Contents lists available at ScienceDirect

CALPHAD: Computer Coupling of Phase Diagrams and Thermochemistry

journal homepage: www.elsevier.com/locate/calphad

Set based framework for Gibbs energy minimization

Jeff Snider [a,*], Igor Griva [a], Xiaodi Sun [b], Maria Emelianenko [a,**]

[a] Department of Mathematical Sciences, George Mason University MS 3F2, 4400 University Dr, Fairfax, VA 22030, USA
[b] Department of Energy, Environmental and Chemical Engineering, Washington University, 1 Brookings Drive, St. Louis, MO 63130, USA

ARTICLE INFO

Article history:
Received 29 March 2014
Received in revised form
6 August 2014
Accepted 15 September 2014
Available online 23 October 2014

Keywords:
Mathematical modeling
Phase diagrams
Optimization
Automation
Miscibility gaps
Sublattices

ABSTRACT

A new unified approach to Gibbs energy minimization is introduced. While it has only been tested on binary and ternary systems so far, it has a built in capability of handling arbitrary multicomponent multiphase systems with any number of sublattices, miscibility gaps, order–disorder transitions, and magnetic contributions. This new unified AMPL set-based Gibbs energy description optimizes the data representation and makes it possible to subject the task of phase diagram calculation to numerous existing general purpose optimization strategies as well as custom-made solvers. The approach is tested on a variety of systems, including Co–Mo, Al–Pt and Ca–Li–Na, all known to be computationally challenging for other approaches. In most of the tested systems, the AMPL code reproduces phase diagrams obtained via Thermo-Calc. In other systems, in-depth comparison of results suggests that in prior work a sub-optimal equilibrium might have been identified as a global one, and re-evaluation of previously published diagrams and databases might be necessary.

1. Introduction

The last few decades have seen substantial development of algorithms and software for phase diagram calculations. These specialized software packages provide much needed functionality and capabilities for handling complex material systems. At the same time, the existing software packages for phase diagram calculation often rely on user input of initial conditions for convergence, and require special handling when it comes to formulations involving miscibility gaps and sublattices [1,2]. A number of remedies have been proposed in recent years, some of which have been quite successful at reducing the risks of running into suboptimal solutions, often with a steep cost premium [3,4]. Here a completely new methodology is proposed that allows a user to tackle these issues with the efficient use of the state-of-the-art constrained optimization algorithms that utilize the generalized form of the Gibbs energy minimization problem.

This paper introduces a *novel, set-based energy formulation*, equivalent to the traditional one, yet possessing a number of advantages in automation and flexibility. The new formulation allows the treatment of any multicomponent multiphase system involving an arbitrary number of sublattices, order–disorder transitions, and magnetic terms by means of a single formula. Automated miscibility gap detection is also achieved with no need for additional sampling as implemented by other existing methodologies [3,4].

The set-based approach serves as a bridge between Thermo-Calc-type databases and cutting edge optimization technologies which have become available in recent years. The approach allows the use of AMPL (a Modeling Language for Mathematical Programming) [5], a low-cost software environment which exposes the problem to a wide variety of general purpose optimization solvers as well as custom implementations taking advantage of the structure of the CALPHAD problem. This black box approach has the potential to reduce or eliminate the need for expert knowledge, enabling a simple and straightforward phase diagram evaluation interface for specialists and non-specialists alike.

In order to produce an AMPL compatible system description from a traditional database (TDB file), a system-independent converter was developed to extract all phase data from a TDB and feed it into the main AMPL model. This is not the first adaptation of Thermo-Calc data for further processing within a separate software module, or introduction of a framework capable of computing phase diagrams from Thermo-Calc-type descriptions. Notable existing work inspiring this effort includes the ESPEI infrastructure [6], OpenCalphad initiative [7], and the Gibbs module [8].

ESPEI is a self-optimizing phase equilibrium software package, which integrates databases (crystallographic, phase equilibrium, thermochemical, and modeled Gibbs energy data, etc) and database development (automation of thermodynamic modeling) with GUI (graphical user interface) designed mainly using Microsoft C# and SQL (structured query language). The data is stored in a matrix

[*] Corresponding author.
[**] Principal corresponding author.
E-mail addresses: jeff@snider.com (J. Snider), igriva@gmu.edu (I. Griva), xiaodi.daniel.sun@gmail.com (X. Sun), memelian@gmu.edu (M. Emelianenko).

http://dx.doi.org/10.1016/j.calphad.2014.09.005
0364-5916/© 2014 Elsevier Ltd. All rights reserved.

format specific to ESPEI and allows for automation of database development [6]. OpenCalphad is an open-source code for performing thermodynamic calculations using the Calphad approach. The code implements several different thermodynamic models which allow the description of thermodynamic state functions, such as the Gibbs energy, as a function of temperature, pressure and composition [7]. Finally, Gibbs module is a tool hosted by nanoHUB [9] that enables rapid prototyping, validation, and comparison of thermodynamic models to describe the equilibrium between multiple phases for binary systems [8].

The approach presented herein complements these efforts and is not meant to introduce yet another solver, or compete with existing packages in terms of speed. It is designed as a research tool allowing scrutiny of phase diagram calculation from a different viewpoint, making it possible to subject the problem to numerous existing general purpose optimization strategies as well as custom-made solvers, revealing similarities and differences between resulting phase diagrams. As seen in Section 6 below, preliminary benchmarking tests suggest that this methodology can lead to some interesting observations when it comes to systematic analysis of phase diagrams obtained by other methods. In particular, new insights into phase stability were obtained by applying the new methodology to several multicomponent test systems containing miscibility gaps, multiple sublattices, and order–disorder transitions. The AMPL based approach precisely mirrored the Thermo-Calc phase diagram for the Co–Mo system, however for the Al–Pt case a lower Gibbs energy associated with a different phase combination was found than that reported by Thermo-Calc using the default settings. Similar findings have been reported earlier, e.g., in [10] for the systems Si–Tb, Al–Nb, Co–Si, and others. While elucidating the source of these discrepancies is outside of the scope of current work, the comparative analysis presented in Section 6 gives evidence to certain inconsistencies in the treatment of Gibbs energies that might warrant further investigation.

The information flow is depicted graphically in Fig. 1. The converter, model, and mapper are “black box” codes which do not depend on the system being investigated. The converter is written in Java, while the model and mapper are implemented in the AMPL modeling language. The model is a description of the problem for any system, completely independent of the number and choice of elements or species. It is described in detail in Sections 3 and 4.

Given a TDB file for a system, the converter generates data and parameter files in the AMPL modeling language. The mapper is an AMPL script which reads the model and data and writes out phase information by temperature and composition. This output constitutes the data for a phase diagram, which is converted (for this paper using MATLAB) into a graphical phase diagram.

The paper is organized as follows. A short description of the AMPL environment is given in Section 2. The standard problem formulation given in Section 3 is followed by the description of the novel set-based formulation in Section 4, with the discussion on the handling of the miscibility gap given in Section 4.6. The details of the use of the novel formulation for the automation of phase diagram calculation are provided in Section 5. Results of numerical tests are shown in Section 6, followed by discussion.

