

NEW OPTIMIZATION BASED METHODOLOGY FOR CALCULATING
THERMODYNAMIC EQUILIBRIA AND QUANTIFYING UNCERTAINTIES

by

Jeffrey H. Snider
A Dissertation
Submitted to the
Graduate Faculty
of
George Mason University
In Partial fulfillment of
The Requirements for the Degree
of
Doctor of Philosophy
Mathematics

Committee:

_____	Dr. Maria Emelianenko, Dissertation Chair
_____	Dr. Igor Griva, Dissertation Co-Chair
_____	Dr. Harbir Antil, Committee Member
_____	Dr. Howard Sheng, Committee Member
_____	Dr. David Walnut, Department Chair
_____	Dr. Donna M. Fox, Associate Dean, Office of Student Affairs & Special Programs, College of Science
_____	Dr. Fernando Miralles-Wilhelm, Dean, College of Science

Date: _____ Summer Semester 2020
George Mason University
Fairfax, VA

New Optimization Based Methodology for Calculating Thermodynamic Equilibria and
Quantifying Uncertainties

A dissertation submitted in partial fulfillment of the requirements for the degree of
Doctor of Philosophy at George Mason University

By

Jeffrey H. Snider
Master of Science
George Mason University, 2012
Bachelor of Science
George Mason University, 2010

Co-Director: Dr. Maria Emelianenko, Professor
Co-Director: Dr. Igor Griva, Professor
Department of Mathematical Sciences

Summer Semester 2020
George Mason University
Fairfax, VA

Copyright © 2020 by Jeffrey H. Snider
All Rights Reserved

Dedication

I dedicate this dissertation to Totis Pittas. His contagious enthusiasm for mathematics, and learning of all forms, and his personal attention to my education set me on the path to graduate mathematics, and his memory still guides me.

Acknowledgments

I would like to thank the following people who made this dissertation possible.

Dr. Stephen Saperstone for welcoming me as an informal auditor of his lectures when the university would not admit me, and whose eager discussions of advanced material fired my interest in pursuing a graduate degree. My parents Jim Snider and Pat Glaunert for their unfailing support and encouragement throughout my life and in my academic career. Susan Snider for her support, patience, and tolerance of the seemingly endless process of completing this degree. Jackson Snider for his understanding that sometimes daddy has to do math work when he'd rather spend time with you than anything else. Andrew and Lorelei Crerar for providing a place to sleep in a time of transition, and a bicycle to ride to campus. The MITRE Corporation for the financial support it has provided as my employer and through its Advanced Graduate Degree Program. And my advisors Dr. Maria Emelianenko and Dr. Igor Griva for their guidance through this long process, the many discussions we had together, and the ideas they shared with me.

Table of Contents

	Page
List of Figures	vi
Abstract	x
1 Background	1
1.1 Definition of Key Terms	3
1.2 Introduction	4
1.3 Prior Work	8
1.4 CALPHAD's Magnetic Model	10
2 Systematic analysis of the modeling aspects in CALPHAD	28
2.1 Lagrangian	34
2.2 Set-based Formulation	36
2.3 Two Models	48
2.4 Computational Framework	51
2.5 Existence of the Lagrange Multipliers	54
2.6 Existence of Feasible Point	57
2.7 Existence of a KKT Point	60
2.8 Equivalence of Models	61
2.8.1 Karush-Kunn-Tucker conditions for two formulations	61
2.8.2 Second-Order Conditions at KKT points	69
2.9 Vacancies	91
2.10 Implementation	99
3 Optimization, Sampling, and Uncertainty Quantification	110
3.1 Special-purpose solver separating iteration into two steps	110
3.2 Sampling	111
3.3 Uncertainty Quantification	116
3.4 Future Work	136
3.5 Summary and Discussion	144
A AMPL model	147
Bibliography	164

List of Figures

Figure		Page
1.1	Heat Capacity by the Inden formula.	12
2.1	Energy curve for a notional phase with a miscibility gap (a). By including two instances for the phase, the miscibility gap is automatically spanned by a linear combination of two compositions of that phase (b), as occurs in physical materials at thermodynamic equilibrium.	48
2.2	Information flow through the framework.	52
2.3	Mapping Algorithm [1]. $P(x, T)$: the set of phases present at composition x and temperature T . $\text{card}(P)$ indicates the cardinality of P (always 1 or 2 in this idealized context). When there are two phases present, one has composition x_{lower} and the other x_{upper} , where $x_{\text{lower}} \leq x_{\text{upper}}$	53
2.4	For various values of G_{Va} , the molar Gibbs energy of single-sublattice phase with a single element “A” mixing with vacancy “Va”. Constants have been set to $T = 2000$; $G_{\text{A}} = -4e4$; $G_{\text{A,Va}}^0 = 1e5$; for $i > 0$, $G_{\text{A,Va}}^i = 0$; and $G_{\text{Va}} \in \{0, 31, 100, 310, 1000, 3100, 10000\}$. An equilibrium point is visible near $x = 1$ (no vacancy) for any selection of values, but for G_{Va} below approximately 50 it is not the global equilibrium: a composition of near-total vacancy ($x \approx 0$) has a lower energy. This includes the generally selected value $G_{\text{Va}} = 0$	95
2.5	The derivative of the curves in Figure 2.4 in the equilibrium region near 2% vacancy. We can see how the vacancy fraction of the equilibrium will change as G_{Va} is adjusted.	96
2.6	A plot of the energy corresponding to solely the pressure-volume portion from Pinwen Guan. The magnitude is 1/10000 that of our other terms: compare to the vertical axis in Figure 2.4.	100

2.7	A plot of the energy of our fictional system with the pressure-volume incorporated. Pay attention to the axes: $y_A \approx 0$, and the energy at the minimizer is far below that near $y_A \approx 1 \Rightarrow G_m \approx -4 \times 10^4$ in Figure 2.4.	101
2.8	Al-Ni Dupin and Sundman [2], the B2 phase at $T = 1000$ and molar composition = 0% nickel. The vacancy fraction x_{Va} is the vertical axis, $y_{0,Al}$ the horizontal axis, and $y_{0,Ni}$ the axis pointing into the page, but the feasible range for this is zero. Red areas are high energy, and blue areas are low energy. You can see there are two stable equilibria, one around $y_{0,Al} = 0.9$ and $x_{Va} = 0$, and the other around $x_{Va} = 0.95$. This demonstrates that a vacancy barrier above 0.25 cannot be prevented from creating local stability points.	102
2.9	Al-Ni Dupin and Sundman [2], the B2 phase at $T = 1000$ and molar composition = 100% nickel. The vacancy fraction x_{Va} is the vertical axis, $y_{0,Ni}$ the horizontal axis, and $y_{0,Al}$ the axis pointing into the page, but the feasible composition range for Al is zero. Red areas are high energy, and blue areas are low energy. You can see there are two stable equilibria, both around $x_{Va} = 0.5$. This demonstrates that choosing a vacancy barrier at or below 0.5 is not advisable.	103
2.10	Four bcc lattice structures: A2, B2, D03, and B32.	108
2.11	Three fcc lattice structures: A1, L1 ₀ , and L1 ₂	108
3.1	Energy of the AlNi system using the new AMPL-only (no solver) method of calculation.	112
3.2	Lower hull of the energy of the AlNi system using the new AMPL-only (no solver) method of calculation.	113
3.3	AlFe Sundman Option 2, bcc _{va} energy at 1400K. The left image shows 50,000 sample points uniform in feasible region, the right image shows 500,000 points. Neither is dense along the lower hull, nor in the extreme left or right edges of the image.	114
3.4	Sampling uniform in space projects non-uniformly on 1-d y subspace.	114
3.5	Uniform sampling in the feasible space, in 1-d, 2-d, and 3-d. N.b., the 3-d view is a projection onto the 3-d simplex from 4-d space.	115
3.6	Al-Pt energy at 1800K, without vacancy corrections.	119

3.7	(a) shows the Nb-Re phase diagram, (b) the corresponding diagram of the additional energy required to reach the lowest metastable phase in the middle, and (c) the ratio of initial conditions that converge to the lowest equilibrium to local equilibria on the right.	120
3.8	The Halton quasi-random sequence 2,3 (left) compared to pseudo-random points (right). The improved uniformity of sampling is visible in the Halton sequence, eliminating heavily over- and under-sampled regions apparent in the random sequence.	122
3.9	Al-Pt binary phase diagrams calculated with different vacancy constraints using Model 2 and <code>snopt</code>	125
3.10	High level sketch of the algorithm for finding UQ values over the T, x space.	127
3.11	Al-Pt energies for <code>bcc_a2</code> and B2 phases, showing many metastable states with energies very near the stable state. This motivates the choice of only taking an energy difference between equilibria that have at least one distinct phase in the mix.	129
3.12	The Al-Pt system, produced with Model 1 and <code>SNOPT</code>	131
3.13	The Al-Pt system, produced with Model 2 and <code>SNOPT</code>	132
3.14	The Al-Pt system, produced with Model 1 and <code>MINOS</code>	133
3.15	The Al-Pt system, produced with Model 2 and <code>MINOS</code>	134
3.16	Two unusual complementary views of energy in the Al-Pt system at composition $x(\text{pt}) = 0.15$, produced with Model 1 using <code>SNOPT</code> . The top graph has vertical axis T , horizontal axis $x(\text{pt})$ of the phase itself in equilibrium, and the relative size of the line shows the frequency with which that phase is found in equilibrium. The bottom graph shows the same data set with T on the vertical axis, and energy in J/mol on the horizontal axis. Since the composition of the overall system is always the same, the phases in equilibrium must be inferred from the other information on the graph – this can be challenging but revealing.	135
3.17	Energy and frequency of convergence for Al-Pt using Model 1 and <code>MINOS</code> at 806K. Smaller disks indicate fewer times converging to that phase in equilibrium.	135

3.18	The Al-Pt system at 1805K. These graphs show only equilibria the solver can converge to with that model, both stable and metastable points – unstable points are never identified. The presence of the L12 phase is apparent in the Model 2 + SNOPT graph, and absent in the Model 1 + MINOS graph, which gives insight into the differences in the corresponding phase diagrams. . . .	136
3.19	This graph is “sideways”, with temperature on the x -axis, fixed at composition $x(\text{pt}) = 0.34$. It shows the uncertainty in the liquidus from the perspective of metastable equilibria of the liquid phase.	137
3.20	The Co-Mo system.	137
3.21	The Co-Mo system, produced with Model 1 and SNOPT.	138
3.22	The Co-Mo system, produced with Model 1 and MINOS	138
3.23	The Co-Mo system, produced with Model 2 and SNOPT	139
3.24	The Co-Mo system, produced with Model 2 and MINOS	139
3.25	Meta-energy of the Co-Mo system, four histograms over the entire composition and temperature range. The first two are Model 1, the last two are Model 2. The first and third are SNOPT, the second and fourth are MINOS.	140
3.26	The Co-Mo system, four histograms over the entire composition and temperature range. The data is binned into bins with width 0.1 and then smoothed with a triangle distribution of width 0.4.	141

Abstract

NEW OPTIMIZATION BASED METHODOLOGY FOR CALCULATING THERMODYNAMIC EQUILIBRIA AND QUANTIFYING UNCERTAINTIES

Jeffrey H. Snider, PhD

George Mason University, 2020

Dissertation Co-Director: Dr. Maria Emelianenko

Dissertation Co-Director: Dr. Igor Griva

This dissertation presents a novel methodology for automatic calculation of thermodynamic equilibria and quantifying numerical uncertainty for arbitrary multicomponent materials systems using the CALPHAD method. The methodology is based on mathematical analysis of two different model formulations implemented using an efficient set-based framework, and then applying state-of-the-art nonlinear optimization algorithms to find thermodynamic equilibria and quantify uncertainty at the stage of Gibbs energy minimization. Unlike previously developed methods that rely on guessing material components present in the equilibria, the new methodology involves all the components of a material system in computations, and thus is more robust. The thesis establishes theoretical equivalence between the models and provides results on extensive numerical experiments that demonstrate the practical importance of the developed methodology.

This page intentionally left blank

Chapter 1: Background

With its origin in the work of J. W. Gibbs in the late 19th century, the CALPHAD method (CALculation of PHAse Diagrams) began to find extensive use as automated computing became increasingly available in the 1960's and later, and its method for extrapolating from empirical data on individual elements and binary and ternary systems to arbitrary systems of elements is now aiding the selection of advanced materials in 21st century fabrication projects. Today, as omnipresent computing power expands the reach of the method to a broader range of scientists, it is coming under increased scrutiny, its shortcomings are enumerated in the literature alongside its extraordinary power, and opportunities for improvement of the method are obvious. We identify crucial approaches to improving the method, and we develop additional improvements of our own.

The CALPHAD method is an extraordinarily valuable technique for determining thermodynamic properties of a chosen ratio of elements, at a given temperature and pressure, computationally rather than empirically. The method is rapidly expanding its reach to a wider range of materials scientists and other researchers, but has attracted relatively little interest from the mathematical community.

In recent years the CALPHAD method has been the subject of many important developments, rapidly broadening its appeal to wider audiences and enabling calculation of more features than free energy and atomic structure. Many old challenges remain incompletely addressed, and new challenges are created by each expansion of its scope and each new feature introduced to the method.

Until very recently even the most prominent thermodynamic software made no attempt to guarantee it was identifying the true minimum configuration and energy of a system. The

existence of optimization algorithms and software that more quickly and reliably identify the minima is a significant improvement for the method.

The Inden-Hillert-Jarl magnetic model has been used almost unquestioned for nearly four decades, but it is inadequate to modern needs and unnecessarily simplified, given current computing power. The possibility of replacing the IHJ model with a more sophisticated one is facilitated by our proposed framework, and enables research in an area of CALPHAD where little has been done recently.

While uncertainties are present at all stages of the method, from empirical measurements, to parameter fitting, and to material quantities determined for unknown systems, no effort has been made to maintain and convey that information to the researcher. Simply making uncertainty information visible to the researcher would present a significant improvement to the method.

We develop methods that solve the CALPHAD problem using existing and custom optimization software, e.g., AMPL and its accompanying solvers. Within that framework we develop various models and sub-models such as different magnetic models, variations on how vacancies are handled, etc., for comparison with published works. The output is visualized with custom plotting packages depending on the particular features being examined. Comparisons with published research for validation is conducted at the level of phase diagrams, energy diagrams, outputs common to thermodynamic software packages, and novel visualizations of uncertainty measurements.

The entire body of source code for database conversion, energy mapping, generation of phase diagram data, and plotting diagrams is in the process of being made publicly available on github. It will remain open source for academic and industrial community research.

1.1 Definition of Key Terms

Many of these terms are familiar to materials scientists, however the audience of this document includes many without that background. Here we give informal definitions of the terms encountered in this document, interpreted from a mathematician’s point of view.

element – a chemical element, i.e., a type of atom.

constituent – a constituent of a phase can be an element or a chemical compound of multiple elements.

crystallographic lattice – the repeated symmetrical arrangement of constituents inside a crystal.

thermodynamic (sub)lattice – a symmetric subset of a lattice corresponding to the identifiable positions in a unit of the parent lattice.

sites – a single identifiable position of a sublattice, sometimes also the total number of constituents in that position per mole of a phase.

end-member – a specific arrangement of constituents on the sublattices of a lattice, used to specify a free energy of a phase in a thermodynamic database. Some end-members are purely hypothetical and cannot be physically present.

microstate – a specific configuration of a system associated with a specific probability in statistical mechanics.

phase – a collection of microstates sharing a common crystal structure and set of constituents, defined by its properties such as enthalpy, volume, etc., and specified in a database by a set of end-members with corresponding energies as functions of temperature and pressure.

system – a fixed set of components; binary systems have two components, ternary have three, and systems with four or more components are termed multi-component.

component – the set of all constituents a system comprises; a phase has constituents, a system has components.

composition – the ratio of components in a system, which necessarily sum to unity.

vacancy – a site containing no constituent, often treated as a hypothetical constituent.

equilibrium – the set and ratio of phases giving the lowest possible energy for a given composition, temperature, and pressure.

global minimizer / stable – the equilibrium of a system, the set and ratio of phases giving the lowest possible energy.

local minimizer / metastable – a set and ratio of phases which has no lower energy in a small neighborhood, but not the lowest possible energy for that system.

phase diagram – a chart of phases present at given temperature, pressure, and composition.

liquidus – the temperature boundary on a phase diagram above which a system is entirely liquid in equilibrium.

solidus – the temperature boundary on a phase diagram below which a system is entirely solid in equilibrium.

1.2 Introduction

The CALPHAD method, developed and named for its ability to generate phase diagrams, is well suited for calculating not only equilibrium energies of systems and closely related features such as heat capacity, but also mechanical features like volume, thermal expansivity, bulk modulus, shear modulus, Young's modulus, Poisson's ratio, and other features such as electrical resistivity or conductivity, thermal conductivity, magnetism, and others.

The ability to extrapolate values for those features rests on a foundation laid originally over one hundred years ago with the work of Josiah Gibbs and later Johannes van Laar, but not put into practical use until the mid 1960's when automatic computation was regularly available to researchers and academics, at which point exponential growth in its use began. Key moments in the development of the CALPHAD method are in 1970 with Kaufman and

Bernstein’s [3] publication of a seminal text on computer calculation, in 1976 when Inden [4] formulated the magnetic model, slightly revised by Hillert and Jarl in 1978 [5] and still used in that form today, and in 1991 with Dinsdale’s [6] regularization of elemental parameters. Founded by Kaufman in 1977, the CALPHAD [7] journal has become a central clearinghouse for research related to the method.

Today “computational experiments” compliment physical gathering of data, for example in the form of Density Functional Theory (DFT) or other first-principles calculations. And the CALPHAD community is looking forward to improvements in its methodology and building on good collaboration with physicists and chemists to give the parameters used more recognizable physical meaning, and to model more features than the basic method allows.

An idealized use of the method is for materials scientists who desire a certain set of features in a novel material to computationally determine likely combinations of constituents, and only then turn to the laboratory or manufacturing floor where they can be assembled and physically tested. There are examples of this procedure being followed, e.g., Olson and Kuehmann [8] [9] describe the integration of CALPHAD into the successful development process of “designer alloys” for airplane landing gear, and Reed et al. [10] applied it to nickel-based “superalloys”.

The most important prerequisite in the foundation of the CALPHAD method is the ability to correctly and reliably identify the minimizer for the Gibbs energy of a system of constituents. Without the ability to identify the phases present, their relative concentrations, and the state of their internal variables, the method has no utility. Yet this fundamental necessity is rarely discussed, even as it becomes increasingly difficult in multicomponent systems containing phases with many internal degrees of freedom.

The extrapolative power of the core CALPHAD concept is valuable: creating equilibrium phase diagrams. However its ability to calculate features beyond Gibbs energy or heat capacity is increasingly tantalizing to the materials science community, and many efforts are being

made to extend the method to incorporate not only additional thermodynamic features, but also electrical features such as conductivity, and many mechanical features such as volume and bulk modulus.

The method has been in practice most often restricted to atmospheric pressure and to temperatures above 298.15K. There is great interest in expanding the pressure and temperature regimes over which CALPHAD can be applied

The ability of CALPHAD software to take input from and especially to provide output to other software and other stages of a material research or development effort is essential to its broad utility, and this has long been accomplished as regards diffusion (atomic mobilities), and grain growth studies. Further integration efforts include calculation of molar volume, elastic properties, thermal conductivity, thermoelectric properties such as electrical resistivity, optical and acoustic properties, and many others. Simplifying and enabling greater integration is an active topic of research in the Integrated Computational Material Engineering (ICME) community.

The magnetic model almost ubiquitously employed today is the Inden-Hillert-Jarl [5] model, a short Taylor expansion of the “computationally difficult” Inden [4] model. In 2010 Xiong [11] identified the need for a more sophisticated model, noting significant inaccuracies in the important Fe-Ni system, and proposed some alterations. It is worthy of notice that the “computationally difficult” Inden model is in fact quite simple to implement in AMPL and poses no challenges to our framework. There is much work taking place using *ab initio* methods to calculate magnetic parameters, and although in its magnetic model CALPHAD uses values which have some physical meaning, β and T_C , some focus has been placed on how the method can be made more directly representative of what occurs physically rather than merely fitting parameters for arbitrary curves to values only indirectly reflecting the underlying physics.

The presence of uncertainty within CALPHAD parameters has been explicitly acknowledged

at least since the 1977 paper by Lukas [12] on parameter fitting, but little has been accomplished in carrying those uncertainties through to the final product of the method and conveying the information to the researcher. The need to quantify the uncertainty at all stages is increasingly apparent, and some important steps are now being taken to do so.

Open-source software is a byword of this internet-driven decade, and its value and even its necessity affect the CALPHAD community as much as the rest of the scientific community. Some efforts have been made to produce completely free open-source CALPHAD software, yet many hurdles remain to producing reliable, extensible software usable by a majority of interested researchers.

Many aspects of the CALPHAD method are under active development, expansion, or integration into further materials science processes, and we cannot hope to cover all of them in this one review. These are areas for the researcher to be aware of, but which are not discussed here.

Data storage is an important and active theme in the literature [13] [14] [15]. In this dissertation we do not discuss data storage techniques or options. The assumption, which at the current moment is only infrequently violated in practice, is that all thermodynamic data is stored according to the *de facto* standard: in a “TDB” file, i.e., a file formatted according to the Thermo-Calc database standard [16], which accompanies a published paper discussing the researchers’ models and methods. The highly valuable line of research on updating how thermodynamic data is stored re-appears many times, especially in regards to reducing it to physically meaningful data rather than un-physical fitted parameters. Never the less, it remains beyond the scope of this review.

While the flexibility to adapt a model rapidly to test new proposals is essential, e.g., the two-state liquid model [17], and while we believe our model [1], elaborated below, to be the most flexible possible, we do not directly compare the ability of various pieces of software to make such adjustments.

There are many thorough and interesting histories of CALPHAD which have been published from different perspectives, and many papers focused on a singular topic include a thorough history as well. In this light, we do not present a detailed history of the method, but refer the reader to these insightful references [18] [19] [15] [20].

The CALPHAD community, including the larger materials science community now benefiting from wide adoption of the CALPHAD approach, have not uniformly adopted modern optimization approaches in their tools and research. We introduce the capabilities provided by modern applied mathematics to the method, in a framework which reproduces existing results from published work. We employ this framework which makes use of modern mathematical tools to evaluate prior research and as a testbed for solutions to outstanding problems identified in the CALPHAD and materials science literature.

1.3 Prior Work

Many themes connected to the CALPHAD method have emerged in the literature of various materials science communities as its extraordinary value has become more widely appreciated:

1. incorporating data from *ab initio* methods into CALPHAD
2. the extension of the method to additional material features
3. the extension to greater temperature ($0\text{K} \leq T < 298.15\text{K}$) and pressure ($P \gg 1\text{atm}$) ranges
4. integrating the CALPHAD method with other methods and processes
5. the need for reliably and quickly identifying global minimizers
6. the inadequacy of the Inden-Hillert-Jarl magnetic model
7. quantifying uncertainties at all stages

8. the benefit and growing ubiquity of open source software

First-principles methods are being increasingly used within CALPHAD, and the ability to determine material features not measurable empirically has provided enormous value to the method. Improving the connection between first-principles methods and CALPHAD is an active area of research. The parameters used in CALPHAD are fitted using empirical data, but have no inherent physical meaning. Taking an alternative approach that ties directly to physically relevant values would be a significant improvement in the method in the eyes of many researchers and scientists from other fields.

The method traditionally was used only to determine thermodynamic features, such as free energy, and heat capacity. It is readily extensible to other types of features, and a flexible framework should be able to incorporate many of those features.

Data for the CALPHAD method has traditionally been collected above 298.15K, while the need to determine material properties below that temperature is obvious and present. Methods have been devised to use alternative approaches to thermodynamic free energy, such as replacing Gibbs energy with Helmholtz energy, but this requires a significant retooling of the method. Similarly, it can be inconvenient to measure thermodynamic properties away from 1 atmosphere, so much of the CALPHAD data is for that fixed pressure only. Extrapolation to much higher pressures is necessary in many contexts, for example in Geophysics where pressures up to 360 GPa exist.

CALPHAD has huge potential for determining material features which are fed into an outside process. The need for integration into a larger research structure has significant implications for how a tool implementing the CALPHAD method can be accessed by other programs, how easily it performs without human intervention, and the speed and accuracy of its results.

Much of the thermodynamic software available today is closed-source, and some can be prohibitively expensive for a single academic license. The advantages of creating software which is open-source are many, not least is the importance of presenting reproducible results.

While this dissertation does not intend to act as an advocate for open-source, we examine its benefits and follow its precepts in developing our framework.

Many other themes are present, such as the flexible data storage techniques necessary to enable rapid development of new capabilities, but we have identified these relevant few beyond the scope of this dissertation to include in this brief review.

1.4 CALPHAD’s Magnetic Model

In the CALPHAD paradigm the parameters fitted to data often have little or no physical meaning, and in some cases this limits the predictive power of the method. For example the magnetic models available depend very little on the physics of magnetism at the atomic level, beyond the use of the β and the short-range-order to long-range-order ratio SRO/LRO labeled f which depends on the material structure. Assessments also typically use a short Taylor expansion of the formula for magnetic contributions, that is the Inden-Hillert-Jarl [5] model rather than the Inden [4] model, for reasons which do not apply to a modern framework. There is a further improved model from Xiong [21] which seems to be an improvement on Inden-Hillert-Jarl, but is in little use at present.

CALPHAD is inherently a non-physical model – it reflects a mathematical model of the free energy of a system, and need not rely on any physical meaning or parameters to do so. However it has become clear that incorporating physically meaningful elements in the model has value, as they allow more accurate energy estimates and better extrapolation.

One area where physical meaning was introduced into the CALPHAD method in the mid 1970’s is magnetism. In 1975 and 1976 Inden [4] formulated and presented a magnetic model for CALPHAD, which was appropriate but too computationally intensive for the computers and software available at the time to be fully incorporated. Following discussion by Miodownik in 1977 [22], in 1978 Hillert and Jarl [5] published a version of the Inden model simplified by using a short Taylor expansion of the computationally challenging functions.

This had the added benefit of smoothing over the pole in the heat capacity which occurs at the Curie temperature, and this model under the label Inden-Hillert-Jarl has been used unaltered in the three decades since its publication. More recently, Xiong in 2010 and 2011 [11] [23] noted that the Inden-Hillert-Jarl model was insufficient to reproduce certain effects in important ferromagnetic materials and proposed some modifications.

Most recently, Körmann et al., in their significant overview of lambda transitions [24] after the 2013 Ringberg workshops [25], cite the possibility of using “microscopic model Hamiltonians (Hubbard model, Heisenberg model and beyond) in combination with DFT-computed parameters” to calculate magnetic contributions. It is interesting to note here the 2013 paper by Körmann et al. [26] that observes that even the atomic model for magnetism is uncertain and subject to errors, and first-principles methods are still being revised and refined. They mention that the effects of electronic, magnetic, and vibrational excitations are not independent, as they are treated in the CALPHAD model. Further, “One of the drawbacks of current magnetic models within CALPHAD is their inability to actually predict a magnetic structure of a phase/alloy.” They do not propose a remedy, but the inadequacies of the current approach are obvious.

Magnetism has long been recognized as one of the most significant factors in the heat capacity of paramagnetic materials [27] – the “magnetic specific heat.” The ability to model a steep curve centered on a singularity by measuring a pair of physical values gives the method improved accuracy and simultaneously improved physical meaning. As more accuracy and precision is created with a physical model within CALPHAD, less importance is placed on the excess model and hence fewer or smaller terms are necessary to achieve the desired degree of overall accuracy and precision.

The necessity to apply a three-decades-old simplified model for computational reasons has been obviated in today’s high-speed computational environments, where even the most sophisticated formulas have their derivatives instantly computed by computer-algebra systems [1] [28], and with comparatively vast amounts of CPU resources available on every desktop

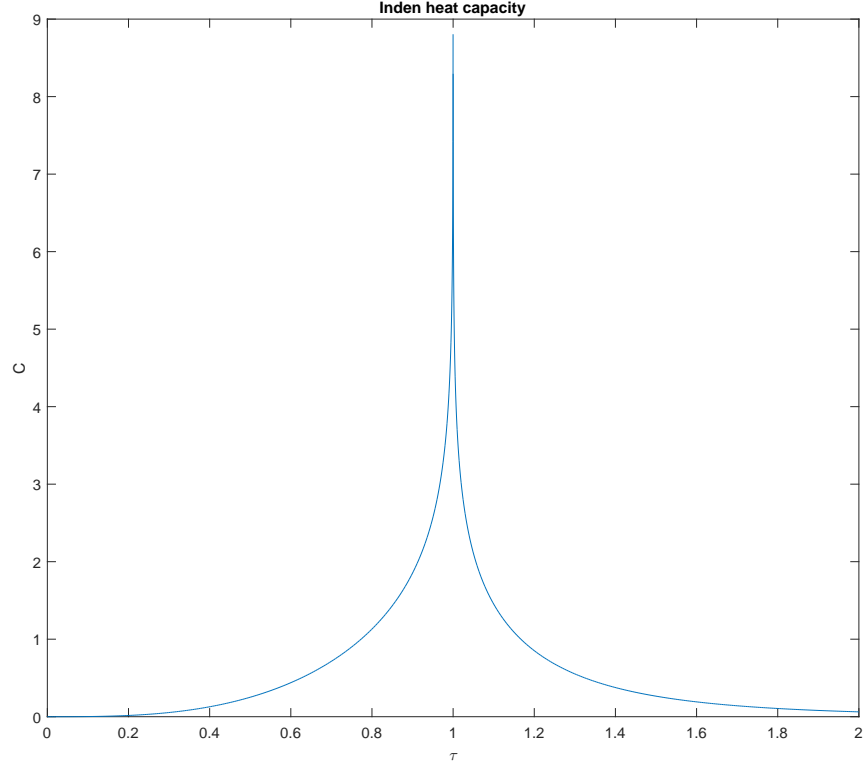


Figure 1.1: Heat Capacity by the Inden formula.

at very modest expense.

Heat capacity according to Inden [4] is

$$\begin{aligned} C_P^{\text{LRO}} &= K^{\text{LRO}} \cdot R \cdot \ln \frac{1+\tau^3}{1-\tau^3}, & \tau < 1, \\ C_P^{\text{SRO}} &= K^{\text{SRO}} \cdot R \cdot \ln \frac{1+\tau^{-5}}{1-\tau^{-5}}, & \tau > 1, \end{aligned} \tag{1.1}$$

where $\tau := T/T_C$, LRO corresponds to ferromagnetic and SRO to paramagnetic states, β is the mean magnetic moment, and $S_{\text{max}}^{\text{magn}} = R \cdot \ln(\beta + 1)$. Recall that $dG = dH - T \cdot dS$, where H is the enthalpy and S is the entropy.

Hillert and Jarl [5] found a short expansion of the above to be sufficient:

$$G_m^{\text{magn}} = RT \cdot \ln(\beta + 1) \cdot g(\tau) \quad (1.2)$$

where

$$g(\tau) := \begin{cases} 1 - \frac{1}{A} \left[\frac{79\tau^{-1}}{140p} + \frac{474}{497} \left(\frac{1}{p} - 1 \right) \left(\frac{\tau^3}{6} + \frac{\tau^9}{135} + \frac{\tau^{15}}{600} \right) \right], & \tau < 1, \\ -\frac{1}{A} \left(\frac{1}{10}\tau^{-5} + \frac{1}{315}\tau^{-15} + \frac{1}{1500}\tau^{-25} \right), & \tau \geq 1, \end{cases} \quad (1.3)$$

where

$$A = \frac{518}{1125} + \frac{11692}{15975} \left(\frac{1}{p} - 1 \right), \quad (1.4)$$

and p is the ratio of magnetic enthalpy in the paramagnetic state to the total magnetic enthalpy, called the “structure factor.” The accepted values are $p = 0.4$ in bcc and $p = 0.28$ in fcc and hcp.

Sundman et al. [29] in 2009 highlighted the inadequacy of the magnetic model in their assessment of the Al-Fe system by writing “The use of Redlich-Kister series are not well suited to describe the Curie temperature and Bohr magneton number as these may vary rapidly with composition and can be zero for an extended composition range. However, in order to calculate chemical potentials it is necessary to have a smooth function and thus they are used.” Here, an assessment of an extremely important system is being knowingly conducted with an inadequate model, as no alternative is ready.

Further, the magnetic model is well known to be inadequate at high pressures, such as those studied in geophysics and mineral physics. Gheribi et al. [30] examined the addition of pressure to the magnetic model and proposed a simple modification of Inden-Hillert-Jarl by adding a few P dependent terms.

The ability of a computer algebra system such as that contained within AMPL to calculate derivatives, and the ability of optimization codes to find minimizers of complex nonlinear

functions seems to eliminate the need for anything but the most precise physically representative functions. Xiong has taken some important steps in reconciling the CALPHAD magnetic model with current modeling needs, but further work is possible.

The framework we have developed readily allows comparison of differing models, including ones involving functions formerly considered “difficult”, such as the pure Inden model. Rapid prototyping of new models is simple, and this would enable a researcher in this field to find the most physically appropriate model in a minimum of time.

Data from first-principles

In the early 20th century as the CALPHAD approach was being developed, the first systems examined were unary, and then later binary, phases had at most one internal degree of freedom, all data was found by physical experiment, and phase diagrams were computed and drawn by hand. By 1970 when the Compound Energy Formalism (CEF) was clearly explicated, the need for parameters matching end-members which could not physically exist was obvious, however those parameters were still being found by extrapolation from stable configurations found in physical experimental data. As *ab initio* methods matured, it became possible to computationally model phases under conditions which are physically impossible. This has allowed CALPHAD researchers to fill in blanks in empirical data sets and has improved the accuracy of the method.

The volume of thermodynamic research being conducted using *ab initio* computational methods has grown dramatically in recent years, and is notably included in the charter of the Materials Genome Initiative [31] under “Goal 2: Integrate Experiments, Computation, and Theory.” The data gathered from these methods is used in parallel with data gathered empirically for parameter-fitting in the CALPHAD method. Further connections between the methods have been suggested, such as a direct coupling of an *ab initio* model to a CALPHAD framework.

The most prominent first-principles method in the CALPHAD literature is Density Functional

Theory (DFT), which models the quantum mechanical electronic structure of many-atom systems. Direct connections between DFT and CALPHAD were already being discussed in 2009 by Liu [32], and comparisons between the results of DFT calculations and CALPHAD calculations go back farther [33] [34]. Due to the complexity of the equations governing interactions, the number of atoms it is possible to simulate with DFT is strongly limited by computational resources, and at present only occasionally exceeds hundreds of atoms.

A significant challenge to implementing a DFT method is that the precise functionals governing exchange and correlation are not known, with the academic exception of the free electron gas. Thus all current codes rely on approximation exchange-correlation functionals, referred to in shorthand as xc functionals. The choice of xc functional determines much of the accuracy of a code and its applicability to a particular need. Further, these are based on a particular model such as the exact-muffin-tin-orbital method (EMTO) applied by Breidi et al. [35] to obtain specific structural properties. There are many such choices available to the researcher.

Many codes implementing DFT exist, including ABINIT, QUANTUM ESPRESSO [36], PAW [37], and VASP [38] – over 70 software packages including DFT are identified today in an online list of quantum chemistry and solid-state physics software [39]. According to Palumbo, “the choice of the program package and thermodynamic tools is determined by criteria such as availability and user friendliness, rather than accuracy and reliability for the specific material system at hand” [40]. Hence, there is no single platonic ideal of DFT which researchers can turn to, but it remains the primary source of first-principles data for CALPHAD researchers at the moment.

DFT was until recently only applied to the $T = 0\text{K}$ ground state, but now is being applied to “finite temperatures”, meaning non-zero absolute temperature with some accuracy [41]. This is a key step in making DFT data applicable to the CALPHAD problem.

Several additional atomic methods exist. These include *ab initio* Monte Carlo methods and

Molecular Dynamics which Becker et al. [17] applied to investigate liquids in the CALPHAD context after the 2013 Ringberg workshops [25].

First-principles methods can calculate features of materials at “regions in the temperature-pressure-composition space where experimental data are hard or impossible to achieve, or are simply absent” [40].

In particular, metastable phases by their very nature may not be possible to measure in the laboratory, as they are never physically present, but having a precise estimate of their theoretical thermodynamic features is essential to implementing the CALPHAD method. Significantly, the compound-energy-formalism (CEF) requires that energies be provided for each end-member in a phase, and some end-members are unstable and can never be measured directly, while others are un-physical, e.g., a mostly-vacant four-sublattice end-member, and can never exist. The energies are fitted as parameters using data from stable regions; however, using *ab initio* methods the energy associated with any end-member can be calculated directly, ideally reducing error introduced by the physical measurement and fitting processes. This is limited by the error of the *ab initio* method, but tests of the accuracy of DFT have generally shown it obtains error less than that of a physical measurement process when care is taken to select the correct DFT method to apply.

Further, first-principles calculations can be used to identify possibly erroneous data gathered by experiment, as Grabowski et al. [42] demonstrated. It is then possible to exclude such unreliable values from thermodynamic databases.

The three excitations of a material which generate its thermodynamic properties are vibrational, electronic (both discussed in [43]), and magnetic (discussed in [24]). Each of these is modeled on a first-principles basis. Further contributions come from point defects (discussed in [44]).

“A phonon is a discrete quantum of vibration” [45], which carries energy and “other quantum numbers”, analogous to a photon being a quantized wave. These lattice vibrations are

usually the major contributor to the enthalpy of a material [43], and a derivation of the heat capacity from first principles is made possible by way of the phonon concept. The precise lattice vibrations can only be calculated at the atomic level, and computationally simulating a large enough number of atoms to correctly determine the phonon values is a computational challenge, especially when considering quasiharmonics, second order terms which are required explain thermal expansion, and anharmonicities, cubic terms and higher in inter-atomic potentials. This is being tackled in unary systems today [43], with the hope of developing a larger database as simulations speed up and researchers develop higher throughput techniques.

Electronic contributions to free energy can be calculated in density-functional perturbation theory (DFPT) or supercell approaches, and several software packages implementing each are available [43]. The electronic contributions to heat capacity are important but smaller than those of harmonic and quasiharmonic contributions, and are negligible above the Debye temperature where all harmonic modes up to the highest frequency are excited.

Magnetic contributions are an example of a “Lambda transition”, which are discussed in depth by Körmann et al. [24]. In a ferromagnetic material the thermal energy required to overcome the electron exchange interactions as the Curie temperature T_C is approached becomes substantial. This results in a characteristic lambda-shaped spike in the heat capacity at T_C . Calculating the required energies from first principles involves solving the same Heisenberg-based Hamiltonians at the electron-electron interaction level, as in lattice vibrations above. Among others, Körmann et al. [26] have been able to model many of the important effects of magnetism using parameter-free methods.

Rogal et al. [44] review the contributions of various types of point-defects. Their effect on *ab initio* calculations is straightforward, assuming a heterogeneous mix of constituents can be modeled efficiently: an anti-site atom replaces the expected one in a lattice site, and a vacancy is the absence of an atom in a site. Other types of defect are uncommon enough to merit little discussion in the first-principles portion of CALPHAD literature at this time.

One interesting direction of research is to “store quantities such as phonon frequencies and the electronic DOS [density of states] and derive thermodynamic quantities at any temperature, pressure and volume directly from these physical meaningful quantities. These quantities can be obtained from first-principles, but experimental data can also be used” [43].

Integrating *ab initio* models directly into CALPHAD would be impossible due to their substantial computational expense. However, the possibility of finding analytical solutions to Heisenberg-based model Hamiltonians is being investigated by various researchers, as well as other promising alternative approaches relying only on first-principles data rather than fitted parameters [24]. For example, Gheribi et al. [46] elaborate a method specific to insulators for applying first-principles calculations directly within the CALPHAD method using alternative formulations based on physical models. Approaches such as these may allow the CALPHAD method to calculate its parameters from first-principles data, or rely directly on that data, but the concept remains in its infancy.

Methods such as DFT not only determine thermodynamic features such as those traditionally included in CALPHAD calculations, but additional features like volume, thermal expansivity, bulk modulus, shear modulus, electrical resistivity or conductivity, thermal conductivity, and magnetism. Basing CALPHAD on Helmholtz energy [47], which is natural in the context of DFT, may facilitate the expansion of CALPHAD into calculation of those other features, as discussed at length in Section 1.4, as well as reaching beyond the typical lower temperature limit of 298.15K, as discussed in Section 1.4.

Coupling the decades of accumulated experimental data with the results of *ab initio* calculations is enabling the CALPHAD method to improve its accuracy and its reach to additional material features. Directly applying *ab initio* methods within CALPHAD is a tantalizing possibility not yet realized.

The question of whether quantities such as phonon frequencies and electronic density of

states are sufficient to derive thermodynamic quantities directly at any temperature, pressure and volume is an interesting direction of research. The separate approach of finding analytical solutions to model Hamiltonians has promise of putting *ab initio* based methods inside CALPHAD. Either of these approaches has the possibility of moving CALPHAD away from fitted parameters with no direct physical meaning toward more physically meaningful databases.

Extension to Additional Material Features

The CALPHAD method was devised originally as a means of generating phase diagrams. This necessitated calculation of free energy, which correlates directly with the calculation of heat capacity and chemical potential. The method, however, is well suited to the calculation of any type of quantitative material feature which can be modeled with fitted curves. With this powerful technique in hand, researchers have recently sought to apply it to many other features.

This extension of the method to additional properties beyond those deduced from the free energy of phases and their driving force [13] is a topic of a substantial amount of research and corresponding literature, and covers a wide range of different features, for example:

- atomic mobilities and diffusion [48] [49] [50] [51]
- molar volume [52] [53] [54] [55] [56]
- elastic properties [53] [55] [57]
- thermal conductivity [46]
- electrical conductivity
- interfacial energies

It is important to distinguish between using the CALPHAD method to calculate a material feature and coupling the output of the method with additional software or algorithms. Here we focus on direct use of the method to determine specific quantitative values, and

in a following section we will discuss integration of the method with other software and algorithms. Very often the two uses overlap, and some references conflate the concepts, but we attempt to keep them separate and provide some clarity on the distinction.

The original concept behind what is now called CALPHAD, although as we see it provides many benefits beyond merely “calculating phase diagrams”, was to fit parameters to the curve representing free energy of each phase, and not to model specific physical processes. Fitting meaningful curves has been accomplished with increasing sophistication and accuracy over many decades, in many cases by including more physical meaning in the model, e.g., magnetism, order-disorder transition. However, the core concept of fitting parameters to match a set of curves can be applied to much more than just the free energy of each phase.

In this section we examine some efforts that have been made to expand the CALPHAD method in this way. There are a number of informative overviews of the subject, including Campbell [13].

The ability of the method to reproduce measured results, or results calculated by *ab initio* methods, has proven invaluable. Having this powerful tool in-hand, researchers naturally have identified other ways it could be applied. Someone interested in how elastic properties vary by changing the composition of a material would find this approach extremely useful – if sufficient data has been compiled on elastic properties of the relevant elemental, binary, and ternary systems. We do not address the very important need to collect such data before the method can be applied in these novel ways, but it is an absolute prerequisite.

CALPHAD is often touted as an ideal method of extrapolating features of unknown alloys. As applied what it does is determine the equilibrium phases present in a given alloy – and knowing the phases allows the researcher to infer physical features. Fortunately, the method can be expanded to model more than phase energies.

An approach making use of the Helmholtz energy [47] in place of the Gibbs energy has

proven effective at calculating thermophysical properties as well as thermodynamic ones, and is particularly effective in low-temperature or high-pressure regimes where CALPHAD is notably weak. A full switch of the CALPHAD method from Gibbs to Helmholtz energy would entail a significant re-surveying of data on the existing thermodynamic databases. It does, however, create a straightforward coupling between CALPHAD and first-principles techniques.

The most valuable CALPHAD-based tool will be able to incorporate these additional features and read the additional datasets as they are developed. To that end, any software tool should be flexible in its design, and allow modification by a variety of researchers and software developers. This implies the use of a language accessible to a wide range of people, and an open-source project. We address open-source in particular in a later section.

The CALPHAD method has been demonstrated at being effective and valuable in extrapolating from empirical measurements of simple systems to determine the features of more complex systems, sometimes with very high precision. Applying this technique to material features beyond the thermodynamics at the origin of CALPHAD is natural. While this approach has already shown important results, there remains much work to collect the relevant data on the features of interest, and to develop and share databases of that information.

Extension to Broader Pressure and Temperature Regimes

This section on extending the practical temperature and pressure regimes in which CALPHAD can be applied is included as an overview of the important subject, to alert the reader to known limitations on the CALPHAD method, and to provide references on work being done in the area. We do not identify a research question which we might address in this section beyond verification that any framework we develop can handle T and P values in the extended range.

The traditional CALPHAD method limits itself to temperatures above 298.15K, and databases generally restrict themselves to the range 298.15K to 6000K. If a researcher needs to predict

features of a material at a lower temperature, those databases will not be useful.

Although the formulas for lattice stabilities (energy of end-members) allow the inclusion of P terms, in practice they are rarely determined, instead assuming 1 atm or sometimes 100 kPa.

