The OpenCalphad thermodynamic software interface

Published version:

DOI:10.1016/j.commatsci.2016.08.045

Elsevier Editorial System(tm) for
Computational Materials Science
Manuscript Draft

Manuscript Number: COMMAT-D-16-00973R1

Title: The OpenCalphad thermodynamic software interface

Article Type: Full Length Article

Section/Category: Large Scale Systems

Keywords: Computational Thermodynamics; CALPHAD; Phase Transformations; Simulations; Parallel Computing; Free Software

Corresponding Author: Prof. Bo Sundman, PhD

Corresponding Author's Institution: -

First Author: Bo Sundman, PhD

Order of Authors: Bo Sundman, PhD; Ursula R Kattner, PhD; Christophe Sigli, PhD; Matthias Stratmann, PhD; Romain Le Tellier, PhD; Mauro Palumbo, PhD; Suzana G Fries, PhD

Abstract: Thermodynamic data are needed for all kinds of simulations of materials processes. Thermodynamics determines the set of stable phases and also provides chemical potentials, compositions and driving forces for nucleation of new phases and phase transformations. Software to simulate materials properties needs accurate and consistent thermodynamic data to predict metastable states that occur during phase transformations. Due to long calculation times thermodynamic data are frequently pre-calculated into ``lookup tables'' to speed up calculations. This creates additional uncertainties as data must be interpolated or extrapolated and conditions may differ from those assumed for creating the lookup table. Speed and accuracy requires that thermodynamic software is fully parallelized and the OpenCalphad (OC) software is the first thermodynamic software supporting this feature.

This paper gives a brief introduction to computational thermodynamics and introduces the basic features of the OC software and presents four different application examples to demonstrate its versatility.

Cover letter

This paper is a collaboration between several scientist to present the possibilities of a free thermodynamic software in different kinds of applications in materials science. This software provides the possibility to calculate equilibria in parallel and we believe this is a new feature not yet available in similar software and this feature is important for improving the future use of thermodynamics in simulations. We provide a brief description of the software and several examples of its use.

Response to reviewer, reviewer text indicated by ->

-> Reviewer #1: In this paper, the authors describe recent developments of the
-> OpenCalphad open-source thermodynamic software package and its
-> interfacing with 'client' programs. Several examples are given
-> to illustrate the possibilities for concrete applications.

-> The subject is interesting, and in my opinion the presented
-> material deserves publication. However, I believe that substantial
-> modifications to the manuscript are necessary before publication.

-> The problem for me is: what is the target audience, and what
-> is the purpose of the paper ? Some parts read almost like a
-> user manual for the program, and the appendices 2 and 3 belong
-> rather to the documentation part of a software website than to
-> a scientific article (appendix 2 is a list of functions -
-> what is its use in an article ? If I can click on each item
-> and get additional instructions, such as is the case in
-> well-designed web sites, then it's useful, but in paper form
-> I do not see what information this list should add). Other parts
-> are intended to explain to a general audience of materials
-> scientists the usefulness of the tools on the example of
-> concrete problems.

The authors are pleased with the careful review and we acknowledge that the paper has been too detailed in some parts and too brief in others.

The audience we are aiming for are all materials scientists who develop or use software using thermodynamic data. We have tried to expand the explanations of the basic functions and algorithm and we have removed the details of the current software, such as Appendices 2 and 3.

-> For me, the latter aspect of the paper should be reinforced.
-> But this requires adding quite some information - in many
-> places, the presented material is partial or sketchy. A few
-> concrete examples and recommendations are:

-> 1) After Eq. (1): what do you mean by 'surface of reference' ?
-> This is a non-standard term (at least for me)

The explanation of the Gibbs energy function for the models are now extended although it is difficult to make this short.

-> 2) Beginning of Section 2.3: The first paragraph is an
-> enumeration of other papers that I should read to understand
-> what is said here. I would expect a little more information
-> to make the paper self-contained. For example, what is the
-> prenciple of Hillert's algorithm and why is it better than
-> others ? How is the stability of a phase prescribed ?

The algorithm is explained in more details including the¨technique to select the set of stable phases.

Comparison with other (commercial) software is not easy as they have not published enough details of their algorithms. One of the authors has long experience working with the Thermo-Calc (TC) software and the OC software is a new implementation of the same algorithm. A very rough attempt to compare the calculation speed between OC and TC is to calculate a phase diagram for a 5 component steel, this takes OC 13 CPU seconds to calculate and the same diagram with TC takes 12 CPU seconds on the same computer. But a phase diagram calculation involves much more than just equilibrium calculations, there are step control, occational global minimizations etc. The authors are convinced the calculation speed and results using OC are comparable with commercial thermodynamic software but we have no "hard facts" to put in a paper.

-> 3) End of Sec. 2.4: there is a standard methodology to
-> demonstrate the scalability of an algorithm, that is, the
-> same calculation is performed with different number of
-> processors, and the execution time is compared. Here, only
-> one statement is made (in addition with the 'wishi-washi'
-> word 'almost'). We need some hard numbers here to make the
-> case for scalability.

We have added a table with a detailed comparison of sequential/parallel calculation. Also in section 4.2 there is a verifiable comparison using sequential/parallel calculations.

-> 4) Section 3, beginning of second paragraph: I do not believe
-> that it is useful to announce things that are 'under development';
-> by the time the article is published, hopefully things will
-> have moved forward. Stated otherwise, things that are not finished
-> should not be published. Same for the next paragraph.

OCASI is fully operational but the documentation is rudimentary and incomplete. There will also be a revision based on the experiences of the use.

-> 5) Section 3.1: it is very unusual to refer to a Ph.D. thesis
-> unless it is already publicly available.

The sentence was reworded and the in-text reference to the thesis was removed.

-> 6) Section 4: 'Some of the examples presented here may require
-> the use to proprietary software'. Since the examples were all
-> calculated, I do not understand the 'may'. Which of the examples ?
-> Which software ?

The proprietary software using OCASI is the one used for the simulation of the aluminium alloy in section 4.2. This is now indicated in the text.

-> 7) Section 4.2, second paragraph (page 14): the simulation setup
-> is not at all described. What does 'a two-dimensional cylindrical
-> geometry' mean ? Are all the potential readers supposed to know
-> what is back-diffusion, the Scheil model, and the secondary
-> dendrite arm spacing ? A more detailed description is needed here.

