{\mathrm{d}B}{\mathrm{d}A}$ is the derivative of the band energy with respect to the Fermi area at the estimated Fermi level, ΔA is the Fermi area tolerance, and ΔB is the desired error for the band energy. For the integration tests in this dissertation, the finite tolerance parameter for the Fermi area is made smaller than necessary so that errors in the Fermi level from the convergence of the Fermi area are inconsequential compared to other sources of band energy error.

Error in the Fermi level for a given approximation of the band structure is determined by calculating the Fermi level using the upper limit of the coefficient intervals and then the lower limit of the coefficient intervals. These two values determine the Fermi level interval. The estimate of the Fermi level is calculated using the least-squares coefficients. Numerical derivatives may also be used to determine tolerances for the allowable Fermi level error for an allowable band energy error:

$$\frac{\mathrm{d}B}{\mathrm{d}l}\Delta l < \Delta B \tag{8.26}$$

where $\frac{\mathrm{d}B}{\mathrm{d}l}$ is the derivative of the band energy with respect to the Fermi level coordinate at the estimated Fermi level, Δl is the Fermi level tolerance, and ΔB is the desired error for the band energy.

In practice, a Fermi level tolerance makes possible mesh refinement driven by Fermi area error until the Fermi level error is less than a tolerance that is determined by the band energy tolerance. Adaptive mesh refinement for results in this dissertation is driven solely by band energy error. Since error in the Fermi level affects band energy error, estimates of band energy errors take into account Fermi level error.

Estimating allowed Fermi level error with derivatives of the band structure may not be very robust because the derivatives change when the representation of the band structure changes. Estimating allowable Fermi area error for a given band energy error is less predictable, especially if the Fermi level happens to lie at a Van Hove singularity or a location where the density of states ($\frac{\mathrm{d}B}{\mathrm{d}A}$) is singular. At Van Hove singularities, the derivatives may change drastically when the representation of the band structure changes, which is shown in Fig. 8.22. Ideally, derivatives would become more consistent as the representation of the band structure changes less and less as more simplices are added to the IBZ mesh but until then, estimates of allowable errors for the Fermi area and Fermi level are inconsistent.

Relative tolerances are necessary because, in the high-throughput approach, typically the band energy tolerance is a fixed, absolute tolerance whereas the magnitude of the band energy varies by material. Materials with larger band energies take longer to converge to within the band energy tolerance, and the benefits of the quadratic integration algorithm are greater for smaller band energy tolerances.

There are many numerical methods to find the roots of a function. The implementation of the quadratic integration algorithm has the option to use the bisection or Chandrupatla's root-finding methods (Chandrupatla 1997). Chandrupatla's method results in far fewer iterations when calculating the Fermi level and is the one used for all results in this dissertation. For adaptive mesh refinement, one may use the Fermi level interval before refinement as a window in which to search for the Fermi level after refinement. This too significantly reduces the number of root-finding

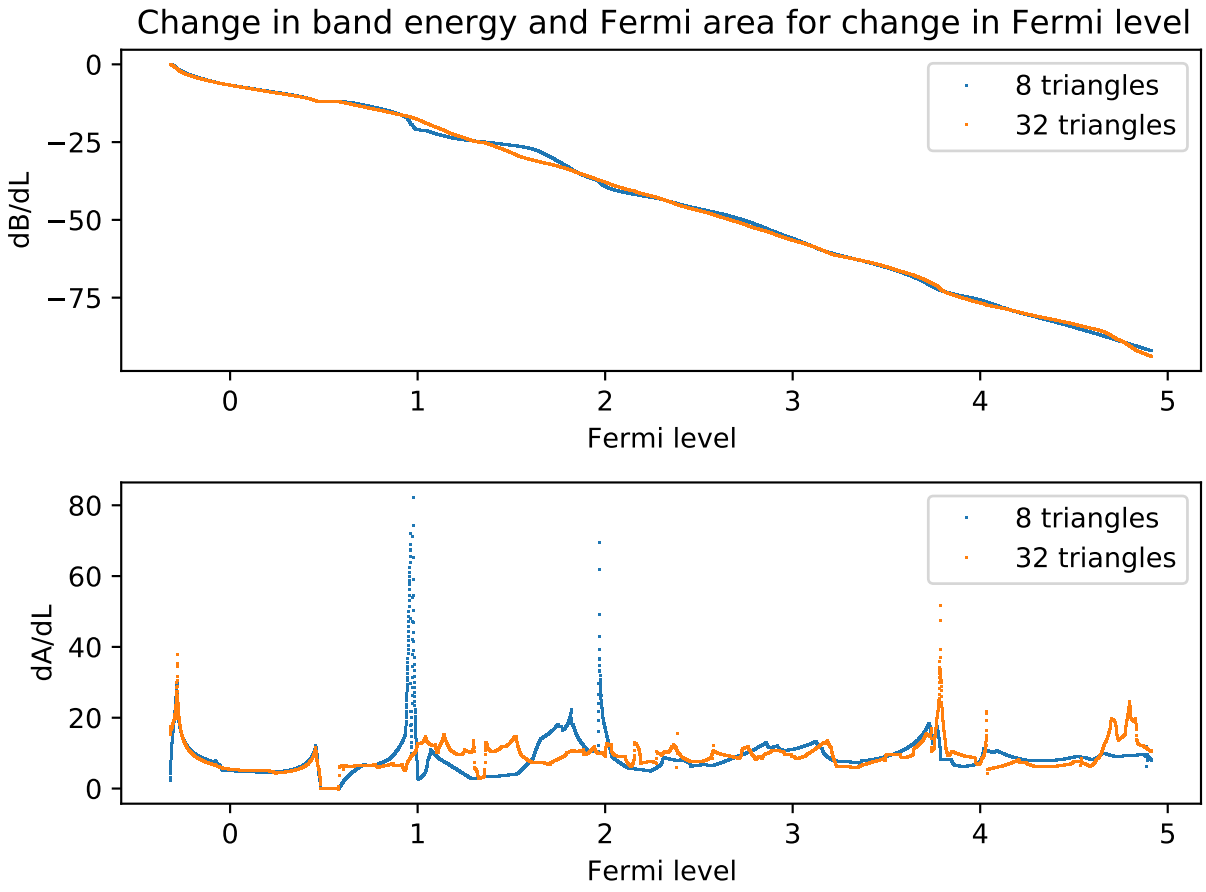


Figure 8.22 Plotted are numerical approximations of $\frac{\mathrm{d}B}{\mathrm{d}A}$ and $\frac{\mathrm{d}B}{\mathrm{d}l}$ for different values of the Fermi level coordinate for a piece-wise quadratic approximation of the band structure for M41 (see Sec. 9.1). The density of states ($\frac{\mathrm{d}B}{\mathrm{d}A}$) changes dramatically at Van Hove singularities when the representation of the band structure changes.

iterations for subsequent calculations of the Fermi level, which is demonstrated in Fig. 8.23 for one of the 2D models (m31) described in Sec. 9.1. The Fermi level interval does not always bracket the root because containment of the eigenvalue sheets is not 100%, in which case extra iterations are required to extend the interval so that it does bracket the root.

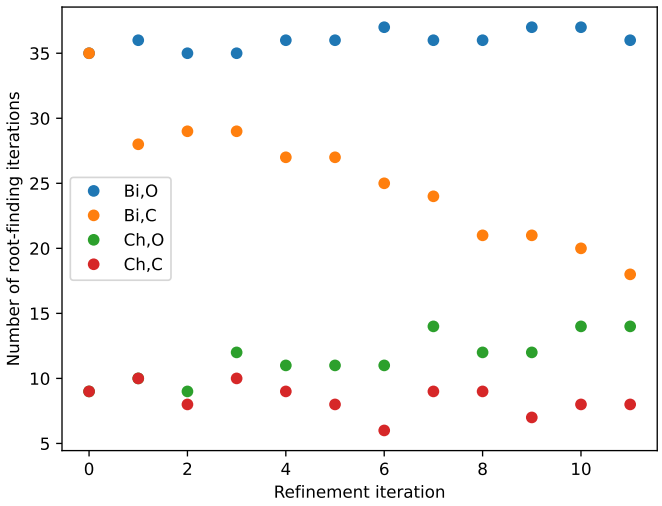


Figure 8.23 The number of root-finding iterations—and the efficiency of the integration algorithm—depends on the size of the interval that brackets the root and on the root-finding algorithm. Bisection and Chandrupatla's root-finding methods are compared along with bracketing or not bracketing the root with the Fermi level interval. In the legend, Bi - Bisection, Ch - Chandrupatla, O - open, and C - closed (bracketed by the Fermi level interval). When the Fermi level interval is not used, the largest and smallest eigenvalues of the sheets included in the computation are used. The number of root-finding iterations going from Bi,O to Ch,C is reduced (on average) by a factor of 4.4 for this example.

8.3.4 Errors of partially occupied sheets

A quadratic within a subelement of the IBZ for a single sheet is completely occupied if the upper quadratic from the upper limit of the interval coefficients is below the lower limit of the Fermi level interval. When this is the case, the band energy error is calculated as described previously in this

chapter. When this is not the case, the band energy error is calculated differently and is such that the sum of band energy errors from multiple sheets over one subelement do not cancel.

The band energy error for partially occupied quadratics is illustrated in Fig. 8.24 in 1D. The true band energy is the area of the true sheet below the true Fermi level. The approximate band energy is the area of the least-squares quadratic (the least-squares quadratic is the quadratic curve obtained from the least-squares coefficients) below the least-squares Fermi level (the Fermi level obtained using least-squares quadratics for all subelements and sheets). The true band energy error is the difference between the areas, which are show in gray. The estimate band energy error is the difference in the areas shown in red. The lower-interval band energy is the area of the upper-interval quadratic (the upper limit of the coefficient intervals) and the lower-interval Fermi level. The upper-interval band energy is the area of the lower-interval quadratic (the lower limit of the coefficient intervals) and the upper-interval Fermi level. The estimated band energy error is difference between the lower-interval and upper-interval band energies. The lower limit of Fermi level interval is obtained from a calculation of the Fermi level with the upper limit of the interval coefficients, and the upper limit of the Fermi level interval is obtained from a calculation of the Fermi level with the lower limit of the interval coefficients. So the estimated band energy error is the difference between the band energies obtained using the upper and lower interval coefficients.

It is critical that Fermi level error be included when calculating the estimated band energy errors if errors are to be estimated accurately. When Fermi level errors are not considered, band energy errors of partially occupied sheets are much less that the true band energy errors. In Fig. 8.25, the distribution of true band energy errors and estimated band energy errors are shown for toy 2D empirical pseudopotential model. Sigma errors, errors of completely occupied sheets are the same in both cases, but partial errors, errors of partially occupied sheets, are much small than the true band energy error. This is seen in the distribution of total errors (sigma + partial error) and the distribution of partial errors.

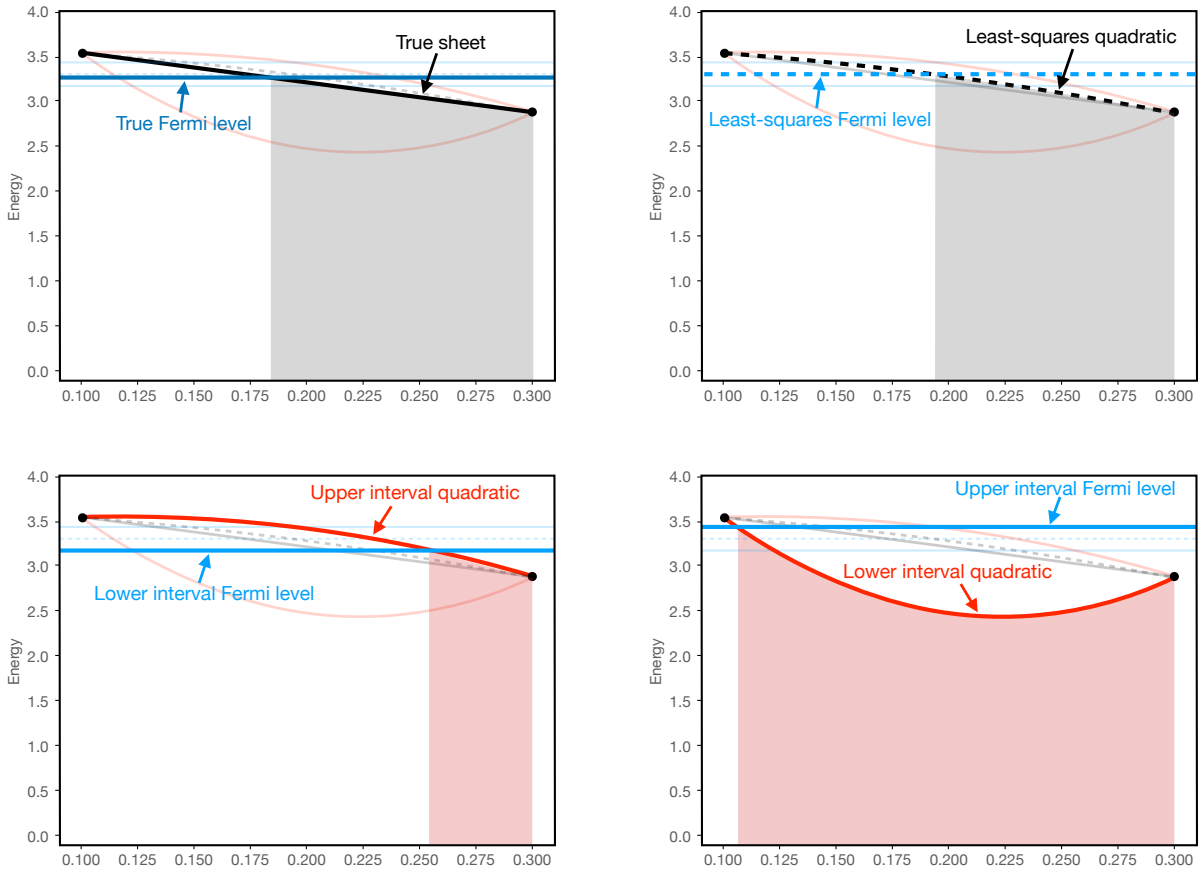


Figure 8.24 A 1D illustration of the true and approximated band energy errors. In the top-left frame, the true band energy is the area shaded in gray. In the top-right frame, the estimated band energy is the area shaded in gray. The true band energy error is difference between the two gray-shaded areas. The lower-interval band energy is the area shaded in red in the lower-left frame. The upper-interval band energy is the area shaded in red in the lower-right frame. The estimated band energy error is the difference between the two red-shaded areas.