These limitations are not inherent to the method. Rather, they are a result of traditional data gathering techniques, due in part to the difficulties of measurements at high pressure or obtaining true equilibria at low temperatures.

The most recent and thorough survey of the expansion into additional pressure and stress regimes is Hammerschmidt et al. [58], following the 2013 Ringberg workshops [25], and Palumbo et al. [43] includes a section on temperatures below 300K. Lu and Chen in 2009 [47] showed that using Helmholtz energy in place of Gibbs energy is an effective means of expanding the temperature regime to 0K and the pressure regime to arbitrarily high levels, although this is not directly compatible with the existing thermodynamic databases.

In the CALPHAD model, magnetism and pressure are treated as independent factors. But they are not in fact independent; after the work of Guillermet in 1987 [59] this has been shown in separate contexts by at least Lu et al. [60], Gheribi et al. [30], and Körmann et al. [61]. Treating them independently does limit the ultimate accuracy achievable in high pressure regimes, but remains the *de facto* standard.

Many applications of the “designer alloys” envisioned by proponents of the CALPHAD method would be at temperatures well below 298.15K. Alloy components of a system operating at very low temperatures are of interest, hence the necessary temperature range examined must include single-digit Kelvin values. The present collection of databases does not make this feasible.

Geophysical processes occur at very high pressures, up to 360 GPa, which is 3.6×10^6 atmospheres. Geophysicists would benefit tremendously from having CALPHAD-type data available, but at the present moment very few systems have been examined at these very

high pressures.

Integrating with Other Methods and Processes

At its inception, the CALPHAD method was devised to create hand-drawn phase diagrams, and researchers, scientists, and industrial practitioners would refer to the diagrams in the course of their work. Today the diagrams may still be used in that visual format, however the quantitative data associated with the diagrams is extremely valuable when applied computationally in processes outside of the method. This integration into other industrial or scientific processes is an ongoing effort, with some noteworthy success stories while much work and standardization of the practice remains to be completed.

This section covers integration of the CALPHAD method with other methods and other algorithms or software such as materials simulation codes, e.g., solidification simulations [62] or phase field simulations [63]. It is important to distinguish between this integration with outside methods and the inclusion of new features in the CALPHAD method, such as the calculation of mobilities, which in itself is insufficient to model diffusion without coupling to software that can solve the relevant PDEs.

Industrial processes involve calculation of different material features, including [18] [8] [13] [9] solid diffusion (mobilities), molar volume, bulk modulus (elastic properties), electrical resistivity, thermal conductivity, thermoelectric properties, optical properties, acoustic properties, interfacial energies, surface tension of the liquid phase, nucleation, grain boundary diffusion, and thermal migration, among others.

Tools designed for the calculation of each of those specific features, both by continuum and atomistic models, have been coupled to greater or lesser extent with the CALPHAD method. The state of the art includes improving the coupling between calculation and application of those properties and tools applying the CALPHAD method, and enabling researchers from disparate fields to include the many kinds of tools within their efforts. There are active efforts ongoing [64] to ensure interoperability in the Integrated Computational Material

Engineering (ICME) [65] setting.

“Integrating materials science, continuum mechanics and quantum physics, the systems design approach has featured a suite of validated computational tools enabled by an iterative interaction of theory, simulation and experiment, spanning the full range of process/structure/property/performance relations,” [8] [9].

If we have reliable models for relevant thermodynamic, physical, electrical, or other properties of a material in its elemental form, in binary pairs and ternary combination, we can hope to extrapolate to larger multicomponent systems and computationally explore the composition space to identify materials with the necessary features for a given application. Computational researchers identify desired features of the material and use a variety of computational tools to model and simulate test materials before they are manufactured in a laboratory or on the production floor. The tools include, among others, precipitation simulators, and macroscopic component-level simulation tools [8], which can be coupled with each other to improve the speed of research.

Another example of solidification simulations being coupled with CALPHAD is Schneider et al. [66], where microstructures present in a final product were determined as different heat-treatments are applied. They compared the DICTRA package, for simulating diffusion reactions, [67] with Scheil simulations.

Although a long list exists of tools which may combine with CALPHAD, specific examples are scarce, and open-source or free code has not been found. This could be due to the complexity of the processes which can make use of CALPHAD, or that their profitable nature requires industry to keep some methods as private intellectual property.

Open-Source Software

“A number of CALPHAD software are available for the calculation of properties of multicomponent systems, and the majority are commercial products.” [68] In contrast, Gibbs [28], OpenCalphad [69], and pycalphad [70] are high quality open-source projects freely available

under typical open-source licenses, and are focused specifically on CALPHAD type thermodynamic calculations. Other open-source projects connected with materials science include FiPy [71] and the Virtual Kinetics of Materials Laboratory [72] incorporating it.

Additional features can be added to open-source projects, and improvements made by researchers only tangentially attached to the project. For example, in 2012 Piro and Simunovic [73] published algorithms for enhancing the calculations of thermodynamic equilibria. In the context of open-source, they could contribute those directly to the relevant projects.

An important element of the Materials Genome [31] project is “open data” and reproducibility of results, meaning that when research is published, data is made available in a format accessible to their peers. Open-source software speeds this process and allows all researchers to disseminate their state-of-the-art and accelerate the overall research effort.

Open-source software delivers certain features which closed-source does not [74] [75]. We discuss a few of these separately below.

With the source examinable by everyone, and especially by interested experts in the field, bugs and logic errors are more likely to be found and fixed than those present in closed-source code. The generally larger group of software developers contributing to the code means that once a bug is identified, or an issue raised, a person is available to fix it more quickly.

Software using proprietary data formats makes it difficult or impossible to switch to alternatives without losing data, and will incur a difficult transition period. Open-source software always makes its data formats public, and translation code between that open format and other formats is often freely available, and can be written by anyone with access to the second format. This means a researcher or research organization is not making a long-term commitment to a given software stack when they begin using an open-source package.

A researcher desiring to modify code for a specific research need can readily do so with open-source code. In contrast, it is not possible to modify closed-source code, and for-profit

organizations often require substantial sums of money to create custom modifications – sums well beyond the financial capacity of any but the very largest research organizations.

With open-source code, the cost is only the amount of time necessary for the research team to make modifications to the code.

Most open-source projects are available free of charge. Contrast this to software sold for-profit, which may cost tens of thousands of US dollars for a license. These for-profit codes often have an academic license available for students, but these are not available to researchers or have a limited time they may be used without charge.

Making a project open-source is an increasingly common decision, allowing outside researchers to evaluate and contribute to the code, and enhances the effort from all perspectives.

Synopsis

The CALPHAD method has demonstrated its value to a wide range of researchers and scientists, and is increasing its reach well beyond its namesake “Calculation of Phase Diagrams” to many other features of multicomponent systems. However, some old challenges to the utility of the method remain incompletely addressed, and fresh challenges arise as the method is stretched to cover new regimes, rely on new and different data sources, and treat disparate new material features.

The incorporation of data from first-principles methods has been invaluable in improving the accuracy and reliability of the method. Further integration and closer coupling between *ab initio* method and CALPHAD seems inevitable, although there are difficulties that stem from the long execution times that *ab initio* methods require. Updating CALPHAD parameters to have physical meaning is another important goal which faces serious obstacles in the near term.

Extending the temperature and pressure range in which the method can be applied is essential for its appeal to many specific types of researcher and scientist. Geophysicists cannot apply most CALPHAD databases due to the absence of high pressure data.

The CALPHAD approach of taking measurements on simple systems and extrapolating to multi-component systems may naturally be applied to many material features beyond the original thermodynamic ones. This work is ongoing, but much data remains to be gathered and validated.

The need to rapidly and reliably identify the minimizer of a system is a subject which has received little discussion in the CALPHAD literature, while many extraordinarily powerful optimization techniques exist in other mathematical literature. Applying these general-purpose techniques to the CALPHAD problem has great potential to speed the method and ensure its reliability without an experienced operator.

The magnetic models being used in CALPHAD are antiquated by today's computational standards. There is a tremendous amount of room to test new models and achieve a better fit to systems which the current models do not reflect accurately.

Uncertainties are present in the CALPHAD process at all stages, but are not displayed in any form to users of the method. This is a serious oversight which even the simplest steps could begin to address. Challenges exist in how uncertainty data would be conveyed visually, but many visualization options exist.

Chapter 2: Systematic analysis of the modeling aspects in CALPHAD

The enormous power of the CALPHAD method relies on its ability to determine at a given, fixed temperature, pressure, and composition: which phases exist, the Gibbs energy of each phase, chemical potentials, and driving forces. This can be extended even to metastable and physically unrealistic phases when desired. This is a straightforward problem readily solved using only basic methods in a system with few phases, especially phases with few or no internal degrees of freedom. Indeed, in the early days of the method researchers would plot energies by hand and use a straightedge to generate the lower convex hull of all phases [76], from which the result would be determined graphically. As the method has matured and become more sophisticated and as computing power increases geometrically with time, the complexity of the models used for phases has steadily grown. For phases modeled with several sublattices, and for systems with several or many such phases, the computational challenge of determining the values of those internal variables which minimize the overall energy can be considerable – even by today’s computing standards. A poorly formulated problem and algorithm could be expected to require more computing resources than theoretically available over the course of the next hundred years.

If a minimizer is found which is only a local minimizer, i.e., a metastable condition, while the global minimizer produces lower overall energy, and is believed to be the global minimum, then the result produced by the method will be incorrect and misleading.

Additionally, there are different formulations of the same problem which can be easier or more difficult to solve. In particular, whether one explicitly calculates the energy of each phase per mole of atoms, requiring a correction for the fraction of sites occupied by vacancy, has an important effect on the method of solution.

Having the means to reliably identify minimizers means the Hillert algorithm [77], while profoundly insightful, and valuable on the slower and simpler computing platforms of the past, is now facing competition. And yet it continues to be used both in well established commercial software like ThermoCalc [78] and new software projects such as OpenCalphad [69]. Modern optimization techniques can facilitate the discovery of a global minimizer, as we do herein and in [1], and other projects such as Pandat [79] and Gibbs [28].

The meaningful results produced by the CALPHAD method rest upon the ability to reliably and quickly find the lowest energy configuration of a system. Software which identifies metastable or unstable phases as if they were stable is misleading and untrustworthy. If the researcher does not obtain repeatable, reliable and accurate results, then the method would have little value in scenarios where much is at stake. Further, if only a highly experienced specialist can produce good results with the method, then it lacks broad applicability outside the relatively small field of CALPHAD researchers.

Recent research has focused in part on high-throughput CALPHAD methods [80] [81] [82], as mentioned in the Materials Genome Initiative’s Objective 2.3 [31], implying a need for extremely rapid convergence to the stable phases without direct human intervention.

Researchers developing software tools have traditionally created a model simple enough for them to find derivatives by hand, to implement themselves even when they are not primarily software developers, and to run quickly – even with today’s extremely fast computational resources some researchers remain focused on streamlining code. Each of those trade-offs, such as selecting a simpler model to facilitate analytical work and reduce development effort, limits the potential of the framework which is ultimately developed. Hence, there is a tremendous advantage in separating the software tools which conduct optimization from the thermodynamic model.

In the framework we describe here, based on AMPL [83], the model is entirely separate from both the data and the wide range of solvers available to find minima [84] [85] [86]. This allows

the researcher to focus on the structure of the model, and rely on a wide range of expertly developed optimization solvers to conduct the difficult and often quite specialized work of finding minima, as well as the computer algebra code within AMPL which finds derivatives automatically, similarly to the symbolic mathematics within the software package Gibbs [28].

In such a framework models are easily and rapidly changed and compared. For example, implementing the original Inden [4] magnetic model is straightforward in AMPL, and comparing it to the Inden-Hillert-Jarl or the more recent Xiong modification, as well as testing additional alternatives a researcher may have in mind, can be as easy as changing one line in the model.

We can also take various approaches to modeling vacancies (e.g., with and without molar correction), to finding the minimizers (the Hillert algorithm [77], or the full sum-of-phases [1]), handling order-disorder transitions. In fact any part of the model is readily modified to suit a researcher’s needs.

Care must still be taken to ensure functions are “smooth enough,” and that they can be evaluated outside the feasible region. For example the entropy terms $y \ln y$ are singular at 0, and complex for $y < 0$, which is handled with the typical glue function, a piecewise function,

$$\text{Ent}(y) := \begin{cases} 0 & y = 0 \\ y \ln y & y > 0 \end{cases} \quad (2.1)$$

which is continuous for the feasible region $y \geq 0$. For practical application we use the

following glue function for the entropy term,

$$\epsilon := 10^{-8} \quad (2.2)$$

$$a := \frac{1}{2\epsilon} \quad (2.3)$$

$$b := \ln(\epsilon) + 1 - 2a\epsilon = \ln(\epsilon) \quad (2.4)$$

$$c := \epsilon \ln(\epsilon) - a\epsilon^2 - b\epsilon = -\frac{1}{2}\epsilon \quad (2.5)$$

$$\text{Ent}(y) := \begin{cases} ay^2 + by + c & y < \epsilon \\ y \ln y & y \geq \epsilon \end{cases} \quad (2.6)$$

which is well-defined and continuously differentiable everywhere on \mathbb{R} . At $y = 0$ the glue function differs from the exact value by $\frac{1}{2}\epsilon$.

Although we can prove that at equilibrium all y are positive, the glue function is necessary as some optimization solvers will make use of the objective value at infeasible points, including negative y values, as they converge toward feasibility and a minimum objective value. The freedom to try many different models does not free the researcher from taking proper mathematical care with the equations.

The solvers used in this research take different paths toward convergence, SNOPT using a quadratic approximation at each iteration, and MINOS taking steps according to a quasi-Newton direction with respect to linearized constraints. The difference in how they take steps and converge results in some equilibria having relatively larger effective basins of convergence with one solver and smaller with another, depending on the parameters present in the thermodynamic database.

The CALPHAD method determines equilibrium phases and their energy by solving the problem that follows, here using notation from Sundman [77]. For clarity some details are omitted

here, such as the precise magnetic formulation, order-disorder transition, and multi-state phases like the liquids of Ågren [87]. It is important to remember at all times that the G terms that follow are functions of T and P as well as Y , however temperature and pressure are held fixed in these examples and are omitted in the interest of space.

$$\min_{\aleph, Y} \sum_{\alpha} \aleph^{\alpha} G_{\text{M}}^{\alpha}(Y^{\alpha}), \quad (2.7)$$

$$\text{subject to } \sum_i y_{is}^{\alpha} = 1, \quad \forall \alpha, s, \quad (2.8)$$

$$\text{subject to } \sum_{\alpha} \aleph^{\alpha} M_A^{\alpha} = \tilde{N}_A, \quad \forall A, \quad (2.9)$$

$$\text{where } M_A^{\alpha} := \sum_s a_s^{\alpha} y_{A,s}^{\alpha}, \quad \forall \alpha, \quad (2.10)$$

$$\text{where } G_{\text{M}}^{\alpha}(Y^{\alpha}) := \sum_h G_h^{\alpha} \prod_{i,s \in h} y_{is}^{\alpha} + RT \sum_{i,s} y_{i,s}^{\alpha} \ln y_{i,s}^{\alpha} + G_{\text{magn}}^{\alpha}(Y^{\alpha}) + G_{\text{E}}^{\alpha}(Y^{\alpha}). \quad (2.11)$$

Here, in Sundman's notation, each phase is indexed by α , end-members in that phase by h , constituents by i and separately by A , and sublattice by s . T is temperature and P is pressure. \aleph^{α} indicates the amount of a phase, $y_{i,s}^{\alpha}$ is the fraction of constituent i on sublattice s in phase α , and Y^{α} represents the set of all y values for phase α . a_s^{α} is the site ratio of sublattice s , \tilde{N}_A indicates the amount of a constituent in the whole system, and M_A^{α} the amount of a constituent in phase α . G_{magn}^{α} is the magnetic component of the energy.

G_{E}^{α} is the excess term, typically modeled by Redlich-Kister polynomials [88] as follows,

$$G_{\text{E}}^{\alpha} = \sum_{h,s,\nu} \left[\left(\prod_{i,t \in h} y_{it}^{\alpha} \right) (y_{A,s}^{\alpha} - y_{B,s}^{\alpha})^{\nu} \right]. \quad (2.12)$$

Here h is an endmember, t is an index over sites and i over constituents, where s is the site with mixing between constituents A and B of order ν .

The standard procedure, as explicated by Sundman [77], is to put the above equations explicitly into a Lagrangian and develop a minimization algorithm by hand.

The alternative we demonstrate herein and in [1], is to use an optimization modeling language to represent the objective and constraints and let a software “black box” efficiently find the solution and corresponding Lagrange multipliers. This frees the developer and researcher from handling the low level software concerns in optimization, and allows them to leverage extremely mature optimization techniques.

As examples used in this work, we look briefly at the commercial solvers SNOPT and MINOS. SNOPT [84] [89] handles nonlinear objective functions with linear and nonlinear, equality and inequality constraints. It solves successive quadratic subproblems with a merit function based on a sparse quasi-Newton approximation of the Hessian of the Lagrangian. Each quadratic subproblem is handled by solving a sequence of linear problems from the reduced Hessian $Z^T H Z$, where Z is treated implicitly from LU factorization of the Jacobian, and the working set of active constraints is updated with each minor iteration. SNOPT has a well-developed set of techniques for handling infeasibility, negative curvature, and producing finite-difference estimates when derivatives are unavailable, and is well suited to large nonlinear problems with many active constraints so that the number of degrees of freedom remains relatively low, although the code can handle a problem of arbitrary size. By default SNOPT identifies that it has converged when the complementarity gap and infeasibility have been reduced to 10^{-6} or less; the values can be controlled by the user.

The solver MINOS (Modular In-core Nonlinear Optimization System) [86] [90] is well suited to linear, quadratic, and general nonlinear problems and all types of constraints. MINOS will use different methods to solve different classes of problems; for nonlinear problems with nonlinear constraints, in major iterations an approximation of the problem with linear constraints is solved using a reduced gradient method. The minor iterations use a reduced Hessian of the augmented Lagrangian with linearized constraints and is solved with a quasi-Newton method. Although MINOS was released in 1980 it remains an effective solver and is

heavily used in academia and industry.

However, there is a benefit from working analytically with the Lagrangian, as it makes the values of the Lagrange multipliers available for use in the equations.

2.1 Lagrangian

The Lagrangian can be written as follows

$$\mathcal{L} = \sum_{\alpha} \aleph^{\alpha} G_{\text{M}}^{\alpha}(Y^{\alpha}) - \sum_{\alpha, s} \lambda_s^{\alpha} \left(\sum_i y_{is}^{\alpha} - 1 \right) - \sum_A \mu_A \left(\sum_{\alpha} \aleph^{\alpha} M_A^{\alpha} - \tilde{N}_A \right). \quad (2.13)$$

Here the terms λ_s^{α} and μ_A are Lagrange multipliers, also called dual variables, corresponding to the two constraints: that the sum of fractions on each sublattice is unity, and that the amount of each constituent in the system matches the specified concentration.

The solution in terms of \aleph and Y must satisfy the Karush-Kuhn-Tucker (KKT) conditions. As this material does not appear in the relevant literature, we explore it as an appendix to the research results chapter.

Hillert [91] showed that the μ_A dual variable is in fact the chemical potential for constituent A . We prove that the Lagrange multipliers exist further below.

A wide range of methods exist for solving this problem [92]. We discussed the commercial solvers SNOPT and MINOS above, but we must note they are equipped only for identifying a local minimum given a certain initial starting point, or initial condition.

An effective means of finding low energy points in the configuration domain, although this does not guarantee the very lowest, is to randomly sample the domain and take the lower convex hull. This is the approach originally used by Gibbs [28] and `pycalphad` [70] [93], and the latter has recently incorporated as a second refinement step of optimization.

Each point is a set of consistent y values for a phase at a given temperature and pressure, and corresponds to a single composition. Many such points are picked according to a random or pseudo-random scheme, and the energy at each point is computed. The lower convex hull in the composition-energy space gives an approximation to the energy curve for that phase. The collection of energy curves for the phases in a system can be used to either generate a lower hull directly, or as input to a common-tangent algorithm.

The difficulties of this approach include its approximate nature, and the large space of consistent y values, sometimes requiring a very large number of points to be sampled to achieve a reasonable density along the lower hull.

The formulation is dramatically affected by how vacancies are handled in the model. The original model does not appear to anticipate the use of vacancies, and they were included later as a physically important phenomenon.

The formula for computing the Gibbs free energy of a phase determines the energy per mole of sites G_M , labeled the mole formula unit, while what the researcher is interested in is the energy per mole of atoms or constituents G_m . The two can be related by the fraction of sites occupied by vacancies x_{Va} :

$$G_m = \frac{G_M}{1 - x_{\text{Va}}}. \quad (2.14)$$

A few methods [1] minimize molar energy directly using G_m , while others [77] calculate chemical potential μ_j for constituents j and use the relation

$$G_m = \sum_j x_j \mu_j. \quad (2.15)$$

Today varying approaches to point defects such as vacancies is a subject of much research, including modeling using first-principles techniques. Rogal et al. [44] demonstrate techniques to determine point defect concentrations, and how defect formation energies can be

formulated in various formulations.

Alternative views on how to handle vacancies exist, such as giving them a non-zero energy [94]. This is a mathematically effective technique, but as it has no physical basis it has not been widely adopted. Ågren and Hillert [95] discuss some of the problems that arise when treating vacancy as a constituent, identify the unphysical high-vacancy equilibria, and among other things propose a variable value of G_{VaVa} rather than Franke’s fixed values. They state it is not possible for present software for CEF to handle varying parameters, while in fact it is possible within our framework. Examination of this new approach is outside the scope of this dissertation and must be left to future work.

2.2 Set-based Formulation

A variety of sources in the literature document pieces of the CALPHAD model, however none is comprehensive and some are contradictory. We developed our approach to minimizing Gibbs energy, capable of handling an arbitrary number of constituents and phases with any number of sublattices, miscibility gaps, order-disorder transitions, and magnetic contributions. Our approach is set-based, avoiding the need to iterate explicitly through many levels of nested loops. We have demonstrated its accuracy on a variety of systems, including Al-Pt, Co-Mo, and Ca-Li-Na, which are known to be computationally challenging. We reproduce phase diagrams created via Thermo-Calc, and identified areas where previously published work may have been based on local equilibria rather than global.

The standard Gibbs energy minimization problem formulation (Model 1) described in [96],

[97], [98] can be written as,

$$\left. \begin{aligned} \min_{f,y} G &= \sum_{\phi} f^{\phi} G^{\phi}(y) \\ 0 &\leq f^{\phi} \leq 1 \quad \text{for each phase } \phi \\ 0 &\leq x_e^{\phi} \leq 1 \quad \text{for each phase-species pair } (\phi, e) \\ 0 &\leq y_{s,e}^{\phi} \leq 1 \quad \text{for each phase-species-sublattice triplet } (\phi, e, s) \\ \sum_e y_{s,e}^{\phi} &= 1, \quad \text{for each phase-sublattice pair } (\phi, s) \\ \sum_{\phi} f^{\phi} x_e^{\phi} &= f_e^0, \quad \text{for each species } e, \text{ such that } \sum_e f_e^0 = 1 \end{aligned} \right\} \quad (2.16)$$

Here x_e^{ϕ} indicates the mole fraction of species e within the phase, $y_{s,e}^{\phi}$ is the mole fraction of species e in sublattice s within the phase, a_s^{ϕ} is the site fraction at sublattice s in the phase, and f^{ϕ} indicates the mole fraction of the phase in the overall composition. Similarly to [96], rather than being the subject of a constraint, x_e is a function of $y_{s,e}$ in this implementation:

$$x_e^{\phi} := \frac{\sum_s a_s^{\phi} y_{s,e}^{\phi}}{\sum_{c \neq \text{Va}} \sum_s a_s^{\phi} y_{s,c}^{\phi}}, \quad (2.17)$$

which reduces the number of variables and constraints.

In 2013 we developed a version of the CALPHAD model in AMPL, derived from a spreadsheet-based implementation. The strict matrix-based implementation placed severe constraints on the flexibility of the model and required a substantial software development effort for every small change that was desired. This motivated a more flexible framework.

We developed the entirely set-based model and implemented it in AMPL with an accompanying Java-based database (TDB) converter. This is a constrained nonlinear problem, and two suitable free solvers were identified (`snopt` and `minos`) and run independently at each T, c point to test for the optimal solution.

Motivating the set-based approach is the complexity of the many and varying number of nested sums in Redlich-Kister polynomials for the excess energy, depending on the phases present in the system, for a traditional indexed approach.

A phase having a single sublattice, where the standard form for excess energy ${}^{\text{xs}}G_m^\Phi$ is given, e.g., in [99], as:

$$\sum_i \sum_{j>i} x_i x_j \sum_k {}^kL_{i,j}^\Phi (x_i - x_j)^k. \quad (2.18)$$

Relabeling kL as kG terms, substituting x for y , and moving the inner sum to the outside, (2.18) is written as,

$$\sum_k \sum_i \sum_{j>i} {}^kG_{i,j} y_i y_j (y_i - y_j)^k. \quad (2.19)$$

If there is mixing between two constituents in one sublattice, the formula can be written,

$$\sum_s \sum_i \sum_{j>i} \sum_\ell \sum_m \sum_n y_{s,i} y_{s,j} y_{r,\ell} y_{t,m} y_{u,n} \sum_k {}^kL_{i,j:\ell:m:n} (y_{s,i} - y_{s,j})^k. \quad (2.20)$$

Relabeling kL as kG terms, replacing x with y , and moving the inner sum outside (2.20) is written as,

$$\sum_k \sum_s \sum_i \sum_{j>i} \sum_\ell \sum_m \sum_n {}^kG_{i,j:\ell:m:n} y_{s,i} y_{s,j} y_{r,\ell} y_{t,m} y_{u,n} (y_{s,i} - y_{s,j})^k. \quad (2.21)$$

Two additional summations enter the formula if there is mixing at two sublattices. Other models require more or fewer summations.

Observe that with appropriate definitions of G and y , (2.19) is identical to (2.18), and (2.20) is identical to (2.21).

Applying this set-based framework with appropriate indices and sets, (2.19) and (2.21) is

generalized as follows:

$$\sum_{(\sigma, \nu)} \left[{}^\nu G_\sigma \left[\prod_{(s, e)} y_{s, e} \right] \sum_{(s_m, e_0, e_1)} (y_{s_m, e_0} - y_{s_m, e_1})^\nu \right]. \quad (2.22)$$

A compound's constituent array, first introduced in [100], denoted by σ , indicates which constituents may be present at which sublattices, and is coupled with ν indicating mixing order. In (2.22), σ ranges over the same constituent arrays as i, j , etc., and ν represents all possible orders similar to the k index in (2.18)–(2.21). The tuple (s, e) defines the sublattice-species pair and depends on the constituent array σ ; while the pairing (s_m, e_0, e_1) reflects the sublattice and species of mixing.

The feature distinguishing (2.18) or (2.20) and (2.22) is that (2.18) or (2.20) can only be used to model a particular fixed number of sublattices and mixing sites, while (2.22) can be used to model arbitrarily varying sublattices and additional mixing sites. Below, we elaborate the set based formulation, the inclusion of order-disorder and magnetic contributions, and finally produce the general energy formula in (2.23) and (2.44).

Set Based Energy formulation

We now write out the details of the set-based formulation. In this discussion the following notation is used: p is a particular phase, e indicates a constituent (element), and e_0, e_1 denote mixing constituents. σ indicates a particular constituent array, s is a sublattice, s_m indicates the sublattice where mixing is occurring with ν being the order of mixing in the Redlich-Kister model – if there is no mixing then ν is zero. ${}^\nu G_\sigma^p$ is the Gibbs coefficient for phase p , constituent array σ , and mixing order ν .

By use of unified index sets, a single formula can apply to an arbitrary number of sublattices and any number of mixing sites per compound. These index sets are discussed below. Before we examine those sets, we can write the energy term without disorder or magnetic

contribution. It is distinguished from the complete energy term G^p and other similar terms by a subscript \star ,

$$G_{\star}^p = \sum_{(\sigma, \nu) \in S^{(p)}} \left\{ {}^{\nu} G_{\sigma}^p \left[\prod_{(s, e) \in T_{\sigma, \nu}^{(p)}} y_{s, e}^p \right] \sum_{(s_m, e_0, e_1) \in X_{\sigma, \nu}^{(p)}} (y_{s_m, e_0}^p - y_{s_m, e_1}^p)^{\nu} \right\} - R T \sum_{(s, e) \in T^{(p)}} \left(a_s^{(p)} y_{s, e}^p \ln y_{s, e}^p \right). \quad (2.23)$$

Summation is conducted over every index from the data file, working from outer to inner sum and product. The first sum is over each constituent array and mixing order pair (σ, ν) for phase p . Then the product is over each sublattice and species pair (s, e) for that (p, σ, ν) . Most often ν is zero, except in first or higher order Redlich-Kister terms, where the final sum is over all mixing site and pair of mixing species (s_m, e_0, e_1) which exist for (p, ν, σ) . The term $\prod_{s, e} y_{s, e}^p$ expands into a product of y values for sublattices s and species e . In the presence of mixing in sublattice s_m between species e_0 and e_1 the Redlich-Kister term is

$$\left[\prod_{s, e} y_{s, e}^p \right] \sum_{s_m, e_0, e_1} (y_{s_m, e_0}^p - y_{s_m, e_1}^p)^{\nu}. \quad (2.24)$$

In the case of a single mixed sublattice, the sum is over a single tuple. In a compound with no mixing the sum is empty (see (5.65) in [96]).

When multiplying other terms, both the sum and product over the empty set equal one: $1 \cdot \sum_{\emptyset} \cdot = 1$, and $1 \cdot \prod_{\emptyset} \cdot = 1$. Similarly, when adding the sum or product over the empty

set they are zero: $0 + \sum_{\emptyset} \cdot = 0$. Hence in the case of no mixing, the sum is empty and

$${}^{\nu}G_{\sigma}^p \left[\prod_{s,e} y_{s,e}^p \right] \sum_{s_m, e_0, e_1} (y_{s_m, e_0}^p - y_{s_m, e_1}^p)^{\nu} = {}^{\nu}G_{\sigma}^p \left[\prod_{s,e} y_{s,e}^p \right]. \quad (2.25)$$

(see (5.67) and (5.70) in [96]).

The energy for a phase will be a sum of terms like (2.24) with their respective coefficients ${}^{\nu}G_{\sigma}^p$, plus the entropy term. In [96] the coefficients are in the deepest part of the sum denoted by ${}^{\nu}L_{ij}$; below they are pulled out front as ${}^{\nu}G_{\sigma}^p$ so that one unified set of coefficients applies to mixing and non-mixing conditions equally. Each ${}^{\nu}G_{\sigma}^p$ coincides with a particular constituent array σ and mixing order ν .

The entropy sum

$$- R T \sum_{(s,e)} \left(a_s^{(p)} y_{s,e}^p \ln y_{s,e}^p \right), \quad (2.26)$$

is over all existing (s, e) pairs for that phase p .

Example 1. *A simple two sublattice phase; $p := \text{Al}_3\text{Pt}_2$. [1]*

In the Al-Pt binary system, consider the phase which is the stoichiometric compound $p := \text{Al}_3\text{Pt}_2$, where the sole constituent array Al:Pt is modeled with two sublattices having site fractions $a_1^{(p)} = 0.6$ and $a_2^{(p)} = 0.4$. In this case the only (σ, ν) pair is (AL:PT, 0). For this (σ, ν) pair the (s, e) pairs are (1, Al) and (2, Pt). Since there is no mixing, the set of (s_m, e_0, e_1) tuples is empty, and the empty product is taken to be 1. There is only one term in the outer sum, $\sigma = \text{AL:PT}$, $\nu = 0$, hence for this phase

$$G_{\star}^p = {}^0G_{\text{Al:Pt}}^p y_{1,\text{Al}}^p y_{2,\text{Pt}}^p - R T \left(a_1^{(p)} y_{1,\text{Al}}^p \ln y_{1,\text{Al}}^p + a_2^{(p)} y_{2,\text{Pt}}^p \ln y_{2,\text{Pt}}^p \right) \quad (2.27)$$

Note that the coefficient ${}^0G_{\text{Al:Pt}}^p$ with a preceding zero is distinct from the pure energy term ${}^\circ G_{\text{Al:Pt}}^p$ in the literature. The zero is superfluous in a context with no higher order mixing, but included here for completeness.

Index Sets

Index sets are used in the sums and products in (2.23), where some sets are themselves indexed by other sets. Define P to be the set of all phases p in the system, and E the set of all species e in the system. Set $S^{(p)}$ lists as tuples (σ, ν) all constituent arrays σ in phase p with all corresponding mixing orders ν . E.g., if a particular constituent array σ has mixing of orders 0, 1, and 2, then $S^{(p)}$ will contain $(\sigma, 0)$, $(\sigma, 1)$, and $(\sigma, 2)$, perhaps among others.

For each constituent array and order (σ, ν) , each set $T_{\sigma, \nu}^{(p)}$ lists as tuples (s, e) all sublattices s in the constituent array and the species e which exist in the data file at that sublattice for that mixing order; the set $T^{(p)}$ lists all (s, e) pairs for the phase irrespective of compound (it is the union of all $T_{\sigma, \nu}^{(p)}$); and for each constituent array and order (σ, ν) , each set $X_{\sigma, \nu}^{(p)}$ contains all mixing sublattices and the species which mix there (s_m, e_1, e_2) .

In other words,

$$\text{constituent arrays of } p \quad S^{(p)} \equiv \{(\sigma, \nu) \mid (\sigma, \nu) \text{ in } p\}, \quad (2.28)$$

$$\text{species sites of } \sigma \quad T_{\sigma, \nu}^{(p)} \equiv \{(s, e) \mid (s, e) \text{ in } \sigma\}, \quad (2.29)$$

$$\text{species sites of } p \quad T^{(p)} \equiv \bigcup_{(\sigma, \nu) \in S^{(p)}} T_{\sigma, \nu}^{(p)}, \quad (2.30)$$

$$\text{mixing sites of } \sigma \quad X_{\sigma, \nu}^{(p)} \equiv \{(s_m, e_0, e_1) \mid (s_m, e_0, e_1) \text{ in } \sigma\}. \quad (2.31)$$

Example 2. *A simple phase with multiple compounds; $p := \text{Laves}$. [1]*

In the Ni-Al system the C14_LAVES phase comprises the four constituent arrays AL:AL,

AL:NI, NI:AL, NI:NI, and has no mixing compounds. The set of (σ, ν) pairs is

$$S^{(p)} = \{(\text{AL:AL}, 0), (\text{AL:NI}, 0), (\text{NI:AL}, 0), (\text{NI:NI}, 0)\}. \quad (2.32)$$

For $(\text{AL:AL}, 0)$ the set of (s, e) pairs is $T_{\text{AL:AL}, 0}^{(p)} = \{(1, \text{Al}), (2, \text{Al})\}$, for $(\text{AL:NI}, 0)$ the (s, e) pairs are $(1, \text{Al})$ and $(2, \text{Ni})$, and so on for the other constituent arrays. Since there is no mixing, all the sets $X_{\sigma, \nu}^{(p)}$ are empty. The empty product is taken to be 1, and there are four terms in the outer sum, so

$$\begin{aligned} G_{\star}^p = & {}^0G_{\text{AL:AL}}^p y_{1, \text{Al}}^p y_{2, \text{Al}}^p + {}^0G_{\text{AL:NI}}^p y_{1, \text{Al}}^p y_{2, \text{Ni}}^p \\ & + {}^0G_{\text{NI:AL}}^p y_{1, \text{Ni}}^p y_{2, \text{Al}}^p + {}^0G_{\text{NI:NI}}^p y_{1, \text{Ni}}^p y_{2, \text{Ni}}^p \\ & - R T \left(a_1^{(p)} y_{1, \text{Al}}^p \ln y_{1, \text{Al}}^p + a_1^{(p)} y_{1, \text{Ni}}^p \ln y_{1, \text{Ni}}^p \right. \\ & \left. + a_2^{(p)} y_{2, \text{Al}}^p \ln y_{2, \text{Al}}^p + a_2^{(p)} y_{2, \text{Ni}}^p \ln y_{2, \text{Ni}}^p \right). \quad (2.33) \end{aligned}$$

Example 3. A simple phase with with mixing; $p := \text{liquid}$ [1]

In the Co-Mo system, the liquid phase is modeled with first order mixing and comprises the three constituent arrays CO, MO, and CO,MO. The set of (σ, ν) pairs is $S^{(p)} = \{(\text{CO}, 0), (\text{MO}, 0), (\text{CO,MO}, 0), (\text{CO,MO}, 1)\}$. The two sets of (s, e) pairs are identical: $T_{\text{Co,Mo}, 0}^{(p)} = T_{\text{Co,Mo}, 1}^{(p)} = \{(1, \text{Co}), (1, \text{Mo})\}$. The two sets of (s_m, e_0, e_1) tuples are again identical: $X_{\text{Co,Mo}, 0}^{(p)} =$

$X_{\text{Co,Mo},1}^{(p)} = \{(1, \text{Co,Mo})\}$. *There are four terms in the outer sum, so*

$$\begin{aligned}
G_{\star}^p &= {}^0G_{\text{Co}}^p y_{1,\text{Co}}^p + {}^0G_{\text{Mo}}^p y_{1,\text{Mo}}^p \\
&\quad + {}^0G_{\text{Co,Mo}}^p y_{1,\text{Co}}^p y_{1,\text{Mo}}^p + {}^1G_{\text{Co,Mo}}^p y_{1,\text{Co}}^p y_{1,\text{Mo}}^p (y_{1,\text{Co}}^p - y_{1,\text{Mo}}^p) \\
&\quad - R T \left(a_1^{(p)} y_{1,\text{Co}}^p \ln y_{1,\text{Co}}^p + a_1^{(p)} y_{1,\text{Mo}}^p \ln y_{1,\text{Mo}}^p \right). \quad (2.34)
\end{aligned}$$

Disordered Contribution

The general formula for $G_{\star}^p(y^{(p)})$ defined in (2.23) is used for modeling order-disorder transition:

$$G^{(p)} = G_{\text{dis}}^{(p)}(x^{(p)}) + G_{\text{ord}}^{(p)}(y^{(p)}) - G_{\text{ord}}^{(p)}(y^{(p)} = x^{(p)}), \quad (2.35)$$

see (5.144) and (5.145) from [96].

First, $G_{\text{ord}}^{(p)}(y^{(p)}) = G_{\star}^p(y^{(p)})$, so the above expression for $G_{\star}^p(y^{(p)})$ can be used “as is” for the middle term in (2.35).

Second, replacing $y_{s,e}^p$ with $x_e^{(p)}$ throughout $G_{\star}^p(y^{(p)})$ creates $G_{\text{ord}}^{(p)}(y^{(p)} = x^{(p)})$ in (2.35).

Finally, if ordered phase p has a disordered contribution from phase \tilde{p} , then replacing the index sets from those that correspond to p with those corresponding to \tilde{p} (as elaborated below) and $y_{s,e}^p$ with $x_e^{(p)}$ throughout $G_{\star}^{\tilde{p}}(y^{(p)})$ creates $G_{\text{dis}}^{(p)}(x^{(p)})$ in (2.35).

Thus a “disordered contribution” is defined, analogous but complimentary to the “ordered contribution” in [96],

$$\Delta G^p := G_{\text{dis}}^{(p)}(x^{(p)}) - G_{\text{ord}}^{(p)}(y^{(p)} = x^{(p)}). \quad (2.36)$$

Using this to update (2.35),

$$G^{(p)} = G_{\star}^p + \Delta G^p \quad (2.37)$$

More rigorously, consider the following expression using q as a dummy variable:

$$G_{(q)}^{(p)} = \sum_{(\sigma, \nu) \in S^{(q)}} \left\{ {}^{\nu}G_{\sigma}^p \left[\prod_{(s, e) \in T_{\sigma, \nu}^{(q)}} x_e^{(p)} \right] \sum_{(s, m, e_0, e_1) \in X_{\sigma, \nu}^{(q)}} (x_{e_0}^{(p)} - x_{e_1}^{(p)})^{\nu} \right\} - R T \sum_{(s, e) \in T^{(q)}} \left(a_s^{(q)} x_e^{(p)} \ln x_e^{(p)} \right). \quad (2.38)$$

The formula for $G_{(q)}^{(p)}$ given in (2.38) is a modified version of the formula for $G_{\star}^p(y^{(p)})$, where p is the phase under consideration, q is the phase providing index sets and coefficients, and $y_{s, e}^p$ is replaced with $x_e^{(p)}$. Observe the location of q in $S^{(q)}$, $T^{(q)}$, $T_{\sigma, \nu}^{(q)}$, and $X_{\sigma, \nu}^{(q)}$, in contrast with p in the coefficients ${}^{\nu}G_{\sigma}^p$, and the replacement of $y^{(p)}$ variables with $x^{(p)}$.

The first and the third terms in (2.35) using $G_{(q)}^{(p)}$ can be written,

$$G_{\text{dis}}^{(p)} = G_{(\tilde{p})}^{(p)}, \quad (2.39)$$

$$G_{\text{ord}}^{(p)}(y^{(p)} = x^{(p)}) = G_{(p)}^{(p)}. \quad (2.40)$$

To specify the phases with disordered contributions, a set D is introduced which indicates all disordered contributions in the system by pairs (p, \tilde{p}) . Using set notation, for a phase p this may be written as

$$D^{(p)} = \{(q, \tilde{q}) \in D \mid q = p\}. \quad (2.41)$$

In all systems and for all phases p the set $D^{(p)}$ will contain zero or one pair (p, \tilde{p}) . This generic set based formulation allows summation over zero or one pairs to act as an “if” condition,

$$\Delta G^p := G_{\text{dis}}^{(p)} - G_{\text{ord}}^{(p)}(y^{(p)} = x^{(p)}) = \sum_{(p, \tilde{p}) \in D^{(p)}} G_{(\tilde{p})}^{(p)} - G_{(p)}^{(p)}. \quad (2.42)$$

A phase p receiving no disordered contribution does not appear in the left hand side of any tuple in D , and in this case $D^{(p)} = \emptyset$, and the sum over the empty set is zero. Hence the disordered contribution received by such a phase is automatically zero: $\Delta G^p = 0$, giving

$$G^{(p)} = G_{\text{ord}}^{(p)}(y^{(p)}) = G_{\star}^p(y^{(p)}).$$

Example 4. Al-Pt B2 with a disordered contribution from bcc. [1]

In the Al-Pt system the two-sublattice ordered B2 phase is present and receives a disordered contribution from bcc with first order mixing.

The energy equation is

$$\begin{aligned} G^{\text{B2}} &= G_{\star}^{\text{B2}}(y^{(\text{B2})}) + G_{\text{dis}}^{\text{B2}}(x^{(\text{B2})}) - G_{\text{ord}}^{\text{B2}}(y = x^{(\text{B2})}) \\ &= G_{\star}^{\text{B2}}(y^{(\text{B2})}) + G_{(\text{bcc})}^{\text{B2}}(x^{(\text{B2})}) - G_{(\text{B2})}^{\text{B2}}(y = x^{(\text{B2})}). \end{aligned}$$

For a disordered phase such as bcc, because there is only one sublattice and no disordered contribution, without considering a magnetic contribution, thus

$$G^{\text{bcc}} = G_{\star}^{\text{bcc}} = G_{\text{dis}}^{\text{bcc}}(x^{(\text{bcc})}) = G_{\text{dis}}^{\text{bcc}}(y^{(\text{bcc})}) = G_{(\text{bcc})}^{\text{bcc}}(y^{(\text{bcc})}).$$

Magnetic Contribution and complete formulation

The model for magnetic contribution used here is given in full generality in [21], and applied, e.g., in [97] for Co-Mo using specific calculated values,

$$G_{\text{mag}}^{(p)} = R T \ln(\beta^{(p)} + 1) g(\tau). \quad (2.43)$$

See [21] for detail on β , $g(\cdot)$, and τ .

Including the disordered and magnetic contribution as defined above, the complete formula is

$$G^{(p)} = G_{\star}^{(p)} + \Delta G^p + G_{\text{mag}}^{(p)}. \quad (2.44)$$

where $G_{\star}^{(p)}$ is defined in (2.23), and ΔG^p in (2.42).

Using the modeled Gibbs energy for each phase $G^{(p)}$, the total Gibbs energy function to be minimized is

$$G = \sum_{p \in P} G^{(p)} f^{(p)}, \quad (2.45)$$

where $f^{(p)}$ indicates the mole fraction of the phase in the overall composition.

Miscibility gap handling

Where the energy curve of a stable phase is concave down, it is called a miscibility gap, see Figure 2.1. This means the lowest stable energy for that phase is a linear combination of two (or more for multicomponent systems) points at distinct composition values. This formulation automatically includes multiple instances for each phase, one for each constituent in the phase including vacancies. This is the maximum possible present at equilibrium due to Gibbs' phase rule. In this manner they will span any possible miscibility gap. If no miscibility gap is present then the two instances will coincide, or one or more of them will have an f value of zero. While other approaches to managing miscibility gaps are well represented in the literature, the automatic creation of multiple instances is not found in the literature and we believe it represents a novel contribution to the state of the art.