This section has now been extended with additional references.

-> 8) End of Section 4.2: again, problems with the documentation of

-> scalability. The ratio between 3.5 days and 3.5 hours is 24;
-> this cannot be explained by the use of 12 processors. How much
-> is the execution time of the SAME software on a single processor ?

A precise figure is now given for the effect of parallelization. The previous value was comparing the old sequential version of the software using a different equilibrium algorithm with the revised using OCASI, i.e. included many other changes than parallelization.

-> In summary, the article should be enriched and made more self-consistent
-> to be useful for a wider audience; appendices 2 and 3 could be deleted.

We have tried to make the article more readable and interesting and removed Appendices 2 and 3.

***Graphical Abstracts (for review)**

No graphical abstract

The OpenCalphad thermodynamic software interface

Bo Sundman[a], Ursula R Kattner[b], Christophe Sigli[c], Matthias Stratmann[d], Romain Le Tellier[e], Mauro Palumbo[f], Suzana G Fries[d]

[a] *INSTN, CEA Saclay, France*
[b] *National Institute of Standards and Technology, Gaithersburg, MD, USA*
[c] *Constellium Technology Center, CS 10027, F-38341 Voreppe Cedex, France*
[d] *Ruhr University Bochum, Germany*
[e] *CEA, DEN, DTN/SMTA/LPMA Cadarache, F-13108 St Paul-lez-Durance, France*
[f] *SISSA, Trieste, Italy*

Abstract

Thermodynamic data are needed for all kinds of simulations of materials processes. Thermodynamics determines the set of stable phases and also provides chemical potentials, compositions and driving forces for nucleation of new phases and phase transformations. Software to simulate materials properties needs accurate and consistent thermodynamic data to predict metastable states that occur during phase transformations. Due to long calculation times thermodynamic data are frequently pre-calculated into "lookup tables" to speed up calculations. This creates additional uncertainties as data must be interpolated or extrapolated and conditions may differ from those assumed for creating the lookup table. Speed and accuracy requires that thermodynamic software is fully parallelized and the OpenCalphad (OC) software is the first thermodynamic software supporting this feature.

This paper gives a brief introduction to computational thermodynam-

Email address: bo.sundman@gmail.com (Bo Sundman)

Preprint submitted to Computational Materials Science *August 9, 2016*

ics and introduces the basic features of the OC software and presents four different application examples to demonstrate its versatility.

Keywords: Computational Thermodynamics, CALPHAD, Phase Transformations, Simulations, Parallel Computing, Free Software,

1. Introduction

The background for the development of a free thermodynamic software for multicomponent calculations using the CALPHAD method (CALculation of PHAse Diagrams) is described by Sundman et al. [1]. A very limited first version of the OpenCalphad (OC) software was released in 2013. In February 2015, a second version, capable also of multicomponent single equilibrium and some phase diagram calculations, was released. During 2015, extensive new facilities were added and the software became much more stable and is available at [2]. Version 3 includes parallel equilibrium calculations using the OpenMP [3] library, a rudimentary assessment module and an Application Software Interface, OCASI, callable from C++ as well as Fortran and other programming languages. OC is written using the new Fortran standard, including 2008 additions.

There is extensive literature describing the use of thermodynamic data for materials and process development. Although somewhat outdated, an excellent review can be found in the book by Saunders and Miodownik [4]. Especially the development of phase field techniques [5, 6, 7] and other techniques, such as fluid dynamics [8] and finite element methods [9] for simulating micro-structure evolution has increased the demand for fast and reliable calculations of chemical potentials, driving forces, mobilities etc. Within the framework of Integrated Computational Materials Engineering (ICME) [10] the necessary tools are coupled to accomplish the desired simulations and the OC software has the intention of becoming one of these tools.

The OC software has been extensively tested on Windows and Linux systems and, although there are occasional problems, the overall performance is comparable with that of commercial thermodynamics software for materials science, such as FactSage [11], Pandat [12] or Thermo-Calc [13]. In addition, the OC software can perform many equilibrium calculations in parallel, including calls to OCASI. The OCASI interface is fully functional and has been used in some applications presented here to demonstrate how it can be integrated in simulations using consistent thermodynamic data. Examples and full documentation of the source code, as well as macro files showing the interactive use of the software, are available on the OC web site [2]. A development version with several new features is available at the github repository [14].

2. Computational Thermodynamics

Computational thermodynamics consists of software and databases for calculating phase diagrams and all other kinds of thermodynamic data. The databases are a collection of parameterized model descriptions of the phases as function of constitution and state variables. The parameterized model descriptions are obtained from the thermodynamic assessment (or optimization) of individual systems. The basic techniques are described in the book by Lukas et al. [15]. An important advantage from using an assessed thermodynamic database is that all data calculated with such a database are consistent, including phase solubilities, melting temperatures, heat capacities, chemical potentials etc.

2.1. State variables

A thermodynamic system is described by a set of state variables such as T, P, V for temperature, pressure, volume. The amounts of the components can be given in moles, N, mole fractions, x, or mass fractions, w. All these variables determine functions of the state, such as internal energy, U, Helmholtz energy, A, or the Gibbs energy, G.

Many properties derived from the state functions can be calculated, or even used to control the system like chemical potentials, μ, enthalpies, H, etc. All state variables available in OC are listed in the Appendix.

2.2. Thermodynamic models

In CALPHAD-type thermodynamic databases the Gibbs energy of each phase is described by a model reflecting its structure. Several different kinds of models are described by Lukas et al. [15]. Many different kinds of models, for example ideal gases, regular solutions, interstitial models and chemical ordering are included in the Compound Energy Formalism (CEF)[16]. This formalism describes the Gibbs energy per mole formula unit, G_M, of the phase α with several sublattices as:

$$G_M^\alpha(T,P,y_{is}) = {}^{\mathrm{srf}}G_M^\alpha - T\,{}^{\mathrm{cfg}}S_M^\alpha + {}^{E}G_M^\alpha + {}^{\mathrm{phys}}G_M^\alpha \qquad (1)$$

where T, P are temperature and pressure and y_{is}^α is the fraction of constitent i on sublattice s and

$${}^{\mathrm{srf}}G_M^\alpha = \sum_I \Pi_I(y_{is}^\alpha)\,{}^\circ G_I^\alpha(T,P) \qquad (2)$$

$$
\begin{aligned}
{}^{\mathrm{cfg}}S^{\alpha}_{M} &= RT\sum_{s} a^{\alpha}_{s} \sum_{i} y^{\alpha}_{is}\ln(y^{\alpha}_{is}) && (3)\\
{}^{E}G^{\alpha}_{M} &= \sum_{J} \Pi(y^{\alpha}_{js}) L^{\alpha}_{J}(T,P,y^{\alpha}_{js}) && (4)\\
& && (5)
\end{aligned}
$$