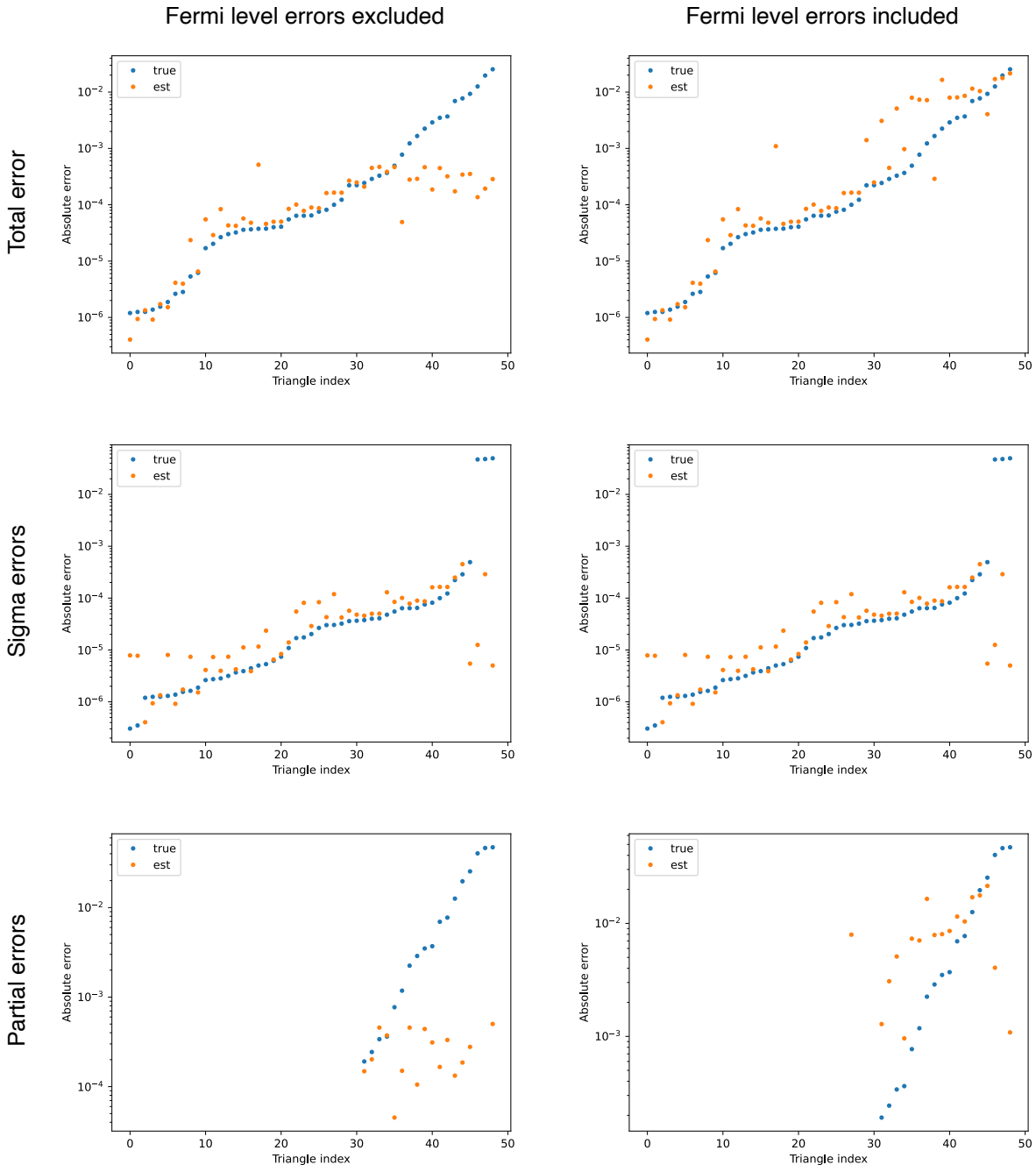


Figure 8.25 The distribution of band energy errors for a quadratic approximation of the band structure of a 2D toy model M13 (see Sec. 9.1) when including and excluding Fermi level error in the calculation of band energy errors. Partial band energy errors (errors of partially occupied sheets) are much smaller than true partial band energy errors when Fermi level error is not considered.

8.3.5 Limitations of estimated band energy errors

Ideally, the interval quadratics are a minimal bound of the true eigenvalue sheets and provide a means of estimating the *maximum* error in the approximation of the eigenvalue sheets. If the points in the calculation of the interval coefficients come from a dense grid inside the tetrahedral subelements (in 3D), the interval coefficients bound the true eigenvalue sheets a large percentage of the time. Since points are selected from outside the triangle or tetrahedron, a parameter is introduced into the algorithm that scales the widths of the interval coefficients in order to make the bounding quadratics contain the true eigenvalue sheets more often. The parameter that scales the widths of the algorithm is chosen such that the initial coarse mesh has a containment percentage (the percentage of the time the true eigenvalue sheet is bounded by the quadratics) near 90%. The value of the parameter needed to reach 90% average containment is 2. Containment percentage is determined by sampling each simplex uniformly of the IBZ tesselation comparing an eigenvalue to quadratic values from each sample point.

An initial containment percentage around 90% is deemed sufficient because the containment percentage improves as the density of mesh points increases. This is observed in Fig. 8.26. Consequently, band energy error bounds may not bound the band energy error for small mesh sizes, and the band energy error bounds bound the true band energy error more often as the mesh is refined.

Regardless of the size of the mesh, when the eigenvalue surface is not bounded by the quadratic surfaces, there are a small number of cases where the eigenvalue surface is not bounded very well by the quadratics. For the example in 2D in Fig. 8.26, the distribution of relative containment errors is shown in Fig. 8.27. To measure relative containment error, at a point within the simplex, the eigenvalue surface and quadratic surfaces are evaluated. If the eigenvalue lies between the quadratic values, the relative containment error is zero. If the eigenvalue does not lie between the quadratics, the relative containment error is the minimum difference between the eigenvalue and quadratics

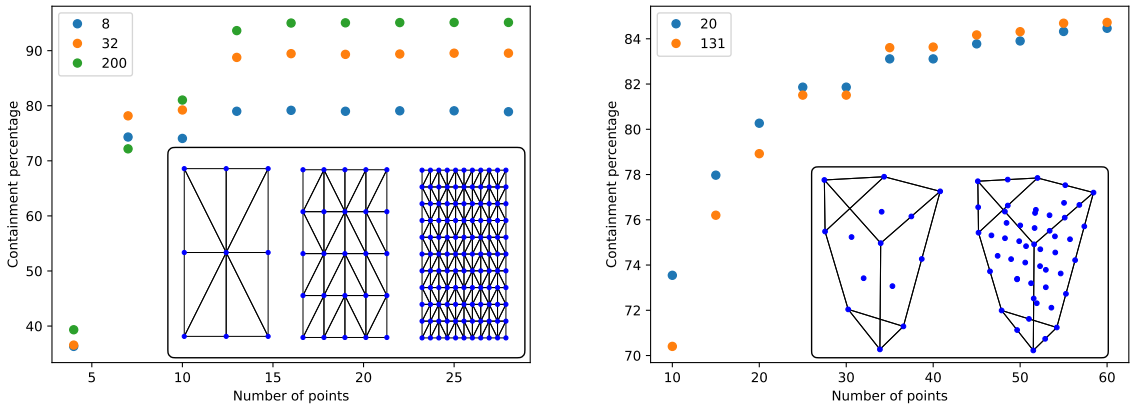


Figure 8.26 The containment percentage depends on the points used in the calculation of interval coefficients. As the density of points increases, mesh points get closer to the triangle and quadratic surfaces bound the eigenvalue surface more often. The number in the legend is the number of simplices in the mesh, and along the x-axis is the number of points used in the calculation of interval coefficients. These percentages are for unscaled interval coefficients. The test was performed on M41 in 2D and an EPM for indium in 3D (see Sec. 9.1 for a description of the models). The insets show the IBZ, mesh simplices (only for 2D), and mesh points.

divided by the difference of the quadratics. To make the quadratic bounds an eigenvalue with a a relative containment error of 1, the interval coefficients would need to scaled at minimum by a factor of 3. This means that the largest relative confinement errors in Fig. 8.27 would need the interval coefficients to be scaled by a factor of around 60 to be bounded.

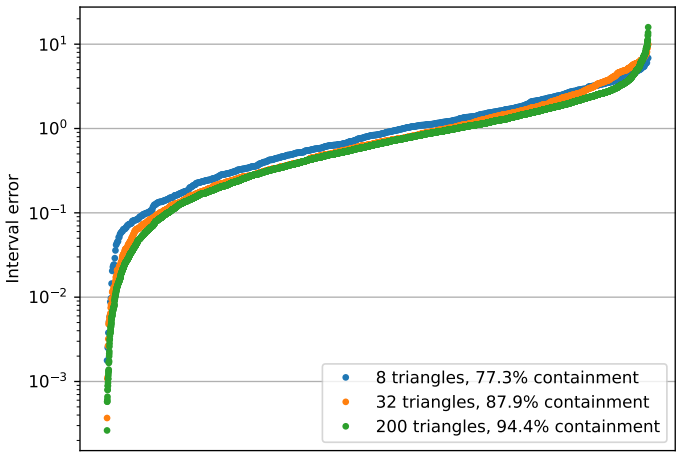


Figure 8.27 The distribution of relative containment errors for M41 for varying sizes of the mesh. Although the containment percentage increases as the size of the mesh increases, the distribution of relative containment errors stays about the same. Regardless of the density of mesh points, the interval coefficients would need to be scaled by a factor of around 60 to guarantee the true eigenvalue sheets are bounded by the quadratic surfaces for the worst cases. The number of points used in calculating interval coefficients was set to 16 for this test.

Where containment usually fails is at sheet crossings. The eigenvalue with the largest relative containment error in Fig. 8.27 comes from a sheet crossing and is shown in Fig. 8.28. When calculating the band energy, sheets entirely below the Fermi level interval are summed and then interpolated. The quadratic surfaces for the sigma sheet (the sheet that results from summing the eigenvalues) bound the sigma sheet and the bound is minimal (see Fig. 8.21). Relative containment errors from sheet crossings are inconsequential because the sheets are summed and then interpolated,

and the sigma sheet does not have sharp corners like those of the sheets that cross. It is for this reason that a containment percentage of less than 100% is justified. For a sheet within the Fermi level interval, the quadratic sheets fail to bound the eigenvalue sheet when the sheet crosses another sheet. Most of the band energy error comes from sheets that intersect the Fermi plane and of those sheets, sheets that cross have the largest band energy errors. Since the eigenvalue sheet with a crossing at the Fermi plane is not bounded by the quadratic sheets, the estimate of the maximum band energy error is not guaranteed to be greater than the true band energy error. In summary, the algorithm gives a poor band energy error bound where the band errors are largest.

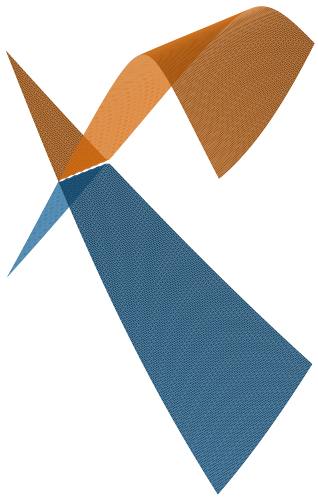


Figure 8.28 The sheet with the largest relative containment errors comes from a sheet crossing.

In Fig. 8.24, one may notice that the true band energy error (the difference between the gray shaded regions) and the estimated band energy error (the difference between the red shaded areas) are not very similar. We have found that the band energy error estimates do well at qualifying the error but do poorly at quantifying the band energy error. Consequently, the adaptive integration algorithm places points where there are large band energy errors but is unable to accurately estimate the amount of error in the band energy. In other words, the algorithm does not know when the band

energy has been calculated to a prescribed accuracy. When the sum of band energy errors over all subelements and sheets is used to estimate the total band energy error, the algorithm obtains a band energy error that is often 100 times more accurate than requested and requires more computational resources and time than necessary.

8.4 Error-driven adaptive mesh refinement algorithm

An outline of the quadratic algorithm follows. The minimum inputs of the algorithm are the lattice vectors, atomic basis, and the number of k-points to use in the computation.

1. Make the provided unit cell primitive. A primitive unit cell has a band structure with fewer sheets and crossings is easier to approximate with quadratic polynomials (see Fig. 7.11).

2. Compute the reciprocal lattice vectors from the real-space lattice. Minkowski reduce the reciprocal lattice vectors. Minkowski reduction of the reciprocal is required to ensure no points of the mesh outside of the IBZ are missed when extending the k-point mesh.

3. Compute the space group of the crystal. The space group makes possible the computation of symmetrically equivalent points outside the IBZ and within the BZ.

4. Compute the irreducible Brillouin zone. Integration is performed within the IBZ to improve the efficiency of integration with adaptive meshes.

5. Split the IBZ into simplices and sample each simplex uniformly. The number of points in each simplex is proportional to its size in an effort to make the mesh uniform.

6. Reduce by symmetry the k-points of the IBZ mesh to obtain a set of unique k-points. The points on the boundary of the IBZ may be equivalent to other points on the boundary. Also, extend by symmetry the k-points on the boundary of the IBZ.

7. Compute the eigenvalues of the unique k-points.

8. Extend the IBZ mesh by symmetry to include points nearby and outside of the IBZ. The point operators of the space group give points with the BZ. Translations by integer multiples of the reciprocal lattice vectors give points outside the BZ. Only keep points close to the IBZ. The k-points outside of the IBZ are needed to compute quadratic polynomial coefficients.

9. Use the Delauney method to tesselate the IBZ mesh and extended mesh with simplices.

10. Compute the least-squares and interval coefficients for each simplex of the IBZ tesselation and for each sheet of the bandstructure.

11. Compute the Fermi level and band energy with the least-squares coefficients and compute the Fermi level interval and band energy interval with the interval coefficients.

12. For each simplex, compute the sum of the band energy errors from occupied sheets and partially occupied sheets.

13. While the number of unique k-points is less than the desired number of k-points for the band energy computation

    (a) Add k-points to simplices with the largest band energy errors. The number of k-points added to each simplex and the number simplices to which k-points are added depend on adjustable parameters. For the results in this dissertation, k-points are added to the midpoints of the simplex edges, and one tenth of of the simplices in the IBZ tesselation are "split".

    (b) Reduce the new k-points by symmetry (points on the IBZ boundary may be equivalent by symmetry). Extend the new k-points on the boundary of the IBZ by symmetry.

    (c) Extend the new k-points by symmetry to equivalent points outside of the IBZ and include these points in the extended IBZ mesh.

(d) Use the Delauney method to tesselate the refined IBZ mesh and extended mesh with simplices.

(e) Compute the least-squares and interval coefficients for each simplex of the IBZ tesselation and for each sheet of the bandstructure. The number of sheets in the computation can possibly be reduced because the highest occupied sheet is now known. One can play it safe by including additional sheets.

(f) Compute the Fermi level and band energy with the least-squares coefficients and compute the Fermi level interval and band energy interval with the interval coefficients.

(g) For each simplex, compute the sum of the band energy errors from occupied sheets and partially occupied sheets.

A large number of parameters and tolerances go into the algorithm. A list of adjustable parameters of the algorithm are given below

- $def\_init\_msize = 3$: The initial size of the mesh over the IBZ
- $def\_num\_near\_neigh = 2$: The number of nearest neighbors included in calculations on interval coefficients
- $def\_fermiarea\_eps = 10^{-10}$: A convergence tolerance for the Fermi area in the calculation of the Fermi level
- $def\_fermilevel\_method = 2$: Chandrupatla's root finding algorithm is used by default
- $def\_frac\_refined = 0.1$: the fraction of simplices refined when k-points are added to the mesh
- $def\_sample\_method = 2$: k-points are added to the edges of the simplices
- $def\_rtol = 10^{-10}$: a relative tolerance for floating-point precision comparisons

- $def\_atol = 10^{-9}$: an absolute tolerance for floating-point precision comparisons

- $def\_fatten = 1.0$: a parameter for scaling the interval coefficients

- $def\_num\_neighbors2D = 12$: The minimum number of neighbors in 2D interval coefficient calculations

- $def\_num\_neighbors3D = 60$: The desired number of neighbors in 3D interval coefficient calculation

- $def\_neighbors\_per\_bin2D = 2$: The number of neighbors per angular bin in 2D

- $def\_neighbors\_per\_bin3D = 3$: The number of neighbors per angular bin in 3D

- $def\_taylor\_exp\_tol = 10^{-2}$: A tolerance for analytic areas and volumes when the control point weight close to one

- $def\_fl\_max\_iters = 50$: The maximum number root-finding iterations for the Fermi-level calculation

- $def\_min\_split\_triangles = 10$: The minimum number of triangles split for refinement

- $def\_bez\_weight\_tol = 10^{-12}$: A smaller tolerance for classifying conic sections

- $def\_min\_simplex\_size = 10^{-12}$: The volume of the smallest simplex that can be split

- $def\_rational\_bezpt\_dist = 10^{6}$: The maximum size of a component of a rational Bezier point before it is considered a straight line.