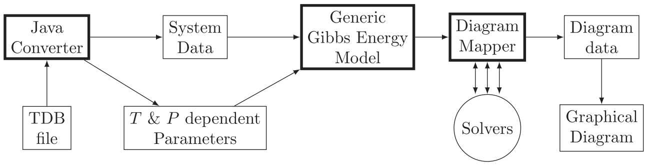

Fig. 1. Data flow.

2. AMPL environment

AMPL is a modeling language and software package providing a unified front-end for an extremely wide variety of solvers, such as LOQO [11], MINOS [12], and SNOPT [13], among others [5,14]. The user can describe the model and separately provide the data, making reuse of the model for alternative systems straightforward. AMPL easily handles any combination of linear and nonlinear objective functions and constraints, and selection of the best solver is left to the user. The “student” version of AMPL with limitations on the number of variables and constraints is freely available online, and a professional version is available for a modest fee. For academic use there are free online services where a job can be submitted with no restriction on size, and the result is returned via email. Many of the most powerful solvers are freely available online. A robust set of documentation is available, and a widespread group of users and AMPL developers actively discuss challenges and advance the state of the art.

In the AMPL modeling language, data is primarily set-based with native commands for set operations on scalars, textual values, and tuples of arbitrary length. Summation, multiplication, and other operations with sets as indices are natural. Set and parameter values may be explicitly defined in the data, or be transparently computed according to their definition in the model when the data is read or updated. It has to be noted that set-based formulation has advantages over other structures traditionally used to store data (e.g. vectors or matrices) due to its flexibility in defining the number and the names of set components. The space is automatically preallocated. The only task left to a user is providing the sets and parameters defined on those sets and then the model will automatically generate the Gibbs energies and the constraints based on this information.

3. Formulation of Gibbs energy minimization problem

First recall the standard Gibbs energy minimization problem formulation [15–17]:

$$\left.\begin{array}{ll}
\min\limits_{f,y} G=\sum\limits_{p} f^{(p)}G^{(p)}(y) & \\
0\le f^{(p)}\le 1 & \text{for each phase } p\\
0\le x_e^{(p)}\le 1 & \text{for each phase} - \text{species pair } (p,e)\\
0\le y_{s,e}^{(p)}\le 1 & \text{for each phase} - \text{species} - \text{sublattice triplet } (p,e,s)\\
\sum\limits_{e} y_{s,e}^{(p)}=1, & \text{for each phase} - \text{sublattice pair } (p,s)\\
\sum\limits_{p} f^{(p)}x_e^{(p)}=f_e^0, & \text{for each species } e, \quad \text{such that } \sum\limits_{e} f_e^0=1
\end{array}\right\} \tag{1}$$

Here $x_e^{(p)}$ indicates the mole fraction of species e within the phase, $y_{s,e}^{(p)}$ is the site fraction of species e in sublattice s within the phase, $a_s^{(p)}$ is the site ratio of sublattice s in the phase, and $f^{(p)}$ indicates the mole fraction of the phase in the overall composition. Similar to [15], rather than being the subject of a constraint, x_e is defined as a function of $y_{s,e}$ in the present implementation:

$$x_e^{(p)}:=\frac{\sum_s a_s^{(p)} y_{s,e}^{(p)}}{\sum_{c\neq \mathrm{Va}}\sum_s a_s^{(p)} y_{s,c}^{(p)}}. \tag{2}$$

This reduces the number of variables and constraints, and for the solvers tested it results in a modest speed improvement.

The overall number of variables and constraints in this problem depends entirely on the system being examined. The smallest system potentially of interest would have two species and two phases each with a single sublattice, hence six variables and a dozen constraints. The largest system examined in this manuscript had three species and twelve phases, resulting in 33 variables and 18 constraints. However, optimization techniques used in our tests and other tools available in AMPL are capable of handing significantly larger systems, so many realistic systems are well within the reach of the methods discussed herein.

Traditionally, the Gibbs energy for each phase $G^{(p)}$ can belong to several classes depending on whether it has sublattices, what type of species interactions are allowed within each sublattice, whether it has an order–disorder transition, etc. The standard formulation is hence strongly parameter dependent. This paper proposes two improvements to the way the problem is set up and solved: (a) generalized Gibbs energy formulation; and (b) a straightforward way to handle miscibility gaps. The goal for the development of a new methodology is to completely automate the task of setting up and accurately solving problem (1).

4. A different look at Gibbs energy minimization

4.1. Motivation

An example of applying the proposed set-based paradigm to modeling the excess energy via Redlich–Kister polynomials [18] motivates the development of the model. Consider the case of a phase having a single sublattice, where the standard form for excess energy ${}^{xs}G_m^{\Phi}$ is given by [15,19]

$$\sum_i \sum_{j>i} x_i x_j \sum_k {}^kL_{i,j}^{\Phi}(x_i - x_j)^k. \tag{3}$$

By relabeling the ${}^kL$ as ${}^kG$ terms, formally replacing x by y and moving the inner sum to the outside, (3) may be equivalently written as

$$\sum_k \sum_i \sum_{j>i} {}^kG_{i,j} y_i y_j (y_i - y_j)^k. \tag{4}$$

Note that in this motivating discussion we are only concerned with the excess energy term, so conversions between x and y are purely formal and do not involve the use of (2). Site ratios of sublattices will naturally appear in the full energy formulation developed in Section 4.2.

In the case of a four sublattice compounds with first order constituent arrays, i.e., mixing between two species in exactly one sublattice that may be represented as (A,B):C:D:E in standard shorthand notation [15], the formula for the excess energy can be written

$$\sum_s \sum_i \sum_{j>i} \sum_l \sum_m \sum_n y_{s,i} y_{s,j} y_{r,l} y_{t,m} y_{u,n} \sum_k {}^kL_{i,j:l:m:n}(y_{s,i} - y_{s,j})^k. \tag{5}$$

Here ${}^kL_{i,j:l:m:n}$ are the kth degree binary interaction (Redlich–Kister) coefficients. Following the notation (A,B):C:D:E given above, the indices i, j correspond to the elements A,B interacting on the first sublattice, and indices m, n stand for the elements C and D respectively.

By relabeling the ${}^kL$ as ^{kG} terms, replacing x with y, and moving the inner sum to the outside, (5) may be equivalently written as

$$\sum_k \sum_s \sum_i \sum_{j>i} \sum_l \sum_m \sum_n {}^kG_{i,j:l:m:n} y_{s,i} y_{s,j} y_{r,l} y_{t,m} y_{u,n} (y_{s,i} - y_{s,j})^k. \tag{6}$$

For second order constituent arrays, e.g., mixing at two sublattices simultaneously, two additional summations would enter the formula. Other models require more or fewer summations.

It is important to note that with the proper definition of G and y, (4) is identical to (3) and (5) is identical to (6).