${}^{\mathrm{srf}}G^{\alpha}_{M}$ is the so-called surface of reference where I is a constituent array specifying one constituent i in each sublattice, s and ${}^{\circ}G^{\alpha}_{I}$ is the Gibbs energy of formation of the compound I with the α structure from the standard states of the elements. ${}^{\mathrm{cfg}}S^{\alpha}_{M}$ is the configurational entropy, assuming random mixing on each sublattice, a^{α}_{s} is the number of sites on sublattice s and R the gas constant. ${}^{E}G^{\alpha}_{M}$ contain modeling parameters that describe the composition dependence due to interactions between constituents specified by J. ${}^{\mathrm{phys}}G^{\alpha}_{M}$ can be used for particular physical contributions such as from ferro-magnetism. The magnetic contribution to the Gibbs energy is usually modeled with a composition dependent Curie or Néel temperature and Bohr magneton number [17].

For liquids with strong short range ordering (SRO), the partially ionic 2-sublattice liquid model [18] is also implemented. Since the source code of OC is available, skilled users can implement their own models.

The mole fraction of a component j in a phase, x^{α}_{j}, can be calculated from the constituent fractions using

$$
x_j = \frac{\sum_s a_s \sum_j b_{ji} y_{is}}{\sum_k \sum_s a_s \sum_i b_{ki} y_{is}} \tag{6}
$$

where b_{ji} is the stoichiometric factor of element j in constituent i.

2.3. Equilibrium calculations

The model of the Gibbs energy of a phase forms a surface in a hyperspace as a function of T, P and its constitution. A stable state, i.e. an equilibrium, for a given set of conditions, is described by the minimum or maximum of the appropriate thermodynamic function. From classical thermodynamics we know that a system at constant T, P and amounts of all elements is at equilibrium at a minimum in the Gibbs energy of the system. The equilibrium Gibbs energy is given by:

$$G_m = \sum_i x_i \mu_i \tag{7}$$

where G_m is Gibbs energy per mole of component and μ_i the chemical potential of element i. If the system contains several phases we have for each stable phase

$$G_m^\alpha = \sum_i x_i^\alpha \mu_i \tag{8}$$

$$x_i = \sum_\varphi \aleph^\varphi x_i^\varphi \tag{9}$$

where $\aleph^\varphi$ is the number of formula units of phase φ.

The equilibrium algorithm used in OC was proposed by Hillert [19] and implemented by Jansson [20] in the POLY module of the Thermo-Calc software and in the PMLFKT software by Lukas et al. [21]. The implementation of this algorithm in OC is presented in [22]. It uses the chemical potentials and the amount of the stable phases as variables to find the equilibrium by iteratively varying the constituent fractions in all phases. If the amount of a stable phase becomes less than zero at an iteration it is removed from the stable phase set. If the Gibbs energy of an unstable phase, at its current composition, becomes lower than the Gibbs energy calculated for this composition using the current values of the chemical potentials, this phase is added to the stable phase set. Such changes in the set of stable phases must be treated with great care as described in[22].

The algorithm minimizes the Gibbs energy for a set of phases, φ:

$$G = \sum_{\varphi} \aleph^{\varphi} G_M^{\varphi}(T, P, y_{is}^{\varphi}) \tag{10}$$

Different types of conditions and constraints can be used by adding the constraints multiplied with Lagrange multipliers. For known T, P and overall composition, the Lagrange function can be written:

$$\begin{aligned} L &= G + \sum_i f_i \mu_i + \sum_{\varphi} \eta_s^{\varphi} g_s^{\varphi} && (11) \\ f_i &= \sum_{\varphi} \aleph^{\varphi} x_i^{\varphi} - \tilde{N}_i && (12) \\ g_s^{\varphi} &= 1 - \sum_j y_{js}^{\varphi} && (13) \end{aligned}$$

where $\tilde{N}_i$ is the prescribed amount of element i and μ_i and η_s^{φ} are Lagrange multipliers. The Lagrange function will have the same extrema as the Gibbs energy when the constrains are fulfilled, i.e. zero. Additional terms with multipliers can be included for other conditions, for example if the volume is known or if the phase has charged constituents.

At equilibrim all partial derivatives of L should be zero and in partcular that with respect to the amount of the phases:

$$\frac{\partial L}{\partial \aleph^{\varphi}} = G_M^{\varphi} - \sum_i x_i \mu_i = 0 \tag{14}$$

and comparing with eq. 8 we find that the Lagrange multiplier μ_i is the chemical potential of element i. Hillert[19] proposed that the equilibrium calculation should be made in two steps which allows for both a change in the set of stable phases as well as the constitutions of the phases to obtain the minimum. Sundman et al.[22] give a detailed description of the implementation of Hillert's algorithm in OC and also show how to calculate additional properties such as heat capacities and slopes of liquidus surfaces from the results of an equilibrium calculation. OC has an additional grid minimizer to ensure that the calculated equilibrium is global and not local and to also to detect miscibility gaps in phases.

2.4. Parallel calculations

The assessment (optimization) of model parameters usually includes a large number of experimental data, each representing a value determined at equilibrium. The assessment procedure calculates these experimental equilibria many times varying the selected model parameters in order to find the best fit. Therefore, the ability to calculate multiple equilibria in parallel is not only useful for simulations but also for the assessment (optimization) of model parameters. The data structure in OC, described in [1], is specifically designed to allow calculation of multiple equilibria in parallel.

The OC source code is distributed with a macro file for calculating 400 separate equilibria of a 6 component steel. When tested on a 4 core Intel(R) i7-4810MQ @ 2.80GHz and 8 GB 64 bit memory using 8 threads, the time for an equilibrium calculation, either sequentially or with OpenMP and OC compiled with or without -O2 optimization, is shown in Table 1. The gain in clock speed using 8 threads is almost 4 i.e. there is very little overhead. Looping the same test for very long time did not show any loss of memory due to leaks and on a Linux system OC has been tested with the Valgrind

Table 1: Times to calculate an equilibrium for a 6 component steel either sequentially or with 8 thread/4 kernels, with or without -O2 compiler option using the GNU Fortran 5.1 version. The times are mean values of 10 calculations of 400 different equilibria.