Chapter 9

Performance of the quadratic algorithm

9.1 Empirical Pseudopotential models

Electronic structure integration methods are commonly tested on either the free electron or tight binding models. These are attractive options because they have analytic solutions for the density of states, number of states, and band energy. They are, however, overly simplistic because, unlike the true electronic band structure, the free electron model is not an algebraic surface but a quadratic polynomial, and the tight binding model has a band structure with only one sheet. We created more realistic models using the empirical pseudopotential method(gro 2014), which in the past were considered the state-of-the-art methods for materials modeling. These models are given by their characteristic equation

$$||(k_i^2 - E)\delta_{ij} \sum_{\mathbf{d}_v} e^{-i(\mathbf{h}_i - \mathbf{h}_j)} V_v^{(\text{pseudo})}(\mathbf{h}_i - \mathbf{h}_j)|| = 0, \tag{9.1}$$

where k_i is the component of a k-point, $\mathbf{d}_v$ is an atom in the atomic basis, $\mathbf{h}_i$ is a reciprocal lattice vector, and $V_v^{(\text{pseudo})}$ is a pseudopotential form factor. We simplified the characteristic equation by making all model systems monoatomic so that the so-called structure phase factor would disappear and the equation would take the following form

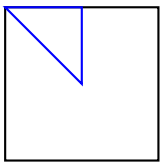
$$||(k_i^2 - E)\delta_{ij} + V_v^{(\text{pseudo})}(\mathbf{h}_i - \mathbf{h}_j)|| = 0. \tag{9.2}$$

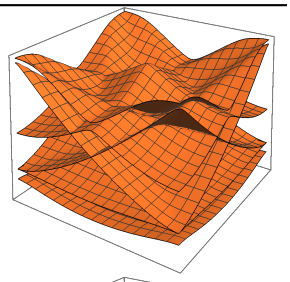
We were unable to locate form factors for 2D metals, so the form factors for the 2D models were chosen at random, and the basis in the Fourier expansion was cut off at around 120 terms for all models. There is a model for all five 2D Bravais lattices, and for each model we test 3 different occupations or Fermi levels. Details of the 2D models are provided in Fig. 9.1.

Form factors for 3D empirical pseudopotenials models for simple metals are available in the literature. The form factors by Cohen and Heine (Cohen & Heine 1970) are used for empirical pseudopotentials in tests of the performance of integration algorithms. Plots of the band structures of the models are shown in Fig. 9.2 and 9.3.

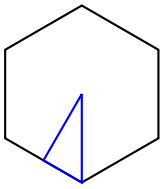
Model Parameters	Brillouin Zone	Band Structure

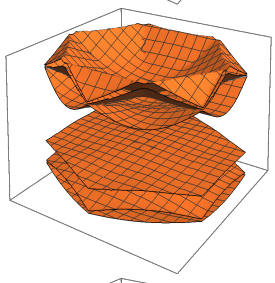
Pseudopotential 1
Lattice type: Square
Number of e^{-1}: (6, 7, 8)
Form factors: (0, −0.23, 0.12)
Lattice basis: $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$



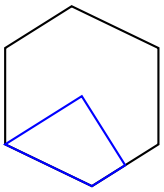


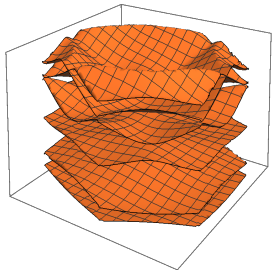
Pseudopotential 2
Lattice type: Hexagonal
Number of e^{-1}: (5, 7, 8)
Form factors: (0.0, 0.39, 0.23, −0.11)
Lattice basis: $\begin{pmatrix} 1. & 0.5 \\ 0 & 0.866025 \end{pmatrix}$



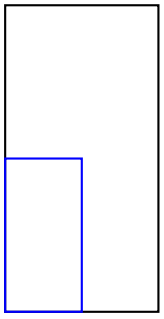


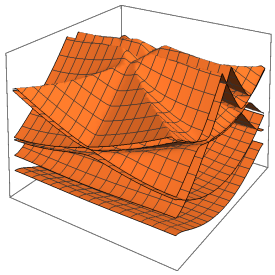
Pseudopotential 3
Lattice type: Centered rectangular
Number of e^{-1}: (5, 7, 8)
Form factors: (0.0, −0.27, 0.2, −0.33)
Lattice basis: $\begin{pmatrix} 1. & 0.433884 \\ 0 & 0.900969 \end{pmatrix}$



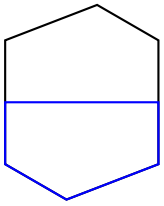


Pseudopotential 4
Lattice type: Rectangular
Number of e^{-1}: (6, 7, 8)
Form factors: (0.0, 0.39, −0.11, 0.11)
Lattice basis: $\begin{pmatrix} 1. & 0 \\ 0 & 2. \end{pmatrix}$





Pseudopotential 5
Lattice type: Oblique
Number of e^{-1}: (5, 7, 9)
Form factors: (0.0, 0.42, 0.02, −0.18)
Lattice basis: $\begin{pmatrix} 1. & -0.4 \\ 0 & 1.03923 \end{pmatrix}$



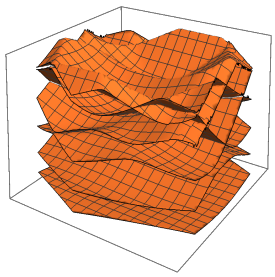


Figure 9.1 2D empirical pseudopotential models used to test the efficiency of the quadratic integration algorithm. The pseudopotential form factors are listed by increasing distance from the origin, starting at the origin. In the second column, the Brillouin zone is outlined in black and the irreducible Brillouin zone in blue.

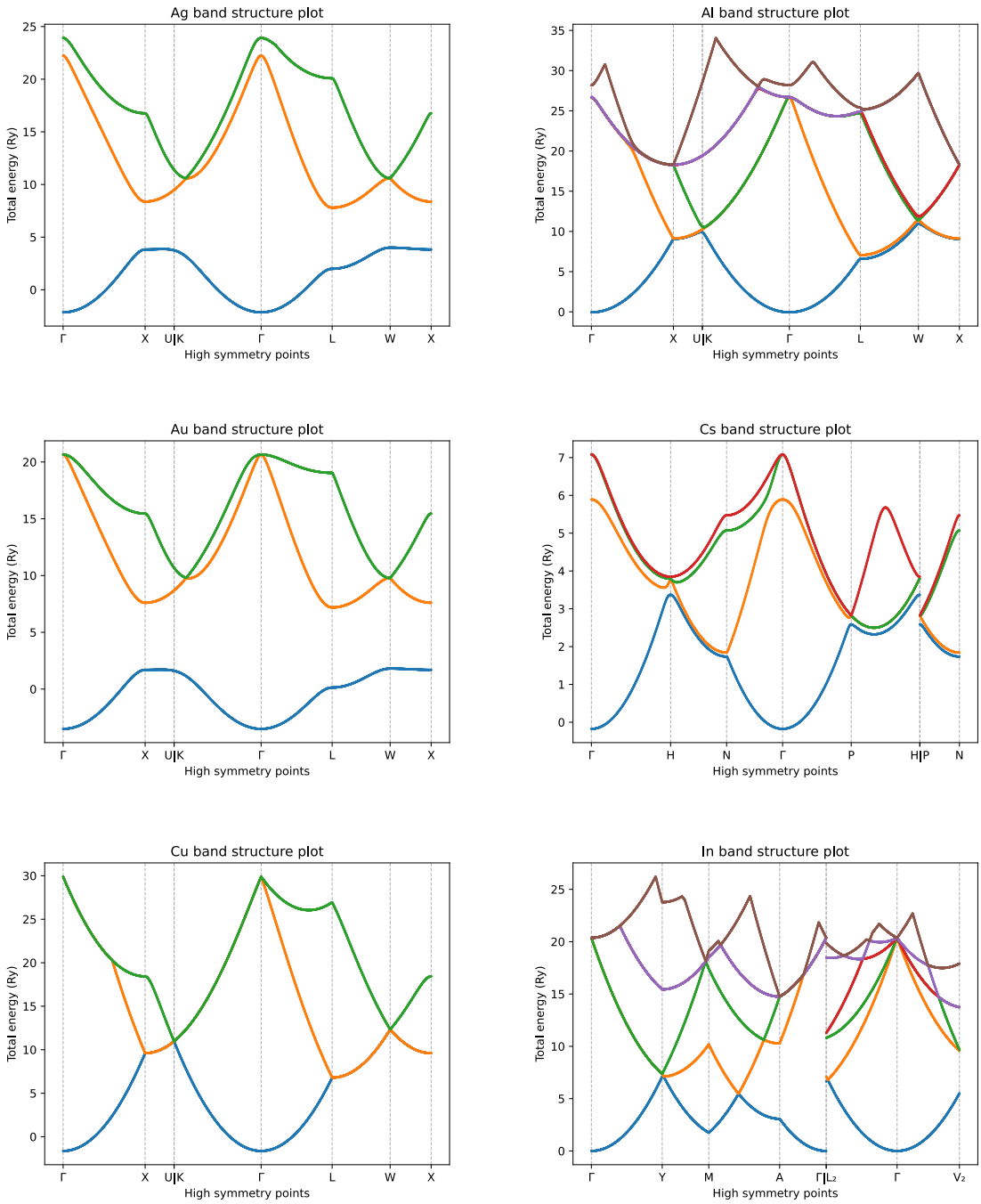


Figure 9.2 Band structure plots of empirical pseudopotentials along paths of high symmetry throughout the Brillouin zone. These models are used to test the performance of the quadratic integration algorithm.

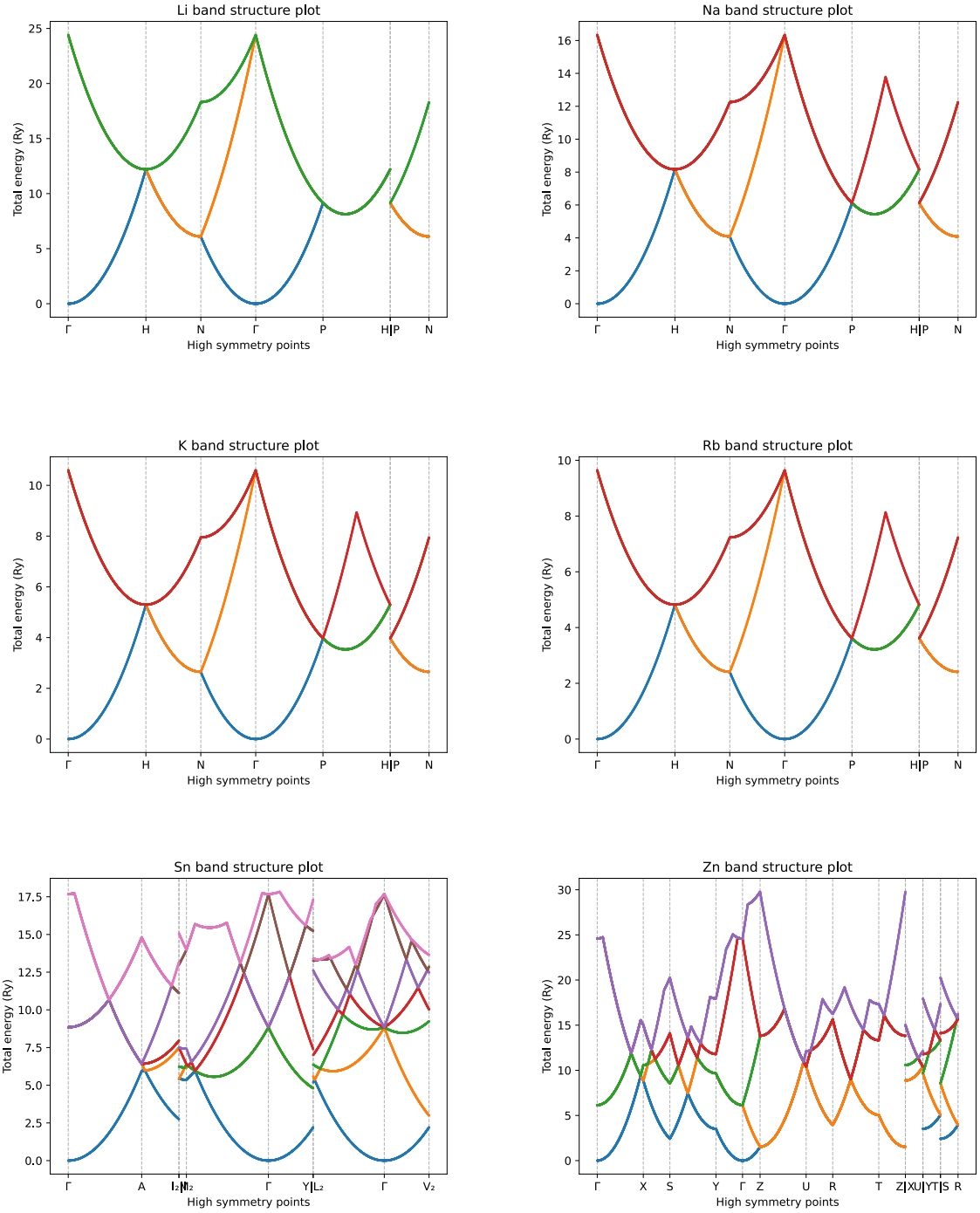


Figure 9.3 Band structure plots of empirical pseudopotentials along paths of high symmetry throughout the Brillouin zone. These models are used to test the performance of the quadratic integration algorithm.

Another factor that affects the band energy accuracy is the expansion size of the plane wave expansion of the wave function that happens at each k-point. If the expansion is too small, attempting to increase the accuracy of the band energy by adding more k-points is ineffective (see Fig. 9.4. The expansion size affects the values of the eigenvalues and if the basis is too small, there are large discontinuities in the band structure when terms in the expansion are added or removed. When the expansion size is large enough, these discontinuities are small enough to not affect the band structure or band energy. The cutoffs for 3D EPMs was chosen so that there were at least 1,000 terms in the expansion at the origin.

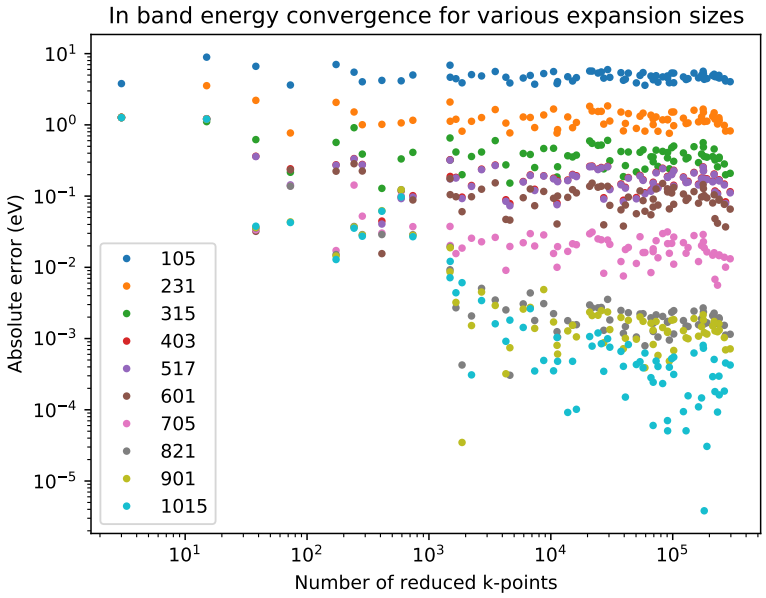


Figure 9.4 The band energy error of an EPM for indium with different numbers of terms in the plane wave expansion of the wave function.

9.2 Quadratic integration performance in 2D

In integration tests, we compare integration over the BZ with the rectangular method to polynomial interpolation and integration of the band structure over the IBZ. The k-points are reduced by symmetry when integrating with the rectangular method; no symmetry reduction is necessary for the polynomial integration methods. We include result for uniform meshes over the IBZ with exact fits (instead of least-square fits) for linear and quadratic polynomials and least-squares fits for quadratic polynomials also with uniform meshes. Integration with adaptive mesh refinement is only shown for interpolation of the band structure with least-squares quadratic polynomials because it is the only method that gives approximated interpolation errors. The integration performance of the various methods is presented as error convergence plots in Fig. 9.5, 9.6, 9.7. The inset in the figures shows the Fermi curve in red and the shadows of the partially occupied sheets in black.