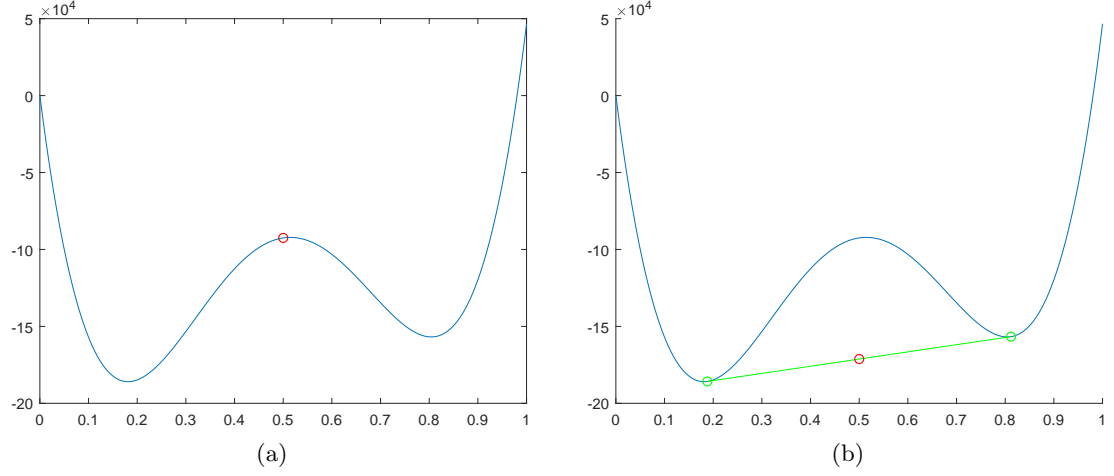


Figure 2.1: Energy curve for a notional phase with a miscibility gap (a). By including two instances for the phase, the miscibility gap is automatically spanned by a linear combination of two compositions of that phase (b), as occurs in physical materials at thermodynamic equilibrium.

2.3 Two Models

Model 1 Variables and Formulation

ϕ is a single phase in the system.

e is a physical constituent of the system (“element”).

s is a sublattice in a phase.

$y_{e,s}^\phi \in [0, 1]$ is the variable constituent fraction of e on sublattice s of phase ϕ .

$G^\phi(y) \in \mathbb{R}$, the energy of phase ϕ with composition y , is detailed further below.

$f^\phi \geq 0$ is the variable fraction of the system that is phase ϕ .

$F_e^0 \geq 0$ is the specified composition of the system.

$$\begin{aligned}
& \min_{f,y} \sum_{\phi} f^{\phi} G^{\phi}(y) \\
& \sum_{\phi} f^{\phi} \xi_e^{\phi} = F_e^0, \quad \forall \text{ constituent } e \\
& \sum_e y_{e,s}^{\phi} = 1, \quad \forall \text{ phase-sublattice pair } (\phi, s) \\
& 0 \leq f^{\phi} \quad \forall \text{ phase } \phi \\
& 0 \leq y_{e,s}^{\phi} \quad \forall \text{ triplet } (\phi, e, s)
\end{aligned} \tag{2.46}$$

where a_s^{ϕ} is the site ratio of sublattice s in phase ϕ , and

$$\xi_e^{\phi} := \frac{\sum_s a_s^{\phi} y_{e,s}^{\phi}}{\sum_c \sum_s a_s^{\phi} y_{c,s}^{\phi}} \tag{2.47}$$

is the real fraction of phase ϕ which is constituent e .

Energy Formulation

Energy of a single phase has four components

$$G(T, P, Y) = G_{\text{entropy}}(T, P, Y) \tag{2.48}$$

$$+ G_{\text{CEF}}(T, P, Y) \tag{2.49}$$

$$+ G_{\text{magnetic}}(T, P, Y) \tag{2.50}$$

$$+ G_{\text{disorder}}(T, P, Y) \tag{2.51}$$

Configurational entropy is standard in the literature,

$$G_{\text{entropy}}(T, P, Y) = -RT \sum_{ij} y_{ij} \ln y_{ij} \tag{2.52}$$

The magnetic and disorder/ordered contributions are also standard from the literature, however are formulated using the set definitions which follow.

Using a set-based approach with the appropriate indices and index sets, the Compound Energy Formalism can be fully generalized as follows:

$$G_{\text{CEF}}(T, P, Y) = \sum_{(\sigma, \nu) \in S^\phi} \left[{}^\nu G_\sigma^\phi \left[\prod_{(s, e) \in T_{\sigma, \nu}^\phi} y_{s, e} \right] \sum_{(s_m, e_0, e_1) \in X_{\sigma, \nu}^\phi} (y_{s_m, e_0} - y_{s_m, e_1})^\nu \right]. \quad (2.53)$$

The flexibility comes from the definition of the sets, which are extracted from the configuration file: constituent arrays of ϕ , $S^\phi \equiv \{(\sigma, \nu) \mid (\sigma, \nu) \text{ in } \phi\}$, species sites of σ , $T_{\sigma, \nu}^\phi \equiv \{(s, e) \mid (s, e) \text{ in } \sigma\}$, species sites of p , $T^\phi \equiv \bigcup_{(\sigma, \nu) \in S^\phi} T_{\sigma, \nu}^\phi$, mixing sites of σ , $X_{\sigma, \nu}^\phi \equiv \{(s_m, e_0, e_1) \mid (s_m, e_0, e_1) \text{ in } \sigma\}$, and disordered contributions, $D^\phi \equiv \{(q, \tilde{q}) \in D \mid q = \phi\}$.

Model 2 Formulation and Variables

Model 2 treats vacancies explicitly, with differences from Model 1 highlighted in red,

$$\begin{aligned} \min_{f, y} \quad & \sum_{\phi} f^{\phi} \frac{G^{\phi}(y)}{1 - \xi_{\text{Va}}^{\phi}} \\ \sum_{\phi} f^{\phi} \textcolor{red}{x}_e^{\phi} = & F_e^0, \quad \forall \text{ constituent } e \\ \sum_e y_{e, s}^{\phi} = & 1, \quad \forall \text{ phase-sublattice pair } (\phi, s) \\ 0 \leq & f^{\phi} \quad \forall \text{ phase } \phi \\ 0 \leq & y_{e, s}^{\phi} \quad \forall \text{ triplet } (\phi, e, s) \end{aligned} \quad (2.54)$$

where a_s^{ϕ} is the site ratio of sublattice s in phase ϕ , and

$$\textcolor{red}{x}_e^{\phi} := \frac{\sum_s a_s^{\phi} y_{e, s}^{\phi}}{\sum_{c \neq \text{Va}} \sum_s a_s^{\phi} y_{c, s}^{\phi}} \quad (2.55)$$

is the real fraction of phase ϕ which is constituent e . The portions of Model 2 which are different from Model 1 have been highlighted in red.

Models 1 and 2 are provably equivalent at equilibrium but alter the composition space away from equilibrium. This change affects the performance of nonlinear solvers, discussed further below.

2.4 Computational Framework

In this framework, the model is completely separated from the data, and any number of systems can share the same model file. For each system being investigated, the converter generates the necessary data files. The model comprises 100 statements across fewer than 200 (non-empty) lines of AMPL code.

The model contains definitions of the data structures, the objective function, and the constraints.

The converter parses a Thermocalc [101] TDB database [16] into the necessary AMPL data format. Figure 2.2 depicts the information flow, and the components of the process. Java was selected as the language for development of the converter in 2013 to enable maximum portability and maintainability by diverse researchers in the field. Since that time Python has eclipsed Java in its widespread use, but the converter has not needed to be rewritten.

The data file is produced by the converter from information in the thermodynamic database (TDB) being examined. It contains data on the 1) elements/species; 2) phases; 3) sublattices and their order of mixing; 4) symmetries; 5) disordered contributions; 6) site fractions; 7) magnetic coefficients; 8) multiplicity of phases (to cover miscibility gaps); and 9) parameters for each named formula in the TDB. The data file is read once by AMPL, after reading the model file.

The parameter file contains temperature dependent information, and fixes parameter values

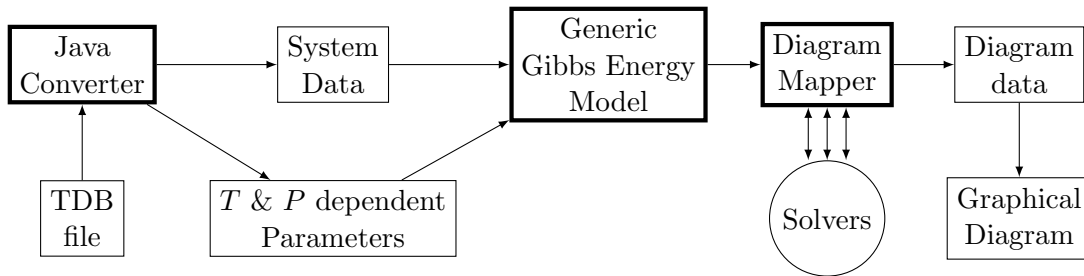


Figure 2.2: Information flow through the framework.

with `let` statements. It must be read once for each temperature row being examined. The possibility of making these parameter values computed functions which change when the parameter T changes exists, but it is not possible with the current implementation of AMPL. The proof-of-concept mapper for a binary phase diagram follows the algorithm shown in 2.3.

Although AMPL is a modeling language, it has enough procedural features to allow development of a “script” that iterates through a specified temperature range, and at each temperature across the composition from 0 to 1. At each T, c point the script solves the problem several times with random initial conditions (the number of trials is configurable) and selects the lowest obtained value at that point. The set of all the obtained lowest values is stored in a csv file for later depiction by a separate plotting mechanism.

An improved algorithm was developed for collecting data points for a binary phase diagram, which begins at a low temperature (300K), in the middle of the composition, and traces the monophasic lines upward in T and tracks topological changes and un-tested regions as it proceeds. It maintains the set of all T, c areas yet to be explored and works until the necessary diagram data has been accumulated. This has the advantage of being able to make direct use of nearby optimal solutions as initial conditions for this current test point, and enabled a 10x or greater improvement in speed.

```

1: function MAPPER()
2:   load model as  $P(x, T)$ 
3:   load system data
4:    $T \leftarrow T_{\min}$ 
5:    $x \leftarrow 0$ 
6:   while  $T \leq T_{\max}$  do
7:     load system parameters
8:     while  $x \leq 1.0$  do
9:       calculate  $P(x, T)$ 
10:      if  $\text{card}(P(x, T)) = 2$  then
11:        mark  $x_{\text{upper}}$  and  $x_{\text{lower}}$  on diagram
12:         $x \leftarrow x_{\text{upper}} + \Delta_x$ 
13:      else
14:         $x \leftarrow x + \Delta_x$ 
15:      end if
16:    end while
17:     $T \leftarrow T + \Delta_T$ 
18:  end while
19: end function

```

Figure 2.3: Mapping Algorithm [1]. $P(x, T)$: the set of phases present at composition x and temperature T . $\text{card}(P)$ indicates the cardinality of P (always 1 or 2 in this idealized context). When there are two phases present, one has composition x_{lower} and the other x_{upper} , where $x_{\text{lower}} \leq x_{\text{upper}}$.

The tracer algorithm was never fully debugged and has remained in a state of near-completion for some time. The complexities of tracking the un-searched area of the T, c space requires careful examination, and other improvements are possible.

An algorithm for mapping the points in a Gibbs triangle at fixed T and P was developed in an AMPL script. It takes the trivial approach of testing every composition point in the triangle with a fixed number of random initial conditions, and recording the lowest result at each point. Many improvements are possible.

A MATLAB script to plot the binary diagram information was created and used for our 2015 paper [1]. This has since been superseded by a python script.

Our paper on the AMPL framework was accepted in late 2014 and published in 2015 [1], detailing the set-based concept and its advantages. It included the analysis of several systems,

including the detection of errors in a prior publication on the Al-Pt system.

The first approach to mapping phase energy in a binary system at fixed T and P is to test each composition point with many random initial conditions and a single fixed phase present. This is inefficient and time consuming. The resulting information is stored in a csv for later plotting.

The phase energy plotter was developed in python. (Earlier, ad-hoc plotting of phase energies was performed in MATLAB.)

A second approach to mapping phase energy was developed based on the “scatter” algorithm in Gibbs and later in pycalphad: for each phase, a large number of feasible Y values each have their energy computed, which requires no optimization step, and then the lower hull if the c, G space is calculated. A later refinement is to track the lower hull within AMPL as the iteration over random values proceeds. This gives a very good approximation to the energy curve, adequate for most needs, and is very fast when compared to the earlier implementation.

2.5 Existence of the Lagrange Multipliers

Define

$$\mathbf{v}^\phi = (f^\phi, y_{j_1, e_1}^\phi, y_{j_1, e_2}^\phi, \dots, y_{j_m, e_n}^\phi). \quad (2.56)$$

which includes the f^ϕ scalar and all the $y_{j,e}^\phi$ scalars for phase ϕ . And

$$\mathbf{v} = (\mathbf{v}^{\phi_1}, \mathbf{v}^{\phi_2}, \dots, \mathbf{v}^{\phi_n}). \quad (2.57)$$

The constraints are,

$$\mathbf{f}^\top (AYI_0)^\top - \mathbf{n}^\top = 0, \quad (2.58)$$

$$Y\mathbf{1} - \mathbf{1} = 0, \quad (2.59)$$

$$\mathbf{f} \geq 0. \quad (2.60)$$

The vector in the Jacobian for the composition constraint (2.58) for constituent e with respect to \mathbf{v} is,

$$\left(\xi_e^{\phi_1}, \frac{a_{j_1}^{\phi_1}}{\sum_i a_i^{\phi_1}} f^{\phi_1}, 0, \dots, 0, \frac{a_{j_2}^{\phi_1}}{\sum_i a_i^{\phi_1}} f^{\phi_1}, 0, \dots \right) \quad (2.61)$$

where the first entry is in the position corresponding to f^{ϕ_1} and the other nonzero entries for phase ϕ_1 are in positions corresponding to the subscript e in $y_{j_1,e}$, matching that this is the composition constraint for element e . Every position in \mathbf{v} corresponding to an f has a nonzero entry, as does every position corresponding to a y value for constituent e . Every constituent e has such a row in the Jacobian.

The vector in the Jacobian for the sublattice unity constraint (2.59) for sublattice j is,

$$(0, 1, 1, 1, 0, 0, \dots) \quad (2.62)$$

where there is a 1 in each y location for sublattice j of phase ϕ , and there is one constraint for each sublattice in each phase. The notional example above with three ones would correspond to a sublattice where three constituents could be present.

For a phase ϕ where $f^\phi = 0$ the non-negativity constraint (2.60) is active and the row is present in the Jacobian with a single nonzero entry of 1 corresponding to the position of f^ϕ in \mathbf{v} ,

$$(1, 0, 0, \dots) \quad (2.63)$$

To conserve space on the page, define an explicit fraction for each site ratio by, e.g.,

$$\underline{a}_{j2}^{\phi_1} = \frac{a_{j2}^{\phi_1}}{\sum_i a_i^{\phi_1}}. \quad (2.64)$$

This gives us a Jacobian matrix, $J =$

$$\begin{pmatrix} \xi_{e_1}^{\phi_1} & \underline{a}_{j_1}^{\phi_1} f^{\phi_1} & 0 & 0 \cdots & \underline{a}_{j_2}^{\phi_1} f^{\phi_1} & 0 & 0 \cdots & \xi_{e_1}^{\phi_2} & \underline{a}_{j_1}^{\phi_2} f^{\phi_2} & 0 & 0 \cdots & \underline{a}_{j_2}^{\phi_2} f^{\phi_2} & 0 & \cdots \\ \xi_{e_2}^{\phi_1} & 0 & \underline{a}_{j_1}^{\phi_1} f^{\phi_1} & 0 \cdots & 0 & \underline{a}_{j_2}^{\phi_1} f^{\phi_1} & 0 \cdots & \xi_{e_2}^{\phi_2} & 0 & \underline{a}_{j_1}^{\phi_2} f^{\phi_2} & 0 \cdots & 0 & \underline{a}_{j_2}^{\phi_2} f^{\phi_2} & \cdots \\ \vdots & \vdots & \vdots & \vdots \ddots & \vdots & \vdots & \vdots \ddots & \vdots & \vdots & \vdots & \vdots \ddots & \vdots & \vdots & \ddots \\ 0 & 1 & 1 & 1 \cdots & 0 & 0 & 0 \cdots & 0 & 0 & 0 & 0 \cdots & 0 & 0 & \cdots \\ 0 & 0 & 0 & 0 \cdots & 1 & 1 & 1 \cdots & 0 & 0 & 0 & 0 \cdots & 0 & 0 & \cdots \\ 0 & 0 & 0 & 0 \cdots & 0 & 0 & 0 \cdots & 0 & 1 & 1 & 1 \cdots & 0 & 0 & \cdots \\ 0 & 0 & 0 & 0 \cdots & 0 & 0 & 0 \cdots & 0 & 0 & 0 & 0 \cdots & 1 & 1 & \cdots \\ \vdots & \vdots & \vdots & \vdots \ddots & \vdots & \vdots & \vdots \ddots & \vdots & \vdots & \vdots & \vdots \ddots & \vdots & \vdots & \ddots \end{pmatrix} \quad (2.65)$$

For a phase where $f^\phi = 0$, meaning the inequality constraint $f^\phi \geq 0$ is active, has a row with a single nonzero entry in the row corresponding to f^ϕ .

The matrix J is full rank, and can be reduced by elementary row operations to

$$J' = \begin{pmatrix} \textcircled{1} & 0 & k_{11} & k_{12} & \cdots & 0 & k_{13} & k_{14} & \cdots & 0 & 0 & k_{15} & k_{16} & \cdots & 0 & k_{17} & \cdots \\ 0 & 0 & k_{21} & k_{22} & \cdots & 0 & k_{23} & k_{24} & \cdots & \textcircled{1} & 0 & k_{25} & k_{26} & \cdots & 0 & k_{27} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots \\ 0 & \textcircled{1} & 1 & 1 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & \cdots \\ 0 & 0 & 0 & 0 & \cdots & \textcircled{1} & 1 & 1 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & \cdots \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & \textcircled{1} & 1 & 1 & \cdots & 0 & 0 & \cdots \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & \textcircled{1} & 1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots \end{pmatrix} \quad (2.66)$$

The circled entries indicate the dependent variables for determining the null space; there is exactly one per row.

Since the Jacobian is full rank, the constraints are linearly independent, and the LICQ condition is satisfied.

The proof for the Molar formulation is analogous with x in place of ξ .

2.6 Existence of Feasible Point

At every temperature T and pressure P to be considered, a well-constructed thermodynamic database must have at least one phase ϕ_0 with y -values capable of attaining $x_e^{\phi_0} = 0$ for each e , and at least one phase ϕ_1 with y -values capable of attaining $x_e^{\phi_1} = 1$ for each e . They may be the same phase, or distinct phases. If they are the same phase, then consider two instances of that phase as distinct phases in what follows.

If two such phases do not exist, then there will exist some composition where no phase can be present, a physical impossibility. (Special-purpose databases covering only a portion

of the composition space are a theoretical possibility, but we are not aware of any in the literature, and in those cases the lower or upper bound should be changed from 0 or 1 respectively to a value suited to the special-purpose database.)

Lemma 2.1. Feasible Point in Binary System *Given n_e at a temperature T and pressure P suited to the thermodynamic database, ϕ_0 with feasible y -values such that $x_e^{\phi_0} \leq n_e$, and ϕ_1 with feasible y -values such that $n_e \leq x_e^{\phi_1}$, then there exist $0 \leq f^{\phi_0}$ and $0 \leq f^{\phi_1}$ such that*

$$f^{\phi_0} x_e^{\phi_0} + f^{\phi_1} x_e^{\phi_1} = n_e. \quad (2.67)$$

Proof. Let

$$a = n_e - x_e^{\phi_0}, \quad (2.68)$$

$$b = x_e^{\phi_1} - n_e, \quad (2.69)$$

$$f^{\phi_0} = \frac{b}{a+b}, \quad (2.70)$$

$$f^{\phi_1} = \frac{a}{a+b}. \quad (2.71)$$

Then,

$$f^{\phi_0} x_e^{\phi_0} + f^{\phi_1} x_e^{\phi_1} = \frac{b}{a+b} (n_e - a) + \frac{a}{a+b} (n_e + b) \quad (2.72)$$

$$= n_e. \quad (2.73)$$

□

This demonstrates the existence of a feasible point for every temperature, pressure, and

composition a binary database covers. It is extended to higher order systems in the next lemma.

There are two complementary assumptions about $\sum_e n_e$ that can be made in specifying a composition,

1. $\sum_e n_e = 1$
2. $\sum_e n_e$ is entirely free

If we allow the sum to be free, then $\sum_i f^{\phi_i}$ must also be free. This presents no theoretical or practical obstacle, and is often the case in software packages and descriptions of algorithms in the literature. As above, we continue below with no restriction on $\sum_e n_e$.

Lemma 2.2. Feasible Point in a Multi-Component System *For any specified vector \mathbf{n} of composition values n_e , where $0 \leq n_e \leq 1$, at a temperature T and pressure P suited to the thermodynamic database, there exist $0 \leq f^{\phi_i} \leq 1$ such that*

$$\sum_i f^{\phi_i} x_e^{\phi_i} = n_e, \forall e. \quad (2.74)$$

Proof. For each e there must exist at least one phase ϕ_e with y -values such that $x_e^{\phi_e} = 1$ and all other $x_d^{\phi_e} = 0$. If not, then the thermodynamic database is incompletely defined for that temperature T and pressure P .

Pick a set of phases P such that for each phase e there is exactly one phase satisfying the above condition.

Let

$$f^{\phi_e} = n_e. \quad (2.75)$$

Then

$$\sum_d f^{\phi_d} x_d^{\phi_d} = x^{\phi_e}. \quad (2.76)$$

Thus a feasible point exists. \square

2.7 Existence of a KKT Point

It is possible to define a thermodynamic database that has no stationary points, hence no equilibria. Such a database would have no physical meaning, and would not be a well-formed database. In any database relevant to materials science, each phase will have at least one minimum with respect to y -values, at least one phase ϕ_0 capable of attaining $x_e^{\phi_0} = 0$ and at least one phase capable of attaining $x_e^{\phi_1} = 1$ for each e , each well-defined and with finite energy at those points.

These assumptions alone are insufficient to prove the existence of a stationary point for any given n_e : the existence of y -values achieving a certain composition does not rule out different Y values at that composition with negative-infinite energy.

From Lemmas 2.1 and 2.2 we have the existence of a feasible point satisfying the constraints.

Since we are given a feasible point, if we assume each $g^\phi(Y)$ is continuous, then the existence of a minimum on the compact domain $f^\phi \in [0, 1]$, $y_{j,e}^\phi \in [0, 1]$ is given by the Extreme Value Theorem.

However $\frac{1}{x_{nv}^\phi} g^\phi(Y)$ is not continuous where $x_{nv} = 0$.

By excluding these points with the non-vacancy constraint, then we again have a continuous function on a compact domain, with continuous constraints, hence a minimum is guaranteed to exist. Since the constraints are linearly independent, we have satisfied the requirements

for the existence of the Lagrange multipliers and a Karush-Kunn-Tucker point.

2.8 Equivalence of Models

2.8.1 Karush-Kunn-Tucker conditions for two formulations

In an optimization problem, the Karush-Kunn-Tucker (KKT) conditions are satisfied at stationary points, and correspond to the first-order necessary conditions.

The second-order sufficient conditions are satisfied at local minima.

We define the parameters, variables, functions, and equations for the problem, then state the KKT conditions with those definitions, and prove the two models are equivalent with respect to points where the KKT conditions are satisfied.

As indices we use ϕ for phase, j for sublattice, and e for constituent (element) in the system. There are l total sublattices across all phases, m phases, and n constituents.

Variables in the system are Y , discussed immediately below, and $\mathbf{f} \in \mathbb{R}^m$ the vector of phase fractions.

Parameters in the system are $A \in \mathbb{R}^{m \times l}$ the matrix of site site ratios, $\mathbf{g}(Y) \in \mathbb{R}^m$ the vector of Gibbs energies of the phases as a function of variable Y , $\mathbf{n} \in \mathbb{R}^m$ the vector specifying the composition of the system, $\boldsymbol{\mu} \in \mathbb{R}^m$, $\boldsymbol{\lambda} \in \mathbb{R}^l$, and $\boldsymbol{\nu} \in \mathbb{R}^m$ vectors of Lagrange multipliers, and $\mathbf{1} \in \mathbb{R}^m$ a vector of all ones.

Each sublattice j of a phase ϕ has a variable $y_{j,e}^\phi$ for each constituent e which can occur there.

We can also consider fixed value $y_{j,e}^\phi \equiv 0$ exist for constituents e which cannot occur on sublattice j (for the entropy function over these fixed zeros we must also declare $0 \ln 0 = 0$).

In a computer algebra system these values are typically stored in a data structure adapted to sparse data and not as matrices or fixed n -dimensional arrays. Here we represent the

complete set of $y_{j,e}^\phi$ values as Y , which for formal manipulations can be envisioned as a 3-dimensional array, one dimension for each of ϕ , j , and e , or as a 2-dimensional matrix where j implicitly indicates the phase as well as sublattice, i.e., if across all phases there are l sublattices, each phase will only have a small contiguous subset of them and no two phases share a sublattice.

Y occurs in all following expressions in one of three ways: as a variable for a function, e.g., $\mathbf{g}(Y)$; as AY where A are the site ratios; or as $Y\mathbf{1}$ indicating the sum of all values on a single sublattice, e.g., $\sum_e y_{j,e}^\phi$. Each parameter in A is a_j^ϕ and $(AY)_e^\phi = \xi_e^\phi = \sum_j a_j^\phi y_{j,e}^\phi$ for each ϕ and e . Similarly in the molar formulation $x_e^\phi = \frac{1}{r^\phi(Y)} \sum_j a_j^\phi y_{j,e}^\phi$.

Given that the y values can be stored or arranged in a variety of ways, for the purposes of formal manipulations below it is always true that $(AY) \in \mathbb{R}^{m \times n}$ is the matrix of ξ_e^ϕ values and $x_e^\phi = \frac{1}{r^\phi(Y)} \xi_e^\phi$.

In the molar formulation the non-vacancy fraction of phase ϕ is,

$$r^\phi(Y) = \frac{\sum_{e \neq \text{Va}} \sum_j a_j^\phi y_{j,e}^\phi}{\sum_e \sum_j a_j^\phi y_{j,e}^\phi}, \quad (2.77)$$

and to conserve space we define

$$R(Y) \equiv \begin{pmatrix} \frac{1}{r^{\phi_0}(Y)} & 0 & \cdots & 0 \\ 0 & \frac{1}{r^{\phi_1}(Y)} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{r^{\phi_m}(Y)} \end{pmatrix}. \quad (2.78)$$

Because the summation that defines $r^\phi(Y)$ includes all non-vacancy y values for a phase in the numerator, the only case where $r^\phi(Y) = 0$ is when phase ϕ is entirely vacant. As

discussed in the vacancy portion of this dissertation, we bound the variables away from that unphysical condition. Hence each $r^\phi(Y)$ is strictly greater than zero, and it follows that as a diagonal matrix with positive entries, $R(Y)$ is always invertible.

AY is the matrix of ξ values for each phase and constituent, arranged so that a column of AY corresponds to a phase and a row of AY corresponds to a constituent. Similarly the matrix of x values in the molar formulation is $X = AYR(Y)$. For this proof we do not need to know how R is calculated, only that it is a function of Y . For the Molar formulation we use $\tilde{\mathbf{f}}$ in place of \mathbf{f} to indicate the phase fraction, and $\tilde{\boldsymbol{\nu}}$ in place of $\boldsymbol{\nu}$ as a Lagrange multiplier.

We choose to construct AY so that vacancy is the last column, and for the composition constraint we define a matrix that removes that last column since there is no specified value of vacancy as a constituent in a system,

$$I_0 = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \\ 0 & 0 & \cdots & 0 \end{pmatrix}.$$

Hillert formulation

The Hillert formulation written as an optimization problem:

$$\min_{\mathbf{f}, Y} \mathbf{f}^\top \mathbf{g}(Y), \quad (2.79)$$

$$\text{s.t. } \mathbf{f}^\top (AYI_0)^\top = \mathbf{n}^\top, \quad (2.80)$$

$$\text{s.t. } Y\mathbf{1} = \mathbf{1}, \quad (2.81)$$

$$\text{s.t. } y_{e,j}^\phi \in [0, 1], \quad \forall \phi, e, j, \quad (2.82)$$

$$\text{s.t. } f^\phi \geq 0, \quad \forall \phi. \quad (2.83)$$

Due to the entropy term $-RTy \ln y$ in $g(Y)$, and the requirement from (2.81) that $\sum_e y_{e,j}^\phi = 1$, if any $y_{e,j}^\phi = 0$ or $y_{e,j}^\phi = 1$, then the derivative $g^\phi(Y)$ is infinite, hence at equilibrium $0 < y_{e,j}^\phi < 1$ for all ϕ , e , and j , and the construction of the Lagrangian can be limited to constraints (2.80) and (2.81), and (2.83), ($\boldsymbol{\mu}$ is used here as the Lagrange multiplier for (2.80) to coincide with its widespread use in the literature as chemical potential),

$$\mathcal{L}_H = \mathbf{f}^\top \mathbf{g}(Y) - \left(\mathbf{f}^\top (AYI_0)^\top - \mathbf{n}^\top \right) \boldsymbol{\mu} - (Y\mathbf{1} - \mathbf{1}) \boldsymbol{\lambda} - \mathbf{f}^\top \boldsymbol{\nu}. \quad (2.84)$$

First-Order Necessary Conditions

The first-order necessary conditions for an equilibrium, the Karush-Kunn-Tucker (KKT) conditions, are that (a) the partials of \mathcal{L} with respect to the primary variables Y and \mathbf{f} are equal to zero,

$$\nabla_Y \mathcal{L}_H = \mathbf{f}^\top \nabla_Y \mathbf{g}(Y) - \mathbf{f}^\top \nabla_Y (AYI_0)^\top \boldsymbol{\mu} - \nabla_Y (Y\mathbf{1}) \boldsymbol{\lambda} = 0, \quad (2.85)$$

$$\nabla_{\mathbf{f}} \mathcal{L}_H = \mathbf{g}(Y) - (AYI_0)^\top \boldsymbol{\mu} - \boldsymbol{\nu} = 0, \quad (2.86)$$

that (b) the constraints are satisfied, that (c) the Lagrange multiplier $\boldsymbol{\nu}$ on the inequality constraint $f^\phi \geq 0$ is non-negative, i.e., $\nu^\phi \geq 0$ for all ϕ , and that (d) complementarity exists for every constraint, i.e.,

$$\left(\mathbf{f}^\top (AYI_0)^\top - \mathbf{n}^\top \right) \boldsymbol{\mu} = 0, \quad (2.87)$$

$$(Y\mathbf{1} - \mathbf{1}) \boldsymbol{\lambda} = 0, \quad (2.88)$$

$$\mathbf{f}^\top \boldsymbol{\nu} = 0. \quad (2.89)$$

Molar formulation

The molar formulation of the Calphad problem written as an optimization problem, using $\tilde{\mathbf{f}}$ and $\tilde{\boldsymbol{\nu}}$ in place of \mathbf{f} and $\boldsymbol{\nu}$:

$$\min_{\tilde{\mathbf{f}}, Y} \tilde{\mathbf{f}}^\top R(Y) \mathbf{g}(Y), \quad (2.90)$$

$$\text{s.t. } \tilde{\mathbf{f}}^\top (AYI_0 R(Y))^\top = \mathbf{n}^\top, \quad (2.91)$$

$$\text{s.t. } Y\mathbf{1} = \mathbf{1}, \quad (2.92)$$

$$\text{s.t. } y_{e,j}^\phi \in [0, 1], \quad \forall \phi, e, j, \quad (2.93)$$

$$\text{s.t. } \tilde{f}^\phi \geq 0, \quad \forall \phi. \quad (2.94)$$

As in the Hillert formulation the construction of the Lagrangian of the molar formulation is limited to constraints (2.91) and (2.92), and (2.94),

$$\mathcal{L}_M = \tilde{\mathbf{f}}^\top R(Y) \mathbf{g}(Y) - \left(\tilde{\mathbf{f}}^\top (AYR(Y))^\top - \mathbf{n}^\top \right) \boldsymbol{\mu} - (Y\mathbf{1} - \mathbf{1}) \boldsymbol{\lambda} - \tilde{\mathbf{f}}^\top \tilde{\boldsymbol{\nu}}. \quad (2.95)$$

The partials with respect to Y and $\tilde{\mathbf{f}}$ are

$$\begin{aligned}\nabla_Y \mathcal{L}_M &= \tilde{\mathbf{f}}^\top R(Y) \nabla_Y \mathbf{g}(Y) + \tilde{\mathbf{f}}^\top \nabla_Y R(Y) \mathbf{g}(Y) \\ &\quad - \tilde{\mathbf{f}}^\top R(Y) \nabla_Y (AY I_0)^\top \boldsymbol{\mu} - \tilde{\mathbf{f}}^\top \nabla_Y R(Y) (AY I_0)^\top \boldsymbol{\mu} - \nabla_Y (Y \mathbf{1}) \boldsymbol{\lambda} = 0, \quad (2.96)\end{aligned}$$

$$\nabla_{\tilde{\mathbf{f}}} \mathcal{L}_M = R(Y) \mathbf{g}(Y) - R(Y) (AY I_0)^\top \boldsymbol{\mu} - \tilde{\boldsymbol{\nu}} = 0. \quad (2.97)$$

One-to-one correspondence of KKT points

A fundamental relationship between the two formulations is $R(Y) \tilde{\mathbf{f}} = \mathbf{f}$ and $\tilde{\boldsymbol{\nu}} = R(Y) \boldsymbol{\nu}$, and we frequently use that substitution below.

The first lemma addresses the complementarity conditions for the composition constraint, (2.80) in the Hillert formulation and (2.91) in the Molar formulation.

Lemma 2.3. $\mathbf{f}^\top \boldsymbol{\nu} = 0$ if and only if $\tilde{\mathbf{f}}^\top \tilde{\boldsymbol{\nu}} = 0$.

Proof. Since $R(Y)$ is a diagonal matrix with positive entries its inverse $R^{-1}(Y)$ exists and $R(Y)^\top = R(Y)$, so $\tilde{\boldsymbol{\nu}} = R(Y) \boldsymbol{\nu}$ is equivalent to $R^{-1}(Y) \tilde{\boldsymbol{\nu}} = \boldsymbol{\nu}$. Substituting $R(Y) \tilde{\mathbf{f}} = \mathbf{f}$ and $R^{-1}(Y) \tilde{\boldsymbol{\nu}} = \boldsymbol{\nu}$,

$$\mathbf{f}^\top \boldsymbol{\nu} = (R(Y) \tilde{\mathbf{f}})^\top R^{-1}(Y) \tilde{\boldsymbol{\nu}} = \tilde{\mathbf{f}}^\top R(Y) R^{-1}(Y) \tilde{\boldsymbol{\nu}} = \tilde{\mathbf{f}}^\top \tilde{\boldsymbol{\nu}}. \quad (2.98)$$

□

Lemma 2.4. (2.86) is true if and only if (2.97) is true.

Proof. Substitute $\tilde{\boldsymbol{\nu}} = R(Y) \boldsymbol{\nu}$ into (2.97). Since $R(Y)$ is a diagonal matrix with positive entries, (2.86) is equivalent to (2.97). □

Lemma 2.5. Every KKT point in the Hillert formulation is a KKT point in the Molar

formulation

Proof. Assume Y, \mathbf{f} is a KKT point of the Hillert formulation, hence (2.85) and (2.86) hold. From (2.85) we have $\mathbf{f}^\top \nabla_Y \mathbf{g}(Y) - \mathbf{f}^\top \nabla_Y (AY I_0)^\top \boldsymbol{\mu} - \nabla_Y (Y \mathbf{1}) \boldsymbol{\lambda} = 0$. Substituting $\tilde{\mathbf{f}} R(Y)$ for \mathbf{f} this coincides with three of the terms in (2.96), leaving this to prove:

$$\tilde{\mathbf{f}}^\top \nabla_Y R(Y) \mathbf{g}(Y) - \tilde{\mathbf{f}}^\top \nabla_Y R(Y) (AY I_0)^\top \boldsymbol{\mu} = 0. \quad (2.99)$$

By assumption, (2.86): $\mathbf{g}(Y) - (AY I_0)^\top \boldsymbol{\mu} - \boldsymbol{\nu} = 0$. By left-multiplying (2.86) with $\tilde{\mathbf{f}}^\top \nabla_Y R(Y)$ we have

$$\tilde{\mathbf{f}}^\top \nabla_Y R(Y) \mathbf{g}(Y) - \tilde{\mathbf{f}}^\top \nabla_Y R(Y) (AY I_0)^\top \boldsymbol{\mu} - \tilde{\mathbf{f}}^\top \nabla_Y R(Y) \boldsymbol{\nu} = 0. \quad (2.100)$$

The first two terms coincide with (2.99). We show the last term is zero, by substituting $\mathbf{f}^\top R^{-1}(Y) = \tilde{\mathbf{f}}^\top$,

$$\mathbf{f}^\top R^{-1}(Y) \nabla_Y R(Y) \boldsymbol{\nu}. \quad (2.101)$$

We can avoid the technical details of multidimensional array multiplication by examining the derivative term by term:

$$\frac{\partial}{\partial y_{e,j}^\phi} R(Y) \quad (2.102)$$

is an $n \times n$ matrix with only a single nonzero entry, in a diagonal position. Substituting this into (2.101), and knowing $R^{-1}(Y)$ is a diagonal matrix,

$$\mathbf{f}^\top \left(R^{-1}(Y) \frac{\partial}{\partial y_{e,j}^\phi} R(Y) \right) \boldsymbol{\nu}, \quad (2.103)$$

the middle factor inside parentheses is a matrix with a single nonzero entry which is on the diagonal. We have assumed the point Y, \mathbf{f} satisfies the KKT conditions for the Hillert formulation, hence \mathbf{f} and $\boldsymbol{\nu}$ are complementary: $f^\phi \nu^\phi = 0$ for all ϕ . Let $B = R^{-1}(Y) \frac{\partial}{\partial y_{e,j}^\phi} R(Y)$

and k, k be the position of its nonzero entry. Then

$$\mathbf{f}^\top B \boldsymbol{\nu} = f^k b_{k,k} \nu^k = 0, \quad (2.104)$$

hence (2.101) is an $l \times m$ matrix of all zeros, thus (2.99) is zero, and so (2.96) is zero. We conclude a point satisfying the KKT conditions in the Hillert formulation always satisfies them in the Molar formulation with the relationships $R(Y) \tilde{\mathbf{f}} = \mathbf{f}$ and $\tilde{\boldsymbol{\nu}} = R(Y) \boldsymbol{\nu}$. \square

Lemma 2.6. *Every KKT point in the Molar formulation is a KKT point in the Hillert formulation*

Proof. Assume $Y, \tilde{\mathbf{f}}$ is a KKT point of the Hillert formulation, hence (2.96) and (2.97) hold and we seek to show this implies (2.85), $\mathbf{f}^\top \nabla_Y \mathbf{g}(Y) - \mathbf{f}^\top \nabla_Y (AY I_0)^\top \boldsymbol{\mu} - \nabla_Y (Y \mathbf{1}) \boldsymbol{\lambda} = 0$, and (2.86), $\mathbf{g}(Y) - (AY I_0)^\top \boldsymbol{\mu} - \boldsymbol{\nu} = 0$. From Lemma 2.4 we have that (2.97) \Leftrightarrow (2.86). Left-multiply (2.86) by $\tilde{\mathbf{f}}^\top \nabla_Y R(Y)$ and substitute $R^{-1}(Y) \tilde{\boldsymbol{\nu}} = \boldsymbol{\nu}$ to find

$$\tilde{\mathbf{f}}^\top \nabla_Y R(Y) \mathbf{g}(Y) - \tilde{\mathbf{f}}^\top \nabla_Y R(Y) (AY I_0)^\top \boldsymbol{\mu} - \tilde{\mathbf{f}}^\top \nabla_Y R(Y) R^{-1}(Y) \tilde{\boldsymbol{\nu}} = 0. \quad (2.105)$$

As in Lemma 2.5, by complementarity of $\tilde{\mathbf{f}}$ and $\tilde{\boldsymbol{\nu}}$ we see the last term is zero, and by subtracting the remaining terms from (2.96) we have

$$\tilde{\mathbf{f}}^\top R(Y) \nabla_Y \mathbf{g}(Y) - \tilde{\mathbf{f}}^\top R(Y) \nabla_Y (AY I_0)^\top \boldsymbol{\mu} - \nabla_Y (Y \mathbf{1}) \boldsymbol{\lambda} = 0. \quad (2.106)$$

Substitute $\mathbf{f}^\top = \tilde{\mathbf{f}}^\top R(Y)$ and we have shown (2.85). \square

Theorem 2.7. *A point Y, \mathbf{f} in the Hillert formulation is a KKT point if and only if the corresponding point $Y, \tilde{\mathbf{f}}$ in the Molar formulation is a KKT point.*

Proof. This follows directly from Lemmas 2.5 and 2.6. \square

Theorem 2.8. *The objective value of Gibbs free energy at a KKT point Y, \mathbf{f} in the Hillert formulation is equal to the objective value at the corresponding KKT point $Y, \tilde{\mathbf{f}}$ in the Molar formulation.*

Proof. The objective in the Hillert formulation is

$$\mathbf{f}^\top \mathbf{g}(Y) \tag{2.107}$$

and the objective in the Molar formulation is

$$\tilde{\mathbf{f}}^\top R(Y) \mathbf{g}(Y). \tag{2.108}$$

Since $R(Y)\tilde{\mathbf{f}} = \mathbf{f}$, by substitution the result follows. \square

2.8.2 Second-Order Conditions at KKT points

If the objective was bounded below, equivalence of KKT points would be sufficient to conclude the global equilibrium is identical in both formulations. However, we have shown in this dissertation that the objective is not guaranteed to be bounded below. Thermodynamic databases exist, such as Al-Pt, where a phase with vacancy requires an artificial constraint to prevent the objective from running to arbitrarily large negative values.

Due to the complexity of representing the higher dimension objects which result from multiple derivatives with respect to variables which are two or three dimensional arrays, we look at partials with respect to individual scalar variables $y_{j,e}^\phi$ and f^ϕ .

In this section to reduce the amount of notation necessary we will use

$$\xi_e^\phi = \sum_j a_j^\phi y_{j,e}^\phi \tag{2.109}$$

and

$$\boldsymbol{\xi}^{\phi^\top} = \left(\sum_j a_j^\phi y_{j,e_1}^\phi, \quad \dots, \quad \sum_j a_j^\phi y_{j,e_{n-1}}^\phi \right), \quad (2.110)$$

which excludes Vacancy if it exists as a constituent of the system. Note that

$$\left(\boldsymbol{\xi}^{\phi_1}, \quad \dots, \quad \boldsymbol{\xi}^{\phi_m} \right) = AYI_0, \quad (2.111)$$

a term which appeared frequently in the preceding sections. So $\boldsymbol{\xi}^\phi$ corresponds one column of AYI_0 . (The alternative notation $(AYI_0)^\phi$ is possible in place of $\boldsymbol{\xi}^\phi$.)

Scalar Second Derivatives of the Hillert Formulation

To help speed comprehension and for ease of reference in proofs below, groups of terms in the following equations are labeled and color coded.