Time version	seq.	seq. -O2	8 threads	8 threads -O2
CPU seconds	1.135310^{-2}	5.33910^{-3}	2.09210^{-2}	9.22010^{-3}
clock cycles	11.321	5.347	2.988	1.213

software [23] system which indicate negligible memory loss during calculations.

It is not possible to make accurate comparisons with commercial software because these softwares do not provide detailed information of the time to calculate an equilibrium. But running the same calculation with a similar commercial software on the same hardware gave the impression that there is no significant difference in time for sequential calculations. There is currently no commercial software which can calculate equilibria in parallel.

2.5. Additional data depending on T, P and phase composition

Thermodynamic data are important but not sufficient for the simulation of phase transformations. For example, to consider the diffusion of elements, their mobilities together with the thermodynamic factor need to be described to provide the diffusion coefficient. In addition, there are also interface energies, interface mobilities, elastic constants and other phase-based properties that may be of interest [24].

In the OC software, such property descriptions can be stored together with the thermodynamic descriptions for each phase as functions of T, P and constitution of the phase. The values of these properties are calculated together with the thermodynamic properties during the equilibrium calculation and can be accessed by the application software.

As an example, the natural logarithm of the mobility of Fe in the face centered cubic (FCC) phase, $M_{\mathrm{Fe}}^{\mathrm{FCC}}$, is denoted in OC as MQ&Fe(FCC), and in the Fe-Ni system it can be described by a linear function:

$$M_{\mathrm{Fe}}^{\mathrm{FCC}} = MQ\&\mathrm{Fe(FCC)} = x_{\mathrm{Fe}}^{\mathrm{FCC}} MQ\&\mathrm{Fe(FCC,Fe)} + x_{\mathrm{Ni}}^{\mathrm{FCC}} MQ\&\mathrm{Fe(FCC,Ni)} \tag{15}$$

where $x_{\mathrm{Fe}}^{\mathrm{FCC}}$ and $x_{\mathrm{Ni}}^{\mathrm{FCC}}$ are the mole fractions of Fe and Ni in FCC, respectively, $MQ\&\mathrm{Fe(FCC,Fe)}$ is the mobility of Fe in pure FCC Fe and $MQ\&\mathrm{Fe(FCC,Ni)}$ is the mobility of a single Fe atom in pure FCC Ni, both of which can be expressed as:

$$MQ\&\mathrm{Fe(FCC,Ni)} = \frac{Q_{\mathrm{Fe}}}{RT} \ln(\nu_{\mathrm{Fe}}) \tag{16}$$

where Q_{Fe} is the activation energy and ν_{Fe} the pre-exponential factor, both of which can depend on T and P.

As mobilities can vary several orders of magnitude, it is a better approximation to describe the composition dependence by a linear relation of its logarithm. If sufficient experimental or theoretical data are available, non-linear terms can be added to eq. 15. Together with the thermodynamic factor, the mobility can be transformed to a diffusion coefficient needed for the simulation of micro-structure evolution in diffusion controlled processes.

3. The OCASI application software interface

Software interfaces for coupling thermodynamic calculations with application software following a proposed standard [25] have been implemented in ChemApp [26] and Thermo-Calc TQ [27]. The idea behind this proposed standard is to overcome the differences between the different thermodynamic software packages which can make the implementation into application software difficult. The methodology with many separate subroutines and functions performing specific tasks makes the implementation straightforward and is also used for the OCASI.

A preliminary version of the OCASI software and documentation is available at the websites mentioned above. As the source code is open and free any user can directly access the whole source code and make modifications. However, any such access should be done with caution as the data structure and source code may change at a later update whereas the OCASI interface should remain stable.

3.1. Interfacing with Java

Based on Fortran and C++ compatibility, it is also possible to interface OC functionalities in the Java language thanks to the Java Native Interface (JNI) [28]. Such an interface has been developed for the PROCOR software [29], implementing a Cahn-Hilliard based multicomponent multiphase diffusion model [30]. This is briefly described in section 4.4. In addition to the Cahn-Hilliard model requirements, equilibrium calculation functionalities have been interfaced for the coupling with other physical models. Some physical parameters needed by these models have to be evaluated from the phase compositions obtained from the OC-based equilibrium calculations.

3.2. Reading the database

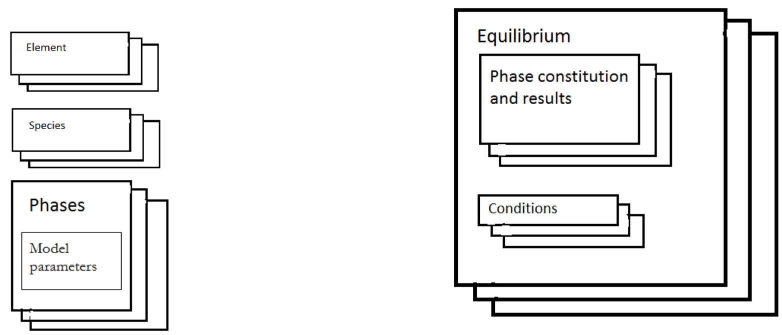
The thermodynamic data are normally read from a database. The OC software supports the TDB format [31] proposed originally by SGTE [32] and used by several other software, most notably by Thermo-Calc and DICTRA[13]. The application software allows selection of elements for which data should be extracted from the database and also allows suspending phases that are not of interest for the application.

The application software must initiate the OC memory structure by a special subroutine call. This returns a pointer to an equilibrium record in the dynamic memory structure which contains information about the constitution and calculated results for all the phases. After this initialization other subroutines, such as reading the database, can be used. The models of the phases and their thermodynamic model parameters read from the database are stored in the static part of the OC memory structure since they are independent of the external conditions that will be applied for a calculation. The external conditions, the constitution of the phases, as well as calculated results are in the dynamic data memory. A schematic representation of both static and dynamic memory is shown in Fig. 1.