In the figure legends, R is rectangles, L is linear polynomials with uniform meshes, Q is quadratic polynomials with uniform meshes, LSQ is least-squares quadratics with uniform meshes, and A is least-squares quadratics with adaptive meshes. The rectangular method uses Monkhorst-Pack grids with a symmetry preserving offsets. There are better-performing uniform grids for the rectangular method (generalized-regular grids) that are about 60% more efficient on average (Morgan et al. 2018). Consequently, in the convergence plots that follow, the polynomial methods need to have 60% fewer k-points than rectangles to have the same performance as rectangles with generalize-regular grids.

We say one integration method performs better than another if it has less band energy error than the other method at the same k-point density. Linear interpolation shows very consistent error reduction with increasing numbers of k-points but consistent poor performance compared to rectangles (this is comparable to the tetrahedron methods in DFT codes shown in Ch. 6. Quadratic interpolation with uniform meshes has better or roughly the same performance as rectangles; the gap in performance increases the higher the k-point density. Least-squares quadratics with

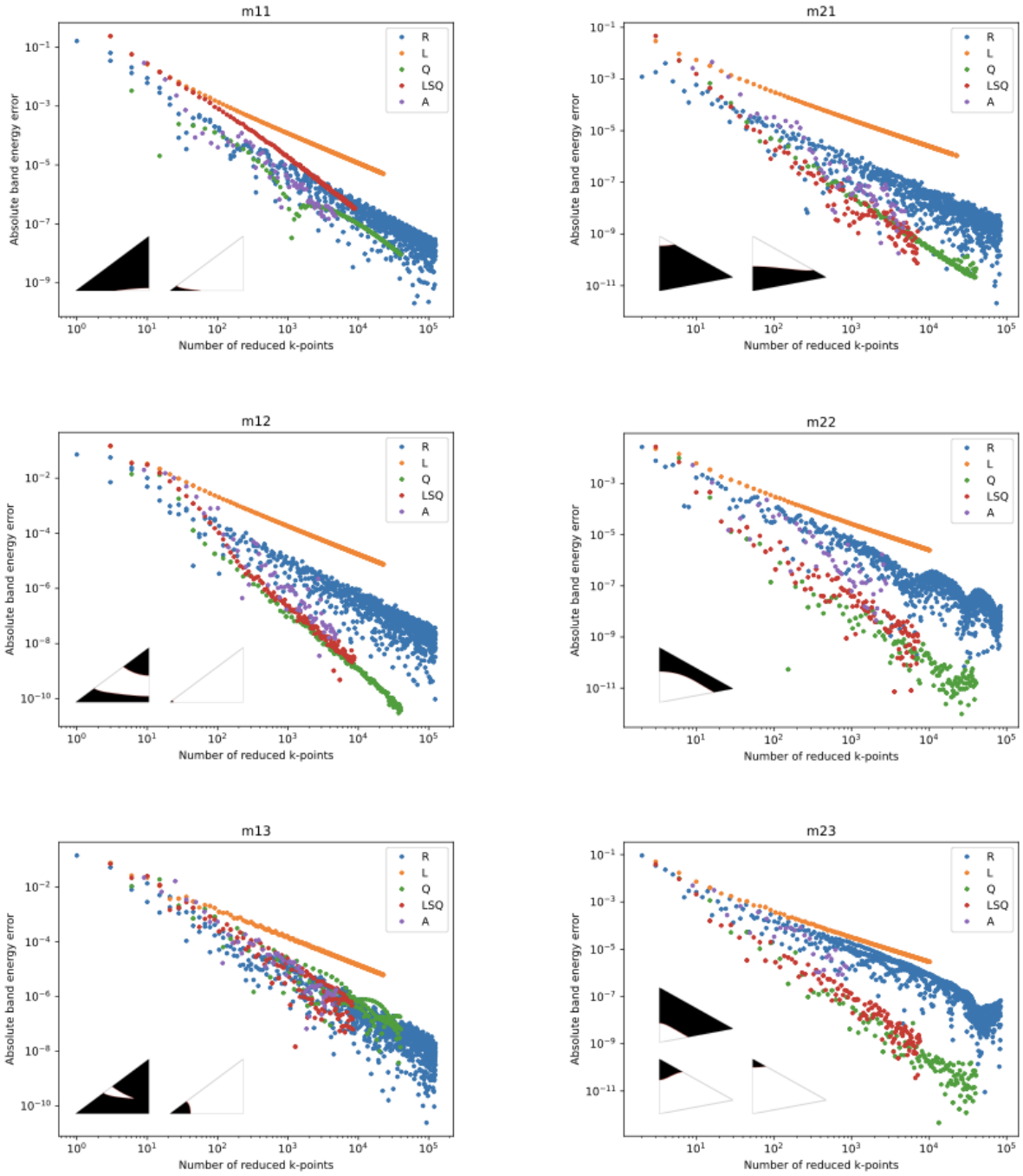


Figure 9.5 A comparison of integration methods for 2D toy models M1 and M2. The integration methods are: R - rectangular method, L - uniform sampling of the IBZ with linear interpolation, Q - uniform sampling of the IBZ with quadratic interpolation, LSQ - uniform sampling of the IBZ with quadratic interpolation and a least-squares fit, A - adaptive sampling of the IBZ with quadratic interpolation and least-squares fitting. The inset shows the the IBZ, the occupied region of the partially occupied sheets as black regions, the unoccupied regions as white regions, and the Fermi curve in red.

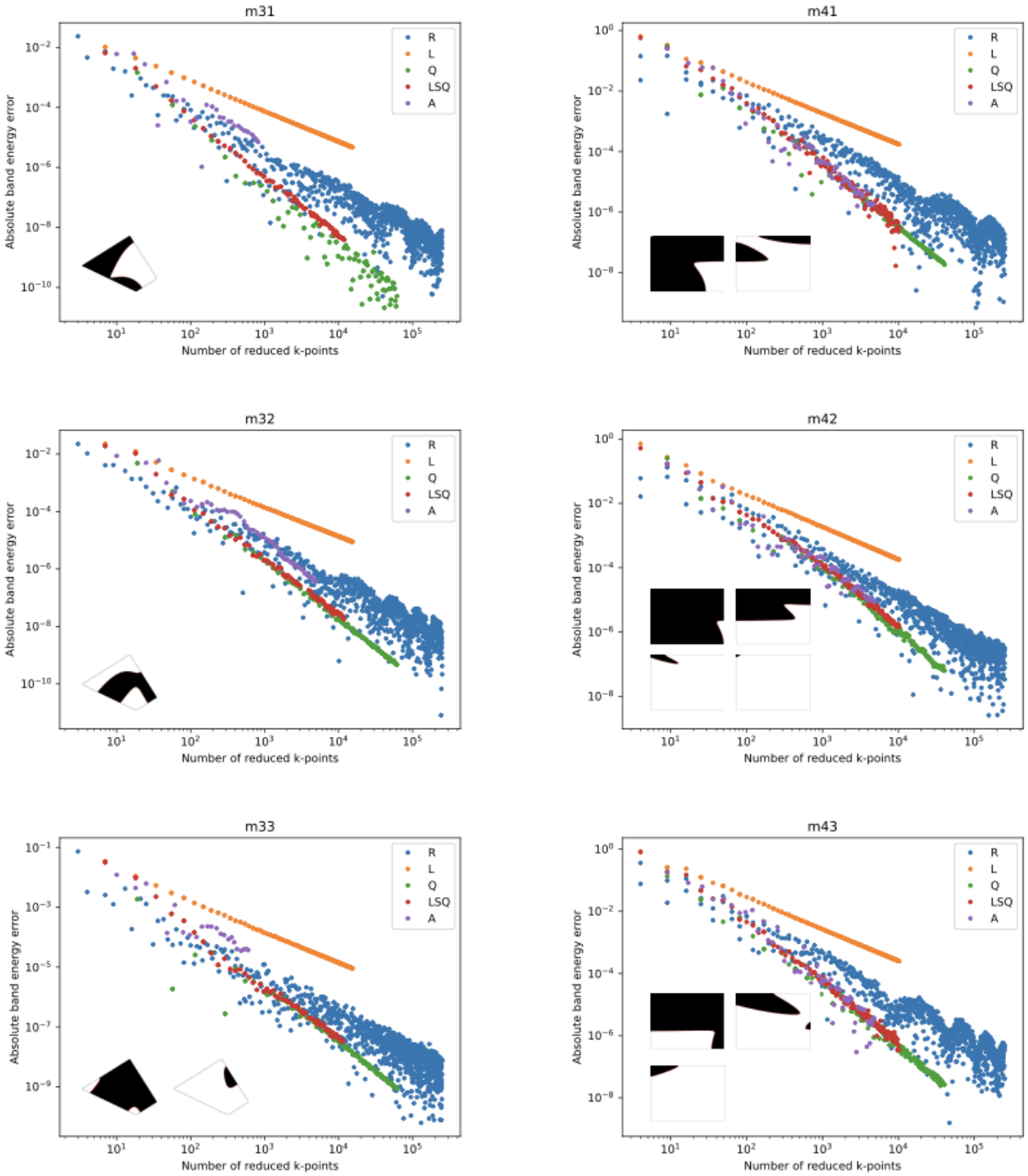


Figure 9.6 A comparison of integration methods for 2D toy models M3 and M4.

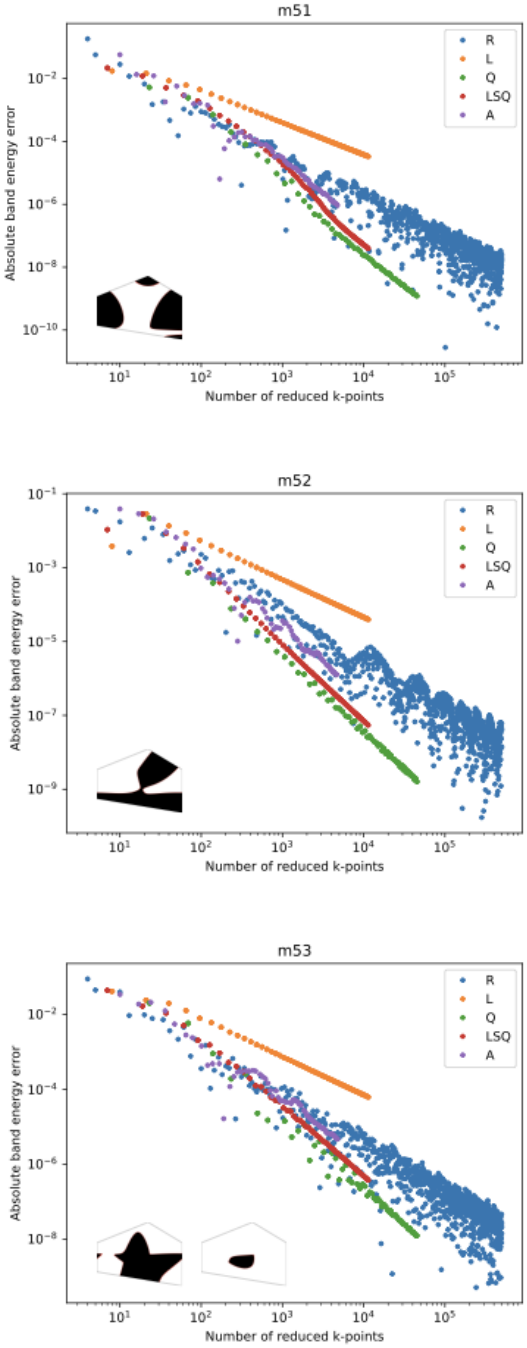


Figure 9.7 A comparison of integration methods for 2D toy model M5.

uniform meshes usually performs similarly to quadratic interpolation; in a few cases and at low k-point densities, least-squares quadratics perform similar to linear polynomials but transition in performance to the performance of quadratic polynomials at higher k-point densities. Least-squares quadratics with adaptive meshes most often perform similar to quadratic polynomials; in a few cases, their performance lies between linear and quadratic polynomials.

In cases where the Fermi surface does not have fine features (high curvature, sharp corners, islands or thin "strips"), we expect uniform meshes to perform as well as adaptive meshes. Integration errors are concentrated at fine features, and we expect adaptive meshes to have superior performance to uniform meshes in the presence of fine features. However, as mentioned in Sec. 8.3.5, containment often fails at sheet crossings, which reduces the estimate of errors where the errors are largest. This is not a problem if the estimated errors at sheet crossings are still large in comparison to other integration errors since simplices with the largest errors relative to other simplices are refined first. Typically, the more atoms or electrons a material has, the more sheets are included in the calculation of the band energy and the more likely is the occurrence of sheet crossings at the Fermi level. Also, the more atoms or electrons in a system, the more basis functions are needed to obtain accurate eigenvalues, so there is a greater cost to evaluating the band structure. We expect adaptive meshes to perform best when the system has many atoms or electrons, many basis functions in the plane-wave expansion of the wavefunction, and high accuracy of the band energy is sought after. We suspect that error bounds are inaccurate if adaptive meshes do not perform well in the presence of fine features when compared to uniform meshes, which is the case for M13.

To get a sense of how well the adaptive algorithm works, a detailed example of mesh refinement for M42 is shown in Fig. 9.8. This figure gives understanding to the poor performance of the adaptive quadratic method at low k-points densities and its superior performance at high k-point densities. When refinement begins, the mesh is so sparse that the estimated Fermi curves (green) and bounding

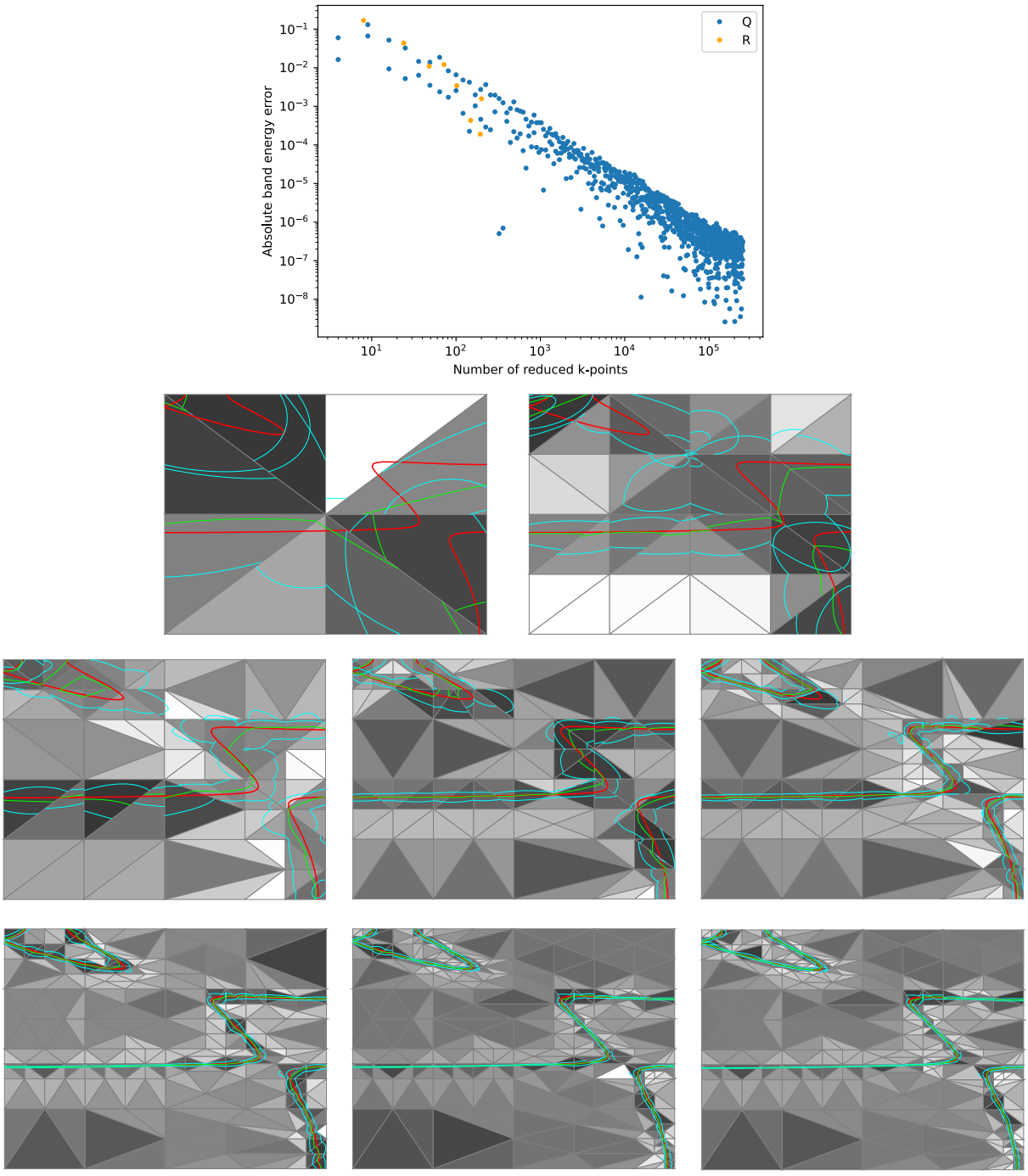


Figure 9.8 A case study of adaptive refinement for M42 with a k-point target of 200 points. The band energy error for each iteration of refinement is shown in the plot at top. The mesh and approximate Fermi curves are shown for each iteration of refinement from top to bottom and left to right. The IBZ is the domain of the plot. The true Fermi curves are the red curves, the approximate Fermi curves are the green curves, and the bounding Fermi curves are the cyan curves. The shading of the triangles indicates relative band energy error where the darker the shading, the larger the error relative to other triangles in the mesh.