Using a set-based approach with proper indices and index sets, (4) and (6) can be fully generalized as follows:

$$\sum_{(\sigma,\nu)} \left[{}^{\nu}G_{\sigma} \left[\prod_{(s,e)} y_{s,e} \right] \sum_{(s_m,e_0,e_1)} (y_{s_m,e_0} - y_{s_m,e_1})^{\nu} \right]. \tag{7}$$

The constituent array of a compound, introduced in [20], here denoted by σ, indicates which species may be present at which sublattices, and is paired with mixing order ν. In (7), σ ranges over the same constituent arrays as i, j, etc., and ν represents all possible orders similar to the k index in (3)–(6). The tuple (s,e) defines the sublattice-species pair and depends on to the constituent array σ; while the pairing (s_m, e_0, e_1) reflects the sublattice and species of mixing.

The essential difference between (3) or (5) and (7) is that (3) or (5) can only be used to model a particular fixed number of sublattices and mixing sites, while (7) needs no modification when used with varying sublattices and additional mixing sites. The following subsections elaborate the set based formulation, and the inclusion of order–disorder and magnetic contributions, resulting in the most general energy formula given in equations (8) and (28).

4.2. Set based energy formulation

The details of the new set-based energy formulation are now introduced. The following notation is used throughout the discussion: p denotes a particular phase, e indicates a species, and e_0, e_1 denote species in mixing. 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.

Using the correct index sets allows a single formula to accommodate any number of sublattices and any number of mixing sites per compound. The necessary index sets are discussed in detail further below, but the energy term without disorder or magnetic contribution can now be written. It is distinguished from the complete energy term $G^{(p)}$ and other similar terms by a subscript $\star$,

$$\begin{aligned} G_{\star}^{(p)} = & \sum_{(\sigma,\nu) \in S^{(p)}} \left\{ {}^{\nu}G_{\sigma}^{(p)} \left[\prod_{(s,e) \in T_{\sigma,\nu}^{(p)}} y_{s,e}^{(p)} \right] \sum_{(s_m,e_0,e_1) \in X_{\sigma,\nu}^{(p)}} (y_{s_m,e_0}^{(p)} - y_{s_m,e_1}^{(p)})^{\nu} \right\} \\ & - RT \sum_{(s,e) \in T^{(p)}} \left(a_s^{(p)} y_{s,e}^{(p)} \ln y_{s,e}^{(p)} \right). \end{aligned} \tag{8}$$

The sums are over all indices existing in the original data file, working from the outer to the inner sum and product. The first sum is over each constituent array and mixing order pair (σ, ν) which exists in the data for phase p. Then the product is over each sublattice and species pair (s,e) which exist in the data for that (p, σ, ν). In most cases 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

$$\left[\prod_{s,e} y_{s,e}^{(p)} \right] \sum_{s_m,e_0,e_1} (y_{s_m,e_0}^{(p)} - y_{s_m,e_1}^{(p)})^{\nu}. \tag{9}$$

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 [15]).

When multiplying other terms, both the sum and product over the empty set equal one: $1 \cdot \sum_{\varnothing} \cdot = 1$, and $1 \cdot \prod_{\varnothing} \cdot = 1$. Similarly, when adding the sum or product over the empty set they are zero: $0 + \sum_{\varnothing} \cdot = 0$. Hence in the case of no mixing, the sum is empty and

$$^{\nu}G_{\sigma}^{(p)}\left[\prod_{s,e} y_{s,e}^{(p)}\right] \sum_{s_m,e_0,e_1} (y_{s_m,e_0}^{(p)} - y_{s_m,e_1}^{(p)})^{\nu} = {}^{\nu}G_{\sigma}^{(p)}\left[\prod_{s,e} y_{s,e}^{(p)}\right]. \tag{10}$$

(see (5.67) and (5.70) in [15]).

The energy for a phase will be a sum of terms like (9) with their respective coefficients $^{\nu}G_{\sigma}^{(p)}$, plus the entropy term. In [15] 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

$$-RT \sum_{(s,e)} \left(a_s^{(p)} y_{s,e}^{(p)} \ln y_{s,e}^{(p)}\right), \tag{11}$$

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

Example 1. A simple two sublattice phase; p:= Al$_3$Pt$_2$.

In the Al–Pt binary system, consider the phase which is the stoichiometric compound p:= Al$_3$Pt$_2$, where the sole constituent array Al:Pt is modeled with two sublattices having site ratios $a_1^{(p)} = 0.6$ 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, $\sigma =$ Al : Pt, ν=0, hence for this phase

$$\begin{aligned} G_{\star}^{(p)} = {} & {}^{0}G_{\text{Al:Pt}}^{(p)} y_{1,\text{Al}}^{(p)} y_{2,\text{Pt}}^{(p)} \\ & - RT\left(a_1^{(p)} y_{1,\text{Al}}^{(p)} \ln y_{1,\text{Al}}^{(p)} + a_2^{(p)} y_{2,\text{Pt}}^{(p)} \ln y_{2,\text{Pt}}^{(p)}\right) \end{aligned} \tag{12}$$

Note that the coefficient $^{0}G_{\text{Al:Pt}}^{(p)}$ 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.

4.3. Index sets

Index sets are used in the sums and products in (8), 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,

$$\text{constituent arrays of } p \quad S^{(p)} \quad \equiv \{(\sigma, \nu) | (\sigma, \nu) \text{ in } p\}, \tag{13}$$

$$\text{species sites of } \sigma \quad T_{\sigma,\nu}^{(p)} \quad \equiv \{(s, e) | (s, e) \text{ in } \sigma\}, \tag{14}$$

$$\text{species sites of } p \quad T^{(p)} \quad \equiv \bigcup_{(\sigma,\nu) \in S^{(p)}} T_{\sigma,\nu}^{(p)}, \tag{15}$$

$$\text{mixing sites of } \sigma \quad X_{\sigma,\nu}^{(p)} \quad \equiv \{(s_m, e_0, e_1) | (s_m, e_0, e_1) \text{ in } \sigma\}. \tag{16}$$

Example 2. A simple phase with multiple compounds; p:= Laves.

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 $S^{(p)} = \{(\text{Al:Al}, 0), (\text{Al:Ni}, 0), (\text{Ni:Al}, 0), (\text{Ni:Ni}, 0)\}$. For (Al:Al, 0) the set of (s,e) pairs is $T_{\text{Al:Al},0}^{(p)} = \{(1, \text{Al}), (2, \text{Al})\}$, for (Al:Ni, 0) the ($s$,$e$) pairs are (1, Al) and (2, 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

$$\begin{aligned} G_{\star}^{(p)} = {} & {}^{0}G_{\text{Al:Al}}^{(p)} y_{1,\text{Al}}^{(p)} y_{2,\text{Al}}^{(p)} + {}^{0}G_{\text{Al:Ni}}^{(p)} y_{1,\text{Al}}^{(p)} y_{2,\text{Ni}}^{(p)} \\ & + {}^{0}G_{\text{Ni:Al}}^{(p)} y_{1,\text{Ni}}^{(p)} y_{2,\text{Al}}^{(p)} + {}^{0}G_{\text{Ni:Ni}}^{(p)} y_{1,\text{Ni}}^{(p)} y_{2,\text{Ni}}^{(p)} \\ & - RT\left(a_1^{(p)} y_{1,\text{Al}}^{(p)} \ln y_{1,\text{Al}}^{(p)} + a_1^{(p)} y_{1,\text{Ni}}^{(p)} \ln y_{1,\text{Ni}}^{(p)}\right. \\ & \left. + a_2^{(p)} y_{2,\text{Al}}^{(p)} \ln y_{2,\text{Al}}^{(p)} + a_2^{(p)} y_{2,\text{Ni}}^{(p)} \ln y_{2,\text{Ni}}^{(p)}\right). \end{aligned} \tag{17}$$