As above, the Lagrangian for the Hillert formulation is

$$\mathcal{L}_H = \mathbf{f}^\top \mathbf{g}(Y) - \left(\mathbf{f}^\top (AYI_0)^\top - \mathbf{n}^\top \right) \boldsymbol{\mu} - (Y\mathbf{1} - \mathbf{1}) \boldsymbol{\lambda} - \mathbf{f}^\top \boldsymbol{\nu}, \quad (2.84)$$

and the partials with respect to scalars are

$$\frac{\partial}{\partial f^\phi} \mathcal{L}_H = \underbrace{g^\phi(Y) - \boldsymbol{\xi}^{\phi^\top} \boldsymbol{\mu} - \nu^\phi}_{(A_H)}, \quad (2.112)$$

and

$$\frac{\partial}{\partial y_{j,e}^\phi} \mathcal{L}_H = f^\phi \underbrace{\frac{\partial}{\partial y_{j,e}^\phi} g^\phi(Y) - f^\phi a_j^\phi \mu_e - \lambda_j}_{(B_H)}. \quad (2.113)$$

The three second derivatives with respect to scalars are

$$\frac{\partial^2}{\partial f^{\phi^2}} \mathcal{L}_H = 0, \quad (2.114)$$

$$\frac{\partial^2}{\partial y_{j,e}^{\phi_1} \partial f^{\phi_2}} \mathcal{L}_H = \begin{cases} 0 & \phi_1 \neq \phi_2, \\ \underbrace{\frac{\partial}{\partial y_{j,e}^{\phi}} g^{\phi}(Y) - a_j^{\phi} \mu_e}_{(C_H)} & \phi_1 = \phi_2, \end{cases} \quad (2.115)$$

and

$$\frac{\partial^2}{\partial y_{j,1,e_1}^{\phi_1} \partial y_{j,2,e_2}^{\phi_2}} \mathcal{L}_H = \begin{cases} 0 & \phi_1 \neq \phi_2, \\ f^{\phi_1} \frac{\partial^2}{\partial y_{j,1,e_1}^{\phi_1} \partial y_{j,2,e_2}^{\phi_2}} g^{\phi_1}(Y) & \phi_1 = \phi_2. \end{cases} \quad (2.116)$$

Knowing that these terms are 0 when ϕ_1 and ϕ_2 don't agree, we improve legibility by writing, e.g.,

$$\frac{\partial^2}{\partial y_1^{\phi} \partial y_2^{\phi}} \mathcal{L}_H = f^{\phi} \underbrace{\frac{\partial^2}{\partial y_1^{\phi} \partial y_2^{\phi}} g^{\phi}(Y)}_{(O_M)}. \quad (2.117)$$

Scalar Second Derivatives of the Molar Formulation

The Lagrangian of the Molar formulation from above is

$$\mathcal{L}_M = \tilde{\mathbf{f}}^{\top} R(Y) \mathbf{g}(Y) - \left(\tilde{\mathbf{f}}^{\top} (A Y R(Y))^{\top} - \mathbf{n}^{\top} \right) \boldsymbol{\mu} - (Y \mathbf{1} - \mathbf{1}) \boldsymbol{\lambda} - \tilde{\mathbf{f}}^{\top} \tilde{\boldsymbol{\nu}}. \quad (2.95)$$

The partials with respect to scalars are

$$\frac{\partial}{\partial \tilde{f}^\phi} \mathcal{L}_M = \underbrace{\frac{1}{r^\phi(Y)} g^\phi(Y) - \frac{1}{r^\phi(Y)} \boldsymbol{\xi}^{\phi^\top} \boldsymbol{\mu}}_{(A_M)} - \tilde{\nu}^\phi, \quad (2.118)$$

and

$$\begin{aligned} \frac{\partial}{\partial y_1^\phi} \mathcal{L}_M = & \underbrace{\tilde{f}^\phi \frac{1}{r^\phi(Y)} \frac{\partial}{\partial y_1^\phi} g^\phi(Y) - \tilde{f}^\phi \frac{1}{r^\phi(Y)} a_j^\phi \mu_e - \lambda_j}_{(B1_M)} \\ & + \underbrace{\tilde{f}^\phi \frac{\partial}{\partial y_1^\phi} \frac{1}{r^\phi(Y)} g^\phi(Y) - \tilde{f}^\phi \frac{\partial}{\partial y_1^\phi} \frac{1}{r^\phi(Y)} \boldsymbol{\xi}^{\phi^\top} \boldsymbol{\mu}}_{(A1_M)}. \end{aligned} \quad (2.119)$$

The second derivatives have many more terms than in the Hillert formulation, and as above if $\phi_1 \neq \phi_2$ they are zero, so we assume the second derivative is with respect to two variables with the same ϕ , and we replace the j, e subscripts $y_{j1, e1}^\phi$ with y_1^ϕ and the corresponding $a_1^\phi = a_{j1}^\phi$ and $\mu_1 = \mu_{e1}$,

$$\frac{\partial^2}{\partial \tilde{f}^{\phi_1} \partial \tilde{f}^{\phi_2}} \mathcal{L}_M = 0, \quad (2.120)$$

$$\begin{aligned}
\frac{\partial^2}{\partial \tilde{f}^\phi \partial y_1^\phi} \mathcal{L}_M = & \underbrace{\frac{\partial}{\partial y_1^\phi} \frac{1}{r^\phi(Y)} g^\phi(Y) - \frac{\partial}{\partial y_1^\phi} \frac{1}{r^\phi(Y)} \boldsymbol{\xi}^{\phi \top} \boldsymbol{\mu}}_{(A2_M)} \\
& + \underbrace{\frac{1}{r^\phi(Y)} \frac{\partial}{\partial y_1^\phi} g^\phi(Y) - \frac{1}{r^\phi(Y)} a_j^\phi \mu_e}_{(C2_M)}. \quad (2.121)
\end{aligned}$$

Writing out the expression in the order it might be found by differentiating the terms in (2.119) one at a time,

$$\begin{aligned}
\frac{\partial^2}{\partial y_1^\phi \partial y_2^\phi} \mathcal{L}_M = & \tilde{f}^\phi \frac{\partial}{\partial y_2^\phi} \frac{1}{r^\phi(Y)} \frac{\partial}{\partial y_1^\phi} g^\phi(Y) + \tilde{f}^\phi \frac{1}{r^\phi(Y)} \frac{\partial^2}{\partial y_1^\phi \partial y_2^\phi} g^\phi(Y) \\
& - \tilde{f}^\phi \frac{\partial}{\partial y_2^\phi} \frac{1}{r^\phi(Y)} a_1^\phi \mu_1 \\
& + \tilde{f}^\phi \frac{\partial^2}{\partial y_1^\phi \partial y_2^\phi} \frac{1}{r^\phi(Y)} g^\phi(Y) + \tilde{f}^\phi \frac{\partial}{\partial y_1^\phi} \frac{1}{r^\phi(Y)} \frac{\partial}{\partial y_2^\phi} g^\phi(Y) \\
& - \tilde{f}^\phi \frac{\partial^2}{\partial y_1^\phi \partial y_2^\phi} \frac{1}{r^\phi(Y)} \boldsymbol{\xi}^{\phi \top} \boldsymbol{\mu} - \tilde{f}^\phi \frac{\partial}{\partial y_1^\phi} \frac{1}{r^\phi(Y)} a_2^\phi \mu_2, \quad (2.122)
\end{aligned}$$

and then grouping for use in proofs,

$$\begin{aligned}
\frac{\partial^2}{\partial y_1^\phi \partial y_2^\phi} \mathcal{L}_M = & \underbrace{\tilde{f}^\phi \frac{1}{r^\phi(Y)} \frac{\partial^2}{\partial y_1^\phi \partial y_2^\phi} g^\phi(Y)}_{(O_M)} \\
& + \underbrace{\tilde{f}^\phi \frac{\partial^2}{\partial y_1^\phi \partial y_2^\phi} \frac{1}{r^\phi(Y)} g^\phi(Y) - \tilde{f}^\phi \frac{\partial^2}{\partial y_1^\phi \partial y_2^\phi} \frac{1}{r^\phi(Y)} \boldsymbol{\xi}^{\phi \top} \boldsymbol{\mu}}_{(A3_M)} \\
& + \underbrace{\tilde{f}^\phi \frac{\partial}{\partial y_1^\phi} \frac{1}{r^\phi(Y)} \frac{\partial}{\partial y_2^\phi} g^\phi(Y) - \tilde{f}^\phi \frac{\partial}{\partial y_1^\phi} \frac{1}{r^\phi(Y)} a_2^\phi \mu_2}_{(C3_M)} \\
& + \underbrace{\tilde{f}^\phi \frac{\partial}{\partial y_2^\phi} \frac{1}{r^\phi(Y)} \frac{\partial}{\partial y_1^\phi} g^\phi(Y) - \tilde{f}^\phi \frac{\partial}{\partial y_2^\phi} \frac{1}{r^\phi(Y)} a_1^\phi \mu_1}_{(C4_M)}. \quad (2.123)
\end{aligned}$$

The second-order necessary condition is that in the Hillert formulation Y, \mathbf{f} is a KKT point and at that point $Z^\top \nabla^2 \mathcal{L} Z$ be positive semi-definite. In the Molar formulation the second-order necessary condition is that $Y, \tilde{\mathbf{f}}$ is a KKT point and at that point $\tilde{Z}^\top \nabla^2 \tilde{\mathcal{L}} \tilde{Z}$ be positive semi-definite. Similarly, the second-order sufficient condition is that the reduced Hessians be positive definite rather than semi-definite.

Proposition 2.9. *A KKT point in the Hillert formulation which meets the second-order necessary conditions also meets the second-order necessary conditions in the Molar formulation.*

We know from Theorem 2.7 that if Y, \mathbf{f} satisfies the KKT conditions in the Hillert formulation then $Y, \tilde{\mathbf{f}}$ satisfies them in the Molar formulation, where $\mathbf{f} = R(Y) \tilde{\mathbf{f}}$ and the Lagrange multipliers have the relationship $\tilde{\boldsymbol{\nu}} = R(Y) \boldsymbol{\nu}$. Hence we have that the first derivatives in both

formulations (2.112), (2.113), (2.118), and (2.119) are zero, and that the complementarity conditions are met.

First observe that (O_H) is equal to (O_M) .

From (2.112) we can conclude

$$\nu^\phi = g^\phi(Y) - \xi^{\phi^\top} \mu, \quad (2.124)$$

and from (2.113) that

$$\lambda_j = f^\phi \frac{\partial}{\partial y_{j,e}^\phi} g^\phi(Y) - f^\phi a_j^\phi \mu_e. \quad (2.125)$$

Under the assumption Y, \mathbf{f} is a KKT point satisfying the second-order sufficiency conditions, (2.115) is non-negative and then from (2.125) we see that $\lambda_j \geq 0$ for all j . From (2.124) and the complementarity of ν^ϕ and f^ϕ we have

$$f^\phi g^\phi(Y) - f^\phi \xi^{\phi^\top} \mu = 0, \quad (2.126)$$

which gives that $(A3_M)$ is zero, and with (2.125) allows us to rewrite (2.123) as

$$\frac{\partial^2}{\partial y_1^\phi \partial y_2^\phi} \mathcal{L}_M = f^\phi \underbrace{\frac{\partial^2}{\partial y_1^\phi \partial y_2^\phi} g^\phi(Y)}_{(O_H)} + \frac{\partial}{\partial y_1^\phi} \frac{1}{r^\phi(Y)} \lambda_2 + \frac{\partial}{\partial y_2^\phi} \frac{1}{r^\phi(Y)} \lambda_1. \quad (2.127)$$

We have assumed that (O_H) is greater than or equal to zero at the point in question.

Derivative of non-vacancy reciprocal

Examining the derivative of $\frac{1}{r^\phi(Y)}$, using dummy variable d for constituent and i for sub-lattice, when $j \neq \text{Va}$,

$$\frac{\partial}{\partial y_{j,e}^\phi} \frac{1}{r^\phi(Y)} = \frac{\partial}{\partial y_{j,e}^\phi} \frac{\sum_d \sum_i a_i^\phi y_{i,d}^\phi}{\sum_{d \neq \text{Va}} \sum_i a_i^\phi y_{i,d}^\phi} \quad (2.128)$$

$$= \frac{a_j}{\sum_{d \neq \text{Va}} \sum_i a_i^\phi y_{i,d}^\phi} - \frac{a_j \sum_d \sum_i a_i^\phi y_{i,d}^\phi}{\left(\sum_{d \neq \text{Va}} \sum_i a_i^\phi y_{i,d}^\phi \right)^2} \quad (2.129)$$

$$= - \frac{a_j \sum_i a_i^\phi y_{i,\text{Va}}^\phi}{\left(\sum_{d \neq \text{Va}} \sum_i a_i^\phi y_{i,d}^\phi \right)^2} \quad (2.130)$$

$$\leq 0. \quad (2.131)$$

For the derivative with respect to a vacancy term $y_{j,\text{Va}}^\phi$,

$$\frac{\partial}{\partial y_{j,\text{Va}}^\phi} \frac{1}{r^\phi(Y)} = \frac{\partial}{\partial y_{j,\text{Va}}^\phi} \frac{\sum_d \sum_i a_i^\phi y_{i,d}^\phi}{\sum_{d \neq \text{Va}} \sum_i a_i^\phi y_{i,d}^\phi} \quad (2.132)$$

$$= \frac{a_j^\phi}{\sum_{d \neq \text{Va}} \sum_i a_i^\phi y_{i,d}^\phi} \quad (2.133)$$

$$\geq 0. \quad (2.134)$$

It is apparent that, for y values dominated by vacancies, (2.130) can obtain negative values of arbitrarily large magnitude, and (2.133) can obtain positive values of arbitrarily large magnitude.

If the value of λ_j is fixed relative to Y , then there does not appear to be a guarantee the second-order necessary conditions of the Molar formulation are met at a point where they are met in the Hillert formulation.

However, λ_j are not fixed relative to Y .

λ_j entered (2.127) as a substitution of (2.125) into (2.123),

$$\lambda_j = f^\phi \frac{\partial}{\partial y_{j,e}^\phi} g^\phi(Y) - f^\phi a_j^\phi \mu_e. \quad (2.125)$$

The above approach to showing that sufficiency in one formulation implies sufficiency in the other has reached challenges making it difficult or impossible.

Taking what we have learned in exploring this approach, instead of pursuing this line to prove Proposition 2.9, we take an alternative approach below.

We begin with preliminary statements about the Hessians, and later state then prove the theorem.

Hessian of the Lagrangian

Above we see that any partial second derivative of either Lagrangian is zero whenever taking the derivative with respect to two variables from distinct phases. Hence, by creating a vector with all \mathbf{f} and Y scalars grouped together by phase the Hessian has a block diagonal structure. In the remainder of this section we examine the Lagrangian and its Hessian on a phase-by-phase basis. We define

$$\mathbf{v}^\phi = (f^\phi, y_{j_1,e_1}^\phi, y_{j_1,e_2}^\phi, \dots, y_{j_m,e_n}^\phi). \quad (2.135)$$

which includes the f^ϕ scalar and all the $y_{j,e}^\phi$ scalars for phase ϕ . In the remainder of this section we drop the ϕ superscript and note an individual sublattice, constituent pair as y_1 .

We use terms developed above to write the Hessian of the two formulations in simplified notation. In order to conserve space we introduce new notation, \mathcal{L} is the Lagrangian of the Hillert formulation and $\tilde{\mathcal{L}}$ is the Lagrangian of the Molar formulation. To indicate derivatives

with respect to f^ϕ and $y_1 \equiv y_{j,e}^\phi$ we define

$$\mathcal{L}_f \equiv \frac{\partial}{\partial f^\phi} \mathcal{L}_H \qquad \tilde{\mathcal{L}}_f \equiv \frac{\partial}{\partial f^\phi} \mathcal{L}_M \qquad (2.136)$$

$$\mathcal{L}_1 \equiv \frac{\partial}{\partial y_1^\phi} \mathcal{L}_H \qquad \tilde{\mathcal{L}}_1 \equiv \frac{\partial}{\partial y_1^\phi} \mathcal{L}_M \qquad (2.137)$$

$$\mathcal{L}_{ff} \equiv \frac{\partial^2}{\partial f^\phi \partial f^\phi} \mathcal{L}_H \qquad \tilde{\mathcal{L}}_{ff} \equiv \frac{\partial^2}{\partial f^\phi \partial f^\phi} \mathcal{L}_M \qquad (2.138)$$

$$\mathcal{L}_{12} \equiv \frac{\partial^2}{\partial y_1^\phi \partial y_2^\phi} \mathcal{L}_H \qquad \tilde{\mathcal{L}}_{12} \equiv \frac{\partial^2}{\partial y_1^\phi \partial y_2^\phi} \mathcal{L}_M \qquad (2.139)$$

$$\mathcal{L}_{12} \equiv \frac{\partial^2}{\partial y_1^\phi \partial y_2^\phi} \mathcal{L}_H \qquad \tilde{\mathcal{L}}_{12} \equiv \frac{\partial^2}{\partial y_1^\phi \partial y_2^\phi} \mathcal{L}_M \qquad (2.140)$$

Using this notation, the Hessian for a single phase in the Hillert formulation can be written

$$\nabla^2 \mathcal{L} = \begin{pmatrix} \mathcal{L}_{ff} & \mathcal{L}_{f1} & \mathcal{L}_{f2} & \cdots & \mathcal{L}_{fn} \\ \mathcal{L}_{f1} & \mathcal{L}_{11} & \mathcal{L}_{12} & \cdots & \mathcal{L}_{1n} \\ \mathcal{L}_{f2} & \mathcal{L}_{12} & \mathcal{L}_{22} & \cdots & \mathcal{L}_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathcal{L}_{fn} & \mathcal{L}_{1n} & \mathcal{L}_{2n} & \cdots & \mathcal{L}_{nn} \end{pmatrix} \qquad (2.141)$$

From (2.114), (2.115), and (2.117) above we have the second derivatives of \mathcal{L}_H :

$$\mathcal{L}_{ff} = 0, \qquad (2.142)$$

$$\mathcal{L}_{f1} = \frac{\partial}{\partial y_1} g(Y) - a_1 \mu_1, \qquad (2.143)$$

$$\mathcal{L}_{12} = f \frac{\partial^2}{\partial y_1 \partial y_2} g(Y). \qquad (2.144)$$

These give the full Hessian of a single phase in the Hillert formulation,

$$\nabla^2 \mathcal{L} = \begin{pmatrix} 0 & \frac{\partial}{\partial y_1} g(Y) - a_1 \mu_1 & \frac{\partial}{\partial y_2} g(Y) - a_2 \mu_2 & \cdots & \frac{\partial}{\partial y_n} g(Y) - a_n \mu_n \\ \frac{\partial}{\partial y_1} g(Y) - a_1 \mu_1 & f \frac{\partial^2}{\partial y_1 \partial y_1} g(Y) & f \frac{\partial^2}{\partial y_1 \partial y_2} g(Y) & \cdots & f \frac{\partial^2}{\partial y_1 \partial y_n} g(Y) \\ \frac{\partial}{\partial y_2} g(Y) - a_2 \mu_2 & f \frac{\partial^2}{\partial y_1 \partial y_2} g(Y) & f \frac{\partial^2}{\partial y_2 \partial y_2} g(Y) & \cdots & f \frac{\partial^2}{\partial y_2 \partial y_n} g(Y) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial}{\partial y_n} g(Y) - a_n \mu_n & f \frac{\partial^2}{\partial y_1 \partial y_n} g(Y) & f \frac{\partial^2}{\partial y_2 \partial y_n} g(Y) & \cdots & f \frac{\partial^2}{\partial y_n \partial y_n} g(Y) \end{pmatrix} \quad (2.145)$$

Similarly the Hessian for a single phase in the Molar formulation is

$$\nabla^2 \tilde{\mathcal{L}} = \begin{pmatrix} \tilde{\mathcal{L}}_{ff} & \tilde{\mathcal{L}}_{f1} & \tilde{\mathcal{L}}_{f2} & \cdots & \tilde{\mathcal{L}}_{fn} \\ \tilde{\mathcal{L}}_{f1} & \tilde{\mathcal{L}}_{11} & \tilde{\mathcal{L}}_{12} & \cdots & \tilde{\mathcal{L}}_{1n} \\ \tilde{\mathcal{L}}_{f2} & \tilde{\mathcal{L}}_{12} & \tilde{\mathcal{L}}_{22} & \cdots & \tilde{\mathcal{L}}_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \tilde{\mathcal{L}}_{fn} & \tilde{\mathcal{L}}_{1n} & \tilde{\mathcal{L}}_{2n} & \cdots & \tilde{\mathcal{L}}_{nn} \end{pmatrix} \quad (2.146)$$

From above we have the second derivatives of \mathcal{L}_M :

(2.120) gives

$$\tilde{\mathcal{L}}_{ff} = 0. \quad (2.147)$$

Recalling (2.121),

$$\begin{aligned}
\frac{\partial^2}{\partial \tilde{f}^\phi \partial y_1^\phi} \mathcal{L}_M = & \underbrace{\frac{\partial}{\partial y_1^\phi} \frac{1}{r^\phi(Y)} g^\phi(Y) - \frac{\partial}{\partial y_1^\phi} \frac{1}{r^\phi(Y)} \boldsymbol{\xi}^{\phi \top} \boldsymbol{\mu}}_{(A2_M)} \\
& + \underbrace{\frac{1}{r^\phi(Y)} \frac{\partial}{\partial y_1^\phi} g^\phi(Y) - \frac{1}{r^\phi(Y)} a_j^\phi \mu_e}_{(C2_M)}, \quad (2.121)
\end{aligned}$$

we have

$$\tilde{\mathcal{L}}_{f1} = \frac{\partial}{\partial y_1} \frac{1}{r(Y)} (g(Y) - \boldsymbol{\xi}^\top \boldsymbol{\mu}) + \frac{1}{r(Y)} \left(\frac{\partial}{\partial y_1} g(Y) - a_1 \mu_1 \right) \quad (2.148)$$

$$= \frac{\partial}{\partial y_1} \frac{1}{r(Y)} \nu + \frac{1}{r(Y)} \mathcal{L}_{f1}, \quad (2.149)$$

Recall that the complementarity of f and ν requires $f\nu = 0$.

From above (2.125) allowed us to rewrite (2.123) as

$$\frac{\partial^2}{\partial y_1^\phi \partial y_2^\phi} \mathcal{L}_M = f^\phi \frac{\partial^2}{\partial y_1^\phi \partial y_2^\phi} g^\phi(Y) + \frac{\partial}{\partial y_1^\phi} \frac{1}{r^\phi(Y)} \lambda_2 + \frac{\partial}{\partial y_2^\phi} \frac{1}{r^\phi(Y)} \lambda_1. \quad (2.127)$$

which in the present notation is

$$\tilde{\mathcal{L}}_{12} = f \frac{\partial^2}{\partial y_1 \partial y_2} g(Y) + \frac{\partial}{\partial y_1} \frac{1}{r(Y)} \lambda_2 + \frac{\partial}{\partial y_2} \frac{1}{r(Y)} \lambda_1. \quad (2.150)$$

Second-Order Necessary Conditions

In addition to the first-order necessary conditions, the KKT conditions, satisfied at Y, \mathbf{f} , the second-order necessary conditions are stated here.

Let $Y, \mathbf{f}, \boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu}$ be primary and dual variables satisfying the first-order necessary conditions.

Let $\mathbf{v} \in \mathbb{R}^h$ be a row vector containing all the elements of \mathbf{f} and Y , with the variables for each phase ϕ_i grouped together as in (2.135), i.e.,

$$\mathbf{v} = (\mathbf{v}^{\phi_1}, \mathbf{v}^{\phi_1}, \dots, \mathbf{v}^{\phi_1}) = (f^{\phi_1}, y_{j_1, e_1}^{\phi_1}, y_{j_1, e_2}^{\phi_1}, \dots, y_{j_m, e_n}^{\phi_1}, f^{\phi_2}, y_{j_1, e_1}^{\phi_2}, y_{j_1, e_2}^{\phi_2}, \dots, y_{j_m, e_n}^{\phi_2}, f^{\phi_3}, \dots). \quad (2.151)$$

Where the Hessian of the Lagrangian is written below, it is with respect to v , i.e., $\nabla^2 \mathcal{L} \equiv \nabla_{vv} \mathcal{L}$.

Z is a basis for the null space of the Jacobian of the constraints. The three constraints are the composition constraint, the sublattice unity constraint, and the phase fraction non-negativity constraint,

$$\mathbf{f}^\top (AYI_0)^\top - \mathbf{n}^\top = 0, \quad (2.152)$$

$$Y\mathbf{1} - \mathbf{1} = 0, \quad (2.153)$$

$$\mathbf{f} \geq 0. \quad (2.154)$$

The second-order necessary condition is that Y, \mathbf{f} is a KKT point and at that point $Z^\top \nabla^2 \mathcal{L} Z$ be positive semi-definite. Similarly, the second-order sufficient condition is that $Z^\top \nabla^2 \mathcal{L} Z$ be positive definite.

Theorem 2.10. *A point Y, \mathbf{f} in the Hillert formulation is a local minimum if and only if $Y, \tilde{\mathbf{f}}$ is a local minimum in the Molar formulation.*

Proof. Given that Y, \mathbf{f} is a KKT point in the Hillert formulation, from Theorem 2.7 it is a

KKT point in the Molar formulation. We will see that

$$Z^\top \nabla^2 \mathcal{L} Z = \tilde{Z}^\top \nabla^2 \tilde{\mathcal{L}} \tilde{Z}, \quad (2.155)$$

where \tilde{Z} is the null space in the Molar formulation at Y, \tilde{f} .

We approach the equivalence in two steps. First we show the difference between the Hessians is zero inside the null space Z , i.e.,

$$Z^\top (\nabla^2 \mathcal{L} - \nabla^2 \tilde{\mathcal{L}}) Z = 0. \quad (2.156)$$

Then we show that

$$(Z - \tilde{Z})^\top \nabla^2 \tilde{\mathcal{L}} (Z - \tilde{Z}) = 0, \quad (2.157)$$

so that by adding (2.156) and (2.157) the equivalence will be demonstrated.

Define a difference matrix $D = \nabla^2 \mathcal{L} - \nabla^2 \tilde{\mathcal{L}}$. The goal is to show that $Z^\top D Z = 0$. Entries in D are indexed by two entries in \mathbf{v} , one for the row and one for the column in D , which are each indexed by ϕ, j, e . Since for any $\phi_1 \neq \phi_2$ the entries in both Hessians are zero, the Hessians and D share a block-diagonal structure with each block corresponding to a single phase ϕ . Examining the entries in D for a single ϕ , using indices f and $y_{j,e}$ for that phase, from (2.115), (2.117), (2.149), and (2.150),

$$D_{f,f} = 0, \quad (2.158)$$

$$D_{f,y_{j_1 e_1}} = \frac{\partial}{\partial y_{j_1 e_1}} \frac{1}{r(Y)} \nu, \quad (2.159)$$

$$D_{y_{j_1 e_2}, y_{j_2 e_2}} = \frac{\partial}{\partial y_{e_1 j_1}} \frac{1}{r(Y)} \lambda_{j_2} + \frac{\partial}{\partial y_{e_2 j_2}} \frac{1}{r(Y)} \lambda_{j_1}. \quad (2.160)$$

To aid in legibility, define, where 1 and 2 are arbitrary, e.g.,

$$r_{12} = \frac{\partial}{\partial y_{j_1 e_2}} \frac{1}{r(Y)}, \quad (2.161)$$

so,

$$D_{f,f} = 0, \quad (2.162)$$

$$D_{f,y_{j_1 e_2}} = r_{12} \nu \quad (2.163)$$

$$D_{y_{j_1 e_2}, y_{j_3 e_4}} = r_{12} \lambda_{j_3} + r_{34} \lambda_{j_1}. \quad (2.164)$$

Now writing out Z the basis of the null space of the Jacobian explicitly. The constraints again are,

$$\mathbf{f}^\top (AYI_0)^\top - \mathbf{n}^\top = 0, \quad (2.165)$$

$$Y\mathbf{1} - \mathbf{1} = 0, \quad (2.166)$$

$$\mathbf{f} \geq 0. \quad (2.167)$$

The vector in the Jacobian for the composition constraint (2.165) for constituent e with respect to \mathbf{v} is,

$$\left(\xi_e^{\phi_1}, \frac{a_{j_1}^{\phi_1}}{\sum_i a_i^{\phi_1}} f^{\phi_1}, 0, \dots, 0, \frac{a_{j_2}^{\phi_1}}{\sum_i a_i^{\phi_1}} f^{\phi_1}, 0, \dots \right) \quad (2.168)$$

where the first entry corresponds to f^{ϕ_1} and the other nonzero entries for phase ϕ_1 correspond to the subscript e in $y_{j_1, e}$ matching that this is the composition constraint for element e . Every position in \mathbf{v} corresponding to an f has a nonzero entry, as does every position corresponding to a y value for constituent e . Every constituent e has such a row in the

Jacobian.

The vector in the Jacobian for the sublattice unity constraint (2.166) for sublattice j is,

$$(0, 1, 1, 1, 0, 0, \dots) \quad (2.169)$$

where there is a 1 in each y location for sublattice j of phase ϕ , and there is one constraint for each sublattice in each phase. The notional example above with three ones would correspond to a sublattice where three constituents could be present.

For a phase ϕ where $f^\phi = 0$ the non-negativity constraint (2.167) is active and the row is present in the Jacobian with a single nonzero entry of 1 corresponding to the position of f^ϕ in \mathbf{v} ,

$$(1, 0, 0, \dots) \quad (2.170)$$

To conserve space on the page, define an explicit fraction for each site ratio by, e.g.,

$$\underline{a}_{j2}^{\phi_1} = \frac{a_{j2}^{\phi_1}}{\sum_i a_i^{\phi_1}}. \quad (2.171)$$

For phases where $f^\phi > 0$ this gives us a Jacobian matrix as we saw above in (2.65) when

proving the independence of the constraints, $J =$

$$\begin{pmatrix} \xi_{e_1}^{\phi_1} & \underline{a}_{j_1}^{\phi_1} f^{\phi_1} & 0 & 0 & \dots & \underline{a}_{j_2}^{\phi_1} f^{\phi_1} & 0 & 0 & \dots & \xi_{e_1}^{\phi_2} & \underline{a}_{j_1}^{\phi_2} f^{\phi_2} & 0 & 0 & \dots & \underline{a}_{j_2}^{\phi_2} f^{\phi_2} & 0 & \dots \\ \xi_{e_2}^{\phi_1} & 0 & \underline{a}_{j_1}^{\phi_1} f^{\phi_1} & 0 & \dots & 0 & \underline{a}_{j_2}^{\phi_1} f^{\phi_1} & 0 & \dots & \xi_{e_2}^{\phi_2} & 0 & \underline{a}_{j_1}^{\phi_2} f^{\phi_2} & 0 & \dots & 0 & \underline{a}_{j_2}^{\phi_2} f^{\phi_2} & \dots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots \\ 0 & 1 & 1 & 1 & \dots & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 & \dots & 0 & 0 & \dots \\ 0 & 0 & 0 & 0 & \dots & 1 & 1 & 1 & \dots & 0 & 0 & 0 & 0 & \dots & 0 & 0 & \dots \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 & 1 & 1 & 1 & \dots & 0 & 0 & \dots \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 & \dots & 1 & 1 & \dots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots \end{pmatrix} \quad (2.172)$$

A phase where $f^\phi = 0$ introduces a row with a single nonzero entry in the row corresponding to f^ϕ .

Next the null space Z will be identified.

Let there be m leading rows of the composition constraint type, with a leading ξ variable.

We can assume basic conditions of regularity from the proof that the constraints are independent. This corresponds to no phase's composition fraction being a linear combination of other phases composition fractions. I.e., an irregular condition would be two stable phases occupying the same point on the phase diagram, a phase being on the line defined by two other phases in a ternary diagram, the plane defined by three others in a quaternary, etc. Having more stable phases than constituents in the system is also irregular. None of these conditions is a strictly stable equilibrium – removal of one of the phases which is a linear combination of the others will produce and equal or lower objective value.

Regularity provides that elementary row operations on J produce a matrix J' sharing the same null space, where k are arbitrary constants produced by the row operations on $\underline{a}_{j_1}^{\phi_1} f^\phi$ values, (note that the two subscripts in k_{12} correspond to a sublattice and a constituent,

respectively),

$$J' = \begin{pmatrix} \textcircled{1} & 0 & k_{11} & k_{12} & \cdots & 0 & k_{13} & k_{14} & \cdots & 0 & 0 & k_{15} & k_{16} & \cdots & 0 & k_{17} & \cdots \\ 0 & 0 & k_{21} & k_{22} & \cdots & 0 & k_{23} & k_{24} & \cdots & \textcircled{1} & 0 & k_{25} & k_{26} & \cdots & 0 & k_{27} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots \\ 0 & \textcircled{1} & 1 & 1 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & \cdots \\ 0 & 0 & 0 & 0 & \cdots & \textcircled{1} & 1 & 1 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & \cdots \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & \textcircled{1} & 1 & 1 & \cdots & 0 & 0 & \cdots \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & \textcircled{1} & 1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots \end{pmatrix} \quad (2.173)$$

The circled entries above indicate the dependent variables for the null space. Hence,

$$Z = \begin{pmatrix} k_{11} & k_{12} & \cdots & k_{13} & k_{14} & \cdots & k_{15} & k_{16} & \cdots \\ 1 & 1 & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots \\ -1 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots \\ 0 & -1 & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots \\ \vdots & \vdots & \ddots & \vdots & \vdots & \cdots & \vdots & \vdots & \ddots \\ 0 & 0 & \cdots & 1 & 1 & \cdots & 0 & 0 & \cdots \\ 0 & 0 & \cdots & -1 & 0 & \cdots & 0 & 0 & \cdots \\ 0 & 0 & \cdots & 0 & -1 & \cdots & 0 & 0 & \cdots \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots \\ k_{21} & k_{22} & \cdots & k_{23} & k_{24} & \cdots & k_{25} & k_{26} & \cdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 1 & 1 & \cdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & -1 & 0 & \cdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & -1 & \cdots \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots \end{pmatrix} \quad (2.174)$$

This is easily generalized to any number of phases, sublattices, and constituents.

Now use this matrix to compute $Z^\top D Z$ for the difference $D = \nabla^2 \mathcal{L} - \nabla^2 \tilde{\mathcal{L}}$. The difference

matrix is block diagonal,

$$D = \begin{pmatrix} D^{\phi_1} & & & \\ & D^{\phi_2} & & \\ & & \ddots & \\ & & & D^{\phi_n} \end{pmatrix}, \quad (2.175)$$

where each array's difference block, written to correspond with the above notional Jacobian J and its null space Z , and for legibility omitting the ϕ terms on variables, $D^\phi =$

$$\begin{pmatrix} 0 & r_{11}\nu & r_{12}\nu & r_{13}\nu & \cdots & r_{21}\nu & r_{22}\nu & r_{23}\nu & \cdots \\ r_{11}\nu & r_{11}\lambda_1 & r_{11}\lambda_1 & r_{11}\lambda_1 & \dots & r_{11}\lambda_2 & r_{11}\lambda_2 & r_{11}\lambda_2 & \dots \\ & +r_{11}\lambda_1 & +r_{12}\lambda_1 & +r_{13}\lambda_1 & & +r_{21}\lambda_1 & +r_{22}\lambda_1 & +r_{23}\lambda_1 & \\ r_{12}\nu & r_{12}\lambda_1 & r_{12}\lambda_1 & r_{12}\lambda_1 & \dots & r_{12}\lambda_2 & r_{12}\lambda_2 & r_{12}\lambda_2 & \dots \\ & +r_{11}\lambda_1 & +r_{12}\lambda_1 & +r_{13}\lambda_1 & & +r_{21}\lambda_1 & +r_{22}\lambda_1 & +r_{23}\lambda_1 & \\ r_{13}\nu & r_{13}\lambda_1 & r_{13}\lambda_1 & r_{13}\lambda_1 & \dots & r_{13}\lambda_2 & r_{13}\lambda_2 & r_{13}\lambda_2 & \dots \\ & +r_{11}\lambda_1 & +r_{12}\lambda_1 & +r_{13}\lambda_1 & & +r_{21}\lambda_1 & +r_{22}\lambda_1 & +r_{23}\lambda_1 & \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots \\ r_{21}\nu & r_{21}\lambda_1 & r_{21}\lambda_1 & r_{21}\lambda_1 & \dots & r_{21}\lambda_2 & r_{21}\lambda_2 & r_{21}\lambda_2 & \dots \\ & +r_{11}\lambda_2 & +r_{12}\lambda_2 & +r_{13}\lambda_2 & & +r_{21}\lambda_2 & +r_{22}\lambda_2 & +r_{23}\lambda_2 & \\ r_{22}\nu & r_{22}\lambda_1 & r_{22}\lambda_1 & r_{22}\lambda_1 & \dots & r_{22}\lambda_2 & r_{22}\lambda_2 & r_{22}\lambda_2 & \dots \\ & +r_{11}\lambda_2 & +r_{12}\lambda_2 & +r_{13}\lambda_2 & & +r_{21}\lambda_2 & +r_{22}\lambda_2 & +r_{23}\lambda_2 & \\ r_{23}\nu & r_{23}\lambda_1 & r_{23}\lambda_1 & r_{23}\lambda_1 & \dots & r_{23}\lambda_2 & r_{23}\lambda_2 & r_{23}\lambda_2 & \dots \\ & +r_{11}\lambda_2 & +r_{12}\lambda_2 & +r_{13}\lambda_2 & & +r_{21}\lambda_2 & +r_{22}\lambda_2 & +r_{23}\lambda_2 & \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}. \quad (2.176)$$

Two indices specify a column of Z using j and e corresponding to the column's first entry k_{ie} . The product $Z^\top D Z$ is made up of entries $z_{je}^\top D z_{kl}$. Due to the block diagonal structure

of D , the product Dz_{je} only contains entries from a single phase. Writing it out,

$$D^\phi z_{je} = \begin{pmatrix} 0 + r_{jj}\nu - r_{je}\nu \\ k_{je}r_{11}\nu + r_{11}\lambda_j + r_{jj}\lambda_1 - r_{11}\lambda_j - r_{je}\lambda_1 \\ k_{je}r_{12}\nu + r_{12}\lambda_j + r_{jj}\lambda_1 - r_{12}\lambda_j - r_{je}\lambda_1 \\ k_{je}r_{13}\nu + r_{13}\lambda_j + r_{jj}\lambda_1 - r_{13}\lambda_j - r_{je}\lambda_1 \\ \vdots \\ k_{je}r_{21}\nu + r_{21}\lambda_j + r_{jj}\lambda_2 - r_{21}\lambda_j - r_{je}\lambda_2 \\ k_{je}r_{22}\nu + r_{22}\lambda_j + r_{jj}\lambda_2 - r_{22}\lambda_j - r_{je}\lambda_2 \\ k_{je}r_{23}\nu + r_{23}\lambda_j + r_{jj}\lambda_2 - r_{23}\lambda_j - r_{je}\lambda_2 \\ \vdots \end{pmatrix} = \begin{pmatrix} (r_{jj} - r_{je})\nu \\ k_{je}r_{11}\nu + (r_{jj} - r_{je})\lambda_1 \\ k_{je}r_{12}\nu + (r_{jj} - r_{je})\lambda_1 \\ k_{je}r_{13}\nu + (r_{jj} - r_{je})\lambda_1 \\ \vdots \\ k_{je}r_{21}\nu + (r_{jj} - r_{je})\lambda_2 \\ k_{je}r_{22}\nu + (r_{jj} - r_{je})\lambda_2 \\ k_{je}r_{23}\nu + (r_{jj} - r_{je})\lambda_2 \\ \vdots \end{pmatrix}, \quad (2.177)$$

Hence,

$$z_{kl}^\top D^\phi z_{je} = k_{kl}(r_{jj} - r_{je})\nu + k_{je}r_{k1}\nu + (r_{jj} - r_{je})\lambda_j - k_{je}r_{kl}\nu - (r_{jj} - r_{je})\lambda_j \quad (2.178)$$

$$= (k_{kl}(r_{jj} - r_{je}) + k_{je}(r_{k1} - r_{kl}))\nu. \quad (2.179)$$

Every term in each $z_{kl}^\top D^\phi z_{je}$ is a multiple of ν^ϕ . Due to complementarity, if $f^\phi > 0$ then $\nu^\phi = 0$. Recalling (2.172), which was row-reduced to (2.173), we see that if $f^\phi = 0$ then $k_{je}^\phi = 0$ for all j and e . Hence the scalar $z_{kl}^\top D^\phi z_{je}$ is zero for all j, e, k, l , and it follows that $Z^\top DZ$ is an all-zero matrix. Thus (2.156) is true.

Next we show that (2.157) is true. Looking at (2.174), we can immediately see that for

unknown values $q_{ij} = k_{ij} - \tilde{k}_{ij}$,

$$Z - \tilde{Z} = \begin{pmatrix} q_{11} & q_{12} & \cdots & q_{13} & q_{14} & \cdots & q_{15} & q_{16} & \cdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots \\ \vdots & \vdots & \ddots & \vdots & \vdots & \cdots & \vdots & \vdots & \ddots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots \\ q_{21} & q_{22} & \cdots & q_{23} & q_{24} & \cdots & q_{25} & q_{26} & \cdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots \end{pmatrix}, \quad (2.180)$$

where the nonzero rows correspond to the position of f values in the Hessian, which is a block diagonal, and each block ϕ_i has $\nabla_{f\phi, f\phi}^2 \tilde{\mathcal{L}} = 0$ in its top-right corner, the f^{ϕ_i}, f^{ϕ_i} position of the matrix. The structure of $Z - \tilde{Z}$ will extract only those top-right corner elements with multiples of q , all other terms are zero. Hence $(Z - \tilde{Z})^\top \nabla^2 \tilde{\mathcal{L}} (Z - \tilde{Z}) = 0$.

By adding (2.156) and (2.157) we find,

$$Z^T(\nabla^2\mathcal{L} - \nabla^2\tilde{\mathcal{L}})Z + (Z - \tilde{Z})^T\nabla^2\tilde{\mathcal{L}}(Z - \tilde{Z}) = 0$$

$$Z^T\nabla^2\mathcal{L}Z - Z^T\nabla^2\tilde{\mathcal{L}}Z + Z^T\nabla^2\tilde{\mathcal{L}}Z - \tilde{Z}^T\nabla^2\tilde{\mathcal{L}}\tilde{Z} = 0$$

$$Z^T\nabla^2\mathcal{L}Z - \tilde{Z}^T\nabla^2\tilde{\mathcal{L}}\tilde{Z} = 0$$

$$Z^T\nabla^2\mathcal{L}Z - \tilde{Z}^T\nabla^2\tilde{\mathcal{L}}\tilde{Z} = 0.$$

In conclusion, $Z^T\nabla^2\mathcal{L}Z = \tilde{Z}^T\nabla^2\tilde{\mathcal{L}}\tilde{Z}$, hence the reduced Hessian in the Hillert formulation is positive definite at Y, \mathbf{f} if and only if the reduced Hessian in the Molar formulation is positive definite at $Y, \tilde{\mathbf{f}}$. Thus a point Y, \mathbf{f} in the Hillert formulation is a local minimizer if and only if the corresponding point $Y, \tilde{\mathbf{f}}$ is a local minimizer in the Molar formulation. \square

2.9 Vacancies

We have encountered diverse Thermo-Calc databases for binary systems for which points of global equilibrium include a phase which is mostly Vacancy. In the context of codes which reliably determine the global energy minimum this creates computational difficulties by driving the vacancy fraction towards unity. Including a phase which is mostly vacant is un-physical and undesirable. Here we explore the root mathematical cause of this difficulty.

To provide a simple example we can work through by hand, we present a phase with one sublattice and a single element “A” mixing with vacancy “Va”. In the CALPHAD formulation it can have its Gibbs energy modeled as follows:

$$G = RTy_A \ln y_A + RTy_{Va} \ln y_{Va} + G_A y_A + G_{Va} y_{Va} + \sum_i G_{A,Va}^i y_A y_{Va} (y_A - y_{Va})^i \quad (2.181)$$

We use the temperature-dependent coefficients G_A , G_{Va} , and $G_{A,Va}^i$ for mixing of order i . (In the literature the mixing coefficients are sometimes labeled L_j^ν for arbitrary integer index j and mixing order ν , but we use the above variation of the notation.)

Equation (2.181) is the “formula energy”, determining the energy per mole of sites. To convert this to energy per mole of atoms, we must divide by the fraction of non-vacancy sites. In this very simple example we have

$$y_{Va} = 1 - y_A, \tag{2.182}$$

hence y_A indicates the non-vacancy fraction we divide by to convert from functional to molar energy. Now we substitute (2.182) into (2.181), and divide by y_A , and find

$$G_m = \frac{1}{y_A} G \quad (2.183)$$

$$= \frac{1}{y_A} \left(RT y_A \ln y_A + RT y_{Va} \ln y_{Va} + G_A y_A + G_{Va} y_{Va} \right. \\ \left. + \sum_i G_{A,Va}^i y_A y_{Va} (y_A - y_{Va})^i \right) \quad (2.184)$$

$$= \frac{1}{y_A} \left(RT y_A \ln y_A + RT (1 - y_A) \ln(1 - y_A) + G_A y_A + G_{Va} (1 - y_A) \right. \\ \left. + \sum_i G_{A,Va}^i y_A (1 - y_A) (y_A - (1 - y_A))^i \right) \quad (2.185)$$

$$= RT \ln y_A + RT \frac{1 - y_A}{y_A} \ln(1 - y_A) + G_A + G_{Va} \frac{1 - y_A}{y_A} \\ + \sum_i G_{A,Va}^i (1 - y_A) (2y_A - 1)^i \quad (2.186)$$

$$= RT \ln y_A + RT \frac{\ln(1 - y_A)}{y_A} - RT \ln(1 - y_A) + G_A + \frac{G_{Va}}{y_A} - G_{Va} \\ + \sum_i G_{A,Va}^i (1 - y_A) (2y_A - 1)^i \quad (2.187)$$

Now consider the condition as $y_A \rightarrow 0$, i.e., the phase becomes dominated by vacancies. It may be helpful to recall that

$$\lim_{y \rightarrow 0} \frac{\ln(1 - y)}{y} = -1. \quad (2.188)$$

Now,

$$\lim_{y_A \rightarrow 0^+} G_m = \lim_{y_A \rightarrow 0^+} RT \ln y_A - RT + G_A + \lim_{y_A \rightarrow 0^+} \frac{G_{Va}}{y_A} - G_{Va} + \sum_i G_{A,Va}^i (-1)^i. \quad (2.189)$$

Four of the terms are simple constants (plus one that becomes zero), and pose no difficulties.