Fermi curves (cyan) are very inaccurate and practically meaningless. As refinement progresses, triangles near the Fermi curve are preferentially refined, the outcome of which is triangles near the Fermi curve are smaller and triangles away from the Fermi curve are larger at the end of refinement. The largest band energy errors occur in smaller triangles with a portion of the Fermi curve or in the largest triangles away from the Fermi curve, as expected. Since the band structure is approximated with quadratic polynomials whose coefficients are obtain by least squares, there is no continuity constraint on the approximation, and the quadratic band structure is discontinuous across triangles. This is manifest as jump discontinuities in the plot of the approximate and bounding Fermi curves.

9.3 Quadratic integration performance in 3D

Integration performance for 3D empirical pseudopotentials are shown in Figs. 9.9 and 9.10. Various uniform quadratic integration methods that vary only in the order of the degree of the polynomial approximation are included in the results. Integration with higher order polynomials is possible with purely numeric integration but the convergence rate was prohibitively expensive, so tests were not performed for polynomials of degree greater than 2. Details of the integration methods are shown in Table 9.1.

Integration methods					
Legend	R	C	L	Q	A
Domain	BZ	IBZ	IBZ	IBZ	IBZ
Degree	0	0	1	2	2
Fit	Exact	Exact	Exact	Exact	Least-squares
Mesh	Uniform	Uniform	Uniform	Uniform	Adaptive

Table 9.1 A summary of integration methods tested on the 3D empirical pseudopotentials

Uniform constant integration over the IBZ is an attempt at a rectangular method over the IBZ instead of BZ. The IBZ is split into simplices and each simplex is sampled at its centroid. Integration is performed the same as rectangular integration except what would be the area of the base of the rectangle in the rectangular method in 2D is the area of a triangle (the volume of a tetrahedron in 3D). Integration performance with uniform constant integration over the IBZ is consistently a little worse than rectangles over the BZ.

Uniform linear integration over the IBZ has similar performance to the uncorrected tetrahedron method in DFT codes: integration error consistently decreases with increasing k-point densities but integration error is orders of magnitude worse than the rectangular method.

Uniform quadratic integration performs as good or better than rectangles for many of the tested EPMs (Ag, Au, Cs, Cu, Li, Na, K, Rb, Zn). We suspect that EPMs with accidental crossing are present in the EPMs where quadratic integration performed no better than rectangles (Al, In, Sn). In several cases, the integration error with uniform quadratic integration is orders of magnitude better than rectangles (Cu, Li, Na, K, Rb, Zn). The flatting of the band energy error is caused by error in the "true" band energy for each EPM, which was limited to around 10^{-5} to 10^{-6} eV on average.

Results for adaptive integration are very poor. In many cases (for example, Ag), the band energy error goes flat over a range of k-point densities. This happens when the estimates of the band energy error in each tetrahedron is inaccurate, causing the the tetrahedra with most error to not be refined. In other cases (for example, Li), the band energy error convergence for adaptive refinement is similar to uniform refinement with linear quadratics. This occurs when the tetrahedra over the IBZ are refined more or less uniformly (the performance is worse than uniform quadratic integration because the coefficients are calculated using constrained least squares; at higher k-point densities, the convergence will be closer to uniform quadratic integration). In both cases, adaptive refinement performs worse than rectangles because the error estimates of the band energy errors are inaccurate.

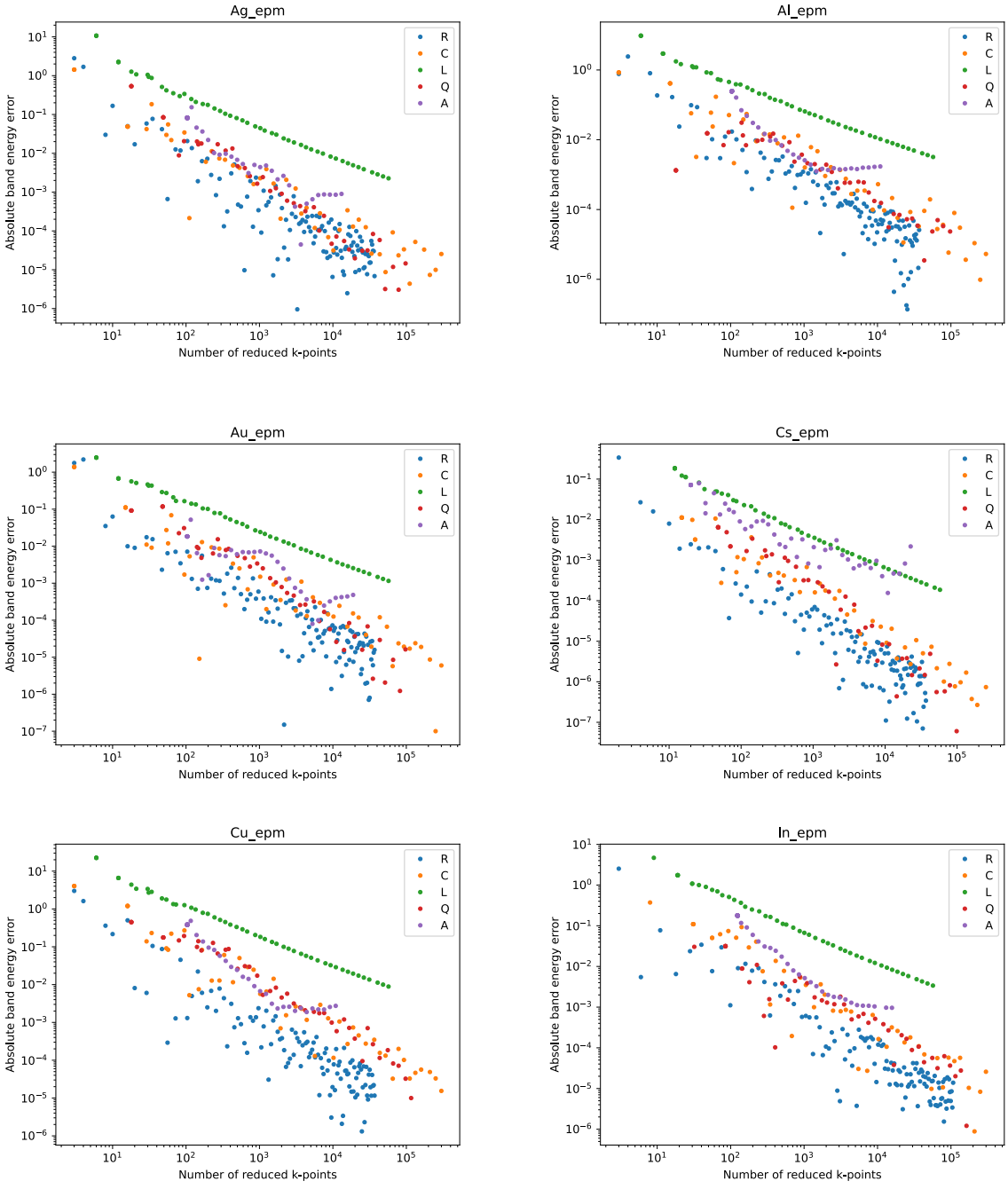


Figure 9.9 The performance of various integration methods on empirical pseudopotentials for Ag, Al, Au, Cs, Cu, and In.

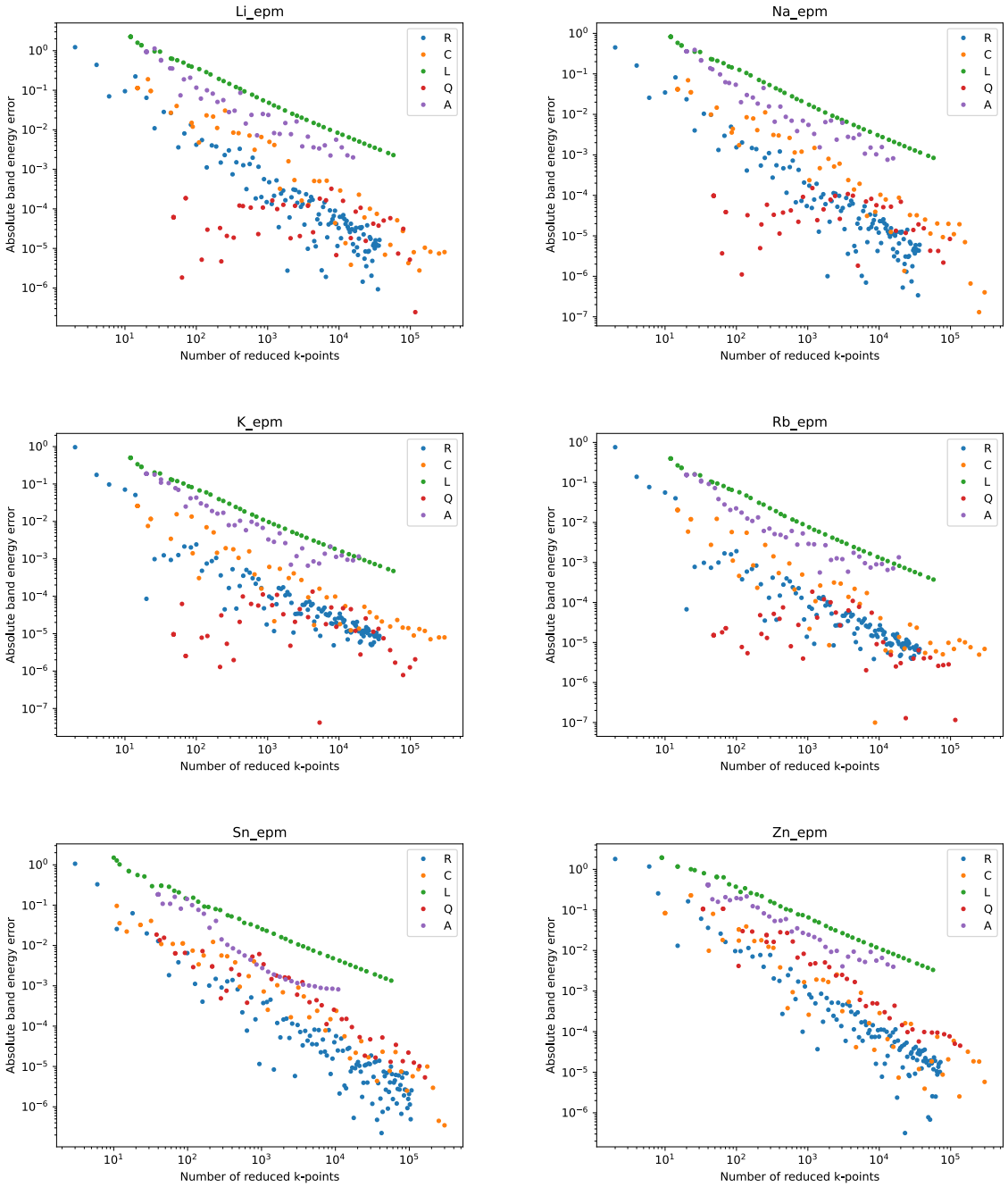


Figure 9.10 The performance of various integration methods on empirical pseudopotentials for Li, Na, K, Rb, Sn, and Zn.

Chapter 10

Conclusion

We have presented an algorithm for computing irreducible Brillouin zones for any crystal structure and a quadratic integration algorithm for computing the band energy over the IBZ. Integration within the irreducible Brillouin zone permits greater freedom in the selection of integration points and subcells and is not limited to quadratic integration or the proposed integration methods. Higher order integration or integration approaches differing from those presented may yield further improvements to the error and efficiency of the band energy computation, which is made possible from integration within the irreducible Brillouin zone.

A quadratic integration algorithm for computing the band energy in density-functional-theory codes has been described. The uniform quadratic algorithm generally requires fewer k-points for a given accuracy than the rectangular method. Consequently, DFT codes that implement quadratic integration have the potential to be more efficient and accurate, which will be reflected in the quality of materials databases, interatomic pseudopotentials, and materials predictions.

In 3D, we suspect adaptive quadratic integration fails to perform better than uniform integration due to poor estimates of the band energy error in each tetrahedron. These inaccurate band energy errors cause refinement of tetrahedra to not occur where band energy errors are largest. Band energy errors are especially inaccurate when there are sheet crossings at the Fermi level. A more accurate

method of estimating band energy error or a method of resolving accidental crossing at the Fermi level could improve the performance of adaptive quadratic integration.

One goal that failed to materialize was estimating the error in the band energy for a given number of k-points. Approximation errors of the quadratic polynomials lead to band energy error bounds that bound the error but are not minimal. Consequently, estimates of the band energy error from summing error bounds are often orders of magnitude greater than the true error. Should a method of estimating errors accurately be developed, the k-point mesh could be made to only be as dense as needed for a provided accuracy. Accordingly, the simulation of a material would take close to as little time as possible for the provided accuracy. In this way, the quality of materials databases can be made guaranteed. If error bounds of materials data are found, machined-learned models will be able to weight the data according to the accuracy of the data. The end result is a more robust, accurate, efficient approach to materials prediction.

Appendix A

Proof of IBZ algorithm

Here we prove that Algorithm 2 is correct for computing an IBZ from the Brillouin zone and the group G of point operators of the space group of a crystal structure. The algorithm works even when the the atomic basis breaks some of the point symmetries of the lattice.

To begin we need a precise definition of an IBZ and the interior of a half-space or polytope.

Definition A.0.1. *For any half-space $H = \{x \in \mathbb{R}^n | d(x,v) \leq d(x,v')\} \subset \mathbb{R}^n$, the interior $\mathring{H}$ of H is the set*

$$\mathring{H} = \{x \in \mathbb{R}^n | d(x,v) < d(x,v')\}.$$

For any closed polytope $Z \subset \mathbb{R}^n$ defined as the intersection $Z = \bigcap_{H \in \mathscr{C}} H$ of a finite collection $\mathscr{C}$ of closed half-spaces, the interior $\mathring{Z}$ of Z is the set

$$\mathring{Z} = \bigcap_{H \in \mathscr{C}} \mathring{H}.$$

Definition A.0.2. *Given a Brillouin zone (BZ) consisting of a closed polytope $P \subset \mathbb{R}^n$ with finite symmetry group G, an irreducible Brillouin zone (IBZ) is a closed polytope $Q \subset P$ such that*

1. *For every point $x \in P$, there exists a $g \in G$ such that $gx \in Q$.*
2. *For every point $y \in \mathring{Q}$ in the interior of Q and every $g \in G$, if $gy \neq y$, then $gy \notin Q$.*

Note that in our definition an IBZ has no interior points that are equivalent under the action of G, but it can have equivalent faces or edges. Equivalent faces or edges may occur when the crystal structure has fewer point symmetries than the lattice, and they may come up even when the atomic basis does not break symmetry. To find an IBZ without equivalent faces or edges, any face of Q that is symmetrically equivalent to another face would need to be removed. See Fig. A.2 for an example in 2D. For BZ integration, the fact that two faces are equivalent under the action of G poses no fundamental problem because the faces have measure zero (contribute nothing to the integral).