Example 3. A simple phase with mixing; p:= liquid 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 $S^{(p)} = \{(\text{Co}, 0), (\text{Mo}, 0), (\text{Co}, \text{Mo}, 0), (\text{Co}, \text{Mo}, 1)\}$. 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: $X_{\text{Co,Mo},0}^{(p)} = X_{\text{Co,Mo},1}^{(p)} = \{(1, \text{Co}, \text{Mo})\}$. There are four terms in the outer sum, so

$$\begin{aligned} G_{\star}^{(p)} = {} & {}^{0}G_{\text{Co}}^{(p)} y_{1,\text{Co}}^{(p)} + {}^{0}G_{\text{Mo}}^{(p)} y_{1,\text{Mo}}^{(p)} \\ & + {}^{0}G_{\text{Co,Mo}}^{(p)} y_{1,\text{Co}}^{(p)} y_{1,\text{Mo}}^{(p)} + {}^{1}G_{\text{Co,Mo}}^{(p)} y_{1,\text{Co}}^{(p)} y_{1,\text{Mo}}^{(p)} \left(y_{1,\text{Co}}^{(p)} - y_{1,\text{Mo}}^{(p)}\right) \\ & - RT\left(a_1^{(p)} y_{1,\text{Co}}^{(p)} \ln y_{1,\text{Co}}^{(p)} + a_1^{(p)} y_{1,\text{Mo}}^{(p)} \ln y_{1,\text{Mo}}^{(p)}\right). \end{aligned} \tag{18}$$

4.4. Disordered contribution

The general formula for $G_{\star}^{(p)}(y^{(p)})$ defined in (8) is used for modeling order–disorder transition:

$$G^{(p)} = G_{\text{dis}}^{(p)}(x^{(p)}) + G_{\text{ord}}^{(p)}(y^{(p)}) - G_{\text{ord}}^{(p)}(y^{(p)} = x^{(p)}), \tag{19}$$

see (5.144) and (5.145) from [15].

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 (19).

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 (19).

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 (19).

Thus a “disordered contribution” is defined, analogous but complimentary to the “ordered contribution” in [15],

$$\Delta G^{(p)} := G_{\text{dis}}^{(p)}(x^{(p)}) - G_{\text{ord}}^{(p)}(y^{(p)} = x^{(p)}). \tag{20}$$

Using this to update (19),

$$G^{(p)} = G_{\star}^{(p)} + \Delta G^{(p)} \tag{21}$$

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

$$\begin{aligned} G_{(q)}^{(p)} = {} & \sum_{(\sigma,\nu) \in S^{(q)}} \left\{ {}^{\nu}G_{\sigma}^{(p)} \left[\prod_{(s,e) \in T_{\sigma,\nu}^{(q)}} x_e^{(p)}\right] \sum_{(s_m,e_0,e_1) \in X_{\sigma,\nu}^{(q)}} (x_{e_0}^{(p)} - x_{e_1}^{(p)})^{\nu} \right\} \\ & - RT \sum_{(s,e) \in T^{(q)}} \left(a_s^{(q)} x_e^{(p)} \ln x_e^{(p)}\right). \end{aligned} \tag{22}$$

The above formula is the modification of $G_{\star}^{(p)}(y^{(p)})$, where p is the phase under consideration, q is the phase providing index sets and coefficients, and $y_{s,e}^{(p)}$ is replaced with $x_e^{(p)}$. Note carefully the placement of q in the sets $S^{(q)}$, $T^{(q)}$, $T_{\sigma,\nu}^{(q)}$, and $X_{\sigma,\nu}^{(q)}$, in contrast with p in the coefficients $^{\nu}G_{\sigma}^{(p)}$, and the replacement of $y^{(p)}$ variables with $x^{(p)}$.

Now the first and the third terms in (19) using $G_{(q)}^{(p)}$ can be expressed as follows

$$G_{\text{dis}}^{(p)} = G_{(\tilde{p})}^{(p)}, \tag{23}$$

$$G_{\text{ord}}^{(p)}(y^{(p)} = x^{(p)}) = G_{(p)}^{(p)}. \tag{24}$$

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

$$D^{(p)} = \{(q, \tilde{q}) \in D \mid q = p\}. \tag{25}$$

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,

$$\Delta G^{(p)} := G_{\text{dis}}^{(p)} - G_{\text{ord}}^{(p)}(y^{(p)} = x^{(p)}) = \sum_{(p,\tilde{p}) \in D^{(p)}} G_{(\tilde{p})}^{(p)} - G_{(p)}^{(p)}. \tag{26}$$

A phase p receiving no disordered contribution does not appear in the left hand side of any tuple in D, and in this case $D^{(p)} = \varnothing$, 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`.

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

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

4.5. Magnetic contribution and complete formulation

The model for magnetic contribution used here is given in full generality in [21], and applied, e.g., in [16] for Co–Mo using specific calculated values,

$$G_{\text{mag}}^{(p)} = RT \ln(\beta^{(p)} + 1) g(\tau). \tag{27}$$

See [21] for detail on β, $g(\cdot)$, and τ. The older Inden–Hillert–Jarl model is easily implemented as well, but not compared here.

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

$$G^{(p)} = G_{\star}^{(p)} + \Delta G^{(p)} + G_{\text{mag}}^{(p)}. \tag{28}$$

where $G_{\star}^{(p)}$ is defined in (8), and $\Delta G^{(p)}$ in (26).

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

$$G = \sum_{p \in P} G^{(p)} f^{(p)}, \tag{29}$$

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

4.6. Miscibility gap handling

The algorithm automatically creates n instances for each of the phases in any n-component system to handle miscibility gaps. For any binary system, the proposed formulation includes two instances for each of the existing phases, so that in the presence of a miscibility gap each is assigned a different phase fraction f and a different equilibrium composition x. Where no miscibility gap is present, the two instances will correspond to the same solution x, or one f will be zero. According to the results of several benchmarking tests run on binary and ternary systems, this simple idea allows for detection of miscibility gaps without a significant increase in computational complexity since the number of variables in the resulting optimization problem is small. Naturally, there is an added cost associated with this approach when it is applied to higher-dimensional multicomponent systems, and its advantage comparing to existing strategies ([4,3,22] etc) still needs to be investigated.

5. Automating phase diagram calculation

5.1. AMPL model

As depicted in Fig. 1, given the TDB database file, the Java converter produces the data and parameter files, which populate the generic model with the required index sets and parameter values in the AMPL syntax.