There are two interesting terms, $RT \ln y_A$, and $\frac{G_{V_a}}{y_A}$. When $G_{V_a} \neq 0$, it dominates the limit as $y_A \rightarrow 0$. Meaning that if $G_{V_a} > 0$ then

$$\lim_{y_A \rightarrow 0^+} \left(RT \ln y_A + \frac{G_{V_a}}{y_A} \right) = +\infty, \quad (2.190)$$

and if $G_{V_a} < 0$ then

$$\lim_{y_A \rightarrow 0^+} \left(RT \ln y_A + \frac{G_{V_a}}{y_A} \right) = -\infty. \quad (2.191)$$

If $G_{V_a} = 0$ then the limit of that term is also zero, and because $RT > 0$ and $\lim_{y \rightarrow 0} \ln y = -\infty$,

$$\lim_{y_A \rightarrow 0^+} \left(RT \ln y_A + \frac{0}{y_A} \right) = -\infty. \quad (2.192)$$

However, even in the case when $G_{V_a} > 0$ but is not large, the sum of these terms can take a significant negative value. The minimum is achieved at $y_A = \frac{G_{V_a}}{RT}$, and for small G_{V_a} you can demonstrate that the sum of the two terms takes a negative value, and as G_{V_a} approaches zero it becomes arbitrarily large.

G_{V_a} is nearly always given the value zero in thermodynamic databases.

The energy diagram in Figure 2.4 visually demonstrates the difficulty for concrete values.

Code which reliably finds the global minimum will select the lowest point on each curve for the phase at this temperature, given that selection of G_{V_a} . When $G_{V_a} = 0$ there is no lowest point: the energy goes to $-\infty$ as $y_{V_a} \rightarrow 1$.

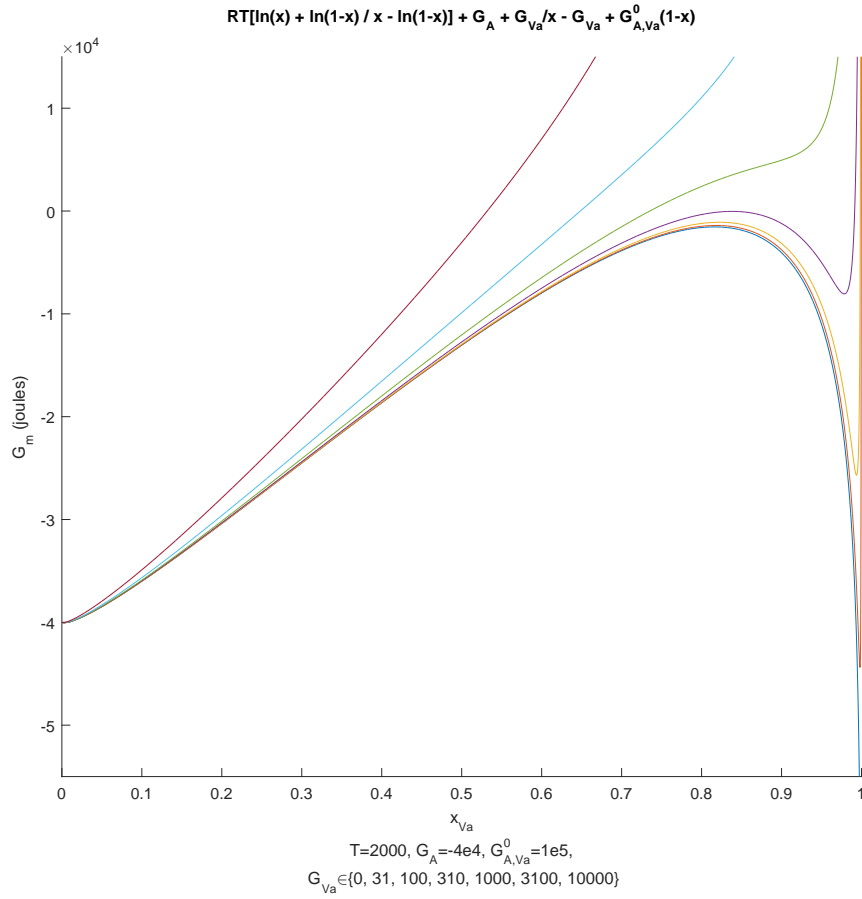


Figure 2.4: For various values of G_{Va} , the molar Gibbs energy of single-sublattice phase with a single element “A” mixing with vacancy “Va”. Constants have been set to $T = 2000$; $G_A = -4e4$; $G_{A,Va}^0 = 1e5$; for $i > 0$, $G_{A,Va}^i = 0$; and $G_{Va} \in \{0, 31, 100, 310, 1000, 3100, 10000\}$. An equilibrium point is visible near $x = 1$ (no vacancy) for any selection of values, but for G_{Va} below approximately 50 it is not the global equilibrium: a composition of near-total vacancy ($x \approx 0$) has a lower energy. This includes the generally selected value $G_{Va} = 0$.

The original implementation of the AMPL-based CALPHAD model used

$$x_e = \frac{\sum_s a_s y_{e,s}}{\sum_s a_s \sum_{e \neq Va} y_{e,s}}, \quad (2.193)$$

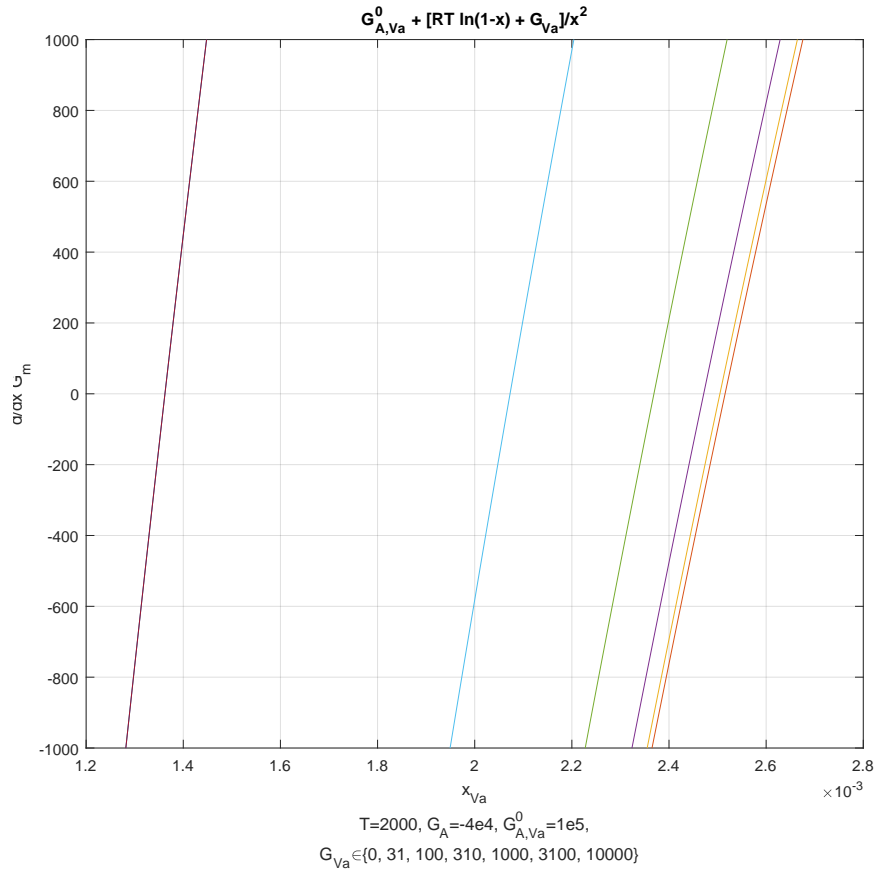


Figure 2.5: The derivative of the curves in Figure 2.4 in the equilibrium region near 2% vacancy. We can see how the vacancy fraction of the equilibrium will change as G_{Va} is adjusted.

as the variable in the composition constraint

$$\sum_{\phi} f^{\phi} x_e^{\phi} = F_e^0. \quad (2.194)$$

In place of x_e it may instead be correct to use

$$\xi_e = \frac{\sum_s a_s y_{e,s}}{\sum_s a_s \sum_e y_{e,s}} = \frac{\sum_s a_s y_{e,s}}{\sum_s a_s}, \quad (2.195)$$

where vacancies are included in the denominator. Using this alternative formulation has initially provided good results – but a thorough comparison of the two must be completed.

Equivalence of the two formulations has been proven. However, the two formulations differ in the search space for the optimization algorithm, and the efficiency and reliability of finding the correct equilibrium differs between them.

One approach to handling the vacancy problem is to place a constraint on the fraction of sites which may be vacant.

In the formulation using x in the composition constraint and dividing the energy by the non-vacancy fraction, this is accomplished by using a variable

$$r^\phi := \frac{\sum_s a_s \sum_{e \neq \text{Va}} y_{e,s}}{\sum_s a_s}, \quad (2.196)$$

and creating the constraint

$$r^\phi \geq r_{\min}, \quad (2.197)$$

for some fixed r_{\min} .

We must then decide whether to keep equilibria found with constraint (2.197) active, or discard those as invalid.

We have had good results with $r_{\min} = 0.85$, $r_{\min} = 0.5$, and $r_{\min} = 0.15$, but the phase energies and diagrams differ.

In the ξ formulation, a constraint such as $f \leq f_{\max}$ may be present in other existing codes. Such a constraint would be similar to using $r^\phi \geq r_{\min}$, but different in that it places an

element-wise minimum rather than a phase-wise minimum. The effectiveness of the two differing constraints needs to be investigated and quantified.

Since there is no consensus on what a physically reasonable maximum amount of vacancy in a single individual phase might be, we hope to identify a means to flexibly but intelligently constrain each phase in the model. This “smart constraint” may allow an individual phase to take a larger fraction of vacancy when the phase represents a smaller portion of the equilibrium, or when it completely mixes with another phase that justifies the amount of vacuum.

Pressure-Volume Equation

An interesting proposal from Pinwen Guan (a student of Zi-Kui Liu) is to counteract the vacancy problem by putting in a pressure-volume condition. The proposal comprises an additional term in G_m :

$$P \left[V_A + \frac{1 - y_A}{y_A} V_{Va} \right] \quad (2.198)$$

It is sound and makes energy go to $+\infty$ as vacancies go to unity.

Units

Pressure at STP is $P = 100 \text{ kPa}$, $V_{Cs} = 7.0732 \times 10^5 \text{ m}^3 \text{ mol}^{-1}$, and we selected an arbitrary $V_A = 1 \times 10^5 \text{ m}^3 \text{ mol}^{-1}$ which is close to the value for Aluminum.

$$1 \text{ m}^3 \text{ mol}^{-1} \times 100 \text{ kPa} = 1 \text{ joule} \cdot \text{mol}^{-1}.$$

Our Gibbs energy is being calculated in J/mol, so the units agree.

Energy Charts

A plot of the energy for these values can be seen in Figure 2.6. The positive result is that

it does prevent the energy from going to $-\infty$ as $y_{Va} \rightarrow 1$. However, its magnitude is so tiny that the global minimum remains close to the all-vacancy point. Compare with Figure 2.4, which has a $\times 10^4$ multiplier on its vertical axis.

For these test values, the global minimum is at $y_A \approx 4 \times 10^{-7}$ and the energy is $G_m \approx -1.8 \times 10^5$. That G_m value is “off the chart” in Figure 2.4. See the plot in Figure 2.7.

The nonvacancy constraint

Among the approaches to handling the vacancy problem is to place a constraint on the fraction of sites which may be vacant. In the literature it is suggested that when 10% of sites are vacant the phase is no longer stable [44] due to rapid vacancy migration and a breakdown of the crystal structure, and other similarly low terms have been proposed in conversation, such as 25%.

Projections of the energy of the B2 phase calculated from the Al-Ni database provided by Dupin and Sundman in 1999 [2] are in Figure 2.8 and Figure 2.9. Together they demonstrate that it is impossible to select a constraint for the nonvacancy fraction which would always be inactive, and that choosing a constraint below 50% would prevent the identification of some mathematically correct equilibria.

2.10 Implementation

We develop an AMPL-based framework for solving the CALPHAD problem. Within this framework we examine the research questions identified in the literature review and enumerated therein. Some questions are answered by comparison of quantitative results with prior published work, while others allow only a qualitative assessment of their success.

The Gibbs project [28] demonstrates many of the desirable features which our framework necessarily reproduces, as well as many other features valuable to the researcher which we cannot hope to have a single software developer implement alone in any reasonable time

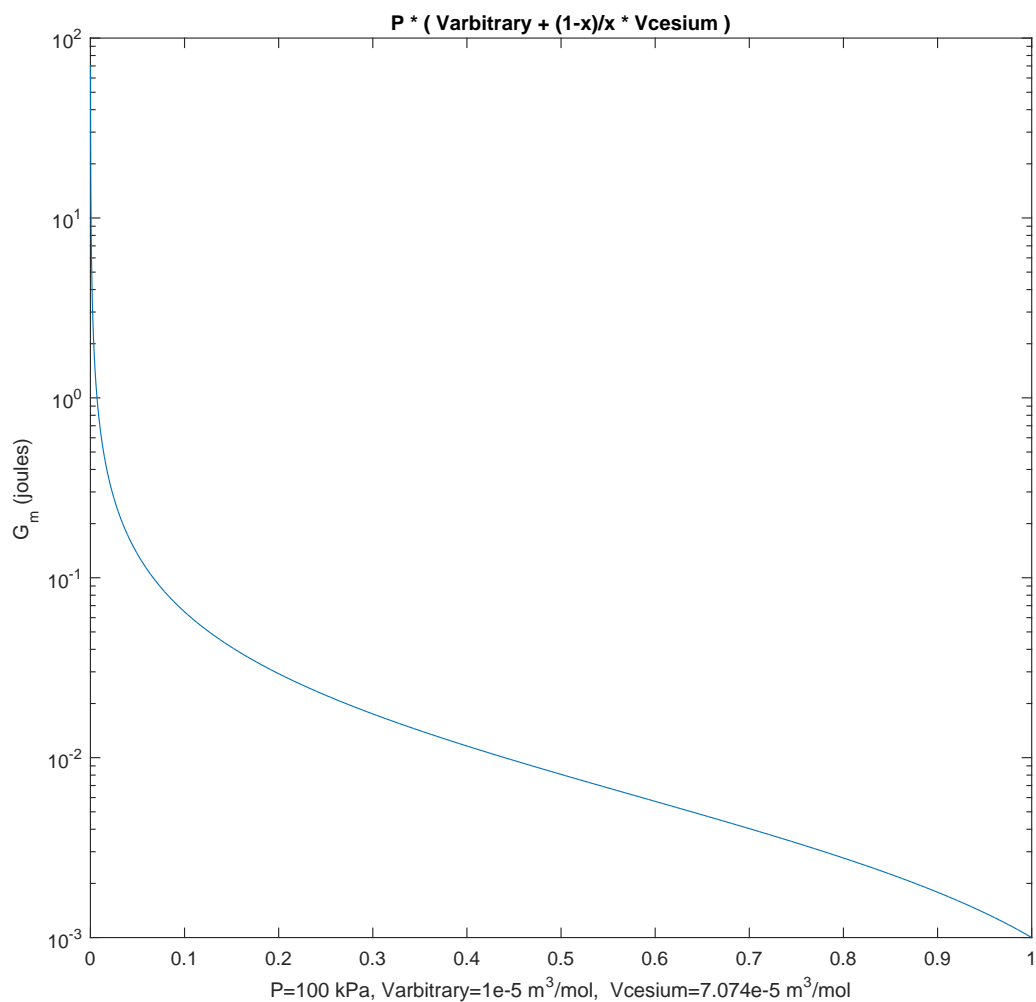


Figure 2.6: A plot of the energy corresponding to solely the pressure-volume portion from Pinwen Guan. The magnitude is 1/10000 that of our other terms: compare to the vertical axis in Figure 2.4.

frame. For example, Gibbs interfaces with FiPy for diffusion and phase field models. A goal of our project may be to gain a footing as public open-source and have multiple students and

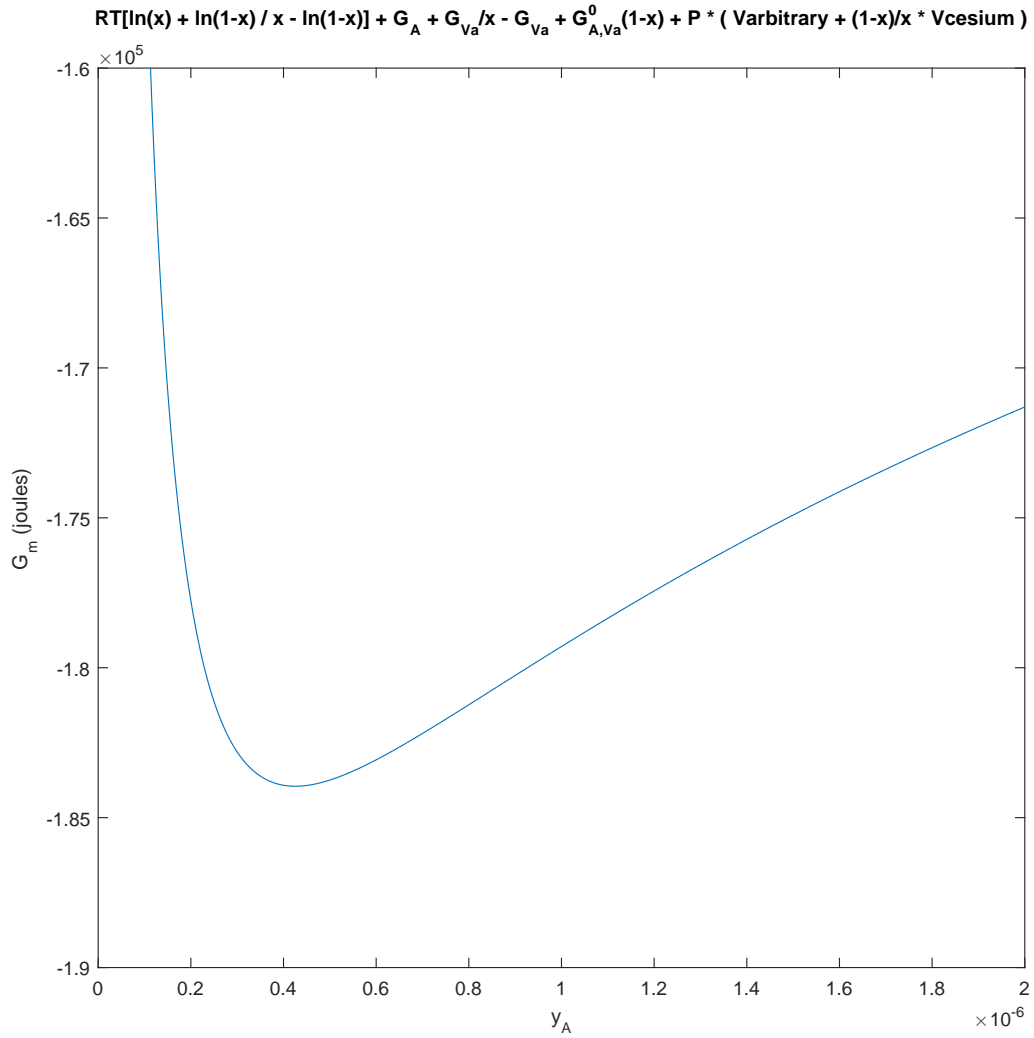


Figure 2.7: A plot of the energy of our fictional system with the pressure-volume incorporated. Pay attention to the axes: $y_A \approx 0$, and the energy at the minimizer is far below that near $y_A \approx 1 \Rightarrow G_m \approx -4 \times 10^4$ in Figure 2.4.

researchers contributing code. However, at the present moment, the repository is private and we have only a single developer contributing code.

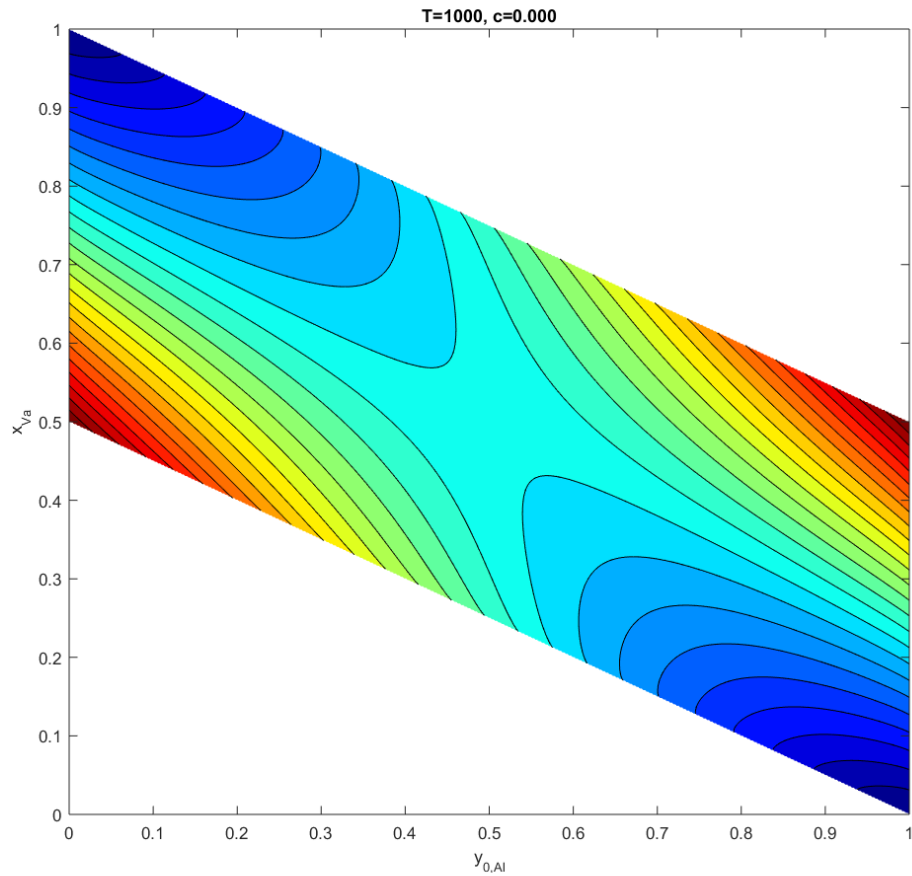


Figure 2.8: Al-Ni Dupin and Sundman [2], the B2 phase at $T = 1000$ and molar composition = 0% nickel. The vacancy fraction x_{Va} is the vertical axis, $y_{0,Al}$ the horizontal axis, and $y_{0,Ni}$ the axis pointing into the page, but the feasible range for this is zero. Red areas are high energy, and blue areas are low energy. You can see there are two stable equilibria, one around $y_{0,Al} = 0.9$ and $x_{Va} = 0$, and the other around $x_{Va} = 0.95$. This demonstrates that a vacancy barrier above 0.25 cannot be prevented from creating local stability points.

Fortunately, we are not attempting to compete in a hypothetical market for CALPHAD software – we are providing academic insights, and an alternative viewpoint from that of the materials science researcher. Some of the advantages of our approach include the wide variety of solvers available, the optimization focus of the work, and enormous flexibility of the model and framework.

A python script was developed reproducing the Hillert algorithm, presumably identical to the core of OpenCalphad’s code. This is a valuable resource, and comprises 457 lines of code,

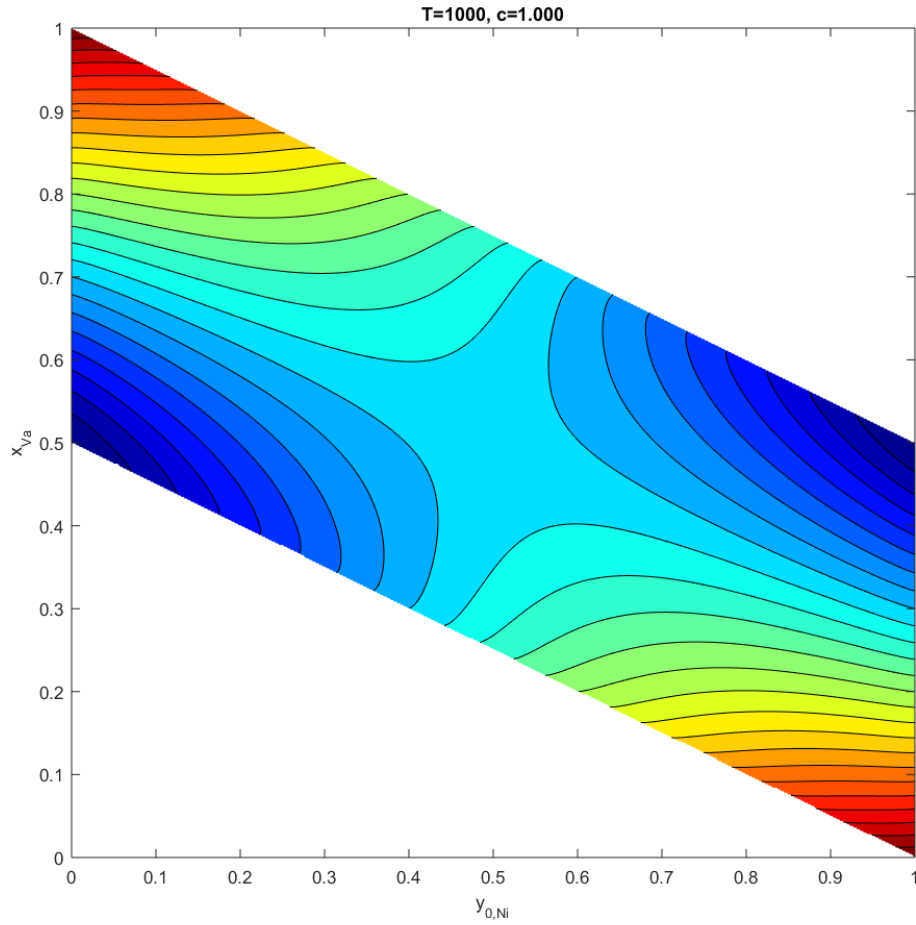


Figure 2.9: Al-Ni Dupin and Sundman [2], the B2 phase at $T = 1000$ and molar composition = 100% nickel. The vacancy fraction x_{Va} is the vertical axis, $y_{0,Ni}$ the horizontal axis, and $y_{0,Al}$ the axis pointing into the page, but the feasible composition range for Al is zero. Red areas are high energy, and blue areas are low energy. You can see there are two stable equilibria, both around $x_{Va} = 0.5$. This demonstrates that choosing a vacancy barrier at or below 0.5 is not advisable.

but it was superseded by the much simpler AMPL implementation of the Hillert-Sundman model described below.

In order to ensure our grasp of the Hillert-Sundman model was correct, it was implemented within a standalone AMPL model for a specific system and set of active phases. This model is entirely distinct from our AMPL framework and from OpenCalphad, having been written exclusively based on the 2015 Sundman paper [77]. This small implementation is simple and

inflexible, but sufficient to ensure our understanding of the model is correct, and it does highlight some of the necessary details absent in the paper.

In the Hillert-Sundman model, the composition of the system is maintained by the constraint, for element A ,

$$N_A = \sum_{\alpha} N^{\alpha} M_A^{\alpha} = \sum_{\alpha} N^{\alpha} \sum_s a_s^{\alpha} y_{As}^{\alpha}. \quad (2.199)$$

The important difference between this constraint and our corresponding constraint

$$F_A^0 = \sum_{\alpha} f^{\alpha} x_A^{\alpha} \quad (2.200)$$

lies in the definition of x :

$$x_A^{\alpha} = \frac{\sum_s a_s^{\alpha} y_{As}^{\alpha}}{\sum_s a_s^{\alpha} \sum_{B \neq \{V_a\}} y_{Bs}^{\alpha}} = \frac{\sum_s a_s^{\alpha} y_{As}^{\alpha}}{\sum_s a_s^{\alpha} (1 - y_{Va,s}^{\alpha})}. \quad (2.201)$$

When no vacancies are present, they are identical. But in the presence of vacancies they differ by the vacancy correction term, the very same term we have placed in our objective function to achieve results in the Al-Fe system where the wide line-phase `al13fe4` is not otherwise corrected calculated.

So we create a new variable

$$\xi_A^{\alpha} = \frac{\sum_s a_s^{\alpha} y_{As}^{\alpha}}{\sum_s a_s^{\alpha}} \quad (2.202)$$

for use in the composition constraint. ξ is equal to x when there are no vacancies, and corresponds directly with Sundman's M_A^{α} variable. By altering the constraint in this way, we have achieved positive results.

Python [102] is a widely used high level programming language generally enabling both short time to complete an implementation (ease of development), and rapid execution times. Many

programming paradigms are supported by python, allowing developers with experience in other languages to understand and write python code relatively quickly. It has a large standard library, and due to the large number of people using python an extremely large number (greater than 72,000) of packages are available providing additional functionalities. The prefix *py-* is used to emphasize that something is python-related, with examples in the CALPHAD literature including `pycalphad` [70] and `FiPy` [71]. June 2016, python was ranked as the fourth most popular programming language by the TIOBE Programming Community Index [103], behind only Java, C, and C++. In 2014, python became “the Most Popular Introductory Teaching Language at Top U.S. Universities” [104]. The combination of ease of development, large number of available packages, and large body of practitioners made python the natural choice for a front-end to our framework.

Originally the mapping algorithm was developed as an AMPL script, which was cumbersome and inflexible. Subsequently a python script was developed to generate AMPL mod files, execute them in parallel, and compile the results into a single output csv. This provides a great deal of flexibility via command-line arguments which control temperature and composition range, number of parallel instances, step sizes, number of random initial conditions to try at each point, the vacancy threshold, and other parameters.

It also allowed later integration of an energy mapper into a single uniform platform.

Under the python script, each time a system is examined a new directory is created where all files are stored, and all execution occurs within that directory. This allows easy review of past experiments, and retains data as long as necessary.

The implementation of parallelization (by executing several `ampl.exe` simultaneously) reduced execution time by up to 95%. With access to a large set of remote computers execution time can be made arbitrarily small.

Further, the plotting of diagrams was moved out of MATLAB and into a python script using `numpy` and `matplotlib`. The inclusion of `numpy` means that on a Windows platform

installation of the correct python packages can be extremely tricky for a novice. At this time Anaconda [105] is the recommended distribution, as it includes all of the most popular high performance science packages in a single free and easy to install distribution and many other beneficial features.

One limitation of the original formulation in AMPL is that the parameters representing energy of end-members of phases, the G terms, cannot be made as “indexed computed parameters” which are functions of T and P and automatically change when T or P is changed. Instead, they are indexed but not computed, and when T or P is changed they must be reset with the `let` command. This means re-reading a mod file every time the temperature or pressure is changed in making a phase diagram, and it prohibits using T or P as a variable to be optimized under certain conditions.

A solution to this difficulty is to formulate the G terms representing energy of end-members as variables with equality constraints on them. This creates a potentially enormous number of variables, and the student (free) AMPL license is limited to a maximum of 500 variables and 500 constraints and objectives.

Fortunately, the AMPL option `presolve` will eliminate fixed variables from the problem before the 500 variable limit is enforced. `presolve` can be set to any whole number, indicating the number of passes AMPL makes through the problem to simplify it before handing it to the solver. As long as it is greater than zero, the fixed variables are all eliminated.

This allows a simpler implementation within AMPL, and also allows experimentation with using T or P as a variable. For example, it may be possible to formulate a problem that directly identifies the liquidus for a given composition.

In the Thermo-Calc version S database specification, a four-sublattice phase may be labeled with a `:F` or `:B` flag, for ordered fcc/hcp symmetry or ordered bcc symmetry respectively. In those cases, a greatly reduced number of end-members needs to be specified, and all its

permutations are automatically created by the software.

For phase-type F, the four sublattice sites indicate the four corners of a regular tetrahedron, so any permutation of a four-sublattice end-member is identical. E.g., these are all the same: A:B:B:B, B:A:B:B, B:B:A:B, and B:B:B:A. Hence, only one of those needs to be presented in the TDB, and its permutations are automatically created [78].

For phase-type B, the tetrahedron is irregular with two pairs of sites as nearest-neighbors. Hence, the end-member A:A:B:B is distinct from A:B:A:B, with the first having two permutations and the second having four.

This presented a challenge to our converter, as it does also for pycalphad and OpenCalphad, neither of which have implemented handling of those flags at the current time.

However, the code was written in the converter to handle the symmetry by automatically ramifying each end-member as appropriate in the AMPL data file it creates. In this regard, this places our database interpreter ahead of either of those two open-source packages. That is not the only meaningful comparison, but it does allow examination of some important systems which are only represented by databases using those symmetry flags.

Examples of phases using the F symmetry are fcc_4sl in the Al-Fe database from Sundman et al. [29], and L1₂ in the Al-Pt database from Kim et al. [98].

Examples of phases using the B symmetry are bcc_4sl and bcc_va in the Al-Fe database from Sundman, et al. [29].

However, aabc is apparently treated differently from abac under :B symmetry.

Fultz [106] provides a good reference on p.336 for bcc and fcc symmetries, including diagrams of B2, D03, L12, L10, and B32, which have been recreated here in Figure 2.10 for bcc and Figure 2.11 for fcc.

B2, D03, and B32 are specific kinds of bcc lattices:

- B2: every corner is the same, every center is the opposite from the corner.

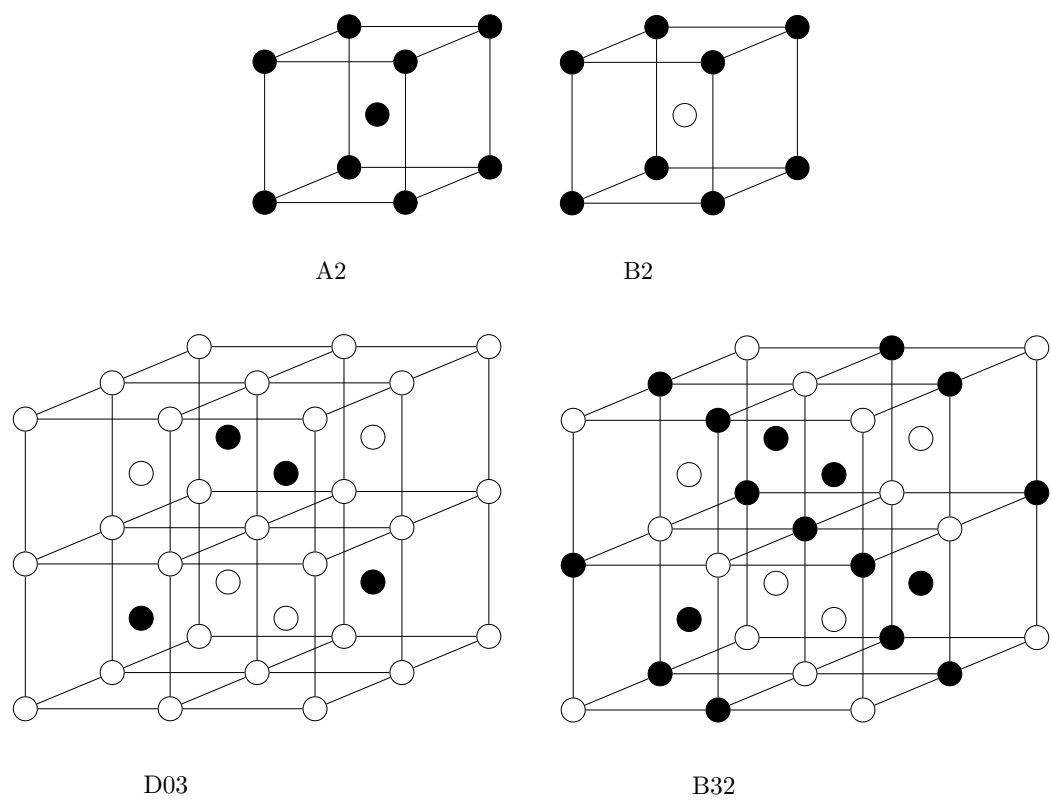


Figure 2.10: Four bcc lattice structures: A2, B2, D03, and B32.

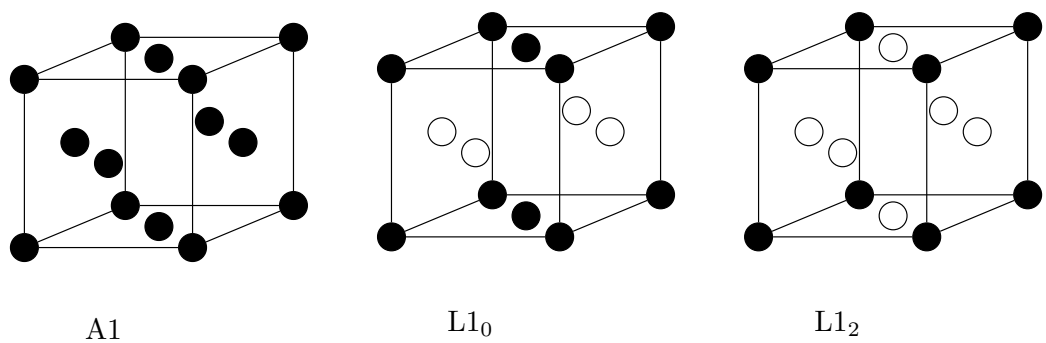


Figure 2.11: Three fcc lattice structures: A1, L1₀, and L1₂.

- D03: every corner is the same, the centers alternate.
- B32: corners alternate, centers alternate. Corner nearest neighbors are opposite to themselves, centers are identical to the 000 corner.

The tests we developed for which type of symmetry an end-member has are:

$$\text{isB2}(e) := (e_0 = e_1 \wedge e_0 \neq e_2 \wedge e_0 \neq e_3) \vee (e_2 = e_3 \wedge e_0 \neq e_2 \wedge e_1 \neq e_2), \quad (2.203)$$

$$\text{isB32}(e) := e_0 \neq e_1 \wedge e_2 \neq e_3 \wedge (e_0 = e_2 \vee e_0 = e_3 \vee e_1 = e_2 \vee e_1 = e_3). \quad (2.204)$$

Chapter 3: Optimization, Sampling, and Uncertainty Quantification

3.1 Special-purpose solver separating iteration into two steps

In late 2014 we investigated a special-purpose optimization algorithm designed with knowledge of the system in mind. We separated the optimization of the y variables from the f variables. We fix f values, solve for optimal y values, and make use of the Lagrange multipliers for the constraint regarding conservation of constituents and the x value for that constituent, as well as the energy of the phase, to make a step in the f space:

If the Lagrangian multipliers for the conservation of constituents are in the vector λ , the matrix of constituent fractions is x , and the computed Gibbs energy in G , then

$$\Delta f := -\lambda^\top x + G. \quad (3.1)$$

This was implemented in AMPL as

```
1 for {phase in Phases, inst in PhaseInstances[phase]} {
2     for {elem in Elements : elem != 'Va'} {
3         let f[phase,inst] := f[phase,inst] -
3             Conservation_Components[elem] * x[phase,elem,
3                 inst];
4     }
5     let f[phase,inst] := f[phase,inst] + phaseGibbs[phase,
5         inst];
6 }
```

f would then need to be normalized so that $\sum f = 1$. There are also complications with `phaseGibbs` being a negative number with large magnitude.

After several months of experimentation, we concluded that this ingenious method of taking steps in f knowing the result of optimizing y turns out to be less efficient than allowing a general-purpose solver to optimize in both variables simultaneously.

3.2 Sampling

Some attention was given to picking appropriate points in Y space. Initial results can be seen in Figure 3.1 and Figure 3.2.

An ideal algorithm would allow the automatic choice of Y values to correspond to a given composition, but this has not yet been achieved.

Even including very many samples does not guarantee a dense collection of points in the lower hull, as can be seen in Figure 3.3.

Better initial conditions

Our random initial conditions are almost always outside the feasible region. We select each y value on a uniform interval, with no regard for the other y values or the fixed composition this test is being conducted at.

If the y values are projected directly into the lower-dimension feasible space it's quite unlikely we would be conducting effective tests of the corners of feasible space. E.g., a two-element single sublattice phase would have a composition space which looks like the segment from $(1, 0)$ to $(0, 1)$, shown as blue in Figure 3.4.

By selecting uniformly from the whole space we are not selecting uniformly from the smaller feasible space. Figure 3.4 shows 50 uniformly random points in the $[0, 1]^2$ square projected onto the feasible space.

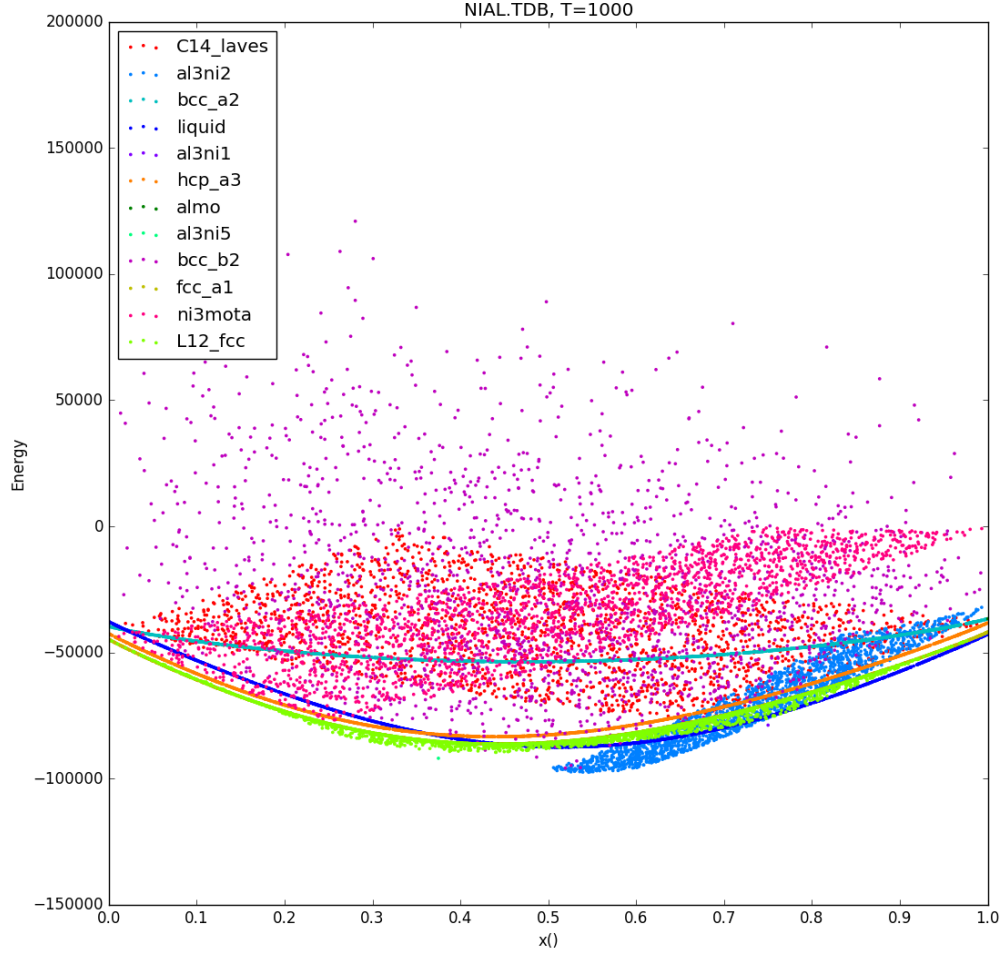


Figure 3.1: Energy of the AlNi system using the new AMPL-only (no solver) method of calculation.

In Figure 3.4 we see that the ends of the blue segment are under-represented, and the middle is over-represented. The distribution on the segment is not uniform, it's triangular.