3.3. Setting conditions

The conditions are set one by one by a call to a subroutine. These represent, for example, the temperature, T, pressure, P, and amount of the components, N_i, in the simplest case. The number of conditions needed to calculate the equilibrium is given by Gibbs phase rule

$$f = n + 2 - p \tag{17}$$



(a) (b)

Figure 1: The static (a) and dynamic (b) parts of the OC memory. Each equilibrium record in (b) is independent and can be calculated in parallel.

where f is the degrees of freedom, n the number of components and p is the number of stable phases. Normally the number of stable phases is the result of the equilibrium calculation but OC allows that one or more phases can be prescribed to be stable. If there are no prescribed stable phases, $f = n + 2$, i.e. a system with a single component must have 3 conditions, normally T, P and N where N is the size of the system. For a binary system 4 conditions must be set and so on. If there is a major component the size of the system can be given as N and the amounts of other components as mole or mass fractions, x_i or w_i. Chemical potentials or activities of the components can be used instead of amounts as conditions as well as many other state variables.

3.4. Creating additional equilibria

It is possible to create additional equilibrium records to represent various local equilibria in a simulation. These may be calculated sequentially or in parallel as they are independent. The software provides a pointer to each equilibrium record with its own dynamic data structure and such a pointer must be supplied in calls to most subroutines to indicate which equilibrium should be used.

3.5. Calculations

When the degrees of freedom according to eq. 17 are zero the equilibrium is calculated by a subroutine call. The calculation procedure in OC is described in detail in [22]. In the call it is possible to specify whether the global minimizer should be used or the calculation should start from the last calculated set of stable phases and constitutions. As already stated each equilibrium has its own set of conditions and can be calculated independently.

3.6. Extracting results

The normal way to obtain calculated results from OCASI is to call a subroutine using the state variable symbols as listed in the Appendix. The values of properties that are modeled separately, like the Curie temperature, mobilities, etc. can be obtained in the same way.

The pointer to the equilibrium data structure that was created when the database was read is used to extract the results of an equilibrium calculation for each phase. For example, with the equilibrium record pointer (called "ceq" below) the user can access the array of records for the phases (called "phase_varres") and, after an equilibrium calculation, extract the value of the Gibbs energy per mole formula unit of a phase with index lokcs as:

```
gfu = ceq%phase_varres(lokcs)%gval(1,1)
```

where the % sign is used in the 2008 Fortran standard to indicate items in a record. The lokcs variable is the index of the phase and provides an easy link between the phase and all its data. If the calculated mobility of component 1 of the same phase is stored in the location “imob”, its value is obtained by

```
mob(1) = ceq%phase_varres(lokcs)%gval(1,imob)
```

Great care must be taken that the relevant data is extracted and the application program must not change any data in the OC data structure.

4. Examples of application software

Several simple examples of how to use the OC software interface are provided with the source code. In one of the examples presented below (section 4.2), the OCASI library is used inside proprietary software.

4.1. The beginnings of a reactor module

A process can sometimes be simulated by dividing it into sections, each of which represent local equilibrium, and a transport of matter occurs between these by diffusion or other means. The equilibrium calculation can be done by OC in a function ”localequil” with the following arguments:

```
integer function localequil(todo,nsel,selel,tp,nmz,outspec,outval)
!
! This routine calculates an equilibrium and return values
! todo    in  character variable          with "commands" to be executed
! nsel    in  integer                     number of components
! selel   in  character*2 array           element names
! tp      in  double precison array(2)    with values of T and P
! nmz     in  double precision array(nsel) with molar content for elements
! outspec in  character variable          information which data to extract
! outval  out double precision array(*)   extracted values
! if there is no error the function value returned is zero
```

The todo argument passes a number of instructions to localequil, nsel gives the number of elements specified in argument selel, tp provides the values of T and P and nmz the element amounts. The argument outspec specifies which values will be returned in outval. A call of this function to calculate a local equilibrium for the system Si-O-C at 1800 K and 1 bar with a specified amount in moles of the elements can be

```
nsel=3
selel(1)='C '; selel(2)='O '; selel(3)='Si'
tp(1)=1.80D2; tp(2)=1.0D5;
nmz(si)=1.0D0; nmz(2)=2.0D0; nmz(3)=1.8D0
todo='verbose open cosi.tdb '
outspec='n(gas,*) mu(*) h h(gas) '
ierr=localequil(todo,nsel,selel,tp,nmz,outspec,outval)
```

The todo argument specifies VERBOSE to generate extra output, then to OPEN the database file COSI.TDB and read the data for the system with the elements in the argument selel and calculate the equilibrium at the specified values of T, P and amounts of the elements. After a successful calculation the amounts of all elements in the gas, n(gas,*), their chemical potentials, mu(*), the total enthalpy, h, and the enthalpy of the gas, h(gas), will be provided in the array outval. The state variables that can be used are listed in the Appendix. If an error occurs the function will return a non-zero number.

The localequil function will remember whether it has already read a database and if no database is specified in a subsequent call the routine will use the static data that it has already stored. In the application software the amounts of elements in the gas and the other phases may be redistributed between different local equilibria to simulate a process. The range of options in the localequil function can easily be extended. One of the intentions of this function is to provide an application programmer simple access to the OCASI interface.

4.2. Simulation of the homogenization of an alloy AA7449

The OpenCalphad Software has been recently coupled to the Constellium thermodynamic package, ProPhase, using OCASI in order to use the faster equilibrium routine included in OC as well as the new parallelization recently implemented. One application of this software is to simulate the solidification path and the segregations within dendrites formed after solidification.

The simplest model to simulate solidification path (liquid concentration evolution), solid fraction versus temperature curves and microsegregation during casting is the Scheil model[33]. The assumptions are threefold: equilibrium at the liquid/solid interface, no diffusion in the solid phases and infinite diffusion in the liquid phase. The coupling of a Scheil solidification routine with a phase diagram subroutine is straightforward and proceeds as follows. The temperature is first set equal to the liquidus temperature corresponding to the alloy composition. It is then decreased by a small decrement and the equilibrium is calculated. The alloy composition is then set equal to the liquid composition, the temperature decreased again by a small decrement and the equilibrium calculated again. This process is iterated until no liquid is found at equilibrium.