Theorem A.0.3. *Assume a finite symmetry group of a polytope P (the BZ) is $G \subset \mathrm{O}(n)$ (every operator $g \in G$ acts linearly and preserves distance in $\mathbb{R}^n$) and that P is the intersection of a*

collection M of half-spaces. Assume further that the group G acts faithfully on the set V of vertices of P, meaning that if $gv = v$ for all $v \in V$, then $gx = x$ for all $x \in P$. Under these assumptions, Algorithm 2 correctly computes an IBZ as the intersection of the half-spaces returned by that algorithm.

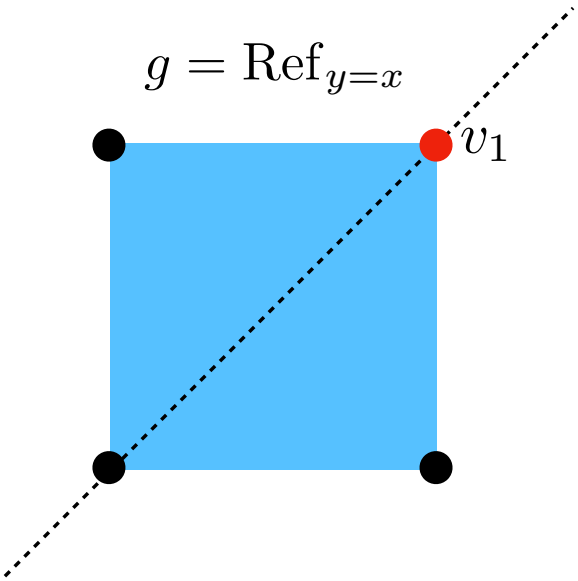


Figure A.1 The stabilizer $\mathrm{stab}_G(v_1)$ is the set of operators in G that fix v_1. Here, for the square lattice, the reflection $g = \mathrm{Ref}_{y=x}$ about the line $y = x$ leaves the point v_1 unmoved ($v_1 = gv_1$), so $g \in \mathrm{stab}_G(v_1)$.

Proof. Begin by giving $V = \{v_1, \ldots, v_m\}$ the ordering used in the algorithm (in the loop starting at line 7). Let $G_0 = G$, and for each $k \geq 1$ let

$$G_k = \mathrm{stab}_{G_{k-1}}(v_k) = \{g \in G_{k-1} | gv_k = v_k\},$$

be the stabilizer, inside of G_{k-1} of v_k; see figure A.1. Let

$$N_k = \{H_{v_k,g}\}_{g \in G_{k-1} \backslash G_k}.$$

Note that N_k is the set of all the hyperplanes associated to vertex v_k that are added to N by the algorithm. Since G acts faithfully on V, only the identify element lies in the stabilizer of every vertex, so we have $F = G \setminus \{I\} = \bigcup_{k=0}^{m-1} G_k \setminus G_{k+1}$.

If $N_0 = M$ is the set of hyperplanes defining P, then the final state of N is

$$N = \bigcup_{k=0}^{m} N_k.$$

Let $P_0 = P$, and for any $\ell \geq 1$ let

$$P_\ell = \bigcap_{k=0}^{\ell} \bigcap_{H \in N_k} H$$

be the polytope constructed by intersecting all the hyperplanes added for all the vertices $v_1, \ldots, v_\ell$. We can characterize the polytope P_1 as

$$\begin{aligned} P_1 &= \{x \in P_0 | d(x, v_1) \leq d(x, gv_1) \, \forall g \in G_0 \setminus G_1\} \\ &= \{x \in P_0 | d(x, v_1) \leq d(x, gv_1) \, \forall g \in G_0\}. \end{aligned}$$

And more generally, we can characterize the polytope P_ℓ as

$$\begin{aligned} P_\ell &= \{x \in P_{\ell-1} | d(x, v_\ell) \leq d(x, gv_\ell) \, \forall g \in G_{\ell-1} \setminus G_\ell\} \\ &= \{x \in P_{\ell-1} | d(x, v_\ell) \leq d(x, gv_\ell) \, \forall g \in G_{\ell-1}\}. \end{aligned} \tag{A.1}$$

The algorithm stops on or before vertex v_m and returns N, from which we can construct P_m, which we show below satisfies the requirements to be an IBZ.

To see that Condition 1 of Definition A.0.2 holds, consider any $x_0 \in P_0 = P$. For each $\ell \geq 1$ we will iteratively choose $g_\ell \in G_{\ell-1} \setminus G_\ell$ such that $x_\ell = g_\ell x_{\ell-1}$ lies in P_ℓ. Therefore, we will have $g_m g_{m-1} \cdots g_2 g_1 x_0 \in P_m$, as required.

For each $\ell \geq 1$, assume we are given $x_{\ell-1} \in P_{\ell-1}$. Since $G_{\ell-1}$ is finite, there exists $g \in G_{\ell-1}$ that minimizes the distance $d(x_{\ell-1}, gv_\ell)$; that is, $d(x_{\ell-1}, gv_\ell) \leq d(x_{\ell-1}, hv_\ell)$ for any $h \in G_{\ell-1}$. Operating by g^{-1} gives $d(g^{-1}x_{\ell-1}, v_\ell) \leq d(g^{-1}x_{\ell-1}, g^{-1}hv_\ell)$ for all $h \in G_{\ell-1}$. But the set $\{g^{-1}h | h \in G_{\ell-1}\}$

is equal to the entire group $G_{\ell-1}$. Thus we have

$$d(g^{-1}x_{\ell-1}, v_\ell) \leq d(g^{-1}x_{\ell-1}, \gamma v_\ell) \qquad \forall \gamma \in G_{\ell-1}. \tag{A.2}$$

Setting $g_\ell = g^{-1}$ and using Equations (A.1) and (A.2) gives $g_\ell x_{\ell-1} \in P_\ell$. Iterating from $\ell = 1$ to $\ell = m$ shows that $g_m g_{m-1} \cdots g_2 g_1 x_0 \in P_m$, as required. Thus Condition 1 holds.

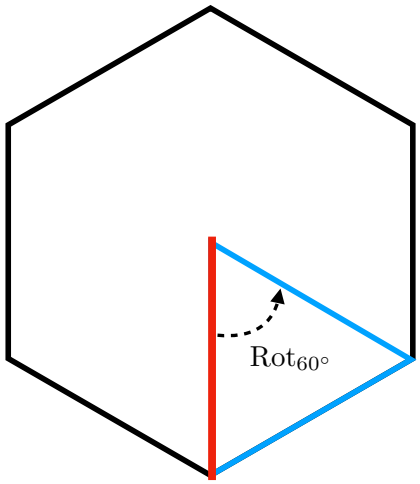


Figure A.2 Example of symmetrically equivalent IBZ edges. In the figure, the BZ is outlined in black and the IBZ in blue. In this example, the atomic basis has broken reflection symmetries; only rotational symmetries remain. The IBZ edge in red is symmetrically equivalent to the opposite edge by a 60° rotation, which is an operation in the point group of a hexagonal lattice.

To see that Condition 2 in Definition A.0.2 holds, first note that for every $v \in V$ if $g \in F$ satisfies $gv \neq v$, then we also have $g^{-1}v \neq v$. This implies that if $H_{v,g} \in N$, then $H_{v,g^{-1}} \in N$. Now consider any $y \in \mathring{P}_m$ and any operator $g \in G$ with $gy \neq y$. Since G acts faithfully on V, there exists at least one vertex $v \in V$ such that $gv \neq v$. Let v be the first such $v \in V$ encountered in the loop (at line 7) over vertices in V, so that $H_{v,g}$ and $H_{v,g^{-1}} \in N$. Since $y \in \mathring{P}_m$, we must have $y \in \mathring{H}_{v,g^{-1}}$; hence

$$d(gy, gv) = d(y, v) < d(y, g^{-1}v) = d(gy, v), \tag{A.3}$$

where the two equalities in (A.3) follow from the fact that any operator $g \in G \subset O(n)$ preserves distances in $\mathbb{R}^n$. This implies that $gy \notin H_{v,g}$, and thus that $gy \notin P_m$. Therefore Condition 2 holds. □

Appendix B

Bézier curves and surfaces

B.1 Parametrization of Conic Section

Our discussion involves points in $\mathbb{R}^2$ (2-dimensional Cartesian coordinates), points in $\mathbb{R}^3$ (3-dimensional Cartesian coordinates), and points in $\mathbb{B}^2 = \{(s,t,u)|s,t,u \in \mathbb{R}, s+t+u \equiv 1\}$ (2-dimensional Barycentric coordinates). Define a projection operatore $\Pi(x,y,z) = (x,y)$ and a selection operator $\mathbb{Z}(x,y,z) = z$.

Given a reference triangle with vertices $\mathbf{v}_s, \mathbf{v}_t, \mathbf{v}_u \in \mathbb{R}^2$, the Barycentric coordinates of an arbitrary point $\mathbf{v} \in \mathbb{R}^2$ uniquely satisfy

$$\mathbf{v} = s\mathbf{v}_s + t\mathbf{v}_t + u\mathbf{v}_u. \tag{B.1}$$

The implicit equation of a line in Barycentric coordinates is

$$z_{100}s + z_{010}t + z_{001}u = 0, \quad (s,t,u) \in \mathbb{B}^2, \quad z_{ijk} \in \mathbb{R} \tag{B.2}$$

Such a line can be thought of as the intersection of the plane $z = 0$ with the plane

$$\begin{aligned} \mathbf{V}(s,t,u) &= \mathbf{V}_{100}s + \mathbf{V}_{010}t + \mathbf{V}_{001}t, \\ (s,t,u) &\in \mathbb{B}^2, \ \ \mathbf{V}_{ijk} \in \mathbb{R}^3 \end{aligned} \tag{B.3}$$

where $\Pi(\mathbf{V}_{100}) = \mathbf{v}_s$, $\Pi(\mathbf{V}_{010}) = \mathbf{v}_t$, $\Pi(\mathbf{V}_{001}) = \mathbf{v}_u$, and $\mathbb{Z}(\mathbf{V}_{ijk}) = z_{ijk}$. The lines bounding the reference triangle have the following implicit equations: line $\mathbf{v}_t$–$\mathbf{v}_u$ is $s = 0$, line $\mathbf{v}_s$–$\mathbf{v}_u$ is $t = 0$, and line $\mathbf{v}_s$–$\mathbf{v}_t$ is $u = 0$.

Similarly, the implicit equation of a degree-two curve (i.e., a conic section) is

$$\begin{aligned} z(s,t,u) = {} & z_{200}s^2 + z_{020}t^2 + z_{002}u^2 + 2z_{110}st \\ & + 2z_{101}su + 2z_{011}tu = 0. \end{aligned} \tag{B.4}$$

This curve can be thought of as the intersection of the plane $z = 0$ with the triangular explicit Bézier

surface patch

$$\begin{aligned}\mathbf{V}(s,t,u) = \mathbf{V}_{200}s^2 + \mathbf{V}_{020}t^2 + \mathbf{V}_{002}u^2 + 2\mathbf{V}_{110}st \\ + 2\mathbf{V}_{101}su + 2\mathbf{V}_{011}tu, \\ (s,t,u) \in \mathbb{B}^2\end{aligned} \tag{B.5}$$

with $\mathbf{V}_{ijk} \in \mathbb{R}^3, \quad \mathbb{Z}(\mathbf{V}_{ijk}) = z_{ijk}$, and

$$\Pi(\mathbf{V}_{ijk}) = \frac{i\mathbf{v}_s + j\mathbf{v}_t + k\mathbf{v}_u}{2}. \tag{B.6}$$

Such a representation has been called a piecewise-algebraic curve of degree two (Sederberg 1984). Eq. (B.6) assures that $\Pi(\mathbf{V}(s,t,u)) \equiv s\mathbf{v}_s + t\mathbf{v}_t + u\mathbf{v}_u$.

The three boundary curves of (B.5) are simply degree-two Bézier curves. For example, the boundary curve corresponding to the edge of the reference triangle $\mathbf{v}_s$–$\mathbf{v}_t$ is obtained by setting $u = 0$ in (B.5). Since $s+t+u = 1$, we can further substitute $s = 1-t$ in (B.5) to obtain the boundary curve expressed as a degree-two Bézier curve $\mathbf{V}(t) = \mathbf{V}_{200}(1-t)^2 + \mathbf{V}_{110}2t(1-t) + \mathbf{V}_{020}t^2$.

For fixed Barycentric coordinates (s_m, t_m, u_m), $\mathbf{V}(s_m, t_m, u_m)$ can be computed using the de Casteljau algorithm, as follows (Sederberg 1984). Set $\mathbf{V}^{m,0}_{ijk} = \mathbf{V}_{ijk}$, then compute

$$\begin{aligned}\mathbf{V}^n_{ijk} = s_m\mathbf{V}^{n-1}_{i+1,j,k} + t_m\mathbf{V}^{n-1}_{i,j+1,k} + u_m\mathbf{V}^{n-1}_{i,j,k+1}, \\ n = 1,2; \quad i+j+k = 2-n.\end{aligned} \tag{B.7}$$

Specifically, for our degree-two case, for $n = 1$, we compute

$$\begin{aligned}\mathbf{V}^1_{100} &= s_m\mathbf{V}_{200} + t_m\mathbf{V}_{110} + u_m\mathbf{V}_{101} \\ \mathbf{V}^1_{010} &= s_m\mathbf{V}_{110} + t_m\mathbf{V}_{020} + u_m\mathbf{V}_{011} \\ \mathbf{V}^1_{001} &= s_m\mathbf{V}_{101} + t_m\mathbf{V}_{011} + u_m\mathbf{V}_{002}\end{aligned} \tag{B.8}$$

and for $n = 2$,

$$\mathbf{V}(s_m, t_m, u_m) = \mathbf{V}^2_{000} = s_m\mathbf{V}^1_{100} + t_m\mathbf{V}^1_{010} + u_m\mathbf{V}^1_{001}. \tag{B.9}$$

The plane $\mathbf{T}$ that is tangent to the surface at that point is defined by the three points in (B.8). If $z^2_{000} = 0$, the point $\mathbf{V}(s,t,u)$ lies on the curve (B.4) and the line that is tangent to the curve at that point is the intersection between $\mathbf{T}$ and the plane $z = 0$. As in (B.2) and (B.3), the equation of that tangent line is

$$z^1_{100}s + z^1_{010}t + z^1_{001}u = 0 \tag{B.10}$$

where, from (B.7),

$$\begin{aligned} z^1_{100} &= sz_{200} + tz_{110} + uz_{101}, \\ z^1_{010} &= sz_{110} + tz_{020} + uz_{011}, \\ z^1_{001} &= sz_{101} + tz_{011} + uz_{002}. \end{aligned} \tag{B.11}$$

A useful property of conic sections is that they can be represented exactly as rational Bézier curves. This is not true in general for implicit curves of degree higher than two. We now explain, without derivation, how to express a segment of a conic section given by (B.4) as a rational Bézier curve

$$\mathbf{P}(\tau) = \frac{\sum_{i=0}^{2} w_i \mathbf{P}_i B^2_i(\tau)}{\sum_{i=0}^{2} w_i B^2_i(\tau)}, \quad \mathbf{P}_i \in \mathbb{B}^2 \tag{B.12}$$

The algorithm is based on implicitization—the process of finding an implicit equation for a given planar parametric curve. A discussion of implicitization that requires no background other than high-school algebra can be found in Ref. 134 and Chapter 17 of Ref. 133.