The mapper is run in AMPL, which uses the model file, the data file, and sources the parameter file each time a new temperature or pressure is to be examined. The mapper produces a database of phase diagram information as a set of CSV files. The diagram is then generated using any available graphical software package (MATLAB was used for this work).

In AMPL, the model and data are separated into three files: model, data and temperature/pressure dependent parameters. The data and parameter files are generated by automatic data conversion from a TDB file. A space sampling (mapping) script will read the model and data only once, while the T and P dependent parameters are read each time those values are adjusted. AMPL then executes the model at each sample point, once for each set of initial conditions to be tested. A variety of mapping scripts are possible and each will assemble the results of the many tests into a coherent image according to its purpose: a temperature/pressure diagram, a fixed-temperature Gibbs triangle, etc.

The *model* has several parts: declaration of sets and parameters, creation of necessary data structures, objective function, and constraints. The model file is read first by AMPL to declare all sets which will be encountered in the data file and how that data will be built into additional data structures. For example, for each phase the data file contains a list of all its possible sublattices. The model file contains definitions of sets such as a list of all possible constituent arrays in the model, and a list of all possible species at a sublattice, both of which are built from the list of constituent arrays in the data file.

The *data* file contains the essential minimum of information from the TDB file, the remaining necessary data structures are created by the model file from that minimum data. The data structures in the data file are: (1) elements/species; (2) phases; (3) sublattices and their order of mixing; (4) symmetries; (5) disordered contributions; (6) site ratios; (7) magnetic coefficients; (8) multiplicity of phases (to cover miscibility gaps); and (9) parameters for each named formula in the TDB.

The temperature dependent *parameter* file has a format similar to the TDB, with minimal sufficient alteration into AMPL syntax. AMPL fixes the parameters with `let` statements which must be called each time the temperature is changed. An equivalent formulation would be to create equality constraints and rely on the AMPL `substout` option to treat them as computed parameters.

This type of implementation would eliminate the need to set the parameters each time the temperature gets changed.

As mentioned in Section 3, in the presence of a miscibility gap the lowest energy phase is nonconvex, but this methodology allows multiple instances of a phase to be present in the sum for the objective function and span the nonconvex region, thus finding the true minimal energy. In this way miscibility gaps are automatically detected, without expert intervention.

5.2. Mapper

As a proof-of-concept algorithm for "mapping" a phase diagram, each temperature is evaluated independently of other temperatures. A minimal version of the algorithm is presented in pseudocode in Fig. 2. Many additional refinements are desirable, but not discussed in this paper.

To solve optimization problem (1) formulated in AMPL, two existing optimization codes were used: MINOS and SNOPT. MINOS implements a linearly constrained Lagrangian algorithm [12] while SNOPT uses Sequential Quadratic Programming [13] to solve a general nonlinear optimization problem [23]. The AMPL model loops over different mole fractions as well as temperatures within some range with a selected step size and calls both MINOS and SNOPT to solve (1) and to build a phase diagram. In most of runs both SNOPT and MINOS return the same solution, while occasionally one of them may find a local minimum. Using this hybrid scheme serves as safeguard and improves the likelihood of finding the global minimum. In the next section, detailed analysis of the computational results produced by this method are presented.

5.3. Converter

The converter handles parsing a Thermo-Calc [24] TDB database [25] into the necessary AMPL data format. Fig. 1 depicts the information flow, and the components of the process.

The converter is written in Java for maximum portability between systems and relative ease of maintenance by a diverse body of researchers. A full description of the converter is not relevant to this paper, however for clarity it must be noted that each PARAMETER statement in the TDB defines an energy function for a particular phase, constituent array, and mixing order. The PARAMETER statements collectively determine which species can be in which sublattices for that phase with that mixing order. The converter is able to handle wildcard notation used in Thermo-Calc databases. In particular, it is able to detect symmetry in binary interaction coefficients and correctly assign same parameter to multiple mixing order pairs in the AMPL description.

```
function MAPPER()
    load model as $P(x,T)$
    load system data
    $T \leftarrow T_{\min}$
    $x \leftarrow 0$
    while $T \le T_{\max}$ do
        load system parameters
        while $x \le 1.0$ do
            calculate $P(x,T)$
            if $\mathtt{card}\big(P(x,T)\big) = 2$ then
                mark $x_{\text{upper}}$ and $x_{\text{lower}}$ on diagram
                $x \leftarrow x_{\text{upper}} + \Delta_x$
            else
                $x \leftarrow x + \Delta_x$
            end if
        end while
        $T \leftarrow T + \Delta_T$
    end while
end function
```

Fig. 2. Mapping algorithm for binary systems. $P(x,T)$: the set of phases present at composition x and temperature T. 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 $x_{\text{lower}} \le x_{\text{upper}}$.

6. Numerical tests

Example systems with varying features have been examined to ensure the phase diagrams calculated using the "push-button" approach made possible by the new unified AMPL-based formulation are in correspondence with the current state of the art.

Two binary systems and one ternary are examined, chosen for their interesting thermodynamic features and for having been recently studied in the literature.

6.1. Co–Mo binary system

The Co–Mo system has phases with one (bcc, fcc, liquid), two (ϵ), three ($\sigma_{10,4,16}$) and four ($\mu_{1,2,6,4}$) sublattices. It also features a magnetic contribution and a miscibility gap as part of the fcc description, as described in [16,26]. The phase diagram produced by the present model appears in Fig. 3, compared against the diagram given in [26].

The diagram shows good correlation between the result produced by this method and that in [26]. The "horn" at 1200 K, 0.05 Mo, is precisely aligned, as are all phases below the liquidus. One discrepancy is that the AMPL model distinguishes different types of μ phases and σ phases which are not distinguished by the diagram in [26].

6.2. Al–Pt binary system

The Al–Pt system has phases with one (liquid, fcc), two (bcc, B2) and four ($\mathrm{L1_2}$) sublattices, it has order–disorder transition and miscibility gaps in B2 and $\mathrm{L1_2}$. fcc and bcc are disordered phases contributing to $\mathrm{L1_2}$ and B2 respectively. $\mathrm{L1_2}$ has a miscibility gap. The Al–Pt phase diagram produced by the present method appears in Fig. 4, compared with the stable diagram produced by running Thermo-Calc 4.0 (Mac version). The diagrams match precisely.

An interesting observation is that the diagram from [17] shows a triangular B2 region, around 1600 K and composition 0.56, which suggests existence of a stable B2 phase in that area. Using the TDB file from [17], the present implementation finds a different minimizer with B2 phase not taking part in the equilibrium. Thermo-Calc produces the same region of $\mathrm{L1_2}$ stability instead of the B2 triangle if the global optimization option is turned on, which is the default setting, but B2 shows up in a wrong place in the left corner of the diagram. The two alternative versions of stable Al–Pt diagram produced by Thermo-Calc with otherwise identical settings are given in Fig. 5. The total Gibbs energy produced by the AMPL code and by the global optimization within Thermo-Calc is GM = −181473.96 (corresponding to the existence of the $\mathrm{L1_2}$ stability region), where as by forcing the B2 phase to be stable one obtains in a slightly higher value of the total Gibbs energy of GM = −178278.65 according to the calculation made in Thermo-Calc.