For fast diffusing atoms in the aluminum solid solution, like magnesium, silicon, copper, zinc and lithium, a more accurate treatment consists in evaluating, at each iterative step "s", the amount of back-diffusion, i.e., solute redistribution, occurring in solid solution[34]. This is done by using a finite difference scheme, by assuming local equilibrium in each volume element of secondary dendrite arms, by calculating the solute concentration in solid solution in each volume element (i.e. applying the phase diagram subroutine for the current temperature and the local alloy concentration) and by solving Fick's laws. For each time step, the finite difference routine performs many independent equilibrium calculations, one for each volume element, and is very well suited for parallelization.

The simulation of a laboratory heat treatment at 472°C of a multicomponent aerospace 7449 alloy (an Al-Mg-Zn-Cu-Fe-Si alloy see Table 2) after casting has been used to measure the gain in speed provided by OpenCalphad when parallelization is activated. The cast was performed in laboratory using the Aluminum Association TP-1 grain refiner test crucible. The corresponding cast structure, as shown in Fig. 2, was simulated using the finite-difference routine described above. For that particular case, a cylindrical geometry was chosen with a diameter of 61 μm corresponding to spacing between secondary dendrite arms. Care must be taken to mesh the dendritic cylinder with sufficiently small enough radius increments especially towards the end of solidification where the concentration gradients may be very steep.

Table 2: Composition range of AA7449 according to Aluminium Association

	Si	Fe	Cu	Mn	Mg	Zn	Ti+Zr	rest
min	0	0	1.4	0	1.8	7.5	0	Al
max	0.12	0.15	2.1	0.2	2.7	8.7	0.25	Al

The evolution of composition gradients has been studied for a heat treatment consisting in a linear heating up of 40°/h up to 472°C followed by a holding of 10h at 472°C. The corresponding predicted and measured incipient melting temperatures and integrated enthalpies for the DSC first melting peak are compared in Fig. 3. Overall, a good agreement is obtained indi-

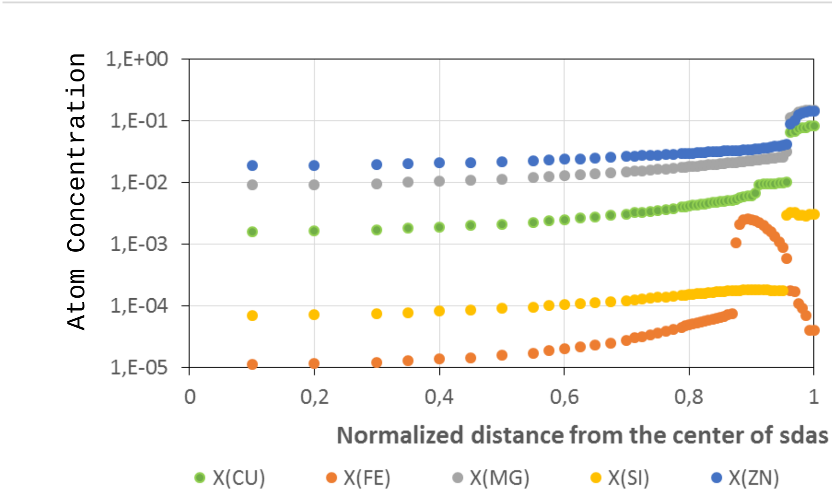


Figure 2: Simulated concentration gradients after solidification of aerospace alloy AA7449

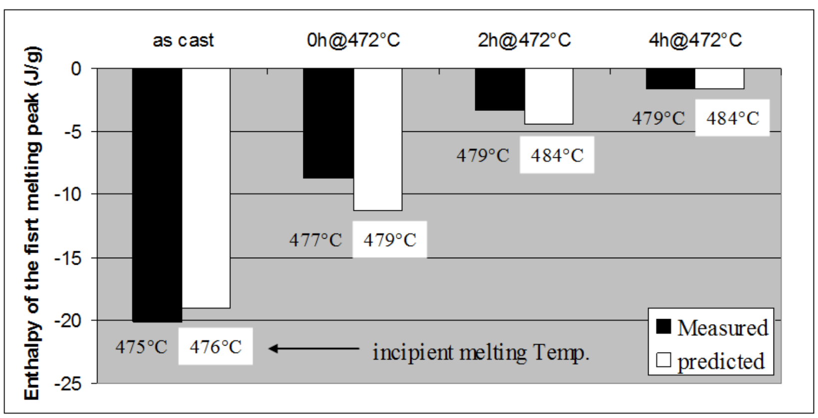


Figure 3: Simulated and predicted incipient melting temperatures and enthalpies of the first melting peak during calorimetric heat up at a rate of 20 °C/min.

cating that the proposed physical description of the phenomena occurring during solidification and homogenizing is acceptable.

This calculation was accelerated by a factor of 2.9 (from 239s to 82s) when using 4 cores instead of 1 (processor was Intel(R) i7-2760QM @2.4GHz).

4.3. Phase-field models and OCASI

Many modern thermodynamically consistent phase-field models [35, 36, 37, 38] rely on thermodynamic and kinetic properties that can be obtained or approximated from CALPHAD calculations. OC as an open source tool is perfectly suited for this purpose. The speed of the OCASI/C++ interface with additional parallelization allows for an efficient way to compute separate phase properties in the diffuse interfaces with equilibrium calculations or with separate phase compositions as input parameters. Coupled with the phase-field software framework, OpenPhase (OP), has been proven to be an efficient tool for mesoscale micro-structure simulations. As extensive phase data can be shared via this interface too, a user friendly simulation setup can be arranged.

The equations governing phase transformations and diffusion in the phase-field model with finite interface dissipation [38], require values of the molar Gibbs energies of the phases, G_m^α, their derivatives with respect to the phase constitution, $\partial G_m^\alpha/\partial y_{is}^\alpha$ and the atomic mobilities M_i^α. These can be calculated for a local equilibrium by the OCASI software by proving the values of the temperature T, pressure P and overall composition x_i.

Parallelization of the code does not only speed up the simulations and thus increasing the accuracy of the results, it also allows for a quick simulation setup of multicomponent, multiphase simulations, as the OP software is independent of the thermodynamic modeling details of the phases.