Ensuring that sampling is uniform in the feasible y space is achieved by a two step algorithm,

$$\hat{y}_j := -\ln(\text{rnd}(0, 1)), \quad \forall j \quad (3.2)$$

$$y_i := \hat{y}_i / \sum_j \hat{y}_j, \quad \forall i. \quad (3.3)$$

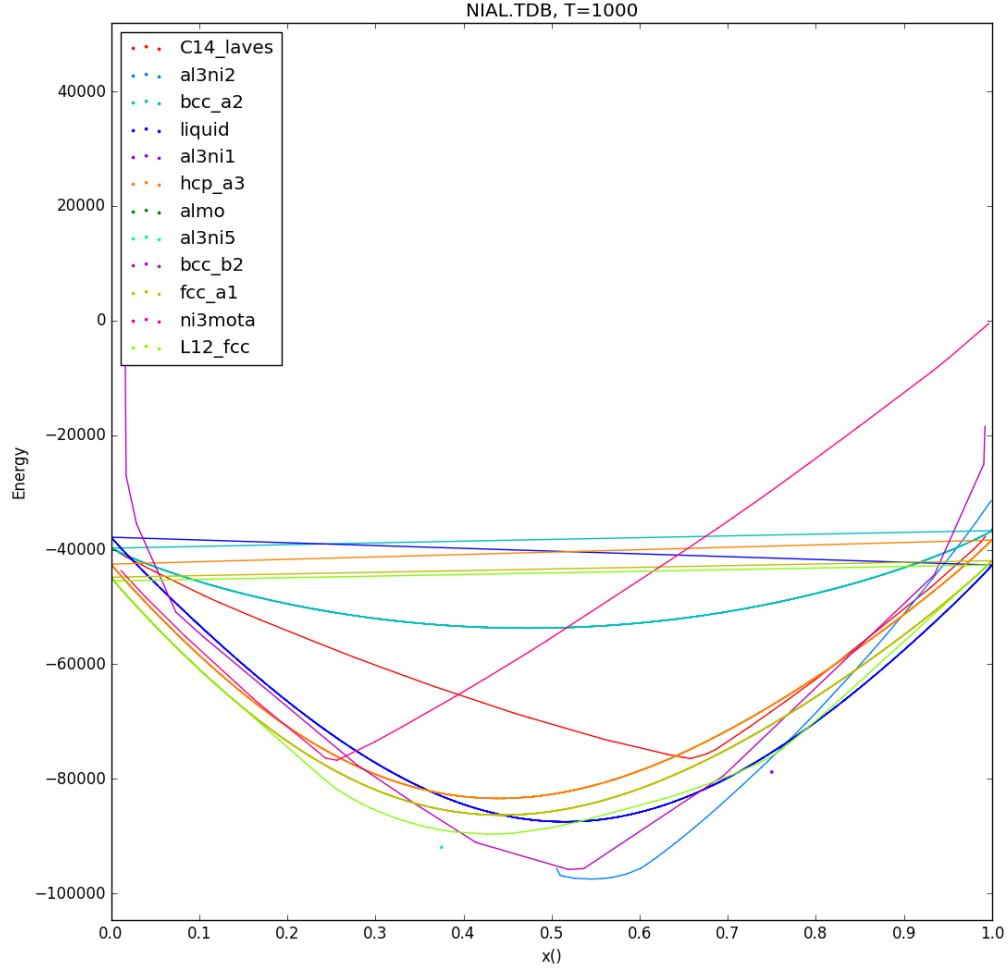


Figure 3.2: Lower hull of the energy of the AlNi system using the new AMPL-only (no solver) method of calculation.

Thus we have a uniform distribution on any unit polytope of dimension $n - 1$ in an n -dimensional space, i.e., the polytope with vertices at points with coordinates consisting of permutations of $(1, 0, \dots, 0)$.

Example projections in 1-, 2-, and 3-d can be seen in Figure 3.5.

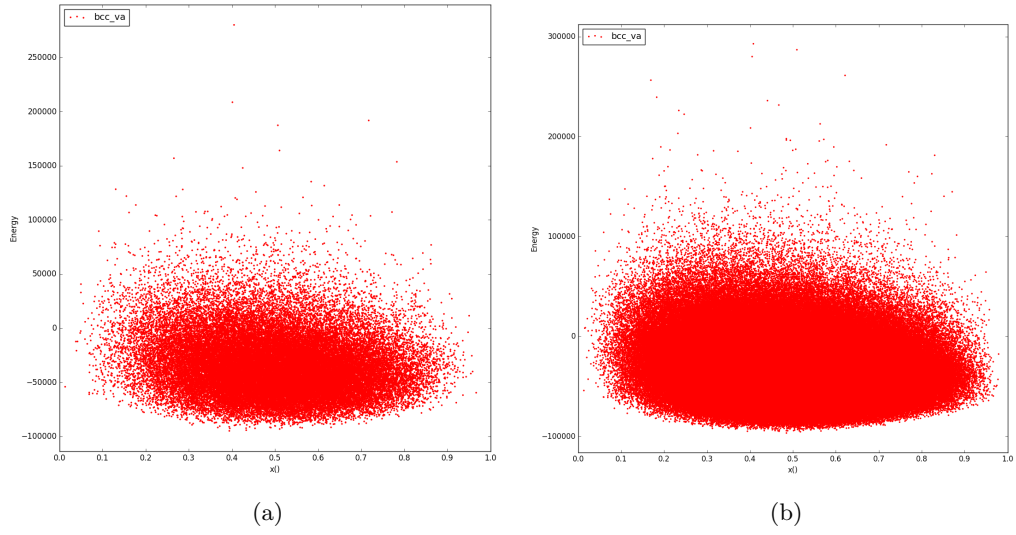


Figure 3.3: AlFe Sundman Option 2, bcc_va energy at 1400K. The left image shows 50,000 sample points uniform in feasible region, the right image shows 500,000 points. Neither is dense along the lower hull, nor in the extreme left or right edges of the image.

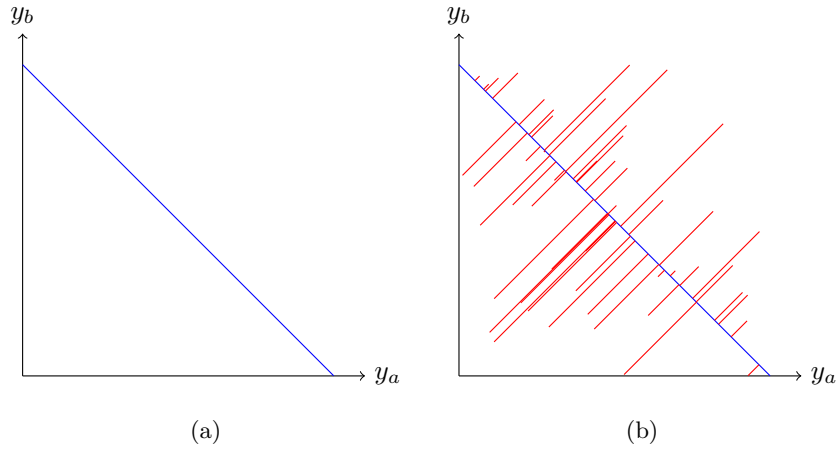
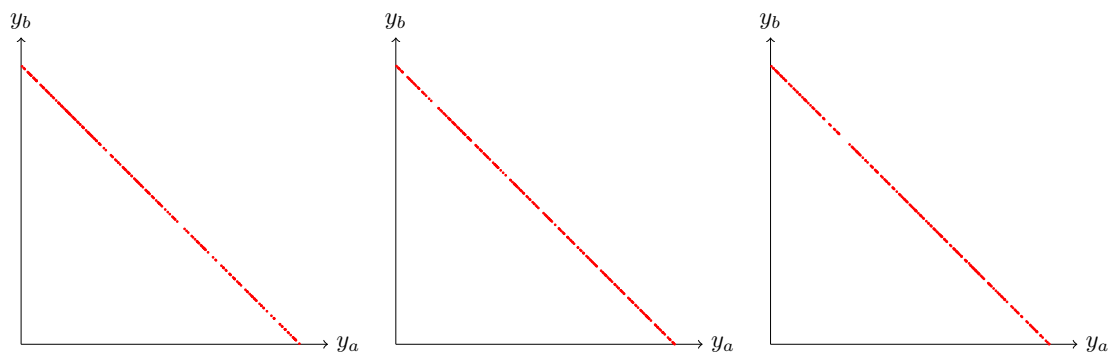
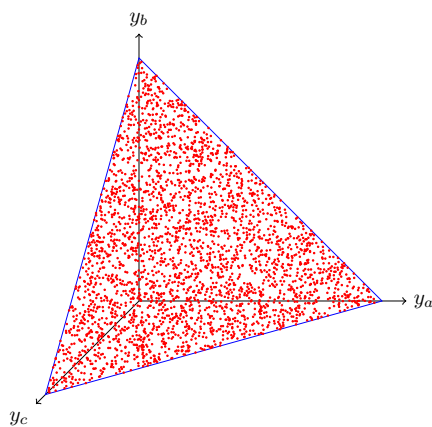


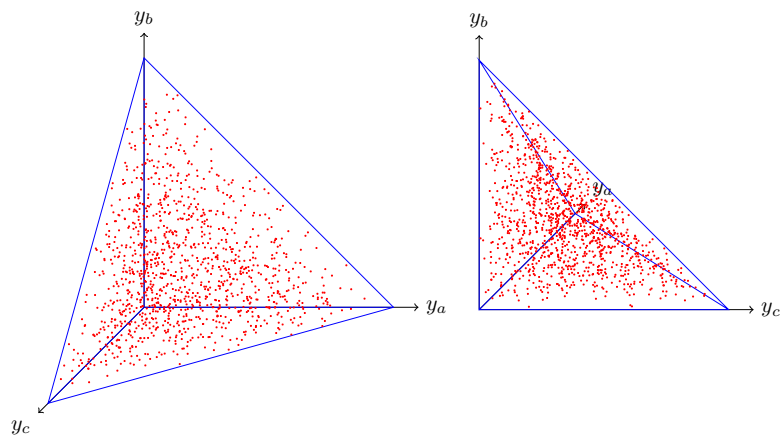
Figure 3.4: Sampling uniform in space projects non-uniformly on 1-d y subspace.



(a)



(b)



(c)

Figure 3.5: Uniform sampling in the feasible space, in 1-d, 2-d, and 3-d. N.b., the 3-d view is a projection onto the 3-d simplex from 4-d space.

3.3 Uncertainty Quantification

Since its inception, CALPHAD has relied on fitting parameters to a collection of sometimes conflicting empirical data of varying quality. Researchers would minimize the error of the fit to the empirical data in various ways, notably a computational least-squares approach [12], but tracking the uncertainty in those fitted parameters and its effect on the final quantitative result has not been introduced into the method. Prior to the inception of our work, no method for quantifying the uncertainty of CALPHAD results existed [18]. Parameters and system assessments are validated by plotting diagrams overlaid with empirical and computational (DFT) results.

The entire CALPHAD method at present is based upon using a combination of empirical and computed values with some inherent error to determining fixed, absolute parameters with no error information propagated forward. The researcher conducting parameter fitting must choose which empirical values to keep and which to discard, and picks a method to minimize the uncertainty, such as least squares [12], or a Bayesian method [107]. Once the errors have been minimized by selecting certain parameters, the error information is discarded.

First-principles computational methods such as DFT are increasingly being relied upon for input into the CALPHAD method, and measuring the uncertainties of values determined by these *ab initio* methods is an active field of research [108] [40] [109]. Those uncertainties provide extremely valuable information which, if they could be incorporated into CALPHAD, would increase the scientific credibility and reliability of its results.

A researcher naturally benefits from knowing when a calculated result is uncertain, or how large the “error bars” are on a value. CALPHAD is valuable in calculating many material features of alloys and other combinations of multiple constituents (elements), originally designed to determine the particular mix of phases in a system at a given temperature, pressure, and composition. Determining which phase is present at a temperature naturally also determines the liquidus of a system. Today the method is actively being extended to

many other features, but here the liquidus provides a convenient example of quantitative values readily generated by the traditional CALPHAD method for which an estimate of error would be immediately meaningful and valuable. Even this simple indication of uncertainty has never been reported by researchers.

Another example of a quantitative value important in industry is the eutectic temperature in steels. Proper heat treating involves crossing the eutectic temperature at a specific rate, and varying the cooling rate can produce vastly different material features, e.g., hardness by quenching (cooling relatively quickly), ductility by annealing (cooling relatively slowly). Assigning uncertainty to the calculated eutectic temperature would improve the value of the CALPHAD method to industry.

Less straightforward is the presentation of uncertainties on the location of phase boundaries (monophasic lines) in phase diagrams. If phase energies as a function of temperature, pressure, and composition can be visualized with error bars, then a researcher could examine such a diagram and visually ascertain the size of regions with uncertainty. Work specifically on this subject does not appear in the literature at this time.

Every system assessment in the literature begins with a survey of all available empirical data and of data from first principles calculations on the material features of interest. For some systems important in industry many studies may have been published over the course of many decades, and the data can be conflicting and of widely varying quality. The researchers may choose to discard some data, and they will assign weights to the remaining data according to its perceived quality, with larger weights implying greater accuracy or importance of the data.

When this wide variety of data is combined, the researchers then fit parameters. For example, they can use techniques which minimize the mean squared error [12] [110]. The least-squares technique is based on the assumption that observed data follows a Gaussian distribution and is greatly swayed by outliers in the data, which is a significant motivation for discarding data which does not fit the overall pattern. The PARROT [111] module

included in Thermo-Calc conducts such a least-squares based parameter fitting. For data presumed to follow a normal distribution, quantification of error is familiar to researchers in the form of variance or standard deviation, and alternative distributions of error may include higher order moments such as skewness and kurtosis. Errors following an exponential or other distribution may be transformed to a normal distribution by appropriate functions, and least-squares can be applied using the transformed values.

An alternative approach which has been applied to parameter fitting is the Bayesian method [107]. This may be less sensitive to outliers, and Königsberger states that it can produce good results with less empirical data. Bayesian methods can include uncertainty quantification [112], and that information can be passed forward to be included in the CALPHAD method.

At the moment, although it is clear in some cases how to determine the uncertainties of parameters, there appear to be no organized attempts to propagate uncertainties through the CALPHAD method to its outputs.

Even a basic approach to tracking and reporting uncertainties to the researcher would be a significant and valuable addition to the CALPHAD method at this time. A simple study of the uncertainty surrounding the liquidus or any other calculable quantity, such as error bars around monophasic lines, would be an advancement of the current state of the art. Testing uncertainty quantification techniques may require fitting parameters for a system to obtain the necessary error data.

In the absence of methods for quantifying uncertainty within CALPHAD, phase diagrams present values as though exact. Researchers relying on these diagrams will benefit from an awareness of the precision, or the lack of precision, in values portrayed on a diagram.

While the ideal approach to UQ may involve tracking the uncertainty of parameters fitted to experimental and computed material data, that lies beyond the reach of this dissertation. Rather, we rely on the fitted parameters and produce a “heat map” of the nearness of a metastable phase to the equilibrium at each point.

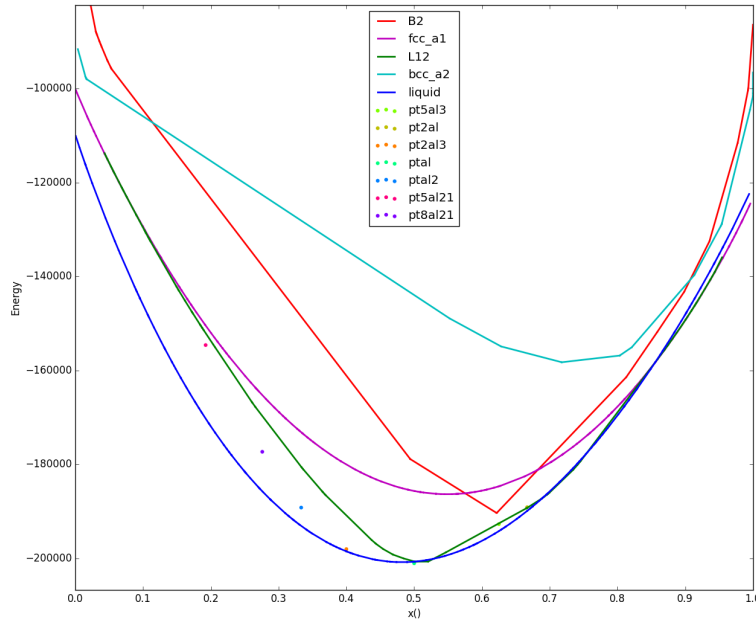


Figure 3.6: Al-Pt energy at 1800K, without vacancy corrections.

In an energy diagram at a fixed temperature, e.g. Figure 3.6, at a given composition there will be one phase or set of phases giving the lowest energy. There will be a phase or set of phases giving the next-lowest energy, which is a metastable solution, potentially a local equilibrium.

By identifying the difference between the equilibrium energy and the next-lowest energy at each point in the temperature-composition space, we can create a “heat map” of the differences as an indication of the uncertainty surrounding the diagram at each point.

Examining Figure 3.6, we can see there is little uncertainty at compositions in the left-hand side of the diagram, but much more uncertainty in the right-hand side.

Figure 3.7 shows an Nb-Re phase diagram and the corresponding diagram of the distance (in Joules per mole) to the next-lowest-energy metastable state.

At the moment, researchers have no indication of the uncertainty in a phase diagram, and this provides a first look at these values.

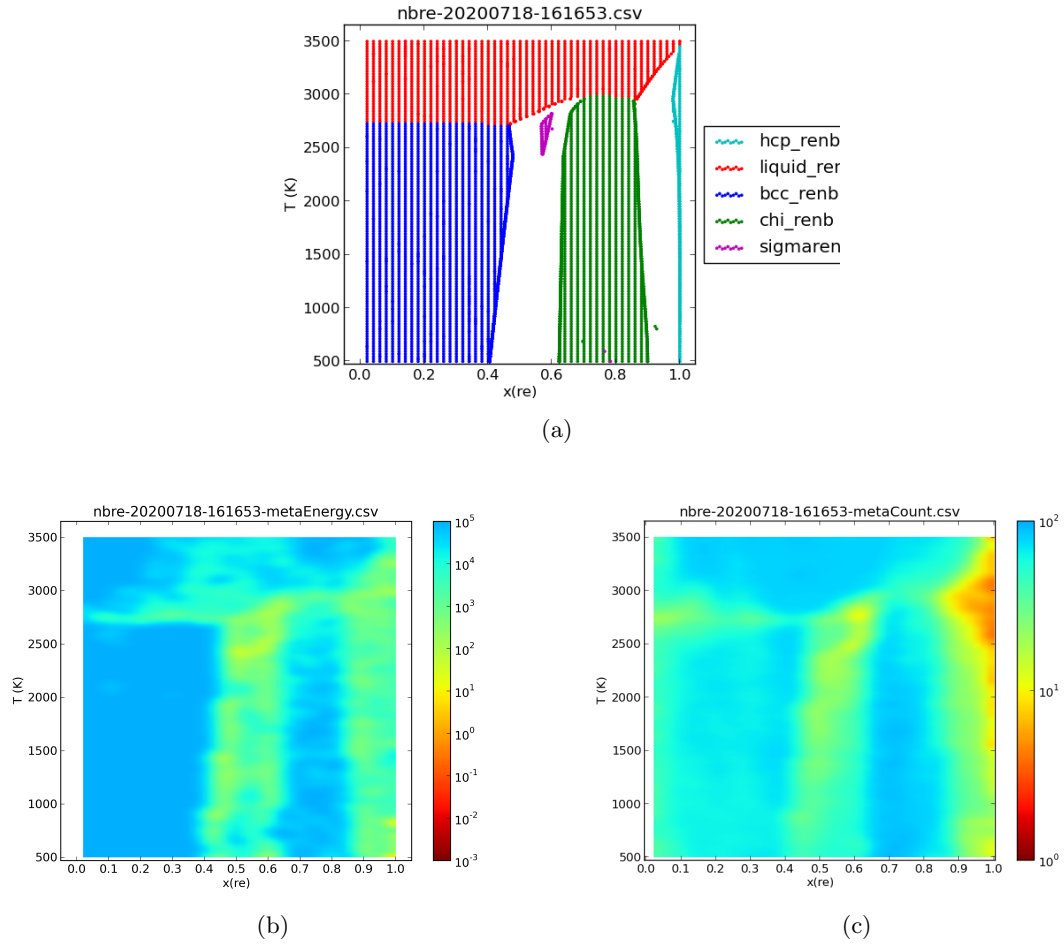


Figure 3.7: (a) shows the Nb-Re phase diagram, (b) the corresponding diagram of the additional energy required to reach the lowest metastable phase in the middle, and (c) the ratio of initial conditions that converge to the lowest equilibrium to local equilibria on the right.

Sampling Approaches

Our framework uses traditional optimization methods to converge to a local equilibrium, and the solver is heavily dependent upon the initial conditions given to it. There are a variety of heuristics for exploring the composition space to improve the odds that we identify the global equilibrium, a few of which are mentioned in the following section. The approach we have explored most thoroughly is to start the optimization several or many times with different

initial conditions, and then select the minimizer producing the lowest system energy.

We call the pattern of initial conditions which are applied a *sampling pattern*. The most naïve approach is to select random values for each y and f variable independently in $[0, 1]$, and then converge from there. This has a number of significant disadvantages: 1) the selected values are almost never feasible, so the method must take one or more initial steps to reach a feasible value, which can be a challenge for some methods; 2) a random pattern leaves some areas of the space under-explored, and concentrates points in some other areas; and 3) which is exacerbated by the projection of the whole $[0, 1]^m$ space being strongly weighted into some portions of the lower-dimension feasible space while making samples in other feasible regions highly unlikely.

We have conducted research in ensuring initial conditions are always selected so they satisfy the constraints of the optimization problem. This is described below.

The application of low-discrepancy techniques to the CALPHAD problem has been described by R. Otis et al., [113]. We apply a Halton sequence to selection of initial conditions for trials, and the improved uniformity of sampling with this sequence compared to a pseudo-random sequence can be seen in Figure 3.8.

Computational materials scientists are working to quantify uncertainties surrounding calculated properties of materials such as those represented directly or indirectly in phase diagrams: phase energies, material structures, liquidus, solidus, etc. One area of focus is the CALPHAD method, which ideally fits parameters to the body of experimental and *ab initio* computational evidence and extrapolates properties of new systems. Within this methodology there are several sources of uncertainty. The natural uncertainty inherent in a physical sample, the measurement error of samples, the uncertainty of the parameters fitted to the sample data, uncertainty of the model itself, and propagation of uncertainty from those sources through the model.

Various approaches are being taken to studying sources of uncertainty and its propagation,

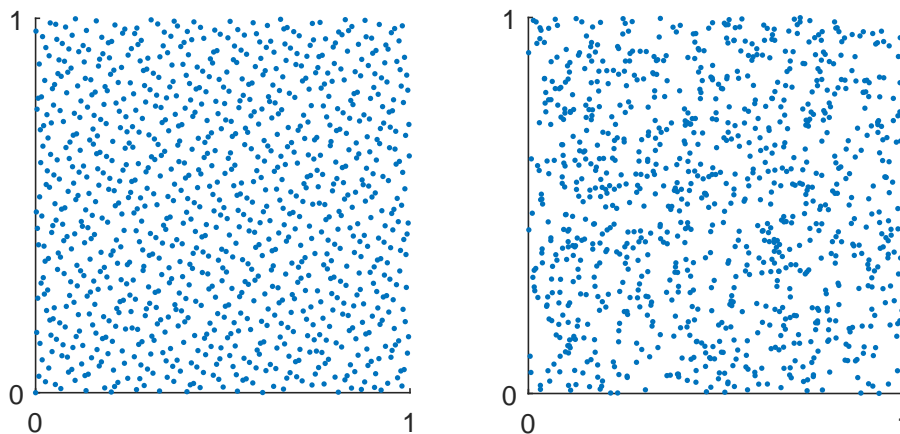


Figure 3.8: The Halton quasi-random sequence 2,3 (left) compared to pseudo-random points (right). The improved uniformity of sampling is visible in the Halton sequence, eliminating heavily over- and under-sampled regions apparent in the random sequence.

[114], [115], [116], [117].

The approach taken here is to accept the parameters as given and evaluate where uncertainties can arise within the methodology, how large those uncertainties are and how sensitive the results of the method are to those uncertainties. Two distinct metrics for UQ within CALPHAD have been identified and developed here. The metrics devised provide insight into existing thermodynamic databases, allowing researchers studying UQ to refine and compare their methods in a new way, and providing practitioners insight into possible errors in phase diagrams.

At a given temperature and composition there is the stable equilibrium, and possibly one or more metastable equilibria at higher energies. The two approaches described here depend on (1) the energy difference between the lowest metastable equilibrium and the stable (global) equilibrium, and (2) the frequency with which an optimization method finds the stable vs a metastable equilibrium.

During the process of creating a phase diagram, we can measure uncertainty in different ways. Two we address here are,

1. The energy difference between the global equilibrium and the lowest energy metastable equilibrium
2. The fraction of initial conditions which converge to the global equilibrium

One consideration is whether a metastable equilibrium should be required to contain a different mix of phases than the apparent global equilibrium.

In this section we compare Model 1 (the Hillert model) which corresponds to literature such as [77], where it is thoroughly explained, and the other labeled Model 2 (the Molar model) as described in this dissertation and in [1].

Equilibria

When examining at a binary system with vacancies and multiple phases, there are several categories of equilibrium, with one, two, or three phases present. A monophasic equilibrium has a single phase present. Most points in a typical binary phase diagram will be a mixture of two phases each with a different composition that combine linearly to create the composition of that T, c point. Given the presence of vacuum as a constituent of the system it is possible for three phases to be present at equilibrium in a binary system.

A miscibility gap is a region where a single phase mixes with itself in more than one different composition. Our framework automatically creates multiple instances of each phase based on the number of constituents it comprises, to ensure any miscibility gap can be spanned by that phase.

Optimization

For each temperature-composition (T, c) point of interest many initial conditions are tried and an equilibrium is identified using one of the solvers available within AMPL. An initial condition comprises a phase fraction, y values for each phase, and dual variables for the optimization solver. The x value of a phase is fully determined by its y values.

Depending on the system being examined, at each T, c point there are three basic categories of metastability: 1) every initial condition converges to the same equilibrium (i.e., no metastable points); 2) there are two or a small number of clearly distinct equilibria; and 3) there may be many closely spaced equilibria. Equilibria are distinguished by the phases which are present and by the y values and resulting energy of the phases.

Vacancies

Vacancy is a constituent used two distinct ways in thermodynamic databases. One corresponds with physical vacancies present in the material, e.g., thermal vacancies. The other is a “modeling vacancy” used to achieve a better mathematical representation of the energy in a phase, which does not correspond to a physical feature of the system.

A database may be formulated in such a way that vacancies become the predominant constituent as a solver seeks equilibrium, even until the phase has almost no real constituents. For physical vacancies this is an unrealistic condition. Figure 2.4 shows how the only stable equilibrium for a hypothetical phase can occur very close to an all-vacant condition.

Various remedies preventing an undesirable excess of vacancy are possible, with various impacts and subtleties. However, for the purposes of this dissertation a simple, static constraint is applied to each phase requiring it to contain a minimum fraction of real (non-vacancy) constituents:

$$\xi_{\text{Va}}^{\phi} \leq C_{\text{Va}}, \forall \phi. \quad (3.4)$$

Figures 2.8 and 2.9 show how there is no consistent vacancy constraint C_{Va} you can pick *a priori*, but applying this technique is adequate for the purposes of the current UQ study. In particular any equilibrium found where the constraint (3.4) is active is discarded.

Figure 3.9 shows the effect of different vacancy barriers on the solution.

The framework has been further adapted to calculate two new types of metric:

1. The energy difference between the stable (global) equilibrium and the lowest-energy

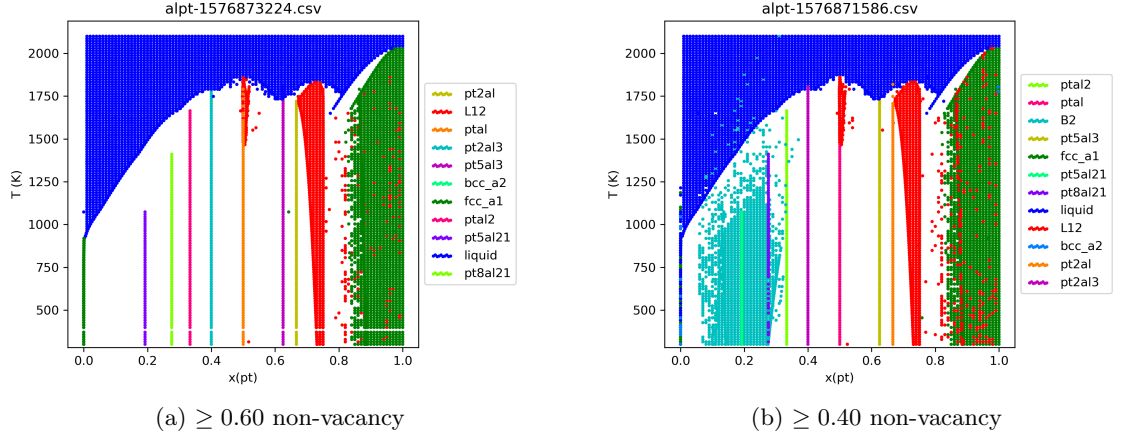


Figure 3.9: Al-Pt binary phase diagrams calculated with different vacancy constraints using Model 2 and snopt.

metastable (local) equilibrium

2. The fraction of distinct initial conditions which converge to the stable (global) equilibrium vs the fraction which converge to a metastable (local) equilibrium

Each is discussed in detail below.

Energy delta

The stable (global) equilibrium has the lowest energy possible in that system at that T, c point. For this research a predefined and fixed number of trials is performed at each point, and assume – sometimes incorrectly – that the lowest energy found corresponds to the true global equilibrium.

In the event that there are other distinct equilibria found, the energy delta between the lowest and second-lowest equilibria provides information about the uncertainty of the result or of the system.

There are several criteria that may be applied to distinguish the stable equilibrium from the lowest metastable equilibrium. Do they contain distinct phases, or is the metastable

equilibrium a different configuration of y values for the same phase(s)? Is the metastable equilibrium sufficiently distinct, or does it differ by only a tiny fraction and produce only a small energy difference? In the initial work shown here, the Gibbs energy is rounded to the nearest integral J/mol value, and the lowest and second-lowest values found are tracked without requiring the phases to be distinct or for there to be a minimum energy difference other rounding to different integers.

This energy delta between the lowest energy result and the next lowest result is plotted in the Results section below.

Fraction of initial-condition space

While AMPL offers some optimization methods labeled “global” methods, they are tailored for smaller combinatorial problems, rather than the continuous nonlinear objective function of the CALPHAD model with Gibbs energy. The optimization methods used in this study are local methods which converge from a specified initial condition to a mathematically stable equilibrium. The configuration space consists of y variables for each phase, and the fraction of the system which comprises each phase, labeled f . Given feasible y and f variables as an initial condition, a local optimization method such as SNOPT or MINOS reliably converges to the same equilibrium – but selection of a different initial condition may converge to a different equilibrium. The regions which converge to a particular equilibrium can be viewed as a “basin of attraction” to that equilibrium, although the optimization methods used here are substantially more sophisticated than a Newton-Raphson type method where the basin terminology is encountered. Given the large number of variables, visualizing these basins is challenging and not revealing. Rather than diagramming basins, the number of initial conditions which result in each equilibrium can be counted to provide a fraction of the configuration space which corresponds to that equilibrium, and this can be identified as a metric of how difficult each equilibrium is to detect. For example, if the lowest-energy equilibrium is the result every time, or 99% of the time, then it may be presumed to be the correct result and it is subjectively *easy* to find the correct result. On the other hand, if

```

pick a model
pick a solver
discretize the  $x$ -axis,  $x_i = i\Delta x$ 
for each  $T$ :
    for each  $x_i$ :
        pick  $n$  low-discrepancy initial conditions
        for each initial condition:
            run solver to equilibrium

```

Figure 3.10: High level sketch of the algorithm for finding UQ values over the T, x space.

after 100 trials the lowest energy result has been found only a single time, or worse if there are 100 distinct results, then it cannot be stated with any certainty that it is the true stable (global) equilibrium, and it is subjectively *hard* to find the correct result.

In order to quantify the difficulty of identifying the global equilibrium, the fraction of the configuration space that converges to the lowest energy equilibrium measured in this way by sampling using initial conditions is plotted in the Results section below.

Technique

The x -axis is discretized and compositions $x_i = i\Delta x$ are used for any choice of Δx suitable to the analysis. In this work $\Delta x = 0.01$ is to generate diagrams which are qualitatively useful and demonstrate the effectiveness of the method. For a given choice of model and solver, at each T and x_i of interest, from n distinct initial conditions covering the composition space the solver runs to equilibrium, producing n results.

The lowest energy found after n trials is assumed to be the stable equilibrium at that T, x_i point. In this work each point is independent from neighboring points and no equilibrium data is shared. For future work, accuracy in estimating the energy and phase mix of the stable state and might be improved by using more sophisticated sampling methods and distributing information, but only the simplest method is applied here to demonstrate the concept.

At each T, x_i point, there are two metrics of uncertainty:

- Energy difference between stable and lowest metastable state
- Fraction of samples that converge to the stable equilibrium

One motivation, and a caveat

A phase in itself may have many metastable states, as you see in Figure 3.11 for Al-Pt phase `bcc_a2`, and phase B2 which receives a disordered contribution from `bcc_a2`.

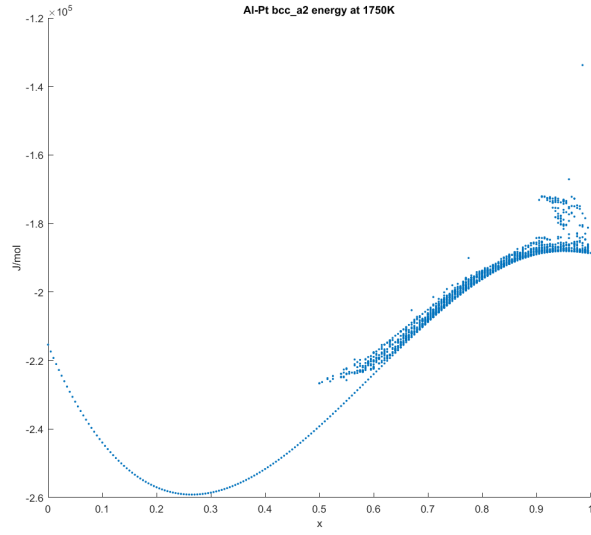
As there can be a continuous range of energy values of metastable states in a single phase there is no minimum positive value energy difference between a stable phase and it's nearest metastable state. Hence, a metastable state for energy difference is used only if it contains at least one distinct phase from the stable state.

Results: Al-Pt

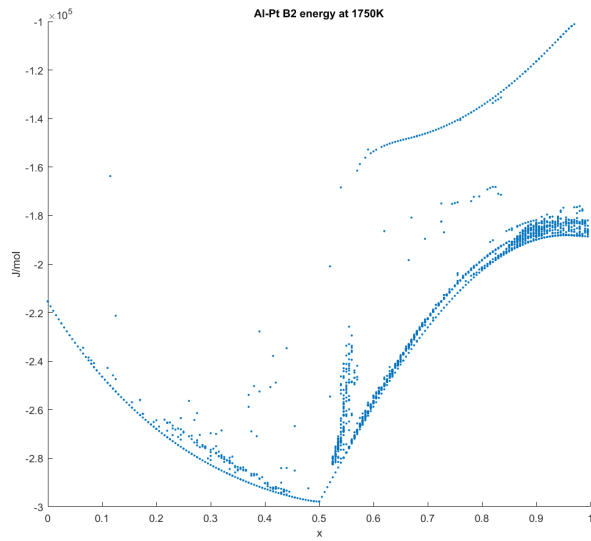
Here the Al-Pt system from the thermodynamic database published in June 2010 is examined [98]. This system has miscibility gap and challenging issues with vacancies.

Results from the various combinations of model and solver are shown in the following figures.

Figure 3.12 shows Model 1 and SNOPT, Figure 3.13 shows Model 2 and SNOPT, Figure 3.14 shows Model 1 and MINOS, and Figure 3.15 shows Model 2 and MINOS. In each figure, the top image is simply the phase diagram itself. The second row is the Uncertainty metric generated by counting the number of initial conditions which converge to different equilibria. The third row is the Uncertainty metric generated by measuring the energy difference between the lowest and second-lowest equilibria. Each represents data from 100 random trials per point, maximum vacancy fraction 0.45, not accepting equilibria with close ordered and disordered phases, and phase-fixing 6 consecutive times. The first image is the phase diagram as generated. The second image indicates the ratio of the number of times the lowest-energy equilibrium is found to the number of random initial conditions tried, where red shows a very small fraction of the attempts and blue a large fraction; red should correspond to an equilibrium that is relatively difficult to detect, hence a high uncertainty



(a) Al-Pt bcc_a2 energy



(b) Al-Pt B2 energy

Figure 3.11: Al-Pt energies for bcc_a2 and B2 phases, showing many metastable states with energies very near the stable state. This motivates the choice of only taking an energy difference between equilibria that have at least one distinct phase in the mix.

that the correct equilibrium has been found. The third image indicates the energy difference (in J/mol) between the lowest energy equilibrium and the next-lowest comprising at least one distinct phase, with red showing a small energy difference and light blue a large energy difference; red corresponds to a metastable equilibrium that is close in energy to the stable equilibrium, hence some uncertainty in the correctness of the identified phases. To generate the relatively coarse UQ diagrams, 129 distinct temperatures were used, with 101 distinct composition points each; since 100 random trials were initiated at each of those 13,029 coordinates, information represented here is the result of 1.3 million independent runs of a solver, which is enormously more time-consuming than simply generating a binary phase diagram.

In the right-hand side of Figure 3.12 you can see the result from using SNOPT and Model 1. In that figure there is a somewhat jagged boundary between fcc and L12 at $x(\text{pt}) \approx 0.85$, more pronounced at lower temperatures. That boundary is examined by running 500 trials at uniform random initial conditions, and the result is seen in Figure 3.16. This diagram is quite different from other diagrams in the literature.

The top image shows temperature along the vertical axis and composition along the horizontal, and the bottom image instead shows energy along the horizontal axis – both images represent the same data. The color indicates the phase which is present, and the size of the circle indicates the relative frequency with which that phase is found at the composition or energy. You can easily see that the lowest energy phase is pt2al, but that has fixed composition $x(\text{al}) = 0.333$ and must pair with a higher energy phase to be present, such as bcc comprising only platinum.

The energy is examined further by looking at a chart showing the energy of phases found in equilibrium (both stable and metastable), and the size of the disc showing the equilibrium phase indicates the relative frequency with which it is found. That chart is in Figure 3.17.

Results: Co-Mo

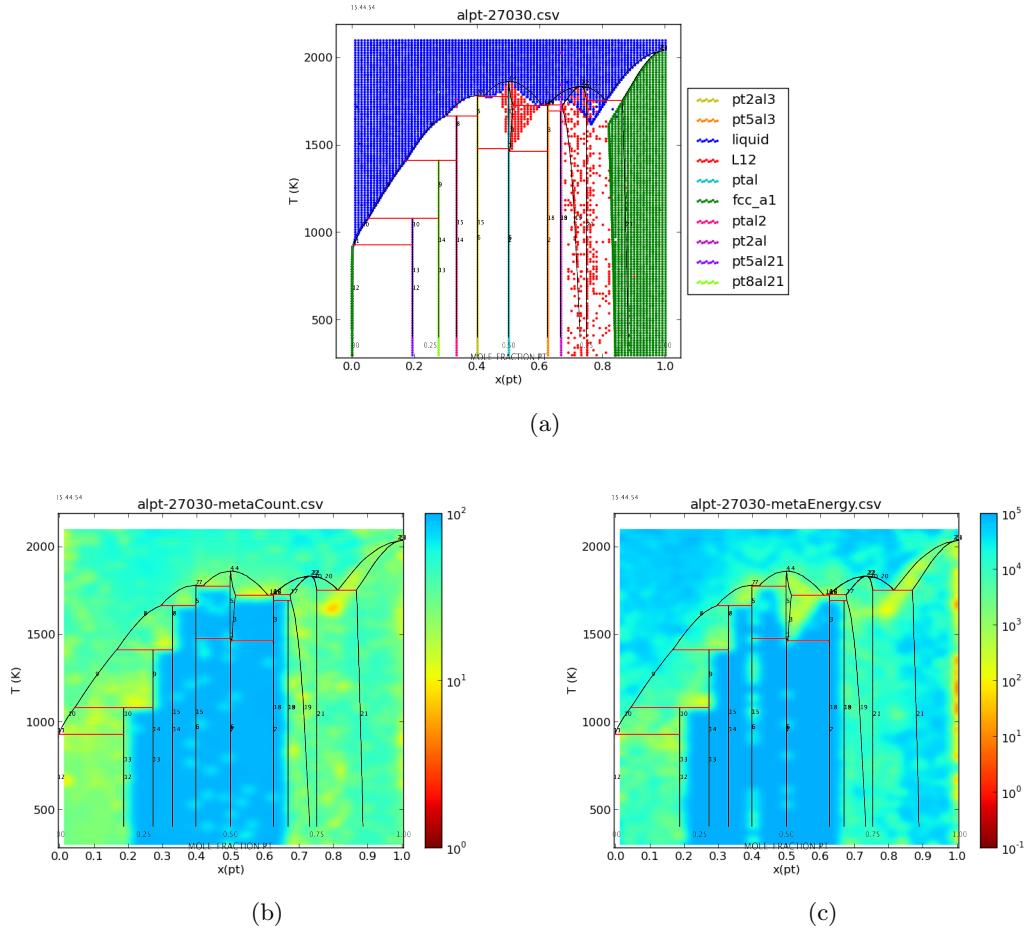


Figure 3.12: The Al-Pt system, produced with Model 1 and SNOPT.

Here the Co-Mo system from the thermodynamic database published in 2009 is examined [118]. This system has a miscibility gap but with no vacancies it presents a simpler problem than the Al-Pt system above.

Each of the figures is generated from data compiled by running 100 random trials per point, not accepting equilibria with close ordered and disordered phase, fixing phases six consecutive times, and 0.55 nonvacancy fraction (irrelevant in this vacancy-free system). The top left image is the phase diagram as generated. The top right image is a histogram of the energy gap between the stable and lowest metastable states over the whole image. The bottom left

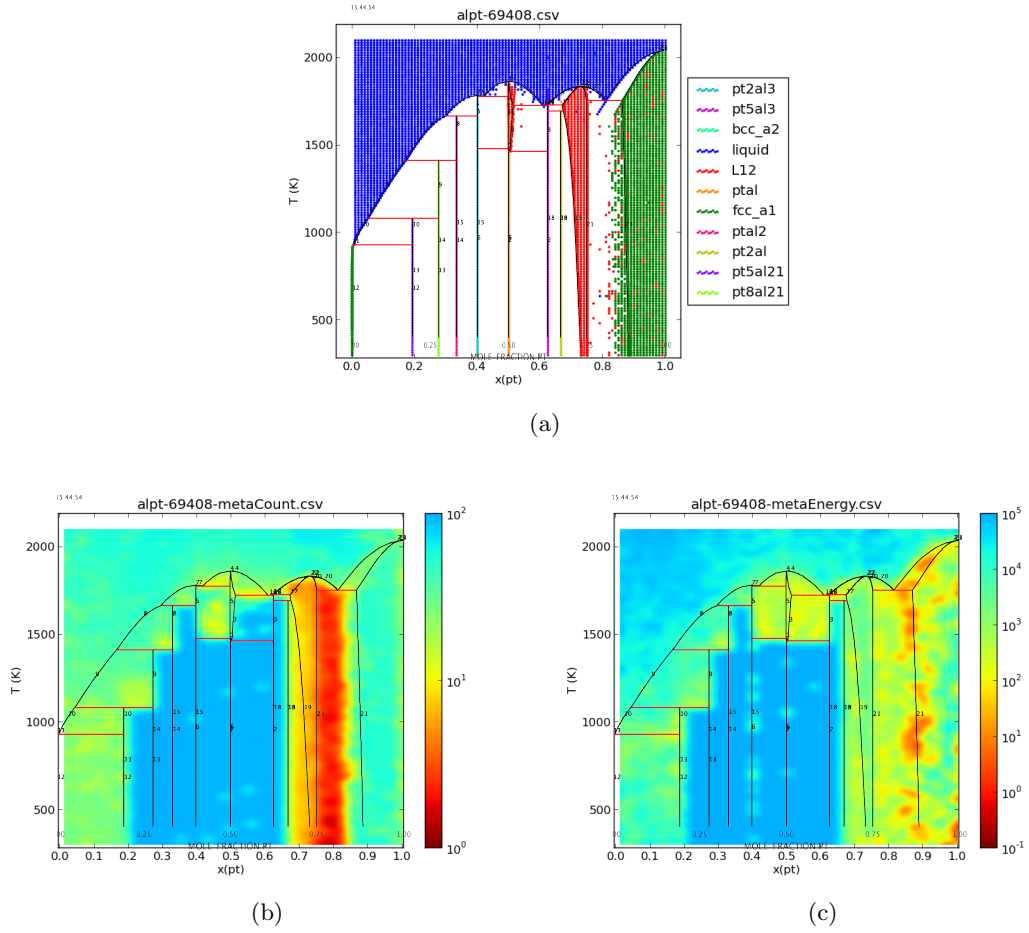
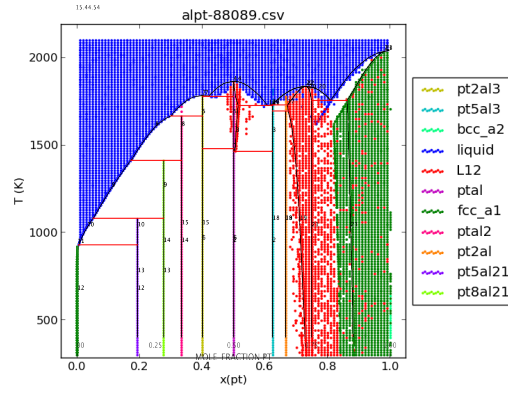
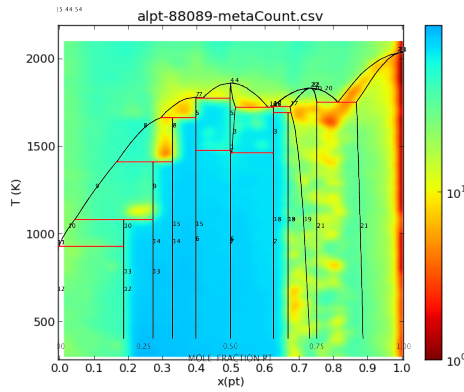


Figure 3.13: The Al-Pt system, produced with Model 2 and SNOPT.