Let $\mathbf{P}_i = (s_i, t_i, u_i)$, $i = 0, 2$ be any two distinct points on the curve. For our application, we generally choose points at which the edges of the triangle intersect the curve (B.4) (hence, at least one of the s_i, t_i, u_i equals 0). Assign $w_0 = w_2 = 1$ and solve for $\mathbf{P}_1 = (s_1, t_1, u_1)$ and w_1 as follows.

We compute $\mathbf{P}_1$ using the property that a quadratic Bézier curve is tangent to its control polygon at $\mathbf{P}_0$ and $\mathbf{P}_2$. So we first find the two lines given by

$$a_i s + b_i t + c_i u = 0, \quad i = 0, 2$$

that are tangent to the curve at $\mathbf{P}_0$ and $\mathbf{P}_2$, respectively. As discussed above, those lines are

$$\begin{aligned} a_i &= z_{200}s_i + z_{110}t_i + z_{101}u_i, \\ b_i &= z_{110}s_i + z_{020}t_i + z_{011}u_i, \\ c_i &= z_{101}s_i + z_{011}t_i + z_{002}u_i. \end{aligned} \tag{B.13}$$

$\mathbf{P}_1$ is the point of intersection between the two lines. Denoting $a \times b = a_0 b_2 - a_2 b_0$ etc,

$$\begin{aligned} \mathbf{P}_1 = (s_1, t_1, u_1) &= (\frac{c \times b}{D}, \frac{a \times c}{D}, \frac{b \times a}{D}), \\ D &= b \times a + a \times c + c \times b. \end{aligned} \tag{B.14}$$

Finally,

$$w_1 = \pm\sqrt{-\frac{\tilde{z}_{110}}{2\tilde{z}_{002}}}$$

where

$$\begin{aligned} \tilde{z}_{002} &= z(s_1, t_1, u_1), \\ zh_{110} &= 2z(\frac{s_0 + s_2}{2}, \frac{t_0 + t_2}{2}, \frac{u_0 + u_2}{2}). \end{aligned} \tag{B.15}$$

Bezout's theorem states that two curves of degree two intersect in four points, and if they intersect in more than four, they are the same curve. So while the derivation of the equations for $\mathbf{P}_1$ and w_1 is beyond the scope of this dissertation, the validity of these results can be verified by showing that $z(\mathbf{P}(t)) = 0$ for at least five values of t.

B.2 Extrema and Saddle Points

A surface (B.5) degenerates to a plane if all $\mathbf{V}_{ijk}$ are coplanar. Otherwise, the surface could be an elliptical paraboloid that contains one extremum, or a hyperbolic paraboloid that contains one saddle point. Extrema or saddle points are identifiable by the fact that their tangent planes have constant

z. A tangent plane is defined by the three points in Eq. B.8. Thus, the Barycentric coordinates (s_e, t_e, u_e) of an extremum or saddle point on (B.5) satisfy

$$s_e z_{200} + t_e z_{110} + u_e z_{101} = s_e z_{110} + t_e z_{020} + u_e z_{011} = s_e z_{101} + t_e z_{011} + u_e z_{002} \tag{B.16}$$

or

$$\begin{bmatrix} z_{200} - z_{110} & z_{110} - z_{020} & z_{101} - z_{011} \\ z_{200} - z_{101} & z_{110} - z_{011} & z_{101} - z_{002} \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} s_e \\ t_e \\ u_e \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

Once (s_e, t_e, u_e) are known, translate the reference triangle to have new vertices $\tilde{\mathbf{v}}_s, \tilde{\mathbf{v}}_t, \tilde{\mathbf{v}}_u$

$$\begin{aligned} \tilde{\mathbf{v}}_s &= s_e \mathbf{v}_s + t_e \mathbf{v}_t + u_e \mathbf{v}_u, \\ \tilde{\mathbf{v}}_t &= \mathbf{v}_t + \tilde{\mathbf{v}}_s - \mathbf{v}_s, \quad \tilde{\mathbf{v}}_u = \mathbf{v}_u + \tilde{\mathbf{v}}_s - \mathbf{v}_s. \end{aligned} \tag{B.17}$$

Changing the reference triangle does not change the overall surface, only the triangular domain of the region of interest. The respective Barycentric coordinates of points $\tilde{\mathbf{v}}_s, \tilde{\mathbf{v}}_t, \tilde{\mathbf{v}}_u$ with respect to the original reference triangle $\mathbf{v}_s \mathbf{v}_t \mathbf{v}_u$ are

$$(s_e, t_e, u_e), \quad (s_e - 1, t_e + 1, u_e), \quad (s_e - 1, t_e, u_e + 1).$$

For the new reference triangle

$$\begin{aligned} \tilde{z}_{200} &= \tilde{z}_{110} = \tilde{z}_{101} = z(s_e, t_e, u_e) \\ \tilde{z}_{020} &= z(s_e - 1, t_e + 1, u_e) \\ \tilde{z}_{002} &= z(s_e - 1, t_e, u_e + 1) \\ \tilde{z}_{011} &= 2z(s_e - 1, t_e + .5, u_e + .5) - (\tilde{z}_{020} + \tilde{z}_{002})/2 \end{aligned}$$

We can classify the surface by computing the discriminant

$$D = (\tilde{z}_{011} - \tilde{z}_{200})^2 - (\tilde{z}_{020} - \tilde{z}_{200})(\tilde{z}_{002} - \tilde{z}_{200}). \tag{B.18}$$

If $D > 0$, the surface is a hyperbolic paraboloid, and if $\tilde{z}_{200} = 0$, the conic section splits into two straight lines that intersect at the saddle point. If $D < 0$ the surface is an elliptical paraboloid that has a minimum or maximum, and if $\tilde{z}_{200} = 0$, the conic section is simply a point. If $D = 0$, the surface is a parabolic cylinder, and if $\tilde{z}_{200} = 0$, the conic section is a double line.

If the extremum or saddle point has an elevation $z = 0$, then the curve $z(s,t,u) = 0$ will have a singularity at that point. This means that the curve can be factored into

$$(a_1 s + b_1 t + c_1 u)(a_2 s + b_2 t + c_2 u) = z_{200}s^2 + z_{020}t^2 + z_{002}u^2 + 2z_{110}st + 2z_{101}su + 2z_{011}tu \quad \text{(B.19)}$$

$$a_1 a_2 = z_{200}, \quad b_1 b_2 = z_{020}, \quad c_1 c_2 = z_{002}$$

Replace $u = w - s - t$ to try and find discriminant

$$z_{200}s^2 + z_{020}t^2 + 2z_{110}st + z_{002}(w-s-t)^2 + 2z_{101}s(w-s-t) + 2z_{011}t(w-s-t) \quad \text{(B.20)}$$

Set $w = 0$,

$$s^2(z_{200} + z_{002} + 2z_{101} + 2z_{011}) + st(2z_{110} - 2z_{002} - 4z_{101} - 4z_{011}) + t^2(z_{020} + z_{002} + 2z_{101} + 2z_{011}) \quad \text{(B.21)}$$

B.3 Validity Test

In our application, $\mathbf{P}_0$ and $\mathbf{P}_2$ will always be points at which the curve $g(s,t,u) = 0$ intersects the boundary triangle. If more than two such points exist, any pair of such points can be chosen for $\mathbf{P}_0$ and $\mathbf{P}_2$, and the Bézier curve given by the equations in Section 1 will exactly define part of the curve; using the negative weight will exactly define the remaining piece of the curve.

For a Bézier curve to be valid for our purposes, no points of the Bézier curve should lie outside of the reference triangle. Thus, one way to determine validity would be to sample a large number of points on the curve and check to make sure they are all inside the reference triangle. A faster, and

more robust, way is to compute the intersection between the candidate Bézier curve $\mathbf{P}(t)$ and each line of the reference triangle. If no such intersection occurs for $t \in [0,1]$, and if, say, $\mathbf{P}(0.5)$ is inside the reference triangle, the entire curve is inside the reference triangle and the curve is valid. If a candidate curve is not valid for $w_1 = +\sqrt{-\frac{h_{110}}{2h_{002}}}$, try again with $w_1 = -\sqrt{-\frac{h_{110}}{2h_{002}}}$. The parameter values at which a degree-two Bézier curve (Eq. B.12) intersects a line $ax+by+c=0$ are the roots of the polynomial equation

$$w_0(ax_0+by_0+c)(1-t)^2 + w_1(ax_1+by_1+c)2t(1-t) + w_2(ax_2+by_2+c)t^2 = 0. \tag{B.22}$$

In our case, we always have $w_0 = w_2 = 1$

B.4 Small loops that do not intersect the triangle

It is possible for the curve $g(s,t,u) = 0$ to be a small loop that does not intersect the triangle. Such loops can be detected as follows. Inside the loop will be a point that is a local minimum or maximum on the surface $z = g(s,t,u)$. The (s,t,u) coordinates of that point will satisfy the equations

$$g_{200}s + g_{110}t + g_{101}u = g_{110}s + g_{020}t + g_{011}u = g_{101}s + g_{011}t + g_{002}u, \tag{B.23}$$
$$s+t+u = 1.$$

This can be re-written

$$\begin{bmatrix} g_{200}-g_{110} & g_{110}-g_{020} & g_{101}-g_{011} \\ g_{200}-g_{101} & g_{110}-g_{011} & g_{101}-g_{002} \\ 1 & 1 & 1 \end{bmatrix} \begin{pmatrix} s \\ t \\ u \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

Solve these equations for (s,t,u) and by checking the sign of $g(s,t,u)$, one can determine the existence of a closed loop.

B.5 Singular Cases

B.5.1 Parallel tangent lines

It might happen that, for a candidate Bézier curve, the lines tangent to $\mathbf{P}_0$ and $\mathbf{P}_2$ are parallel. For example, this would happen if the curve is the unit circle $x^2+y^2-1=0$ and $\mathbf{P}_0=(1,0)$ and $\mathbf{P}_2=(-1,0)$. This case may arise frequently due to the symmetry of the eigenvalues. This case is detected when D in (Eq. B.18) equals zero. As a result, $\mathbf{P}_1$ would lie at infinity. This case is handled by splitting the triangle.

B.5.2 $g(s,t,u)$ is reducible

It is possible for $g(s,t,u)$ to be *reducible*, i.e., expressible as the product of two polynomials:

$$g(s,t,u)=g_{200}s^2+g_{020}t^2+g_{002}u^2+2g_{110}st+2g_{101}su+2g_{011}tu=(a_0s+b_0t+c_0u)(a_1s+b_1t+c_1u) \tag{B.24}$$

If this happens, the curve splits into two lines, defined by $a_0s+b_0t+c_0u=0$ and $a_1s+b_1t+c_1u=0$. If we use the equations in Section 1 to try to represent $g(s,t,u)=0$ with a Bézier curve, this case is detected when $h_{002}=0$, or a very small number. This case is resolved by repeatedly splitting the triangle until all subtriangles contain a single line segment. Should the the two lines intersect, we place one of the vertices of the subtriangles at the position where the lines intersect.

The reducible case can also mean that the portion of $g(s,t,u)=0$ within a given reference triangle is a single line segment. This would mean that the lines tangent to $\mathbf{P}_0$ and $\mathbf{P}_2$ are in fact the same line. In this case we ignore the intersections and treat the Bézier patch as if it were completely below or above the cutting plane.

B.5.3 Floating point arithmetic

A challenge in handling singular cases is that, in floating point arithmetic, we can only declare a case to be singular to within floating point precision. For example, rarely is D exactly zero. Our code uses a relative and absolute tolerance of 10^{-9}.

B.6 Finding a parametric equation for a degree-2 algebraic curve

B.6.1 Solution using Cartesian Coordinates

Given a degree-two curve given by the implicit equation

$$f(x,y) = \sum_{i+j\leq 2} f_{ij} x^i y^j = 0$$

and two points $\mathbf{P}_i = (x_i, y_i)$, $i = 0, 2$ that lie on the curve. Find a rational Bézier curve

$$\mathbf{P}(t) = \frac{\sum_{i=0}^{2} w_i \mathbf{P}_i B_i^2(t)}{\sum_{i=0}^{2} w_i B_i^2(t)}$$

that is equivalent to a portion of the curve $f(x,y) = 0$. We assign $w_0 = w_2 = 1$ and solve for x_1, y_1, w_1 as follows.

First, find the two lines given by

$$a_i x + b_i y + c_i = 0, \quad i = 0, 2$$

that are tangent to the curve at $\mathbf{P}_0$ and $\mathbf{P}_2$, respectively. The coefficients are

$$\begin{aligned} a_i &= f_{20} x_i + f_{11} y_i + f_{10}, \\ b_i &= f_{11} x_i + f_{02} y_i + f_{01}, \\ c_i &= -x_i a_i - y_i b_i. \end{aligned} \tag{B.25}$$

Control point $\mathbf{P}_1 = (x_1, y_1)$ lies at the intersection of those two lines:

$$x_1 = \frac{b_0 c_2 - b_2 c_0}{a_0 b_2 - a_2 b_0}, \quad y_1 = -\frac{a_0 c_2 - a_2 c_0}{a_0 b_2 - a_2 b_0}. \tag{B.26}$$

Using Mathematica, and without explaining the derivation,

$$\begin{aligned} w_1 = &\sqrt{(}(2f_{00} + f_{01}y_0 + f_{01}y_2 + 2f_{02}y_0y_2 + f_{10}x_0 + f_{10}x_2 + f_{11}x_0y_2 + f_{11}x_2y_0 + 2f_{20}x_0x_2)/ \\ &-4(f_{00} + f_{01}y_1 + f_{02}y_1^2 + f_{10}x_1 + f_{11}x_1y_1 + f_{20}x_1^2)) \end{aligned} \tag{B.27}$$

B.6.2 Example

Let the curve be a unit circle

$$f(x,y) = x^2 + y^2 - 1 = 0$$

and $(x_0, y_0) = (1,0)$ and $(x_2, y_2) = (0,1)$. From Eq. B.25,

$$a_0 = 1, b_0 = 0, c_0 = -1$$

and

$$a_2 = 0, b_2 = 1, c_2 = -1$$

and $(x_1, y_1) = (1,1)$. From Eq. B.27,

$$w_1 = \pm\sqrt{\frac{1}{2}}$$

Using Mathematica, we obtain a similar equation for w_1 in terms of Barycentric coordinates. Interestingly, w_1 is independent of the reference triangle

$$\begin{aligned} w_1 = &\sqrt{(} - (g_{200}s_0s_2 + g_{110}s_2t_0 + \\ &g_{110}s_0t_2 + g_{020}t_0t_2 + g_{101}s_2u_0 + \\ &g_{011}t_2u_0 + g_{101}s_0u_2 + g_{011}t_0u_2 + g_{002}u_0u_2)/ \\ &(2g_{200}s_1^2 + 4g_{110}s_1t_1 + 2g_{020}t_1^2 + 4g_{101}s_1u_1 + \\ &4g_{011}t1u_1 + 2g_{002}u_1^2)) \end{aligned} \tag{B.28}$$

Bibliography

2014, Solid state physics (Academic Press)

Abu, M. J., Mohamed, J. J., & Ahmad, Z. A. 2014, International Journal of Refractory Metals and Hard Materials, 47, 86

Adamo, C., & Barone, V. 1998, The Journal of chemical physics, 108, 664

André, J.-M., Delhalle, J., Kapsomenos, G., & Leroy, G. 1972, Chemical Physics Letters, 14, 485

Aroyo, M. I., Orobengoa, D., de la Flor, G., Tasci, E. S., Perez-Mato, J. M., & Wondratschek, H. 2014, Acta Crystallographica Section A: Foundations and Advances, 70, 126

Barber, C. B., Dobkin, D. P., & Huhdanpaa, H. 1996a, ACM Transactions on Mathematical Software, 22, 469

—. 1996b, ACM Transactions on Mathematical Software (TOMS), 22, 469

Becke, A. D. 1986, The Journal of Chemical Physics, 84, 4524

—. 1988, Physical review A, 38, 3098

—. 1993, The Journal of chemical physics, 98, 1372

Bell, G., Hey, T., & Szalay, A. 2009, Science, 323, 1297

Benjamin, J. 1976, Scientific American, 234, 40

Bezanson, J., Edelman, A., Karpinski, S., & Shah, V. B. 2017, SIAM Review, 59, 65