As an example we simulate the diffusion process between two austenitic steel samples with different alloy composition. The two samples, containing iron, manganese, silicon and carbon, are chosen to represent the uphill diffusion experiment by Darken[39]. Apart from the chosen thermodynamic and kinetic databases [40] under the Open Database License, only the temperature, the time step and the grid spacing had to be specified as in Tab. 3

Table 3: Simulation parameters

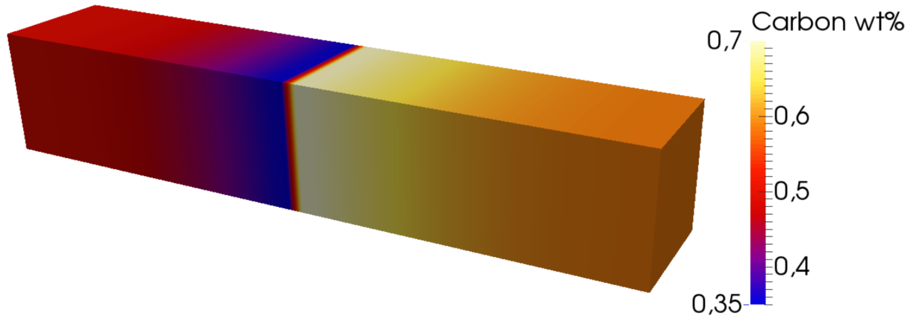
Box size	50x10x10
Grid spacing	0.001 m
Time step	1000 s
Temperature T	1323 K
Thermodynamic database	MatCalc mc_fe_v2.057.tdb [40]
Mobility database	MatCalc mc_fe_v2.008.ddb [40]

The results of the quaternary simulation for the duration of 10 days at constant 1323 K are shown in Fig. 4 and are compared to the results of a DICTRA sharp-interface simulation as well as measurements from [39]. The results obtained with only openly available tools and data show excellent agreement with the results from the commercial DICTRA software[13].

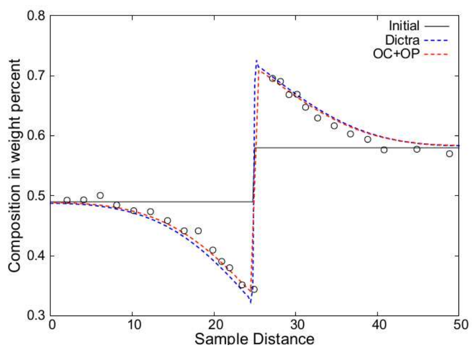
Beyond that, the coupling of OC and phase-field software like OP has been proven to be an efficient, accurate and easy to use tool for mesoscopic micro-structure simulations.

4.4. Mesoscopic modeling of a liquid miscibility gap

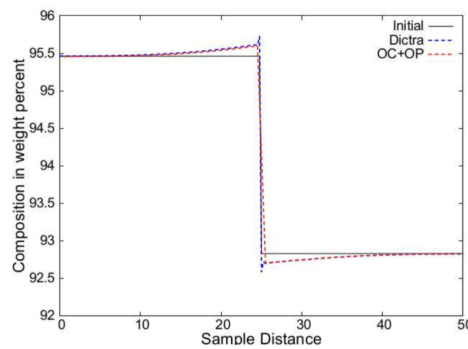
For the study of severe accidents in light water nuclear reactors (LWR), the OpenCalphad software has been used in the development of a Cahn-



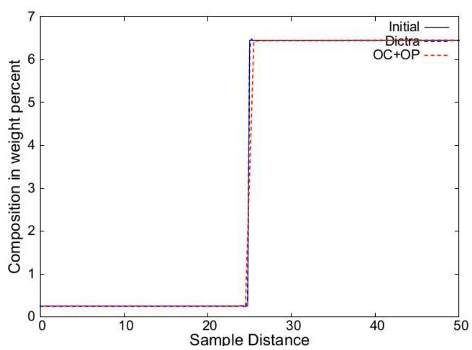
(a) Carbon composition over the whole simulation sample



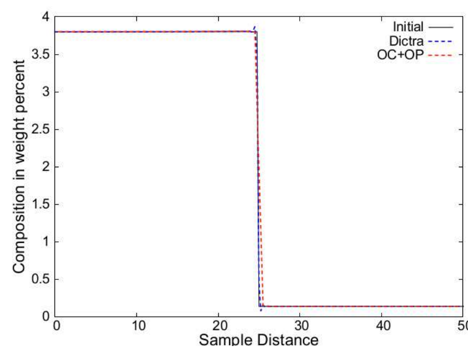
(b) Carbon composition



(c) Iron composition



(d) Manganese composition



(e) Silicon composition

Figure 4: Simulated composition profiles after 10 days of tempering at 1050 °C, with experimental measurements by [39] as circles.

Hilliard based diffusion model for treating the liquid-liquid miscibility gap associated with the corium (oxidic and metallic liquid materials present during a reactor core meltdown) in a reactor vessel lower head.

In-vessel corium is a complex thermodynamic system because of the phases that are present. In addition to partially oxidized cladding and fuel materials (UO_2, ZrO_2 and Zr), the corium contains stainless steel elements (Fe, Ni, Cr) from the reactor's internal structures. In particular, the U-O-Zr-Fe system exhibits a liquid-liquid miscibility gap responsible for a liquid-phases stratification of tremendous importance on the corium propagation [41].

This stratification phenomenon results from the combination of multicomponent multiphase species diffusion and gravitational material movement due to buoyancy effects initiated by Rayleigh-Taylor instabilities. The natural convection in a corium pool can be simulated by Computational Fluid Dynamics (CFD) and Cahn-Hilliard based mesoscopic modeling of the species diffusion appears to be a promising way to account for the stratification phenomenon in detailed simulations. This R&D effort regarding a detailed modeling of the thermohydraulic/thermochemical behavior of in-vessel corium has been undertaken at CEA Cadarache in support of integral models development in the context of the PROCOR software platform for sensitivity/uncertainty studies regarding corium propagation.

A Cahn-Hilliard based diffusion model was first developed for the U-O binary case [30] and is under development for U-O-Zr and U-O-Zr-Fe. Under the assumption of constant molar volume, the model for a n-component system is based on Cahn-Hilliard evolution equations for molar fractions of n-1 components with diffusion potentials, $\tilde{\mu}_i$, and is expressed as:

$$\tilde{\mu}_i = a\left(\frac{\partial \tilde{g}}{\partial x_i}\right)_{x_{j\neq i}} - \sum_j \kappa_{i,j}\nabla^2 x_j \tag{18}$$

where a and $\kappa_{i,j}$ are parameters of the model related to the interface thickness (a "numerical" parameter in this mesoscopic modeling) and interface tension while $\tilde{g}$ is the free energy density (in Jm^{-3}) of the homogeneous system.