An energy diagram of the region in question appears in Fig. 6, and a straight edge shows that the tangent between $\mathrm{Al_3Pt_5}$ and $\mathrm{L1_2}$ lies below the energy curve of B2, justifying the result shown in the diagram. The plot shows perfect agreement between Thermo-Calc and AMPL models in terms of the Gibbs energies.

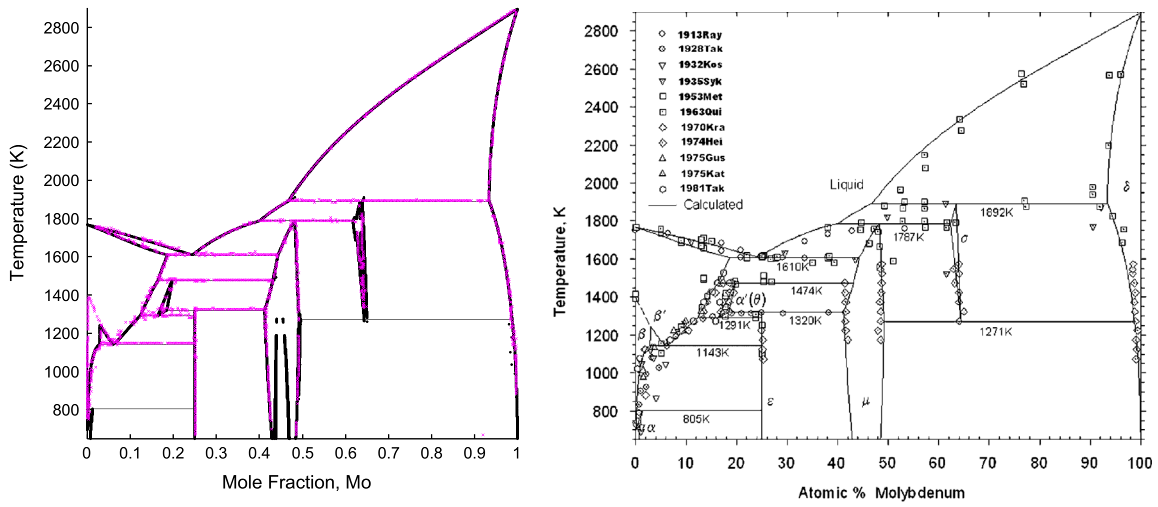

Fig. 3. The Co–Mo phase diagram produced by the new AMPL model (○) overlapped with the diagram from [26](×). (right). The Co–Mo phase diagram from [26].

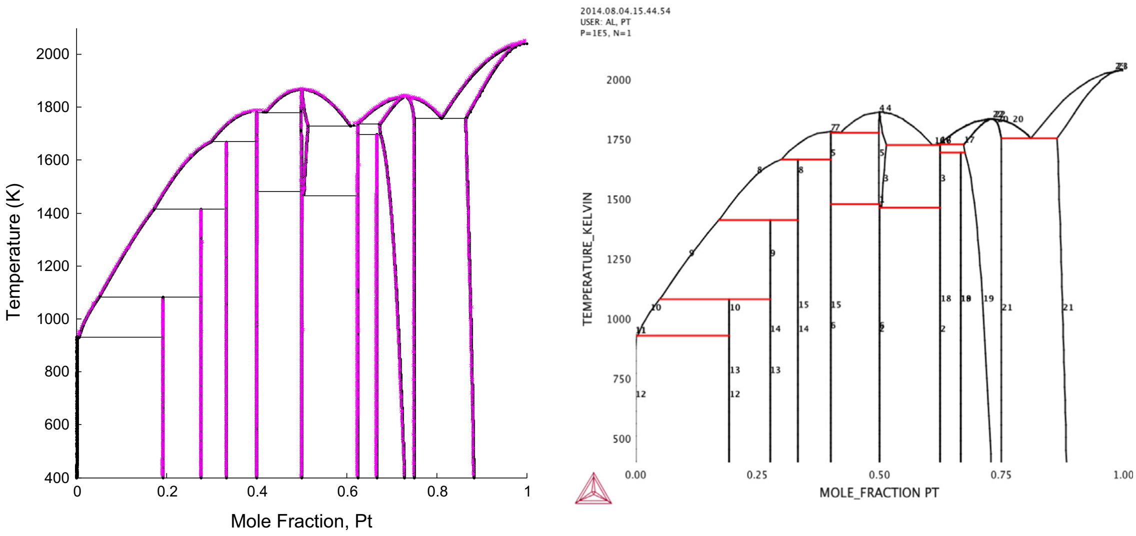

Fig. 4. (Left) The Al–Pt phase diagram produced by the AMPL model (○) overlapped with the diagram from Thermo-Calc (×). (Right) The Al–Pt diagram produced by Thermo-Calc, where $\mathrm{L1_2}$ phase is stable around $x(\mathrm{PT}) = 0.56$.

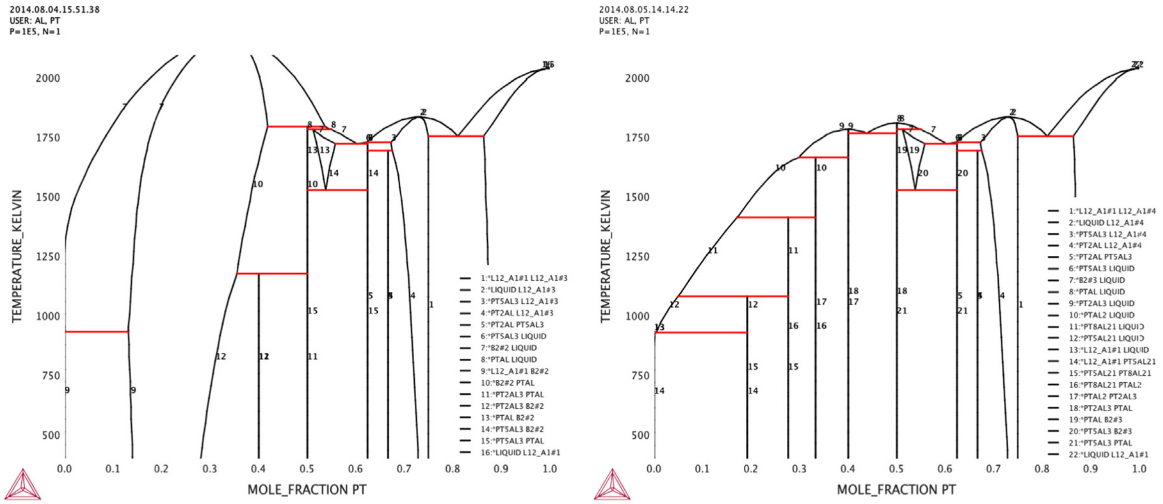

Fig. 5. The AlPt diagram produced by Thermo-Calc using: (left) `global optimization on` option; (right) `global optimization off` option, with same settings used otherwise.