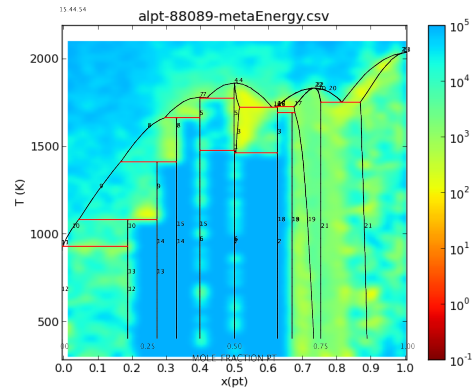
image indicates the energy difference (in J/mol) between the lowest energy equilibrium and the next-lowest comprising at least one distinct phase, with red showing a small energy difference and light blue a large energy difference; red corresponds to a metastable equilibrium that is close in energy to the stable equilibrium, hence some uncertainty in the correctness of the identified phases. The bottom right image indicates the ratio of the number of times the lowest-energy equilibrium is found to the number of random initial conditions tried, where red shows a very small fraction of the attempts and blue a large fraction; red should correspond to an equilibrium that is relatively difficult to detect, hence a high uncertainty that the correct equilibrium has been found. To generate the relatively coarse UQ diagrams,



(a)



(b)



(c)

Figure 3.14: The Al-Pt system, produced with Model 1 and MINOS.

129 distinct temperatures were used, with 101 distinct composition points each; since 100 random trials were initiated at each of those 13,029 coordinates, information represented here is the result of 1.3 million independent runs of a solver, which is enormously more time-consuming than simply generating a binary phase diagram.

Figure 3.21 shows the results from Model 1 and SNOPT, Figure 3.22 shows Model 1 and MINOS, Figure 3.23 shows Model 2 and SNOPT, and Figure 3.24 shows Model 2 and MINOS.

Each figure shows the Uncertainty metric generated by counting the number of initial conditions which converge to different equilibria, the Uncertainty metric generated by measuring

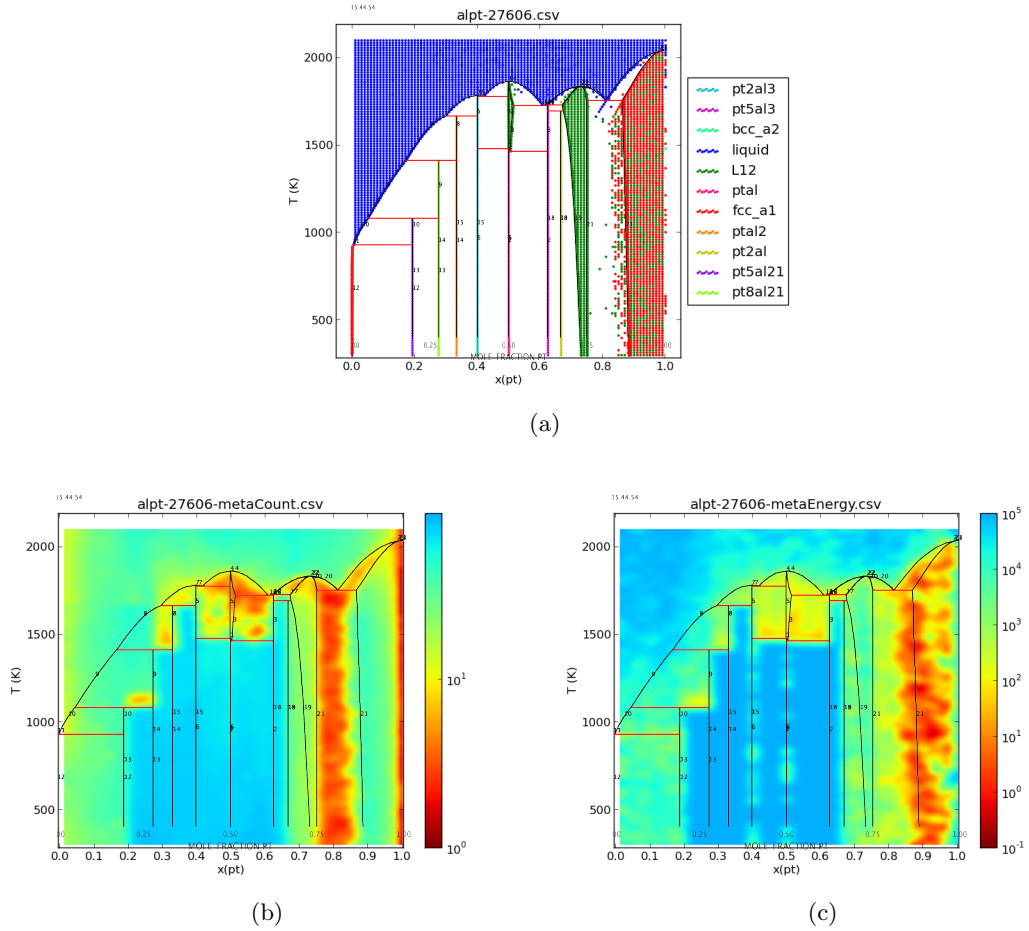


Figure 3.15: The Al-Pt system, produced with Model 2 and MINOS.

the energy difference between the lowest and second-lowest equilibria, and a histogram of the energy differences over the whole temperature and composition range.

Figure 3.21 and Figure 3.22 show results from Model 1, using SNOPT and MINOS respectively. Each of these phase diagrams agrees with other published phase diagrams for the Co-Mo system, and is clean except for a few stray points where a metastable equilibrium has been identified as stable. The stray points are a type of error inherent to creating a UQ diagram in the manner proposed here, where each point is examined in isolation and in this case exactly 100 trials are conducted. The “horn” typical in Co-Mo is visible in the phase

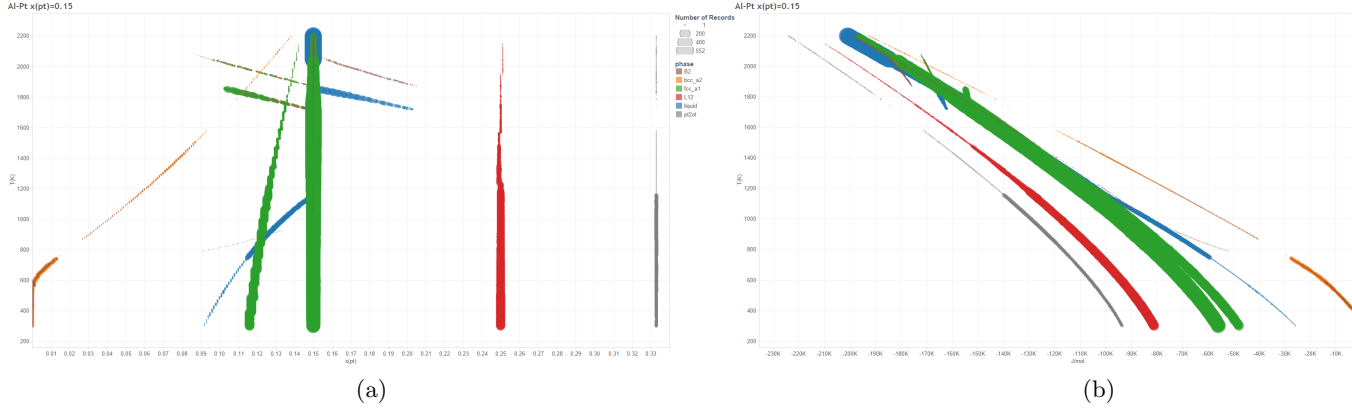


Figure 3.16: Two unusual complementary views of energy in the Al-Pt system at composition $x(\text{pt}) = 0.15$, produced with Model 1 using SNOPT. The top graph has vertical axis T , horizontal axis $x(\text{pt})$ of the phase itself in equilibrium, and the relative size of the line shows the frequency with which that phase is found in equilibrium. The bottom graph shows the same data set with T on the vertical axis, and energy in J/mol on the horizontal axis. Since the composition of the overall system is always the same, the phases in equilibrium must be inferred from the other information on the graph – this can be challenging but revealing.

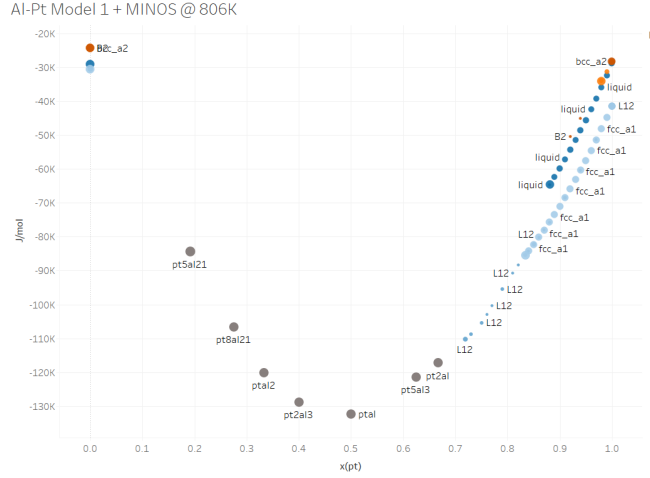
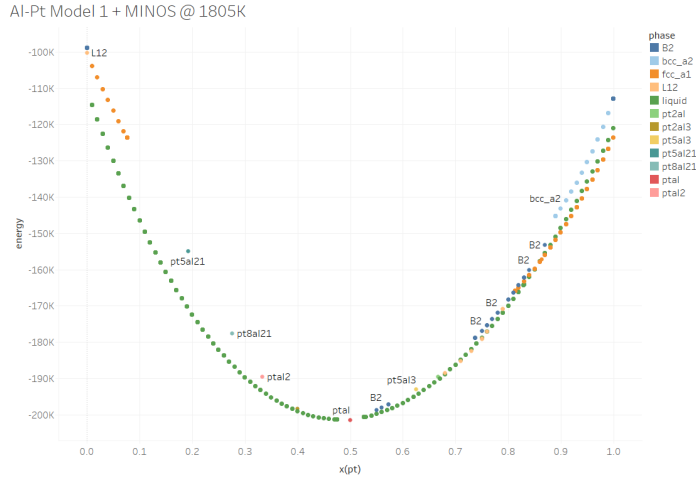
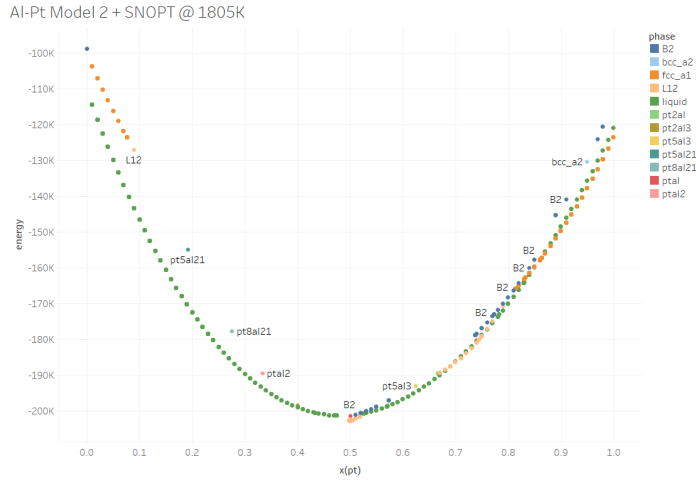


Figure 3.17: Energy and frequency of convergence for Al-Pt using Model 1 and MINOS at 806K. Smaller disks indicate fewer times converging to that phase in equilibrium.

diagram around $x(\text{mo}) \approx 0.05$ and $T \approx 1150$, demonstrating that the model has correctly spanned the miscibility gap in the system.



(a) Model 1 + MINOS



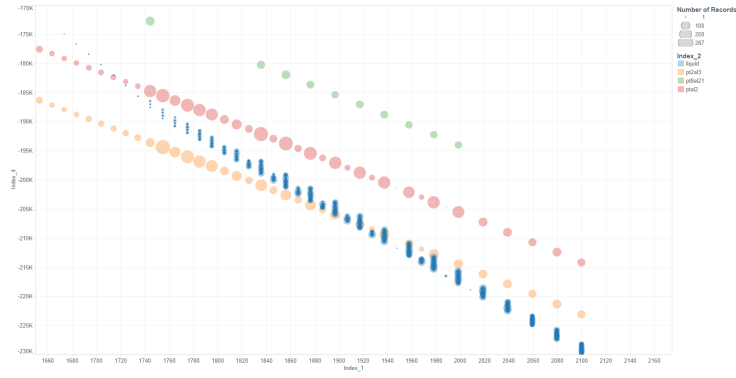
(b) Model 2 + SNOPT

Figure 3.18: The Al-Pt system at 1805K. These graphs show only equilibria the solver can converge to with that model, both stable and metastable points – unstable points are never identified. The presence of the L12 phase is apparent in the Model 2 + SNOPT graph, and absent in the Model 1 + MINOS graph, which gives insight into the differences in the corresponding phase diagrams.

3.4 Future Work

Optimization Techniques

There is a vast and growing body of optimization techniques available; some are recent



(a) Model 2 + SNOPT

Figure 3.19: This graph is “sideways”, with temperature on the x -axis, fixed at composition $x(\text{pt}) = 0.34$. It shows the uncertainty in the liquidus from the perspective of metastable equilibria of the liquid phase.

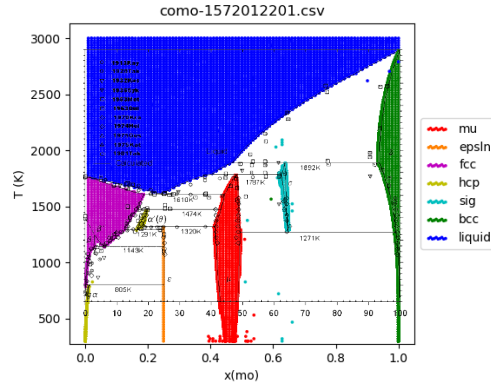


Figure 3.20: The Co-Mo system.

products of academic research, many are heuristics designed for industry, and a third level of “meta-heuristics” proposes to quickly identify techniques appropriate to a problem.

In our framework we use AMPL as the interface between the model and the optimization algorithms used to identify an equilibrium at a point in the composition and temperature space, and this allows us to leverage state of the art solvers which very rapidly converge to a feasible solution. However, the solutions returned by the solver are often not the strict

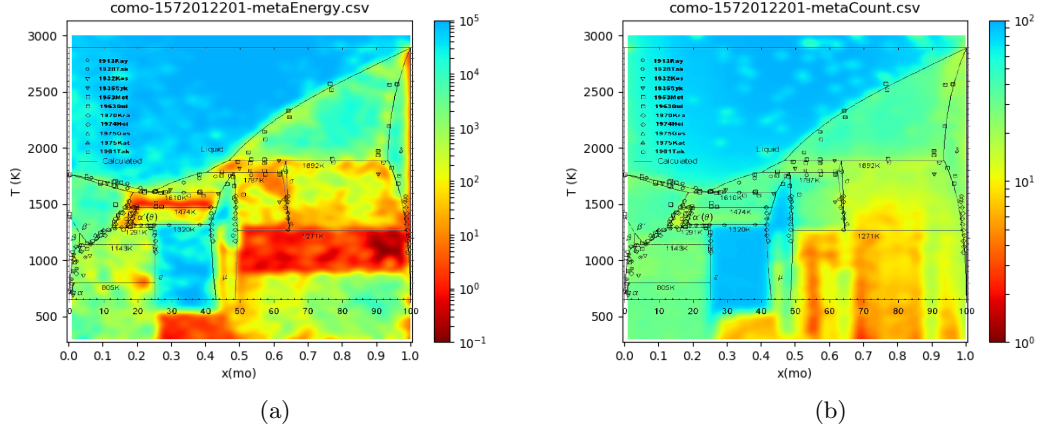


Figure 3.21: The Co-Mo system, produced with Model 1 and SNOPT.

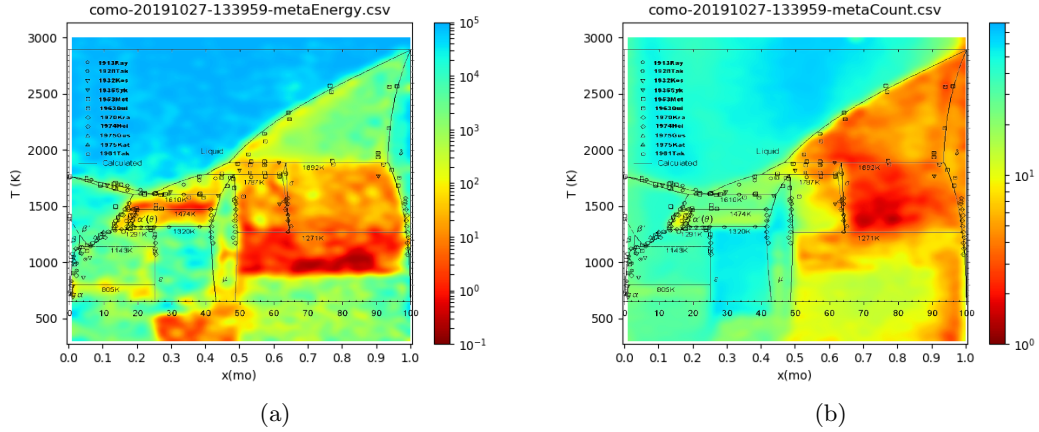
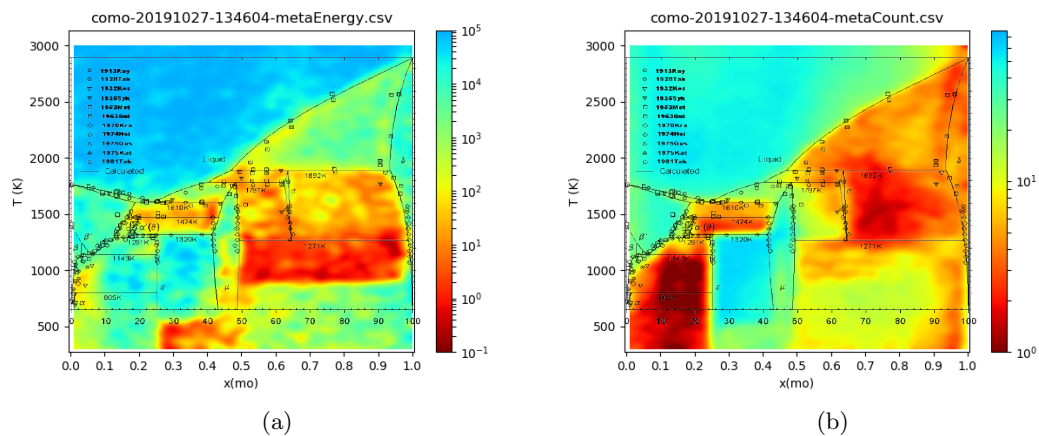
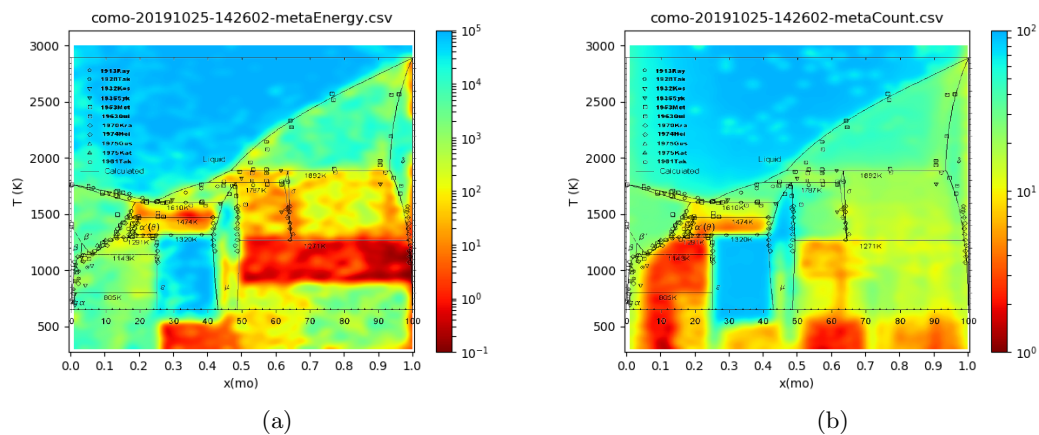


Figure 3.22: The Co-Mo system, produced with Model 1 and MINOS

minimizer, but rather represent a local solution. Our challenge is in identifying a means to more rapidly and with greater assurance identify the global minimizer, or at least an equilibrium very likely to be the global minimizer.

One approach we have already discussed is how we sample the space. By assuring effective coverage of the y space as initial conditions for a given composition we can increase the assurance that all equilibria have been identified, hence the global equilibrium has also been



identified. However when the space becomes large, having many degrees of freedom due to a large number of possible phases with internal y variables, we cannot effectively explore every region. The problem space can be very large: a 4-sublattice phase in a ternary system with vacancies is 81-dimensional, and a multicomponent system may comprise dozens of such phases. In these instances we must rely on techniques to direct sampling in a meaningful way rather than attempting to cover the whole space.

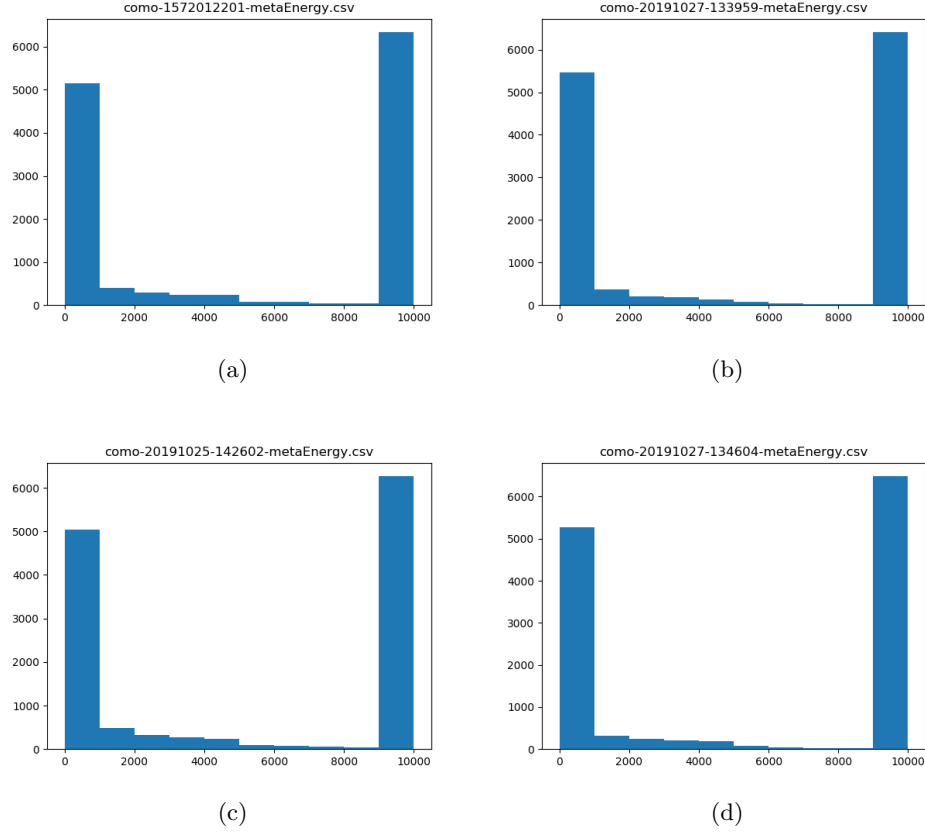


Figure 3.25: Meta-energy of the Co-Mo system, four histograms over the entire composition and temperature range. The first two are Model 1, the last two are Model 2. The first and third are SNOPT, the second and fourth are MINOS.

Each call to the solver and return to AMPL is relatively time-consuming, as the solver runs to convergence, requiring many evaluations of the energy function. An alternative is to evaluate the energy function in AMPL without calling the solver, and from those samples of the space pick one or a few as promising initial conditions to initiate the solver. An extension of this approach is to create a solver which applies this technique without returning to AMPL until it has reached some degree of assurance that it has the global equilibrium. By handling all of this within the solver, likely written in a faster low-level language such as C, it should execute much more quickly than the regular call and return between AMPL and solver. The development time necessary to develop, test, debug, and revise any new solver in a

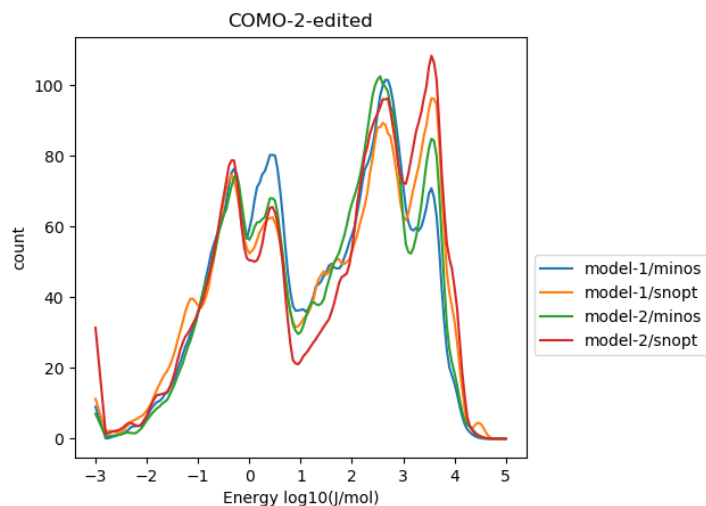


Figure 3.26: The Co-Mo system, four histograms over the entire composition and temperature range. The data is binned into bins with width 0.1 and then smoothed with a triangle distribution of width 0.4.

low-level language is, however, quite substantial and beyond the time budget of a single developer such as a PhD student. With that practical constraint in mind, we can test solver heuristics written in the AMPL language coupled with its state of the art solvers for final convergence with the desired numerical accuracy, and leave full development of a solver for later work. Here we need to identify specific directions of research without overcommitting to a development effort.

We have tested the simplest approach: sample the space a fixed number of times, using each of those samples as an initial condition for the solver, and take the lowest energy result as the final supposedly global equilibrium. A number of straightforward improvements can be made:

- Initially calculate the energy only within AMPL and use a fixed number of the lowest energies as initial conditions for the solver

Among the wide variety of modern global/stochastic optimization techniques/heuristics

which exist, a few appear promising for this framework: evolutionary, swarm, simulated annealing, and stochastic tunneling. Each of these techniques, if thoroughly tested on the CALPHAD problem, could produce a standalone peer reviewed paper.

Global Optimization

Global optimization, specifically in this case global minimization, is finding the minimum value over all possible configurations, in contrast to converging on a local minimum as a quasi-Newton or other nonlinear optimization method will do. A direct analytic solution cannot be applied to an optimization problem of this complexity, and numerical strategies are subject to substantial difficulties since a configuration space with even a small number of dimensions (degrees of freedom) rapidly becomes quite large in measure and demands significant computational expense to explore completely. Except in very small problems it is impossible to practically provide a mathematical guarantee that a minimizer is the global minimizer. Many so-called global optimization techniques use stochastic methods to achieve a certain desired probability that the identified point is the global minimizer, but do not offer absolute certainty.

The sampling methods mentioned above are an example of a direct Monte-Carlo method, a global optimization technique giving good but approximate answers. Since the CALPHAD problem is almost everywhere smooth these approximate answers can be used in a hybrid metaheuristic as initial conditions for a nonlinear method using the gradient, e.g., a quasi-Newton method, to converge on the precise minimizer. One modification of the Monte-Carlo approach in that hybrid context is to implement a tabu-type method [119] to prevent re-searching a previously explored part of the configuration space, although delineating tabu regions the non-discrete configuration space is a difficult challenge.

Often a heuristic method or a metaheuristic is applied to an optimization problem to provide greater assurance that the global minimum has been achieved, and to identify it more rapidly. Among the many machine learning techniques are evolutionary algorithms [120], and swarm-based algorithms [121] which are known to be resilient to problems with many

local minima, as well as a wide range of (meta-)heuristics such as simulated annealing [122] and stochastic tunneling [123] that attempt to explore the configuration space in a more intelligent or efficient way. Any of those methods may have some application to this problem.

The combination of a metaheuristic method and mathematical programming (i.e., optimization algorithms) has been given the label *matheuristics*, or *math-heuristics*, more clearly called *Model-Based Metaheuristics* [124]. With publications being made under those labels, it may provide a clearinghouse for the type of hybrid methods which would apply well to the CALPHAD problem.

Due to the superficial simplicity of the CALPHAD problem, it has not attracted much research from the optimization community, and little literature exists connecting even well-known optimization techniques to CALPHAD. Various approaches are possible which are not examined in the literature. For example, the problem can be explicitly modeled as a combinatorial one in the selection of stable phases, and mixed-integer methods [125] can be applied. This is “mixed integer non-linear programming” (MINLP) in contrast to the more frequently encountered “mixed integer linear programming” (MILP). Mixed-integer nonlinear methods include branch-and-bound, cutting plane, and outer approximation methods. In older CALPHAD literature the combinatorial approach is sometimes dismissed out of hand as impracticable, but modern optimization techniques may change this assessment. AMPL does include access to MINLP solvers which may be extremely efficient in solving a problem formulated in this way.

Inner and Outer Approximation are global optimization techniques for linear and convex problems, as well as certain very restricted classes of nonlinear problems [126]. An Outer-Inner Approximation is possible on problems separable into univariate functions [127], but it is not immediately apparent how this could be applied to the non-separable, non-convex CALPHAD problem.

Determination of Criteria for Phase Stability

As discussed in the section on the existence of KKT points and minimizers, a phase can be constructed so that it has no stable points anywhere. Such a phase is physically meaningless and should not be included in a database. A clear theoretical basis for determining whether selected parameters guarantee the existence of stationary and stable points for a phase and for a database would be meaningful and hold potential value to researchers who construct databases from empirical data.

3.5 Summary and Discussion

In this dissertation we have motivated the study of the CALPHAD problem from a modeling and optimization viewpoint. We have developed a new approach to the CALPHAD problem, fully developed the molar model applicable to every phase, proven its equivalence with the existing formula unit model, demonstrated the effectiveness of the approach in generating phase diagrams and energy plots, developed two new approaches to calculating and visualizing uncertainties in the CALPHAD method, providing a novel collection of methodologies and tools for researchers to investigate and apply the CALPHAD method.

In Chapter 2 we took a systematic approach to the modeling aspects in CALPHAD. We developed a set based formulation for modeling, and a complete computational framework for generating and plotting phase diagram data and investigating phase energies. The use of AMPL provides a connection between materials science modeling and a wide variety of state of the art optimization solvers. We proposed the use of a molar model for energy of each phase, in contrast to the functional unit energy, where energy is explicitly calculated per mole of constituents. This model was fully developed and the single unified model is used for every phase regardless the number of sublattices and constituents, and the presence or absence of magnetic component or disordered contribution. We proved the independence of the constraints in the models, the existence of a feasible point, KKT points, and minimizers. We proved the equivalence of the two models at equilibrium, with identity between KKT

points, objective values, and minimizers, by examining the Lagrangian, the Jacobian and its null space Z , and the reduced Hessian $Z^T \nabla^2 \mathcal{L} Z$.

We then examined the challenge of modeling vacancies while retaining consistency of solutions. We proposed a solution to the run-away vacancy problem of putting a constant constraint on the amount of vacancy in each phase, and demonstrated its effectiveness in producing consistent phase diagrams from challenging thermodynamic databases such as Al-Pt. Concluding Chapter 2, we discuss the implementation of the framework and some of the challenges in correct interpretation of a database.

In Chapter 3 we discuss approaches to optimization, sampling, and two novel methodologies for uncertainty quantification. We developed a special-purpose solver separating steps in the Y and \mathbf{f} variables. We apply logarithmic projection and low-discrepancy sampling to ensure complete and uniform coverage of the feasible space. We provide several views of the energy of phases at fixed temperatures. We develop two metrics of uncertainty in a phase diagram: a measure of the energy difference between the global (stable) equilibrium and the lowest local (metastable) equilibrium; and a measure of the fraction of the composition space which converges locally to the global equilibrium. These two metrics are presented in heat maps overlaying the phase diagram, and provide the researcher with a direct indication of regions where the solution presented by a diagram may be subject to uncertainty. We create histograms of the uncertainty values in a diagram, and by comparing the histograms generated when selecting differing solvers and models we provide a comparison not conceivable in any other thermodynamic framework.

For future application of the framework and models we have developed, there are a number of lines of research possible. Development of more sophisticated mapping algorithms that distribute information between temperature and composition regions to identify global equilibria more rapidly across the diagram, and automating the number of initial conditions tested based on the difficulty in identifying global equilibrium will increase reliability and ease of use of the framework. A variety of optimization techniques exist which more

broadly attempt to find the global equilibrium, and many of these could be tested on the model, such as evolutionary, swarm, simulated annealing, and stochastic tunneling. Additionally, meta-heuristic techniques may be usefully tested on this problem. Application of the variable value of G_{VaVa} proposed by Ågren and Hillert is a possibility unique within the framework we have developed and of potential wide interest to the CALPHAD community. Development of methods for automatically determining appropriate values for the vacancy constraint would be a direct improvement to the model. Characterizing the criteria that parameters for a phase in a thermodynamic database must meet to have a guaranteed bounded equilibrium, especially in the presence of modeling vacancies, would be a benefit to researchers fitting those parameters.

Chapter A: AMPL model

```
1 # General standards used in writing and maintaining this file:
2 #
3 # # Naming of sets, parameters, variables:
4 # - all set/param/var names are descriptive so the code is easy
   to read
5 # - set/param/var names are generally IndexContentContent --
   e.g., PhaseSiteElement is indexed by Phase and contains (Site,
   Element)
6 # - sets start with a captial letter
7 # - parameters and variables start with a lower case letter
8 # - every word after the first in a set/param/var name is
   capitalized
9 # - no special characters are used in set/param/var names (i.e.,
   no _ or -)
10 # - dummy variables are lower case (e.g., "for {phase in Phase}
   {..." uses dummy variable 'phase')
11 # - exceptions to capitalization are single letter parameters or
   variables
12 # - single letter dummy variables are avoided
13 #
14 # # Sets:
15 # - wherever possible a set is constructed within ampl instead
   of by the file converter
16 #
```

```

17
18 model;
19
20 option abs_boundtol 1e-14; # just trying to be reasonable
21 option rel_boundtol 1e-8; # we'd like 6 decimal places at the
    most
22 option show_boundtol 0; # silence numerous messages about
    boundtol affecting deduced dual values
23
24 # ideal gas constant [Joule/(mole*Kelvin)]
25 param R := 8.3144598;
26
27 # Temperature
28 param T;
29
30 # Pressure
31 param P default 100;
32
33 # set of all phases
34 set Phases;
35
36 # set of all sublattices in each phase
37 set PhaseSublattice{Phases};
38
39 # set of all sublattices
40 set AllSublattices = union {phase in Phases} PhaseSublattice[
    phase];
41

```

```

42 # set of all sites and elements in each sublattice
43 # multiple elements at one site indicates mixing
44 set SublatticeSiteElement{AllSublattices} dimen 2;
45
46 # set of all elements at each site in a phase
47 set PhaseSiteElement {phase in Phases} = union {subl in
    PhaseSublattice[phase]} SublatticeSiteElement[subl];
48
49 # set of elements in each phase
50 set PhaseElementsVa {phase in Phases} = (setof {subl in
    PhaseSublattice[phase], (site,elem) in SublatticeSiteElement[
    subl]} elem) union {indexElem};
51 set PhaseElements {phase in Phases} = PhaseElementsVa[phase] diff
    {'Va'};
52
53 # instances for each phase (to handle miscibility gap)
54 set PhaseInstances{phase in Phases} = 1..card(PhaseElementsVa[
    phase]);
55
56 # sets of elements -- one with Va (if it's in the TDB file), one
    without.
57 set ElementsVa ordered; # = union {phase in Phases}
    PhaseElementsVa[phase];
58 set Elements ordered by ElementsVa = ElementsVa diff {'Va'};
59
60 # SER for each element
61 param ElementSER {ElementsVa} symbolic;
62

```

```

63 # integer number of sites for each phase
64 param nSites {phase in Phases} = max {subl in PhaseSublattice[
    phase], (site,elem) in SublatticeSiteElement[subl]} site;
65
66 # maximum number of sites over all phases
67 param maxSites = max {phase in Phases} nSites[phase];
68
69 # site fractions
70 param a{Phases,1..maxSites};
71
72 set PhaseDisorder dimen 2;
73
74 set phaseMagneticParam dimen 3; # defined in ..._data.dat
75
76 set PhasesMagnetic = setof {(phase,AF,SF) in phaseMagneticParam}
    phase;
77
78 # the Structure Factor is variable 'p' in Xiong's 2012 paper --
    we use SF here
79 # in the absence of a clear name we use AF for the second
    coefficient, meaning Anti-Ferromagnetic
80 # in the following lines 'max' is irrelevant since there should
    be only one member matching 'phase', but it is required to
    extract a single member of the set
81 param phaseMagneticSF {phase in PhasesMagnetic} = max {(phase,AF,
    SF) in phaseMagneticParam} SF;
82 param phaseMagneticAF {phase in PhasesMagnetic} = max {(phase,AF,
    SF) in phaseMagneticParam} AF;

```

```

83
84 set PhaseSublatticeMagnetic {PhasesMagnetic} dimen 3; # defined
    in ..._data.dat
85
86 set PhaseMagneticSublattices {phase in PhasesMagnetic} = setof {(
    subl, Curie, Bohr) in PhaseSublatticeMagnetic[phase]} subl;
87
88 param phaseSublatticeCurie {phase in PhasesMagnetic, subl in
    PhaseMagneticSublattices[phase]} = max {(subl, Curie, Bohr) in
    PhaseSublatticeMagnetic[phase]} Curie;
89 param phaseSublatticeBohr {phase in PhasesMagnetic, subl in
    PhaseMagneticSublattices[phase]} = max {(subl, Curie, Bohr) in
    PhaseSublatticeMagnetic[phase]} Bohr;
90
91
92 set PhaseSublatticeOrder {Phases, AllSublattices} default {0};
93 param maxOrder = max {phase in Phases, subl in PhaseSublattice[
    phase], order in PhaseSublatticeOrder[phase, subl]} order;
94
95 param gibbs {Phases, AllSublattices, 0..maxOrder};
96
97 # set of sites in each sublattice
98 set SublatticeSite {subl in AllSublattices} = setof {(site, elem)
    in SublatticeSiteElement[subl]} site;
99
100 # all we care about are the sites where exactly 2 elements mix

```

```

101 set SublatticeMixSite {subl in AllSublattices} = setof {site in
    SublatticeSite[subl] : card( {(site,elem) in
    SublatticeSiteElement[subl]} ) == 2} site;
102 set SublatticeSiteMixElem {subl in AllSublattices, site in
    SublatticeMixSite[subl]} ordered = setof {(site,elem) in
    SublatticeSiteElement[subl]} elem;
103 set SublatticeMixSiteElement {subl in AllSublattices} dimen 3 =
104     setof {site in SublatticeMixSite[subl]} (site,first(
        SublatticeSiteMixElem[subl,site]),last(
        SublatticeSiteMixElem[subl,site]));
105
106 # set of sites with symmetric sublattice
107 set PhaseSymmetric;
108
109 data (dataFile);
110
111 model;
112
113 ##### parsing sublattices with AMPL instead of converter
114
115 set SublatticeElement {AllSublattices};
116 set SublatticeComp {AllSublattices} dimen 2;
117 param sublatticeSymmetries {AllSublattices};
118
119 # increase to whatever value you need.
120 param factMax := 12;
121 param factorial {0..factMax};
122 let factorial[0] := 1;

```



```

123 for {n in 1..factMax} {
124     let factorial[n] := n * factorial[n-1];
125 }
126
127 param whatsleft symbolic;
128 param thisPiece symbolic;
129 param thisSite;
130 param numComp;
131 param numSym;
132 param numSites;
133 param subpiece symbolic;
134 for {subl in AllSublattices} {
135     let SublatticeElement[subl] := {};
136     let SublatticeComp[subl] := {};
137     let SublatticeSiteElement[subl] := {};
138
139     # This is an implementation of "split(':', ...)", which does
140     # Less lengthy methods may be possible, feel free to
141     # implement a better one.
142     let thisSite := 1;
143     let whatsleft := subl;
144     repeat {
145         # This is an implementation of "split(',', ...)", which
146         # does not exist in AMPL.
147         let thisPiece := sub(whatsleft, ".*$", "");

```

```

147     let SublatticeComp[subl] := SublatticeComp[subl] union { (
        thisSite,thisPiece));
148 repeat {
149     let subpiece := sub(thisPiece, ",.*$", "");
150     let SublatticeSiteElement[subl] :=
        SublatticeSiteElement[subl] union {(thisSite,
        subpiece)};
151     let SublatticeElement[subl] := SublatticeElement[subl
        ] union {subpiece};
152
153     if (match(thisPiece, ",") == 0) then
154         break;
155
156     let thisPiece := sub(thisPiece, "^[^,]*", "");
157 }
158
159 if (match(whatsleft, ":") == 0) then
160     break;
161
162 let whatsleft := sub(whatsleft, "^[^:]*:", "");
163 let thisSite := thisSite + 1;
164 }
165
166 # Here we count the symmetries of each sublattice: (not used
    for anything right now)
167 # symmetries = N! / (a!b!c!d!), where a,b,c,d are the number
    of each component

```

```

168   let numSites := card(SublatticeComp[subl]); # number of sites
      in sublattice
169   let numSym := factorial[numSites];
170   for {component in union {(site,comp) in SublatticeComp[subl]}
      {comp}} {
171     let numComp := card( {(site,component) in SublatticeComp[
      subl]} );
172     let numSym := numSym / factorial[numComp];
173   }
174   let sublatticeSymmetries[subl] := numSym;
175 }
176
177 #display SublatticeElement, SublatticeComp, SublatticeSiteElement
      , sublatticeSymmetries;
178
179
180 # glue function to avoid log of nonpositive values
181 param lowerBound := 1e-8;
182 param a_glue := 1/(2*lowerBound);
183 param b_glue := log(lowerBound) + 1 - 2*a_glue*lowerBound;
184 param c_glue := lowerBound*log(lowerBound) - a_glue*(lowerBound
      ^2) - b_glue*lowerBound;
185
186 param A {phase in Phases};
187 let {phase in Phases} A[phase] := sum {i in 1..nSites[phase]} a[
      phase,i];
188

```

```

189 var y{phase in Phases, (site,elem) in PhaseSiteElement[phase],
    inst in PhaseInstances[phase]} := 1/(card(Phases)*maxSites*
    card(Elements));
190
191 var phaseNonVacancy {phase in Phases, inst in PhaseInstances[
    phase]}
192     = (sum {(site,elem) in PhaseSiteElement[phase] : elem not in
        {'Va'}} (a[phase,site] * y[phase,site,elem,inst])) / A[
        phase];
193
194 var f{phase in Phases, inst in PhaseInstances[phase]} := 1/card(
    Phases);
195
196 # x:
197 var x {phase in Phases, elem in PhaseElementsVa[phase], inst in
    PhaseInstances[phase]}
198     = ((sum {(site,elem) in PhaseSiteElement[phase]} (a[phase,
        site]*y[phase,site,elem,inst]))/A[phase]) /
        phaseNonVacancy[phase,inst];
199
200 # xi:
201 var xi {phase in Phases, elem in PhaseElementsVa[phase], inst in
    PhaseInstances[phase]}
202     = (sum {(site,elem) in PhaseSiteElement[phase]} (a[phase,site
        ]*y[phase,site,elem,inst]))/A[phase];
203
204 param F_0 {elem in Elements} in [0,1];
205

```

```

206 ### these variables are separate just for legibility
207
208 var phaseEntropy {phase in Phases, inst in PhaseInstances[phase]}
    =
209     #(1/(phaseNonVacancy[phase,inst])) *
210     R * T * sum {(site,elem) in PhaseSiteElement[phase]}
211         a[phase,site]
212         * ( if (y[phase,site,elem,inst] > lowerBound)
213             then
214                 y[phase,site,elem,inst]*log(y[phase,site,elem
215                     ,inst])
216             else
217                 a_glue*y[phase,site,elem,inst]^2 + b_glue*y[
218                     phase,site,elem,inst] + c_glue
219             );
220
221 var phaseEnthalpy {phase in Phases, inst in PhaseInstances[phase
222     ]} =
223     #(1/(phaseNonVacancy[phase,inst])) *
224     sum {subl in PhaseSublattice[phase], order in
225         PhaseSublatticeOrder[phase,subl]}
226         gibbs[phase,subl,order]
227         * ( prod {(site,elem) in SublatticeSiteElement[subl]} y[
228             phase,site,elem,inst] )
229         * ( if (card(SublatticeMixSiteElement[subl]) == 0)
230             then
231                 1