Blackman, M. 1937, Proceedings of the Royal Society of London. Series A-Mathematical and Physical Sciences, 159, 416

Bligaard, T., & Nørskov, J. K. 2007, Electrochimica Acta, 52, 5512

Blöchl, P. E. 1994, Physical Review B, 50, 17953

Blöchl, P. E., Jepsen, O., & Andersen, O. K. 1994, Physical Review B, 49, 16223

Boon, M., Methfessel, M., & Mueller, F. 1986, Journal of Physics C: Solid State Physics, 19, 5337

Booth, W. 1987, Science (Washington, DC), 238

Bostrom, N. Accessed Oct, 2021, https://www.nickbostrom.com/superintelligence.html

Bouckaert, L. P., Smoluchowski, R., & Wigner, E. 1936, Physical Review, 50, 58

Brandenburg, J. G., & Grimme, S. 2013, in Prediction and Calculation of Crystal Structures (Springer), 1–23

Bridges, C. R., Stolar, M., & Baumgartner, T. 2020, Batteries & Supercaps, 3, 268

Brust, D. 1965, Physical Review, 139, A489

Cai, Z.-L., Crossley, M. J., Reimers, J. R., Kobayashi, R., & Amos, R. D. 2006, The Journal of Physical Chemistry B, 110, 15624

Calderon, C. E., et al. 2015, Computational Materials Science, 108, 233

Cancès, É., Ehrlacher, V., Gontier, D., Levitt, A., & Lombardi, D. 2020, Numerische Mathematik, 1

Car, R., & Parrinello, M. 1985, Physical Review Letters, 55, 2471

Chadi, D. J., & Cohen, M. L. 1973, Physical Review B, 8, 5747

Chandrupatla, T. R. 1997, Advances in Engineering Software, 28, 145

Chen, A.-B. 1977, Physical Review B, 16, 3291

Cohen, M. L., & Heine, V. 1970, in Solid state physics, Vol. 24 (Elsevier), 37–248

Commons, W. 2019, File:Ritual Food Cauldron (Ding) with Spirals LACMA AC1998.251.4.jpg — Wikimedia Commons, the free media repository, [Online; accessed 14-December-2021]

—. 2021, File:Titanic in her way to Cherbourg.jpg — Wikimedia Commons, the free media repository, [Online; accessed 14-December-2021]

Connolly, J., & Williams, A. 1983, Physical Review B, 27, 5169

Cooke, J., & Wood, R. 1972, Physical Review B, 5, 1276

Cramer, C. J., & Truhlar, D. G. 2009, Physical Chemistry Chemical Physics, 11, 10757

Creely, A., et al. 2020, Journal of Plasma Physics, 86

Curtarolo, S., Hart, G. L., Nardelli, M. B., Mingo, N., Sanvito, S., & Levy, O. 2013, Nature Materials, 12, 191

Curtarolo, S., et al. 2012, Computational Materials Science, 58, 218

Delgado, M. C. R., Kim, E.-G., Filho, D. A. d. S., & Bredas, J.-L. 2010, Journal of the American Chemical Society, 132, 3375

Dirac, P. A. 1930in , Cambridge University Press, 376–385

El-Sherik, A. 2017, Trends in oil and gas corrosion research and technologies: Production and transmission (Woodhead Publishing)

Eyert, V. 2012, The augmented spherical wave method: a comprehensive treatment, Vol. 849 (Springer)

Eyert, V., Christensen, M., Wolf, W., Reith, D., Mavromaras, A., Freeman, C., & Wimmer, E. 2018, Computation, 6, 63

Field, M. J. 2002, Journal of Computational Chemistry, 23, 48

Finney, J. L. 1979, Journal of computational physics, 32, 137

Foecke, T. 2008, Materials Today, 11, 48

Frenzel, J., Zhang, Z., Neuking, K., & Eggeler, G. 2004, Journal of Alloys and Compounds, 385, 214

Froyen, S. 1989, Physical Review B, 39, 3168

Galántai, A. 2000, Journal of Computational and Applied Mathematics, 124, 25

Gaughan, R. 2010, Accidental Genius

Giannozzi, P., et al. 2009, Journal of Physics: Condensed Matter, 21, 395502

—. 2017, Journal of Physics: Condensed Matter, 29, 465901

Gibney, E. 2021

Gilat, G. 1972, Journal of Computational Physics, 10, 432

Gilat, G., & Dolling, G. 1964, Physics Letters, 8, 304

Gilat, G., & Raubenheimer, L. 1966, Physical Review, 144, 390

Giustino, F. 2014, Materials modelling using density functional theory: properties and predictions (Oxford University Press)

Gracia, L., Beltrán, A., & Andrés, J. 2007, The Journal of Physical Chemistry B, 111, 6479

Grayce, C. J., & Harris, R. A. 1994, Physical Review A, 50, 3089

Hafner, J. 2008, Journal of Computational Chemistry, 29, 2044

Hart, G. L., Jorgensen, J. J., Morgan, W. S., & Forcade, R. W. 2019, Journal of Physics Communications, 3, 065009

Hart, G. L., Mueller, T., Toher, C., & Curtarolo, S. 2021, Nature Reviews Materials, 1

Hautier, G., Fischer, C. C., Jain, A., Mueller, T., & Ceder, G. 2010, Chemistry of Materials, 22, 3762

Herbst, M. F., Levitt, A., & Cancès, E. 2021in , 69

Herring, C. 1937, Physical Review, 52, 365

Hey, A. J., Tansley, S., Tolle, K. M., et al. 2009, The fourth paradigm: data-intensive scientific discovery, Vol. 1 (Microsoft research Redmond, WA)

Hohenberg, P., & Kohn, W. 1964, Physical Review, 136, B864

Horgan, J. 2012, Scientific American Blogs, Nature Publishing Group, London, United Kingdom

Hu, H., & Yang, W. 2008, Annual Review of Physical Chemistry, 59, 573

Imbalzano, G., Anelli, A., Giofré, D., Klees, S., Behler, J., & Ceriotti, M. 2018, The Journal of chemical physics, 148, 241730

Jain, A., et al. 2013, APL Materials, 1, 011002

Jepson, O., & Anderson, O. 1971, Solid State Communications, 9, 1763

Jorgensen, J. J., & Hart, G. L. 2021, Modelling and Simulation in Materials Science and Engineering, 29, 065014

Kawamura, M., Gohda, Y., & Tsuneyuki, S. 2014, Physical Review B, 89, 094515

Kharina, A., & Rutherford, D. 2015

Kohn, W., & Sham, L. J. 1965, Physical Review, 140, A1133

Kratzer, P., & Neugebauer, J. 2019, Frontiers in Chemistry, 7

Kresse, G., & Furthmüller, J. 1996, Computational Materials Science, 6, 15

Kresse, G., & Hafner, J. 1993, Physical Review B, 47, 558

—. 1994a, Physical Review B, 49, 14251

—. 1994b, Journal of Physics: Condensed Matter, 6, 8245

Kresse, G., & Joubert, D. 1999, Physical Review B, 59, 1758

Kumar, S., Xu, Q., & Suryanarayana, P. 2020, Chemical Physics Letters, 739, 136983

Legat, B., et al. 2019, JuliaPolyhedra/Polyhedra.jl: v0.5.1

Lehmann, G., & Taut, M. 1972, Physica Status Solidi (B), 54, 469

Lherbier, A., Blase, X., Niquet, Y.-M., Triozon, F., & Roche, S. 2008, Physical Review Letters, 101, 036808

Louail, L., Maouche, D., Roumili, A., & Sahraoui, F. A. 2004, Materials Letters, 58, 2975

Martin, N. 2021

Marzari, N. 1996, PhD thesis, University of Cambridge

Mattsson, A. E., Schultz, P. A., Desjarlais, M. P., Mattsson, T. R., & Leung, K. 2004, Modelling and Simulation in Materials Science and Engineering, 13, R1

Matulka, R., & Wood, D. 2013, The history of the light bulb, https://www.energy.gov/articles/history-light-bulb, [Online; accessed 28-September-2021]

Mehl, M. J. 2000, Physical Review B, 61, 1654

Methfessel, M., Boon, M., & Mueller, F. 1983, Journal of Physics C: Solid State Physics, 16, L949

Methfessel, M., & Paxton, A. 1989, Physical Review B, 40, 3616

Meyer, B., & Marx, D. 2003, Physical Review B, 67, 035403

Monkhorst, H. J., & Pack, J. D. 1976, Physical Review B, 13, 5188

Morales-García, Á., Valero, R., & Illas, F. 2017, The Journal of Physical Chemistry C, 121, 18862

Moreno, J., & Soler, J. M. 1992, Physical Review B, 45, 13891

Morgan, W. S., Christensen, J., Hamilton, P., Jorgensen, J. J., Campbell, B., Forcade, R. W., & Hart, G. L. 2019, to be submitted

Morgan, W. S., Hart, G. L. W., Foulk, N. L., Rosenbrock, C. W., & Christensen, J. E. 2021, symlib, https://github.com/msg-byu/symlib

Morgan, W. S., Jorgensen, J. J., Hess, B. C., & Hart, G. L. 2018, Computational Matererials Science, 153, 424

Mulholland, A. J. 2005, Drug discovery today, 10, 1393

Munro, J. M., Latimer, K., Horton, M. K., Dwaraknath, S., & Persson, K. A. 2020, npj Computational Materials, 6, 1

Museum, T. N. W. 2021, Innovating for Victory

Neese, F. 2009, Coordination Chemistry Reviews, 253, 526

Nguyen, P. Q., & Stehlé, D. 2009, ACM Transactions on Algorithms (TALG), 5, 46

Nolan, M., Grigoleit, S., Sayle, D. C., Parker, S. C., & Watson, G. W. 2005, Surface Science, 576, 217

Nørskov, J. K., Abild-Pedersen, F., Studt, F., & Bligaard, T. 2011, Proceedings of the National Academy of Sciences, 108, 937

Nørskov, J. K., et al. 2002, Journal of Catalysis, 209, 275

Nyshadham, C. 2019, Materials prediction using high-throughput and machine learning techniques (Brigham Young University)

Nyshadham, C., Oses, C., Hansen, J. E., Takeuchi, I., Curtarolo, S., & Hart, G. L. 2017, Acta Materialia, 122, 438

Obot, I., Macdonald, D., & Gasem, Z. 2015, Corrosion Science, 99, 1

Ong, S. P., et al. 2013, Computational Materials Science, 68, 314

Oriade, J. 1980, International Journal of Quantum Chemistry, 18, 597

OSHA. 1988, Inspection: 100478866 - Shell Oil Company, Tech. rep., Occupational Health and Safety Administration

Otero-de-la Roza, A., & Luaña, V. 2011, Journal of computational chemistry, 32, 291

Perdew, J. P. 1986, Physical Review B, 33, 8822

Perdew, J. P., Burke, K., & Ernzerhof, M. 1996, Physical Review Letters, 77, 3865

Poffenberger, L. 2021, APS News, 30

Popovich, N. 2019, The New York Times

Prautzsch, H., Boehm, W., & Paluszny, M. 2013, Bézier and B-spline techniques (Springer Science & Business Media)

Pribram-Jones, A., Gross, D. A., & Burke, K. 2015, Annual review of physical chemistry, 66, 283

Rieken, K. 2016

Rincon, P. 2014

Roberts, R. M. 1989, Serendipity: Accidental discoveries in science

Saal, J. E., Kirklin, S., Aykol, M., Meredig, B., & Wolverton, C. 2013, JOM, 65, 1501

Saal, J. E., Oliynyk, A. O., & Meredig, B. 2020, Annual Review of Materials Research, 50, 49

Schleder, G. R., Padilha, A. C., Acosta, C. M., Costa, M., & Fazzio, A. 2019, Journal of Physics: Materials, 2, 032001

Scott, A. P., & Radom, L. 1996, The Journal of Physical Chemistry, 100, 16502

Sederberg, T. W. 1984, Computer Aided Geometric Design, 1, 241

—. 2012, Computer Aided Geometric Design, course notes, https://scholarsarchive.byu.edu/facpub/1

Sederberg, T. W., Anderson, D. C., & Goldman, R. N. 1984, Computer Vision, Graphics, and Image Processing, 28, 72

Setyawan, W., & Curtarolo, S. 2010, Computational materials science, 49, 299

Setyawan, W., Gaume, R. M., Feigelson, R. S., & Curtarolo, S. 2009, IEEE Transactions on Nuclear Science, 56, 2989

Sha, W., & Edwards, K. 2007, Materials & design, 28, 1747

Shen, Z., Johnsson, M., Zhao, Z., & Nygren, M. 2002, Journal of the American Ceramic Society, 85, 1921

Sorg, L. 2021

Sousa, S. F., Fernandes, P. A., & Ramos, M. J. 2007, The Journal of Physical Chemistry A, 111, 10439

Steinmann, P., & Runesson, K. 2021, The Catalogue of Computational Material Models: Basic Geometrically Linear Models in 1D (Springer Nature)

Stephens, P., Devlin, F., Chabalowski, C., & Frisch, M. J. 1994, The Journal of Physical Chemistry, 98, 11623

Sundararaghavan, V., & Zabaras, N. 2005, Computational Materials Science, 32, 223

Tassone, F., Mauri, F., & Car, R. 1994, Physical Review B, 50, 10561

Tirado, F. L. R., Toinin, J. P., & Dunand, D. C. 2018, Acta Materialia, 151, 137

Torralba, J. M., Alvaredo, P., & García-Junceda, A. 2019, Powder Metallurgy, 62, 84

Tullo, A. H. 2016, Chemical and Engineering News, 94

University, Y. 2020

van Mourik, T., Bühl, M., & Gaigeot, M.-P. 2014, Density functional theory across chemistry, physics and biology

Van Noorden, R., Maher, B., & Nuzzo, R. 2014, Nature News, 514, 550

Verma, D. K. 2018, in Advanced Engineering Testing (IntechOpen)

Verma, P., & Truhlar, D. G. 2017, The Journal of Physical Chemistry Letters, 8, 380

Vishnu, K. G., & Strachan, A. 2010, Acta materialia, 58, 745

Vuckovic, S., & Gori-Giorgi, P. 2017, The journal of physical chemistry letters, 8, 2799

Weideman, J. A. C. 2002, The American Mathematical Monthly, 109, 21

Wende, F., Marsman, M., Kim, J., Vasilev, F., Zhao, Z., & Steinke, T. 2019, International Journal of Quantum Chemistry, 119, e25851

Wiesenekker, G., & Baerends, E. 1991, Journal of Physics: Condensed Matter, 3, 6721

Wiesenekker, G., Te Velde, G., & Baerends, E. 1988, Journal of Physics C: Solid State Physics, 21, 4263

Wikipedia contributors. 2021, ITER — Wikipedia, The Free Encyclopedia, [Online; accessed 14-December-2021]

Wisesa, P., McGill, K. A., & Mueller, T. 2016, Physical Review B, 93, 155109

Wong, M. W. 1996, Chemical Physics Letters, 256, 391

Woodley, S. M., & Catlow, R. 2008, Nature Materials, 7, 937

Woods, N. 2018, arXiv preprint arXiv:1803.01763

Xia, S., Qiu, M., Yu, L., Liu, F., & Zhao, H. 2008, Corrosion Science, 50, 2021

Xiao, H., Tahir-Kheli, J., & Goddard III, W. A. 2011, The Journal of Physical Chemistry Letters, 2, 212

Yazyev, O. V., Kudin, K. N., & Scuseria, G. E. 2002, Physical Review B, 65, 205117

Zhan, C.-G., Nichols, J. A., & Dixon, D. A. 2003, The Journal of Physical Chemistry A, 107, 4184

Zhao, Y., & Truhlar, D. G. 2006, The Journal of Chemical Physics, 125, 194101

Zuo, T., Yang, X., Liaw, P. K., & Zhang, Y. 2015, Intermetallics, 67, 171