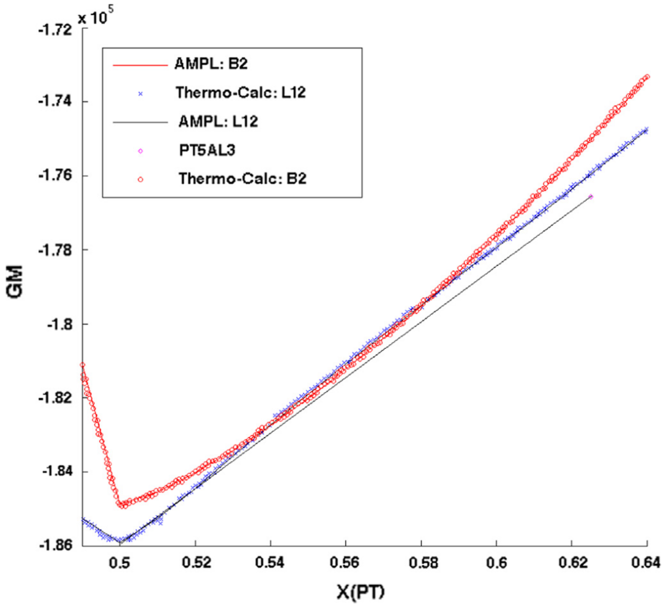

Fig. 6. The Al–Pt Gibbs energy detail. Around x(Pt) = 0.56 the B2 energy curve extends below the $L1_2$ energy curve. However the tangent between Al_3Pt_5 at composition 0.625 and $L1_2$ near 0.5 lies below B2 at all points, so B2 does not appear in the phase diagram near this temperature. Values obtained by Thermo-Calc match those obtained via AMPL.

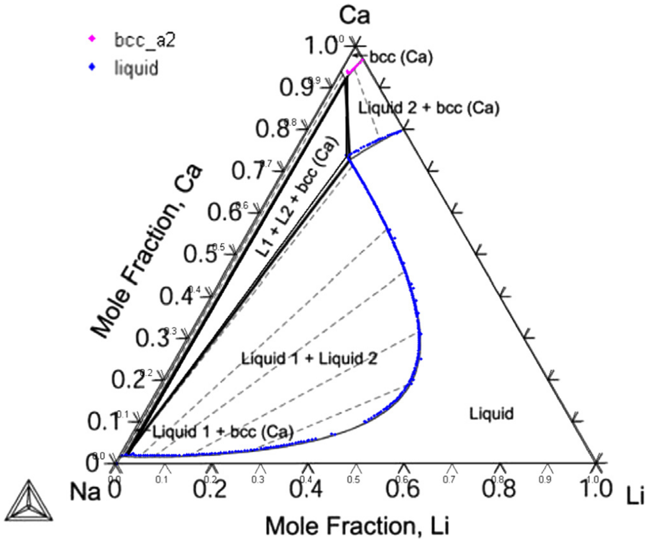

Fig. 7. The Ca–Li–Na phase diagram at 900 K, showing correct triple points and miscibility gap, overlapped with the diagram from [19].

It demonstrates the fact that AMPL code automatically found a lower minimum energy than the previous Thermo-Calc-based calculation.

The discovery of such conflicting data highlights the value of the present method. It allows scrutiny of earlier results obtained using other types of minimization routines and does not require careful choice of initial conditions to identify the correct minimum energy and corresponding phases.

6.3. Ca–Li–Na ternary system

The single three-species system Ca–Li–Na is examined, chosen for its challenging miscibility gap in the liquid phase. This system was thermodynamically examined using Thermo-Calc in [19] and later was subjected to automatization techniques in [4], where adaptive sampling method was shown to successfully detect the miscibility gap. Here same TDB description is used for testing the ability of the AMPL model to handle ternary systems. Note that the method of replicating each phase to detect miscibility gaps described in Section 4.6 is used here without resorting to any mesh adaptation, distinguishing it from the earlier approaches.

The Ca–Li–Na phase diagram produced by this method appears in Fig. 7, with the diagram from [19] on top of it for comparison. The diagrams match, which shows that the triple points and liquidus are correctly detected, and that the miscibility gap has been handled without any intervention from the user.

Note that all of these features are automatically handled in the conversion and modeling process, and the calculated energy levels and resulting diagram match the state-of-the-art Thermo-Calc diagrams with the exception of several discrepancies mentioned above.

7. Discussion

The result of this effort is an approach which solves a Gibbs energy minimization problem using an existing Thermo-Calc database without requiring expert input. One advantage of the model is its flexibility. It enables a “push button” approach requiring no input from the user, ideal for demo or student use. For more experienced users it allows modification of the model and the underlying solver, both readily accessible within the framework. If the user is unaware of the presence of a miscibility gap the result is still correctly produced. The novelty of this new strategy lies in representing the objective function in a completely system-independent way by means of a set-based formalism compatible with state-of-the-art optimization tools. The constrained optimization formulation using this unified paradigm also allows other important investigations such as finding the stable phases at a certain composition or identifying the minimal temperature with certain phase compositions, without changing the model. The model is well suited for other types of calculations, such as energy diagrams including energies of metastable phases; the energy diagram in Fig. 6 is an example of using the model for this purpose. The set based framework of global free energy minimization is readily expandable to other energy models, e.g., Helmholtz free energy [27], and for the computation of other thermodynamic quantities of interest, such as enthalpies, activities etc. The built-in capability of AMPL to symbolically compute partial derivatives is a big bonus for these types of calculations. Further, it is natural to extend the use of this approach for computing the cooling path or other dynamic properties of the materials similarly to the way other software are being used in this context, e.g., as described in [2].

Here three test systems were investigated, with the Co–Mo and Ca–Li–Na systems performing as expected based on previously known data, and Al–Pt exhibiting interesting deviations in the B2 region, which prompts further investigation. As discussed above, a high sensitivity to the optimization method being chosen for investigation is observed in the case of close-lying energies like in the Al–Pt case, which suggests a re-evaluation of this and other phase diagrams and databases using new tools might be necessary. The availability of custom built solver options within this framework opens a pathway for new algorithmic developments which may help shed light on some of these issues. Different numerical methods can be tested and compared. The tools developed herein make identification of discrepancies in diagrams produced by different codes easier. A significant advantage of this open source methodology is that it separates the modeling and computational aspects, allowing concentration on each of them independently. The AMPL model developed in this paper due to its versatility significantly reduces the modeling effort associated with Gibbs energy formulation allowing to concentrate on computational aspects.

The benefits of the AMPL approach outlined above come with certain challenges. While AMPL has control structures such as `for` loops and `if` statements, it has no scoping of variables (i.e., sets and parameters are all global), nor function declaration in a programmatic sense. The only approximation to a subroutine is using the `include` statement to load a separate file from storage. The much needed Integrated Development Environment for AMPL was released earlier this year, too late for evaluation for this paper. These limitations make the implementation of sophisticated numerical algorithms difficult. However, as mentioned above, a large database of numerical solvers available within AMPL makes its use more straightforward.