```

```

228         else
229             ( sum {(mixSite,elem0,elem1) in
                    SublatticeMixSiteElement[subl]} (y[phase,
                    mixSite,elem0,inst]-y[phase,mixSite,elem1,inst
                    ]) ) ^order
230         )
231     ;
232
233 # Disordered Contribution: G = G_dis(x) + G_ord(y) - G_ord(y=x)
    => phaseDisorder = G_dis(x) - G_ord(y=x)
234 # there will only be one disPhase -- if there are none, this
    block is 0
235 var phaseDisorderDelta {phase in Phases, inst in PhaseInstances[
    phase]}
236     = sum {(plusMinus,thePhase) in union {(phase,disPhase) in
        PhaseDisorder} {(-1,phase), (1,disPhase)} }
237     plusMinus *
238     (      R*T* sum {(site,elem) in PhaseSiteElement[thePhase]}
239         ( a[thePhase,site] *
240             (if (xi[phase,elem,inst] > lowerBound)
241                 then
242                     xi[phase,elem,inst]*log(xi[phase,elem,
                        inst])
243                 else
244                     a_glue*(xi[phase,elem,inst])^2 + b_glue*
                        xi[phase,elem,inst] + c_glue
245             )
246         )

```

```

247         + sum {subl in PhaseSublattice[thePhase], order in
           PhaseSublatticeOrder[thePhase,subl]}
248         ( gibbs[thePhase,subl,order]
249           * ( prod {(site,elem) in SublatticeSiteElement[
               subl]} xi[phase,elem,inst] )
250           * ( if (card(SublatticeMixSiteElement[subl]) ==
               0)
251               then
252                   1
253               else
254                   ( sum {(mixSite,elem0,elem1) in
                       SublatticeMixSiteElement[subl]}
255                     (xi[phase,elem0,inst]-xi[phase,
                       elem1,inst]) )^order
256               )
257           )
258     );
259
260     ## Magnetic Contribution, largely from Xiong 2012
261
262     var magneticCurie {phase in PhasesMagnetic, inst in
       PhaseInstances[phase]} =
263         sum {subl in PhaseMagneticSublattices[phase]}
           phaseSublatticeCurie[phase,subl] * prod {elem in
             SublatticeElement[subl]} x[phase,elem,inst];
264

```

```

265 var magneticAbsCurie {phase in PhasesMagnetic, inst in
    PhaseInstances[phase]} = if (magneticCurie[phase,inst] > 0)
    then magneticCurie[phase,inst]
266     else abs(magneticCurie[phase,inst] / phaseMagneticAF[phase]);
267
268 var magneticBeta {phase in PhasesMagnetic, inst in PhaseInstances
    [phase]} =
269     sum {subl in PhaseMagneticSublattices[phase]}
        phaseSublatticeBohr[phase,subl] * prod {elem in
            SublatticeElement[subl]} x[phase,elem,inst];
270
271 var magneticAbsBeta {phase in PhasesMagnetic, inst in
    PhaseInstances[phase]} = if (magneticBeta[phase,inst] > 0)
    then magneticBeta[phase,inst]
272     else abs(magneticBeta[phase,inst] / phaseMagneticAF[phase]);
273
274 var magneticTau {phase in PhasesMagnetic, inst in PhaseInstances[
    phase]} = if (magneticAbsCurie[phase,inst] <> 0) then T /
    magneticAbsCurie[phase,inst] else Infinity;
275
276 param magneticA {phase in PhasesMagnetic} = 518/1125 +
    11692/15975 * (1/phaseMagneticSF[phase] - 1);
277
278 var magneticWeight {phase in PhasesMagnetic, inst in
    PhaseInstances[phase]} = if (magneticTau[phase,inst] ==
    Infinity) then 0
279     else if (magneticTau[phase,inst] < 1) then

```



```

280      (1 - (1/magneticA[phase]) * (79 * magneticTau[phase,inst
      ]^(-1) / (140 * phaseMagneticSF[phase]) + 474/497 *
      (1/phaseMagneticSF[phase]-1)
281      * (magneticTau[phase,inst]^3 / 6 + magneticTau[phase,
      inst]^9 / 135 + magneticTau[phase,inst]^15 / 600))
      )
282  else
283      (-(1/magneticA[phase]) * (1/10 * magneticTau[phase,inst
      ]^(-5) + 1/315 * magneticTau[phase,inst]^(-15) +
      1/1500 * magneticTau[phase,inst]^(-25) ));
284
285  var phaseMagnetic {phase in Phases, inst in PhaseInstances[phase
  ]} = if (phase not in PhasesMagnetic) then 0 else
286      R * T * log(magneticAbsBeta[phase,inst] + 1) * magneticWeight
      [phase,inst];
287
288
289  var phaseGibbs{phase in Phases, inst in PhaseInstances[phase]} =
290      (
291          phaseEntropy[phase,inst]
292          + phaseEnthalpy[phase,inst]
293          + phaseDisorderDelta[phase,inst]
294          + phaseMagnetic[phase,inst]
295      ) / (A[phase] * phaseNonVacancy[phase,inst]); # OPTION --
      multiply times phaseNonVacancy for Model 2, do not
      multiply for Model 1
296
297  var cons_comp {elem in Elements : elem != 'Va'} =

```

```

298     sum{phase in Phases, inst in PhaseInstances[phase] : elem in
        PhaseElements[phase]} f[phase,inst]*x[phase,elem,inst]; #
        OPTION xi (Model 1) or x (Model 2)
299
300 ##### Objective Function
301
302 minimize Energy:
303     sum{phase in Phases, inst in PhaseInstances[phase]} (f[phase,
        inst] * phaseGibbs[phase,inst]);
304
305 ##### Constraints
306
307 subject to Non_Vacant_Phases {phase in Phases, inst in
        PhaseInstances[phase]}:
308     phaseNonVacancy[phase, inst] >= minNonVacancy; # arbitrary,
        but ensures continuity in the definition of x -- when this
        is active at equilibrium we should discard the point
309
310 subject to yI0 {phase in Phases, (site,elem) in PhaseSiteElement[
        phase], inst in PhaseInstances[phase]}:
311     y[phase,site,elem,inst] >= 0;
312
313 subject to yI1 {phase in Phases, (site,elem) in PhaseSiteElement[
        phase], inst in PhaseInstances[phase]}:
314     y[phase,site,elem,inst] <= 1;
315
316 subject to fI0 {phase in Phases, inst in PhaseInstances[phase]}:
317     f[phase,inst] >= 0;

```

```

318
319 subject to fI1 {phase in Phases, inst in PhaseInstances[phase]}:
320     f[phase,inst] <= 1;
321
322 subject to Conservation_Components {elem in Elements : elem != '
    Va'}:
323     cons_comp[elem] == F_0[elem];
324
325 subject to Conservation_Sublattice {phase in Phases, site in 1
    ..nSites[phase], inst in PhaseInstances[phase]}:
326     (sum {(site,elem) in PhaseSiteElement[phase]} y[phase,site,
        elem,inst]) = 1;
327
328 option substout 1;
329 option presolve 0;
330
331 # For Debugging
332 #     for {phase in Phases} {
333 #         for {subl in PhaseSublattice[phase]} {
334 #             print subl & " order " & PhaseSublatticeOrder[
                phase,subl];
335 #             for {(site,elem0,elem1) in
                SublatticeMixSiteElement[subl]} {
336 #                 print site & " " & elem0 & " " & elem1;
337 #             }
338 #         }
339 #     }

```

Bibliography

- [1] J. Snider, I. Griva, S. Xiaodi, and M. Emelianenko, “Set based framework for gibbs energy minimization,” *CALPHAD*, vol. 48, pp. 18–26, March 2015. [Online]. Available: <http://dx.doi.org/10.1016/j.calphad.2014.09.005>
- [2] N. Dupin and B. Sundman, “On the sublattice formalism applied to the b2 phase,” *Zeitschrift fur Metallkunde*, vol. 90, pp. 76–85, 1999. [Online]. Available: <https://www.tib.eu/de/suchen/id/BLSE%3ARN055916172/On-the-Sublattice-Formalism-Applied-to-the-B2-Phase/>
- [3] L. Kaufman and H. Bernstein, *Computer Calculation of Phase Diagrams with Special Reference to Refractory Metals*, 1970.
- [4] G. Inden, “Approximate description of the configurational specific heat during a magnetic order-disorder transformation,” *Proceedings of CALPHAD V*, 1976.
- [5] M. Hillert and M. Jarl, “A model for alloying in ferromagnetic metals,” *CALPHAD*, vol. 2, pp. 227–238, 1978. [Online]. Available: [http://dx.doi.org/10.1016/0364-5916\(78\)90011-1](http://dx.doi.org/10.1016/0364-5916(78)90011-1)
- [6] A. Dinsdale, “Sgte data for pure elements,” *CALPHAD*, vol. 15, pp. 317–425, 1991. [Online]. Available: [http://dx.doi.org/10.1016/0364-5916\(91\)90030-N](http://dx.doi.org/10.1016/0364-5916(91)90030-N)
- [7] L. Kaufman, “Calphad journal,” 1977.
- [8] G. Olson and C. Keuhmann, “Materials genomics: From calphad to flight,” *Scripta Materialia*, vol. 70, pp. 25–30, 2014. [Online]. Available: <http://dx.doi.org/10.1016/j.scriptamat.2013.08.032>
- [9] G. Olson, “Genomic materials design: The ferrous frontier,” *Acta Materialia*, vol. 61, pp. 771–781, 2013. [Online]. Available: <http://dx.doi.org/10.1016/j.actamat.2012.10.04>
- [10] R. Reed, T. Tao, and N. Warnken, “Alloys-by-design: Application to nickel-based single crystal superalloys,” *Acta Materialia*, vol. 57, no. 19, pp. 5898–5913, 2009. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1359645409005187>

- [11] W. Xiong, M. Selleby, and Q. Chen, "Phase Equilibria and Thermodynamic Properties in the Fe–Cr System," *Critical Reviews in Solid State and Materials Science*, vol. 35, pp. 125–152, 2010. [Online]. Available: <http://dx.doi.org/10.1080/10408431003788472>
- [12] H. Lukas, E. Henig, and B. Zimmerman, "Optimization of phase diagrams by a least squares method using simultaneously different types of data," *CALPHAD*, vol. 1, pp. 225–236, 1977.
- [13] C. Campbell, U. Kattner, and Z. Liu, "The development of phase-based property data using the calphad method and infrastructure needs," *IMMI*, vol. 3, 2014. [Online]. Available: <http://immijournal.springeropen.com/articles/10.1186/2193-9772-3-12>
- [14] Y.-B. Kang, C. Aliravci, P. J. Spencer, G. Eriksson, C. D. Fuerst, P. Chartrand, and A. D. Pelton, "Thermodynamic and volumetric databases and software for magnesium alloys," *JOM*, vol. 61, no. 5, pp. 75–82, 2009. [Online]. Available: <http://dx.doi.org/10.1007/s11837-009-0076-9>
- [15] L. Kaufman and J. Ågren, "Calphad, first and second generation – birth of the materials genome," *Scripta Materialia*, vol. 70, pp. 3–6, January 2014. [Online]. Available: <http://dx.doi.org/10.1016/j.scriptamat.2012.12.003>
- [16] T.-C. S. AB, *Thermo-Calc Database Guide*, http://www3.thermocalc.se/res/Manuals/TC_databaseguide.pdf, Norra Stationsgatan 93, 5 tr. SE-113 64 Stockholm, Sweden, 2010.
- [17] C. Becker, J. Ågren, M. Baricco, Q. Chen, S. A. Decterov, U. R. Kattner, J. H. Perepezko, G. R. Pottlacher, and M. Selleby, "Thermodynamic modelling of liquids: Calphad approaches and contributions from statistical physics," *Physica Status Solidi (b)*, vol. 251, pp. 33–52, January 2014. [Online]. Available: <http://onlinelibrary.wiley.com/doi/10.1002/pssb.201350149/abstract>
- [18] U. Kattner, "The calphad method and its role in material and process development," *Tecnologia en Metalurgia, Materiais e Mineracao*, 2016. [Online]. Available: <http://dx.doi.org/10.4322/2176-1523.1059>
- [19] G. Olson, "Preface to the viewpoint set on: The materials genome," *Scripta Materialia*, vol. 70, pp. 1–2, 2014. [Online]. Available: <http://dx.doi.org/10.1016/j.scriptamat.2013.09.013>
- [20] P. Spencer, "A brief history of calphad," *CALPHAD*, vol. 32, pp. 1–8, March 2008. [Online]. Available: <http://dx.doi.org/10.1016/j.calphad.2007.10.001>
- [21] W. Xiong, Q. Chen, P. Korzhavyi, and M. Selleby, "An improved magnetic model for thermodynamic modeling," *CALPHAD*, vol. 39, pp. 11–20, December 2012. [Online]. Available: <http://dx.doi.org/10.1016/j.calphad.2012.07.002>
- [22] A. Miodownik, "The calculation of magnetic contributions to phase stability," *Calphad*, vol. 1, no. 2, pp. 133–158, 1977. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/0364591677900141>
- [23] W. Xiong, H. Zhang, L. Vitos, and M. Selleby, "Magnetic phase diagram of the

- Fe–Ni system,” *Acta Materialia*, vol. 59, pp. 521–530, 2011. [Online]. Available: <http://dx.doi.org/10.1016/j.actamat.2010.09.055>
- [24] F. Körmann, A. A. H. Breidi, S. L. Dudarev, N. Dupin, G. Ghosh, T. Hickel, P. Korzhavyi, J. A. Muñoz, and I. Ohnuma, “Lambda transitions in materials science: Recent advances in calphad and first-principles modelling,” *Physica Status Solidi (b)*, vol. 251, pp. 53–80, January 2014. [Online]. Available: <http://onlinelibrary.wiley.com/doi/10.1002/pssb.201350136/abstract>
- [25] T. Hickel, U. Kattner, and S. Fries, “Computational thermodynamics: Recent developments and future potential and prospects,” *physica status solidi (b)*, vol. 251, pp. 9–13, 2014. [Online]. Available: <http://onlinelibrary.wiley.com/doi/10.1002/pssb.201470107/abstract>
- [26] F. Körmann, B. Grabowski, P. Söderlind, M. Palumbo, S. Fries, T. Hickel, and J. Neugebauer, “Thermodynamic modeling of chromium: strong and weak magnetic coupling,” *Journal of Physics: Condensed Matter*, vol. 25, no. 42, p. 425401, 2013. [Online]. Available: <http://dx.doi.org/10.1088/0953-8984/25/42/425401>
- [27] C. Zener, *JOM*, vol. 7, pp. 619–630, 1955.
- [28] T. Cool, A. Bartol, M. Kasenga, and R. García, “Gibbs: Phase equilibria and symbolic computation of thermodynamic properties,” *CALPHAD*, vol. 34, pp. 393–404, December 2010. [Online]. Available: <http://dx.doi.org/10.1016/j.calphad.2010.07.005>
- [29] B. Sundman, I. Ohnuma, N. Dupin, U. Kattner, and S. Fries, “An assessment of the entire al-fe system including d03 ordering,” *Acta Materialia*, vol. 57, pp. 18–26, June 2009. [Online]. Available: <http://dx.doi.org/10.1016/j.actamat.2009.02.046>
- [30] A. Gheribi, J. Rogez, and J. Mathieu, “Magnetic contribution to the gibbs energy of elements versus temperature and pressure,” *Journal of physics and chemistry of solids*, vol. 67, pp. 1719–1723, August 2006. [Online]. Available: <http://dx.doi.org/10.1016/j.jpcs.2006.03.019>
- [31] U. Gov”, “Materials genome initiative,” 2011. [Online]. Available: <https://www.mgi.gov/>
- [32] “First-principles calculations and calphad modeling of thermodynamics,” *Journal of Phase Equilibria and Diffusion*, vol. 30, no. 5, pp. 517–534, 2009. [Online]. Available: <http://dx.doi.org/10.1007/s11669-009-9570-6>
- [33] V. Ozolins, “First-principles calculations of free energies of unstable phases: The case of fcc w,” *Phys. Rev. Lett.*, vol. 102, p. 065702, Feb 2009. [Online]. Available: <http://link.aps.org/doi/10.1103/PhysRevLett.102.065702>
- [34] Y. Wang, S. Curtarolo, C. Jiang, R. Arroyave, T. Wang, G. Ceder, L.-Q. Chen, and Z.-K. Liu, “Ab initio lattice stability in comparison with calphad lattice stability,” *CALPHAD*, vol. 28, pp. 79–90, March 2004. [Online]. Available: <http://dx.doi.org/10.1016/j.calphad.2004.05.002>
- [35] A. Breidi, S. Fries, M. Palumbo, and A. Ruban, “First-principles modeling of energetic and mechanical properties of ni-cr, ni-re and cr-re random alloys,”

- Computational Materials Science*, vol. 117, pp. 45 – 53, 2016. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0927025616000264>
- [36] S. Ponc, G. Antonius, P. Boulanger, E. Cannuccia, A. Marini, M. Ct, and X. Gonze, “Verification of first-principles codes: Comparison of total energies, phonon frequencies, electron-phonon coupling and zero-point motion correction to the gap between {ABINIT} and qe/yambo,” *Computational Materials Science*, vol. 83, pp. 341 – 348, 2014. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0927025613007143>
 - [37] P. E. Blöchl, “Projector augmented-wave method,” *Phys. Rev. B*, vol. 50, pp. 17 953–17 979, Dec 1994. [Online]. Available: <http://link.aps.org/doi/10.1103/PhysRevB.50.17953>
 - [38] G. Kresse and J. Furthmüller, “Efficient iterative schemes for *ab initio* total-energy calculations using a plane-wave basis set,” *Phys. Rev. B*, vol. 54, pp. 11 169–11 186, Oct 1996. [Online]. Available: <http://link.aps.org/doi/10.1103/PhysRevB.54.11169>
 - [39] “List of quantum chemistry and solid-state physics software,” https://en.wikipedia.org/wiki/List_of_quantum_chemistry_and_solid-state_physics_software, June 2016.
 - [40] M. Palumbo, S. G. Fries, A. D. Corso, F. Körmann, T. Hickel, and J. Neugebauer, “Reliability evaluation of thermophysical properties from first-principles calculations,” *Journal of Physics: Condensed Matter*, vol. 26, no. 33, pp. 335–401, 2014. [Online]. Available: <http://stacks.iop.org/0953-8984/26/i=33/a=335401>
 - [41] T. Hickel, B. Grabowski, F. Krmann, and J. Neugebauer, “Advancing density functional theory to finite temperatures: methods and applications in steel design,” *Journal of Physics: Condensed Matter*, vol. 24, no. 5, p. 053202, 2012. [Online]. Available: <http://stacks.iop.org/0953-8984/24/i=5/a=053202>
 - [42] B. Grabowski, P. Söderlind, T. Hickel, and J. Neugebauer, “Temperature-driven phase transitions from first principles including all relevant excitations: The fcc-to-bcc transition in ca,” *Phys. Rev. B*, vol. 84, p. 214107, Dec 2011. [Online]. Available: <http://link.aps.org/doi/10.1103/PhysRevB.84.214107>
 - [43] M. Palumbo, B. Burton, A. C. e Silva, B. Fultz, B. Grabowski, G. Grimvall, B. Hallstedt, O. Hellman, B. Lindahl, A. Schneider, P. E. A. Turchi, and W. Xiong, “Thermodynamic modelling of crystalline unary phases,” *Physica Status Solidi (b)*, vol. 251, pp. 14–32, January 2014. [Online]. Available: <http://onlinelibrary.wiley.com/doi/10.1002/pssb.201350133/abstract>
 - [44] J. Rogal, S. V. Divinski, M. W. Finnis, A. Glensk, J. Neugebauer, J. H. Perepezko, S. Schuwalow, M. H. F. Sluiter, and B. Sundman, “Perspectives on point defect thermodynamics,” *physica status solidi (b)*, vol. 251, pp. 99–129, January 2014. [Online]. Available: <http://onlinelibrary.wiley.com/doi/10.1002/pssb.201350155/abstract>
 - [45] S. H. Simon, *The Oxford Solid State Basics*. Oxford University Press, August 2013.
 - [46] A. Gheribi and P. Chartrand, “Application of the calphad method to predict the

- thermal conductivity in dielectric and semiconductor crystals,” *CALPHAD*, vol. 39, pp. 70–79, 2012. [Online]. Available: <http://dx.doi.org/10.1016/j.calphad.2012.06.002>
- [47] X.-G. Lu and Q. Chen, “A CALPHAD Helmholtz energy approach to calculate thermodynamic and thermophysical properties of fcc Cu,” *Philosophical Magazine*, vol. 89, pp. 2167–2194, 2009. [Online]. Available: <http://dx.doi.org/10.1080/14786430903059004>
- [48] U. Kattner and C. Campbell, “Invited review: Modelling of thermodynamics and diffusion in multicomponent systems,” *Materials Science and Technology*, vol. 25, pp. 443–459, 2009. [Online]. Available: <http://dx.doi.org/10.1179/174328408X372001>
- [49] J. Andersson and J. Ågren, “Models for numerical treatment of multicomponent diffusion in simple phases,” *Journal of Applied Physics*, vol. 72, 1992. [Online]. Available: <http://dx.doi.org/10.1063/1.351745>
- [50] J. Ågren, “Numerical treatment of diffusional reactions in multicomponent alloys,” *Journal of Physics and Chemistry of Solids*, vol. 43, no. 4, pp. 385 – 391, 1982. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/0022369782902098>
- [51] J. Ågren, “Diffusion in phases with several components and sublattices,” *Journal of Physics and Chemistry of Solids*, vol. 43, pp. 421–430, 1982.
- [52] B. Zhang, X. Li, and D. Li, “Assessment of thermal expansion coefficient for pure metals,” *Calphad*, vol. 43, pp. 7 – 17, 2013. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0364591613000928>
- [53] B. Hallstedt, N. Dupin, M. Hillert, L. Höglund, H. Lukas, J. Schuster, and N. Solak, “Thermodynamic models for crystalline phases. composition dependent models for volume, bulk modulus and thermal expansion,” *Calphad*, vol. 31, no. 1, pp. 28 – 37, 2007. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0364591606000162>
- [54] B. Hallstedt, “Molar volumes of al, li, mg and si,” *Calphad*, vol. 31, no. 2, pp. 292 – 302, 2007. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0364591606000861>
- [55] X. Lu, M. Selleby, and B. Sundman, “Theoretical modeling of molar volume and thermal expansion,” *Acta Materialia*, vol. 53, pp. 2259–2272, 2005. [Online]. Available: <http://dx.doi.org/10.1016/j.actamat.2005.01.049>
- [56] —, “Assessments of molar volume and thermal expansion for selected bcc, fcc and hcp metallic elements,” *CALPHAD*, vol. 29, pp. 68–89, 2005. [Online]. Available: <http://dx.doi.org/10.1016/j.calphad.2005.05.001>
- [57] Z.-K. Liu, H. Zhang, S. Ganeshan, Y. Wang, and S. Mathaudhu, “Computational modeling of effects of alloying elements on elastic coefficients,” *Scripta Materialia*, vol. 63, pp. 686–691, 2010. [Online]. Available: <http://dx.doi.org/10.1016/j.scriptamat.2010.03.049>
- [58] T. Hammerschmidt, I. A. Abrikosov, D. Alfè, S. G. Fries, L. Höglund,

- M. H. G. Jacobs, J. Koßmann, X.-G. Lu, and G. Paul, “Including the effects of pressure and stress in thermodynamic functions,” *Physica Status Solidi (b)*, vol. 251, pp. 81–96, January 2014. [Online]. Available: <http://onlinelibrary.wiley.com/doi/10.1002/pssb.201350156/abstract>
- [59] A. Fernández Guillermet, “Critical evaluation of the thermodynamic properties of cobalt,” *International Journal of Thermophysics*, vol. 8, no. 4, pp. 481–510, 1987. [Online]. Available: <http://dx.doi.org/10.1007/BF00567107>
- [60] X. Lu, M. Selleby, and B. Sundman, “Implementation of a new model for pressure dependence of condensed phases in thermo-calc,” *CALPHAD*, vol. 29, pp. 49–55, 2005. [Online]. Available: <http://dx.doi.org/10.1016/j.calphad.2005.04.001>
- [61] F. Körmann, A. Dick, T. Hickel, and J. Neugebauer, “Pressure dependence of the curie temperature in bcc iron studied by ab initio simulations,” *Phys. Rev. B*, vol. 79, p. 184406, 2009. [Online]. Available: <http://dx.doi.org/10.1103/PhysRevB.79.184406>
- [62] J. Guo and M. T. Samonds, “Alloy thermal physical property prediction coupled computational thermodynamics with back diffusion consideration,” *Journal of Phase Equilibria and Diffusion*, vol. 28, no. 1, pp. 58–63, 2007. [Online]. Available: <http://dx.doi.org/10.1007/s11669-006-9005-6>
- [63] I. Steinbach, B. Böttger, J. Eiken, N. Warnken, and S. G. Fries, “Calphad and phase-field modeling: A successful liaison,” *Journal of Phase Equilibria and Diffusion*, vol. 28, pp. 101–106, 2007. [Online]. Available: <http://dx.doi.org/10.1007/s11669-006-9009-2>
- [64] G. Schmitz, A. Engstrom, R. Bernhardt, U. Prahl, L. Adam, J. Seyfarth, M. Apel, C. A. de Saracibar, P. Korzhavyi, J. Ågren, and B. Patzak, “Towards interoperability of software solutions in icme settings,” in *International Workshop on Multiscale Simulation: from Materials through to Industrial Usage*, Dublin, Ireland, September 2016.
- [65] ICME, “Integrated computational materials engineering,” 2008. [Online]. Available: https://en.wikipedia.org/wiki/Integrated_computational_materials_engineering
- [66] A. Schneider, C. Stallybrass, J. Konrad, A. Kulgemeyer, H. Meuser, and S. Meimeth, “Formation of primary tin precipitates during solidification of microalloyed steels – scheil versus dictra simulations,” vol. 99, pp. 674–679, 2008.
- [67] A. Borgenstam, L. Höglund, J. Ågren, and A. Engström, “Dictra, a tool for simulation of diffusional transformations in alloys,” *Journal of Phase Equilibria*, vol. 21, no. 3, pp. 269–280, 2000. [Online]. Available: <http://dx.doi.org/10.1361/105497100770340057>
- [68] B. Sundman, U. Kattner, M. Palumbo, and S. Fries, “Openalphanad – a free thermodynamic software,” *IMMI*, vol. 4, 2015. [Online]. Available: <http://link.springer.com/article/10.1186/s40192-014-0029-1>
- [69] “Open Calphad project,” <http://www.openalphanad.com/>, February 2014.
- [70] R. Otis, “picalphanad,” 2015. [Online]. Available: <https://github.com/picalphanad>

- [71] D. Wheeler, J. Guyer, and J. Warren, *FiPy: Users Guide*, 2007. [Online]. Available: <http://www.ctcms.nist.gov/fipy/download/fipy-1.2.3.pdf>
- [72] A. Bartol, R. Garcia, D. Ely, and J. Guyer, “The virtual kinetics of materials laboratory,” 2015. [Online]. Available: <http://nanohub.org/resources/vkmlive>
- [73] M. Piro and S. Sumunovic, “Performance enhancing algorithms for computing thermodynamic equilibria,” *CALPHAD*, vol. 39, pp. 104–110, 2012. [Online]. Available: <http://dx.doi.org/10.1016/j.calphad.2012.09.005>
- [74] J. Wilson, “Benefits of open source code,” <http://oss-watch.ac.uk/resources/whoneedssource>, 2013. [Online]. Available: <http://oss-watch.ac.uk/resources/whoneedssource>
- [75] K. Noyes, “10 reasons open source is good for business,” http://www.pcworld.com/article/209891/10_reasons_open_source_is_good_for_business.html, 2010. [Online]. Available: http://www.pcworld.com/article/209891/10_reasons_open_source_is_good_for_business.html
- [76] O. Kubaschewski and E. L. Evans, *Metallurgical thermochemistry*, 3rd ed., ser. International series of monographs on metal physics and physical metallurgy, v. 1. Oxford: Pergamon Press, 1965.
- [77] B. Sundman, X.-G. Lu, and H. Ohtani, “The implementation of an algorithm to calculate thermodynamic equilibria for multi-component systems with non-ideal phases in a free software,” *Computational Materials Science*, vol. 101, pp. 127–137, 2015.
- [78] T.-C. S. AB, *TCC Thermo-Calc Software User’s Guide*, <http://www3.thermocalc.se/res/Manuals/Thermo-Calc.UsersGuide.pdf>, Norra Stationsgatan 93, 5 tr. SE-113 64 Stockholm, Sweden, 2013.
- [79] pandat, “Pandat,” 2000.
- [80] M. Perrut, “Thermodynamic modeling by the calphad method and its applications to innovative materials,” *Aerospace Lab*, no. 9, pp. 1–11, July 2015. [Online]. Available: <http://www.aerospacelab-journal.org/al9/thermodynamic-modeling-by-calphad-method-and-its-applications-to-innovative-materials>
- [81] Z.-K. Liu, “High throughput calphad modeling and the materials genome,” <https://www.sfb-vicom.at/events/single-view-news-events/article/high-throughput-calphad-modeling-and-the-materials-genomeR/>, 2014. [Online]. Available: <https://www.sfb-vicom.at/events/single-view-news-events/article/high-throughput-calphad-modeling-and-the-materials-genomeR/>
- [82] S.-L. Shang, Y. Wang, and Z.-K. Liu, “ESPEI: Extensible, Self-optimizing Phase Equilibrium Infrastructure for Magnesium Alloys,” *Magnesium Technology*, pp. 617–622, 2010.
- [83] “Algebraic Modeling and Programming Language,” <http://www.ampl.com/>, 2012.
- [84] P. Gill, W. Murray, and M. Saunders, *User’s Guide for SNOPT Version 7: Software*

- for *Large-Scale Nonlinear Programming*, 2008, university of California, San Diego, Stanford University, Technical Report.
- [85] R. Vanderbei, *LOQO User's Manual-Version 4.05*, <http://www.princeton.edu/~rvdb/tex/loqo/loqo405.pdf>, 2006, princeton University, Technical Report.
 - [86] B. Murtagh and M. Saunders, *MINOS 5.5 User's Guide*, 1983–1998, stanford University, Technical Report SOL 83-20R.
 - [87] J. Ågren, B. Cheynet, M. T. Clavaguera-Mora, K. Hack, J. Hertz, F. Sommer, and U. Kattner, “Workshop on thermodynamic models and data for pure elements and other endmembers of solutions,” *Calphad*, vol. 19, no. 4, pp. 449 – 480, 1995. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/036459169600003X>
 - [88] O. Redlich and A. Kister, “Algebraic representation of thermodynamic properties and classification of solutions,” *Ind. Eng. Chem*, pp. 345–348, 1948. [Online]. Available: <http://pubs.acs.org/doi/abs/10.1021/ie50458a036>
 - [89] P. E. Gill, W. Murray, and M. A. Saunders, “Snopt: An sqp algorithm for large-scale constrained optimization,” *SIAM Review*, vol. 47, no. 1, pp. 99–131.
 - [90] B. Murtagh and M. Saunders, “Large-scale linearly constrained optimization,” *Mathematical Programming*, vol. 14, pp. 41–72, 1978.
 - [91] M. Hillert, “Some viewpoints on the use of a computer for calculating phase diagrams,” *Physica B+C*, vol. 103, pp. 31–40, 1981. [Online]. Available: [http://dx.doi.org/10.1016/0378-4363\(81\)91000-7](http://dx.doi.org/10.1016/0378-4363(81)91000-7)
 - [92] I. Griva, S. Nash, and A. Sofer, *Linear and Nonlinear Optimization: Second Edition*. SIAM, Philadelphia, 2009. [Online]. Available: <http://books.google.com/books?id=uOJ-Vg1BnKgC>
 - [93] R. Otis and Z. Liu, “Pycalphad: open-source software for computational thermodynamics,” 2015. [Online]. Available: http://matscitech.org/wp-content/uploads/2015/10/MST15.FINAL_PROGRAM_B2.pdf
 - [94] P. Franke, “Modeling of thermal vacancies in metals within the framework of the compound energy formalism,” *Journal of Phase Equilibria and Diffusion*, vol. 35, no. 6, pp. 780–787, October 2014. [Online]. Available: <http://dx.doi.org/10.1007/s11669-014-0348-0>
 - [95] J. Ågren and M. Hillert, “Thermodynamic modelling of vacancies as a constituent,” *CALPHAD*, vol. 67, 2019. [Online]. Available: <https://doi.org/10.1016/j.calphad.2019.101666>
 - [96] H. Lukas, S. Fries, and B. Sundman, *Computational Thermodynamics, The Calphad Method*. New York: Cambridge University Press, 2007.
 - [97] A. Davydov and U. Kattner, “Thermodynamic assessment of the co-mo system,” *Journal of Phase Equilibria*, vol. 20, pp. 5–16, 1999.
 - [98] D. Kim, V. Manga, S. Prins, and Z.-K. Liu, “First-Principles Calculations and

- Thermodynamic Modeling of the Al-Pt Binary System,” *CALPHAD*, vol. 35, pp. 20–29, 2011. [Online]. Available: <http://dx.doi.org/10.1016/j.calphad.2010.10.008>
- [99] S. Zhang, D. Shin, and Z.-K. Liu, “Thermodynamic modeling of the Ca-Li-Na system,” *CALPHAD*, vol. 27, pp. 235–241, 2003. [Online]. Available: <http://dx.doi.org/10.1016/j.calphad.2003.09.001>
- [100] B. Sundman and J. Ågren, “A regular solution model for phases with several components and sublattices, suitable for computer applications,” *Journal of Physics and Chemistry of Solids*, vol. 42, no. 4, pp. 297–301, 1981. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/002236978190144X>
- [101] J. Andersson, T. Helander, L. Höglund, P. Shi, and B. Sundman, “Thermo-calc and dicta, computational tools for materials science,” *CALPHAD*, vol. 26, no. 2, pp. 273–312, 2002. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0364591602000378>
- [102] “Python,” <https://www.python.org/>, 2016.
- [103] “TIOBE Programming Community Index,” http://www.tiobe.com/tiobe_index, 2016.
- [104] P. Guo, “Python is Now the Most Popular Introductory Teaching Language at Top U.S. Universities,” <http://cacm.acm.org/blogs/blog-cacm/176450-python-is-now-the-most-popular-introductory-teaching-language-at-top-us-universities/fulltext>, July 2014.
- [105] “Anaconda,” <https://www.continuum.io/why-anaconda>, 2016.
- [106] B. Fultz, *Phase Transitions in Materials*. Cambridge University Press, August 2014. [Online]. Available: <https://books.google.com/books?id=ChdEBAAAQBAJ>
- [107] E. Königsberger, “Improvement of excess parameters from thermodynamic and phase diagram data by a sequential bayes algorithm,” *CALPHAD*, vol. 15, pp. 69–78, January 1991. [Online]. Available: [http://dx.doi.org/10.1016/0364-5916\(91\)90027-H](http://dx.doi.org/10.1016/0364-5916(91)90027-H)
- [108] K. Lejaeghere, V. V. Speybroeck, G. V. Oost, and S. Cottenier, “Error estimates for solid-state density-functional theory predictions: An overview by means of the ground-state elemental crystals,” *Critical Reviews in Solid State and Materials Sciences*, vol. 39, pp. 1–24, 2014. [Online]. Available: <http://dx.doi.org/10.1080/10408436.2013.772503>
- [109] A. Dick, F. Körmann, T. Hickel, and J. Neugebauer, “*Ab initio* based determination of thermodynamic properties of cementite including vibronic, magnetic, and electronic excitations,” *Phys. Rev. B*, vol. 84, p. 125101, Sep 2011. [Online]. Available: <http://link.aps.org/doi/10.1103/PhysRevB.84.125101>
- [110] D. Marquardt, “An algorithm for least-squares estimation of nonlinear parameters,” *Journal of the Society for Industrial and Applied Mathematics*, vol. 11, pp. 431–441, 1963. [Online]. Available: <http://dx.doi.org/10.1137/0111030>
- [111] T.-C. S. AB, “TCC Thermo-Calc Data Optimization User’s

- Guide,” <http://www.thermocalc.com/media/19866/data-optimisation-user-guide-for-thermo-calc.pdf>, Norra Stationsgatan 93, 5 tr. SE-113 64 Stockholm, Sweden, 2015. [Online]. Available: <http://www.thermocalc.com/media/19866/data-optimisation-user-guide-for-thermo-calc.pdf>
- [112] M. G. Shirangi and A. A. Emerick, “An improved tsvd-based levenberg–marquardt algorithm for history matching and comparison with gauss–newton,” *Journal of Petroleum Science and Engineering*, vol. 143, pp. 258–271, 2016. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0920410516300687>
 - [113] R. Otis, M. Emelianenko, and Z.-K. Liu, “An improved sampling strategy for global energy minimization of multi-component phases with internal degrees of freedom,” *Computational Materials Science*, 2017.
 - [114] P. Honarmandi, T. Duong, S. Ghoreishi, D. Allaire, and R. Arroyave, “Bayesian uncertainty quantification and information fusion in calphad-based thermodynamic modeling,” *Acta Materialia*, vol. 164, pp. 636 – 647, 2019. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1359645418308838>
 - [115] P. Honarmandi, N. H. Paulson, R. Arróyave, and M. Stan, “Uncertainty quantification and propagation in CALPHAD modeling,” *Modelling and Simulation in Materials Science and Engineering*, vol. 27, no. 3, p. 034003, mar 2019. [Online]. Available: <https://doi.org/10.1088%2F1361-651x%2F19030303>
 - [116] R. Otis and Z.-K. Liu, “High-throughput thermodynamic modeling and uncertainty quantification for icme,” *JOM*, vol. 69, no. 5, pp. 886–892, 2017, cited By 12. [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85015820064&doi=10.1007%2F11837-017-2318-6&partnerID=40&md5=93d9461bcca329820033e0e687753d92>
 - [117] M. Sanghvi, P. Honarmandi, V. Attari, T. Duong, R. Arroyave, and D. L. Allaire, “Uncertainty propagation via probability measure optimized importance weights with application to parametric materials models,” *AIAA Scitech 2019 Forum*, 2019. [Online]. Available: <https://arc.aiaa.org/doi/abs/10.2514/6.2019-0967>
 - [118] K. Oikawa, U. R. Kattner, J. Sato, T. Omori, M. Jiang, K. Anzai, and K. Ishida, “Experimental determination and thermodynamic assessment of phase equilibria in the como system,” *Materials Transactions*, vol. 53, no. 8, pp. 1425 – 1435, 2012.
 - [119] F. Glover, “Tabu search–part i,” *ORSA Journal on Computing*, vol. 1, no. 3, pp. 190–206, 1989. [Online]. Available: <http://dx.doi.org/10.1287/ijoc.1.3.190>
 - [120] D. Ashlock, *Evolutionary Computation for Modeling and Optimization*. Springer New York, 2006. [Online]. Available: <http://link.springer.com/book/10.1007/0-387-31909-3>
 - [121] J. Kennedy and R. Eberhart, “Particle swarm optimization,” in *Neural Networks, 1995. Proceedings., IEEE International Conference on*, vol. 4, Nov 1995, pp. 1942–1948 vol.4.
 - [122] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, “Optimization by simulated

- annealing,” *Science*, vol. 220, no. 4598, pp. 671–680, 1983. [Online]. Available: <http://science.sciencemag.org/content/220/4598/671>
- [123] W. Wenzel and K. Hamacher, “Stochastic tunneling approach for global minimization of complex potential energy landscapes,” *Phys. Rev. Lett.*, vol. 82, pp. 3003–3007, Apr 1999. [Online]. Available: <http://link.aps.org/doi/10.1103/PhysRevLett.82.3003>
 - [124] V. Maniezzo, T. Stützle, and S. V. (eds.), *Matheuristics: Hybridizing Metaheuristics and Mathematical Programming*. Springer US, 2010. [Online]. Available: <http://dx.doi.org/10.1007/978-1-4419-1306-7>
 - [125] P. Belotti, C. Kirches, S. Leyffer, J. Linderoth, J. Luedtke, and A. Mahajan, “Mixed-integer nonlinear optimization,” *Acta Numerica*, vol. 22, pp. 1–131, 5 2013. [Online]. Available: http://journals.cambridge.org/article_S0962492913000032
 - [126] M. A. Duran and I. E. Grossmann, “An outer-approximation algorithm for a class of mixed-integer nonlinear programs,” *Mathematical Programming*, vol. 36, no. 3, pp. 307–339, 1986. [Online]. Available: <http://dx.doi.org/10.1007/BF02592064>
 - [127] H. Hijazi, P. Bonami, and A. Ouorou, “An outer-inner approximation for separable mixed-integer nonlinear programs,” *INFORMS Journal on Computing*, vol. 26, no. 1, pp. 31–44, 2014. [Online]. Available: <http://dx.doi.org/10.1287/ijoc.1120.0545>

Curriculum Vitae

Jeffrey H. Snider graduated from Heritage High School [1989], Lynchburg, Virginia. He enrolled at George Mason University in 1989, and was dismissed in 1992.

He was employed by UUNET Technologies as a Network Engineer and a Software Engineer from 1992 until 2001. In 1993 he developed, deployed, and maintained the internet service provider's first automated data gathering and fault reporting system. Among many other activities, he developed and maintained configuration and monitoring tools, and designed and deployed components of the internet backbone. In 1994 he developed predictive targeting algorithms for an Xtank bot that enabled a single immobile robot to consistently defeat all but the most specialized opposing force. In 1996 he participated in the design, development, testing, and deployment of the first Virtual Private Network product. In 1998 he participated in the design, development, and demonstration of a functioning prototype of broadcast quality streaming video over the internet, including a library of available videos.

In 2002 he developed software for Voice XML at Cambridge VoiceTech, Reston, Virginia. In 2003 he developed web front-end software for Team Askin, Falls Church, Virginia. At various times from 2001 through 2005 he was a Patent Researcher for Swidler Berlin Shereff Friedman, Washington, DC.

In 2002 he was on the United States National Fencing Team and represented the USA in the Fencing World Championships in Lisbon, Portugal. He placed 6th in the World Cup in Bogotá, Columbia in 2001, 8th in the World Cup in Havana, Cuba, in 2002, and was a member of the team placing 3rd in the World Cup in Sydney, Australia, in 2002. As a fencer, he has been ranked at the top of his age group in the United States continuously since 2016. He is an internationally certified fencing referee, and in addition to many North American Cups and National Championships he has refereed World Cups throughout Europe. He was selected as a sponsored athlete by Triplette Competition Arms based on moral character and athletic accomplishments.

He was Chairman of the Virginia Division of US Fencing from 2003 to 2007, leading an executive committee of 7 members, serving over 400 Virginia athletes and hundreds of additional visiting competitors, and regularly operating the largest local and regional fencing competitions.

He was Head Coach at the Richmond Fencing Club from 2005 to 2009, managing all business operations, serving several hundred clients and cadre of 10 coaches, and teaching the skill and art of Fencing in group and individual settings

After being denied readmission to George Mason University in 2009, he received an Associate of Science in Mathematics [2009] and an Associate of Science in Engineering [2009] from

Northern Virginia Community College, Annandale, Virginia. He received his Bachelor of Science in Mathematics [2010] and Master of Science in Mathematics [2012] from George Mason University.

He was a Graduate Teaching Assistant in Spring 2011, leading recitation for Analytic Geometry and Calculus II. He was a Research Assistant in Fall 2011, investigating Fractional Derivatives and Fractional Partial Differential Equations.

He has given numerous presentations on mathematical topics in weekly graduate seminars, including the history and application of square roots, GPGPU distributed processing, fractional derivatives and nonlocal operators and their application, and many others. He presented “Energy Minimization Challenges: Metastability, Vacancies and Uncertainty” at the 2018 SIAM Conference on Mathematical Aspects of Materials Science in Portland, Oregon, July 2018.

Since 2011 he has been employed as an Operations Research Analyst by The MITRE Corporation, McLean, Virginia, a Federally Funded Research and Development Center performing not-for-profit work in the public interest. At MITRE he was selected for the highly competitive Advanced Graduate Degree Program, allowing him to pursue his graduate work 20% of his paid work hours for several years. He lead research programs in Risk Analysis and communication policy for Autonomous Agents, in addition to being a team member on numerous projects applying modeling and simulation, optimization heuristics, queue theory, machine learning and artificial intelligence, big data and distributed processing, and many others.

He holds certifications in Advanced Amazon Web Services Development [2019], and is skilled with C, Java, Python, JavaScript, Perl, SQL, Shell scripting, Tcl/Tk, Postscript, PHP, Pig, R, Tableau, git, MATLAB, Office, Sybase, X-Windows, CVS, AWS, Accumulo, Hadoop, GPGPU (OpenCL, CUDA), and many other technologies.

In addition to being a native English speaker, he has basic competence in written and spoken French, Spanish, and Italian. He plays trumpet and is the Boru Basha and Music Director for the premier North-American Mehteran, a recreation of a 16th century Ottoman Military Marching Band. He plays lute and recorder, reading original medieval and renaissance notation as well as modern notation, and is Secretary of the Washington Recorder Society. In addition to fencing he is an avid runner, and he has qualified to the 2021 Amateur 50+ Disc Golf World Championships.