It must be noted that there are multiple ways to sample points in the composition-temperature space and record phase data to create the diagram efficiently. This work has adhered to the simplest sampling strategy to demonstrate the feasibility of the present methodology. Optimal sampling schemes and their impact on performance of this algorithm will be discussed elsewhere.

It is also worth mentioning that the solvers used in this work may converge in theory to other stationary points such as saddles or local maxima, while from practical consideration the likelihood of that is near zero as all the used optimization methods employ descent strategies. Such a scenario has never been observed in our calculations. By testing various initial conditions the mapping algorithm increases the likelihood that the global minimum is identified. Employing global optimization strategies to ensure that a global minimum is found is another promising future research direction.

Future work will also include development of a graphical user interface allowing switching between various tasks, extensions of the framework to address other minimization problems such as liquidus calculations, and development of a custom-built solver to be used in conjunction with the AMPL model presented here.

Acknowledgments

The authors are grateful to anonymous referees for carefully reading the manuscript and offering insightful comments and suggestions that helped improve the presentation. We also wish to thank Zi-Kui Liu, ShunLi Shang, and Cuiping Guo of Pennsylvania State University for providing TDB files and resulting phase diagrams produced by Thermo-Calc for Al–Pt and Ca–Li–Na systems. We are indebted to Albert Davydov of NIST for sharing Co–Mo database descriptions and for the valuable discussions that contributed to the authors' ability to correctly interpret TDB files. The work conducted by undergraduate students Robert Hill (supported by the NSF CSUMS grant DMS-0639300) and Sandra Varela (supported by the NSF REU grant DMS-1062633) at early stages of the project is also gratefully acknowledged. ME was supported in part by NSF grant DMS-1056821. JS is grateful to The MITRE Corporation for the support of its Advanced Graduate Degree Program.

Appendix A. Supplementary material

Supplementary data associated with this article can be found in the online version at http://dx.doi.org/10.1016/j.calphad.2014.09.005.

References

[1] U. Kattner, The thermodynamic modeling of multicomponent phase equilibria, J. Med. JOM 49, 14–19, http://www.tms.org/pubs/journals/JOM/JOMhome.aspx.

[2] Y.A. Chang, S. Chen, F. Zhang, X. Yan, F. Xie, R. Schmid-Fetzer, W.A. Oates, Phase diagram calculation: past, present and future, Progr. Mater. Sci. 49 (2004) 313–345.

[3] S.-L. Chen, S. Daniel, F. Zhang, Y.A. Chang, X.-Y. Yan, F.-Y. Xie, R. Schmid-Fetzer, W.A. Oates, The pandat software package and its applications, CALPHAD 26 (2002) 175–188.

[4] M. Emelianenko, Z.-K. Liu, Q. Du, A new algorithm for the automation of phase diagram calculation, Comput. Mater. Sci. 35 (2006) 61–74.

[5] R. Fourer, D.M. Gay, B.W. Kernighan, AMPL A Modeling Language for Mathematical Programming, second edition, Brooks/Cole, Belmont, CA, 2003.

[6] S.-L. Shang, Y. Wang, Z.-K. Liu, ESPEI: extensible, self-optimizing phase equilibrium infrastructure for magnesium alloys, Magnes. Technol. (2010) 617–622.

[7] Open Calphad project. ⟨http://www.opencalphad.com/⟩, February 2014.

[8] T. Cool, A. Bartol, M. Kasenga, R.E. García, Gibbs: phase equilibria and symbolic computation of thermodynamic properties, CALPHAD 34 (December) (2010) 393–404.

[9] NanoHUB. ⟨http://nanohub.org/⟩.

[10] S.L. Chen, S. Daniel, F. Zhang, Y.A. Chang, W.A. Oates, R. Schmid-Fetzer, On the calculation of multicomponent stable phase diagrams, J. Phase Equilib. 22 (2001) 373–378.

[11] R.J. Vanderbei, LOQO User's Manual–Version 4.05., ⟨http://www.princeton.edu/∼rvdb/tex/loqo/loqo405.pdf⟩, Technical Report, Princeton University, 2006.

[12] B.A. Murtagh, M.A. Saunders, MINOS 5.5 User's Guide, Technical Report SOL 82-20R, Stanford University, 1983–1998.

[13] P.E. Gill, W. Murray, M.A. Saunders, User's Guide for SNOPT version 7: software for Large-Scale Nonlinear Programming, Technical Report. University of California, San Diego, Stanford University, 2008.

[14] AMPL website. ⟨http://www.ampl.com/⟩.

[15] H.L. Lukas, S.G. Fries, B. Sundman, Computational Thermodynamics, The Calphad Method, Cambridge University Press, New York, 2007.

[16] A. Davydov, U.R. Kattner, Thermodynamic assessment of the Co–Mo system, J. Phase Equilib. 20 (1) (1999) 5–16.

[17] D.E. Kim, V.R. Manga, S.N. Prins, Z.-K. Liu, First-Principles calculations and thermodynamic modeling of the Al–Pt binary system, CALPHAD 35 (2011) 20–29.

[18] M. Hillert, Phase Equilibria, Phase Diagrams and Phase Transformations: Their Thermodynamic Basis, Cambridge University Press, Cambridge, UK, 2008.

[19] S. Zhang, D. Shin, Z.-K. Liu, Thermodynamic modeling of the Ca–Li–Na system, CALPHAD 27 (2003) 235–241.

[20] B. Sundman, J. Ågren, A regular solution model for phases with several components and sublattices, suitable for computer applications, J. Phys. Chem. Solids 42 (4) (1981) 297–301.

[21] W. Xiong, Q. Chen, P. Korzhavyi, M. Selleby, An improved magnetic model for thermodynamic modeling, CALPHAD 39 (December) (2012) 11–20.

[22] C.W. Bale, E. Belisle, P. Chartrand, S.A. Degterov, G. Eriksson, K. Hack, I.-H. Jung, Y.-B. Kang, J. Melanon, A.D. Pelton, C. Robelin, S. Petersen, FactSage thermochemical software and databases—recent developments, CALPHAD 33 (June) (2009) 295–311.

[23] I. Griva, S.G. Nash, A. Sofer, Linear and Nonlinear Optimization, Second Edition, SIAM, Philadelphia, 2009.

[24] J.-O. Andersson, T. Helander, L. Höglund, P. Shi, B. Sundman, THERMO-CALC & DICTRA, computational tools for materials science, CALPHAD 26 (June) (2002) 273–312.

[25] Thermo-Calc Software AB., Thermo-Calc Database Guide, ⟨http://www3.thermocalc.se/res/Manuals/TC_databaseguide.pdf⟩, 2010.

[26] A. Davydov, U.R. Kattner, Revised thermodynamic description for the Co–Mo system, J. Phase Equilib. 24 (3) (2003) 209–211.

[27] X.-G. Lu, Q. Chen, A CALPHAD Helmholtz energy approach to calculate thermodynamic and thermophysical properties of fcc Cu, Philos. Mag. 89 (2009) 2167–2194.