In the binary U-O case, the $\left(\frac{d\tilde{g}}{dx_O}\right)$ term was calculated applying the chain rule to the liquid Gibbs energy G_m^{liq} and its derivatives $\partial G_m^{liq}/\partial y_i^{liq}$ obtained from the OCASI interface [30].

For the ternary and quaternary systems, the additional hypothesis that local equilibrium of the redox chemical reactions is instantaneously reached is made. Therefore, $\left(\frac{\partial \tilde{g}}{\partial x_i}\right)_{x_{j\neq i}}$ can be related to the chemical potentials obtained from "local" equilibrium calculations performed using the OCASI interface where the phase separation associated with the miscibility gap is not taken into account (*i.e.* the grid minimizer is turned off, *cf.* Section 2.3).

In order to illustrate this modeling, a 1D calculation for the U-O binary case is presented here; the domain length is $L = 10.0$ cm and the temperature and pressure conditions are $T = 3200$ K and $p = 1$ bar. Fig. 5 schematically depicts the initial configuration and the steady-state of the system. The interface thickness has been set to 1 cm.

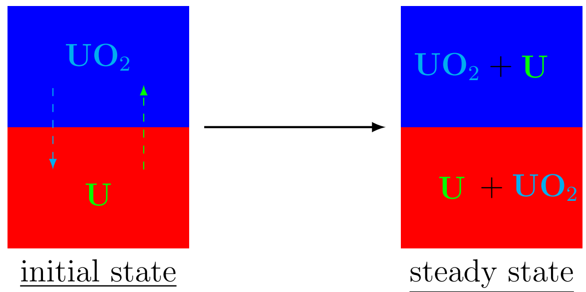


Figure 5: U-O system initial configuration and steady state

The initial configuration corresponds to an oxidic phase (uranium dioxide UO_2) above a metallic phase (in this case metallic uranium U). The diffusion process leads to a steady state configuration with two suboxidized phases. Fig. 6 shows the profile of oxygen molar fraction x_O in the initial state and at steady state where it is verified that, far from the interface, the oxygen molar fraction is equal to the values given by an equilibrium calculation.

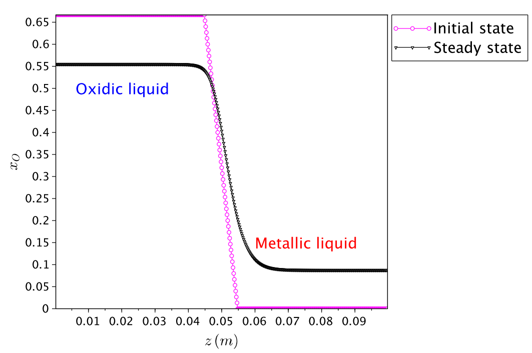


Figure 6: Profile of molar oxygen fraction x_O versus position z

5. Summary

Thermodynamic data needed in simulations, such as melting temperatures, solubilities, heat capacities etc., are often collected from different sources. In many cases little or no efforts are made to verify that these data are consistent. Data from calculations using thermodynamic databases have the advantage of providing consistent data but the fact that millions of equilibrium calculations that are needed in simulations takes considerable time poses a severe problem.

The possibility to calculate the local equilibrium for each point in space in parallel is a major improvement. Still, it may not be possible to completely avoid the use of interpolation methods in very large simulations with several thousand grid-points. In this case, the availability of parallel processing offers the the opportunity to update the results in real time during the simulation avoiding the need for pre-calculated look-up tables and thus allowing more accurate thermodynamic input for the simulation. The flexibility of software has been demonstrated with selected examples from very different application fields.

6. Acknowledgement

One author (BS) is grateful for a senior research grant from the Humboldt Foundation. Commercial products are identified for reference purposes this does not imply an endorsement by the National Institute of Standards and Technology.

References

[1] B. Sundman, U.R. Kattner, M. Palumbo and S.G. Fries, OpenCalphad - free thermodynamic software, Integrating Materials and Manufacturing Innovation, **4** (2015) 1, open access.

[2] OpenCalphad web page at http://www.opencalphad.org, visited 1 August 2016.

[3] An API for multi-platform shared-memory parallel programming at http://www.openmp.org

[4] N. Saunders, A.P. Miodownik, *Calphad*, in: Cahn R. W. (Ed.), Pergamon Materials Series, Vol 1, Oxford, 1997.

[5] Z.-K. Liu, L.-Q. Chen, P. Raghavan P, Q. Du, J.O. Sofo, S.A. Langer , C. Wolverton C, An integrated framework for multi-scale materials simulations and design. J Comput-Aided Mater Design 11 (2004) 183 - 199

[6] U. Grafe, B. Böttger, J. Tiaden, S.G. Fries, Coupling of multicomponent thermodynamic databases to a phase field model: application to solidification and solid state transformations of superalloys. Scripta Mater 42 (2000) 1179-1186.

[7] N. Warnken, D. Ma, A. Dreverman, R.C. Reed, S.G. Fries, I. Steinbach, Phase-field modeling of as-cast microstructure evolution in nickel-based superalloys. Acta Mater 57 (1997) 5862-5875.

[8] M.C. Schneider, J.P. Gu, C. Beckermann, W.J. Boettinger, U.R. Kattner, Modeling of micro- and macrosegregation and freckle formation in single-crystal nickel-base superalloy directional solidification. Metall Mater Trans A 28A (1997) 1517-1531.

[9] D.K. Banerjee, M.T. Samonds, U.R. Kattner, W.J. Boettinger, Coupling of phase diagram calculations for muliticomponent alloys with solidification micromodels is casting simulation software, in: J. Beech, H. Jones (Eds). Solidification Processing 1997: Proceedings of the 4th Decennial International Conference on Solidification Processing, Department of Engineering Materials, University of Sheffield, Sheffield, 1997, pp 354-357

[10] National Research Council, Committee on Integrated Computational Materials Engineering, Integrated computational materials engineering: a transformational discipline for improved competitiveness and national security, National Research Council, Committee on Integrated Computational Materials Engineering, Washington, DC, 2008.

[11] C.W. Bale, E. Bélisle, P. Chartrand, S. Decterov, G. Eriksson, K. Hack, I.-H. Jung, Y.-B. Kang, J. Melan con, A.D. Pelton, C. Robelin, S. Petersen, FactSage thermochemical software and databases - recent developments, Calphad 33 (2009) 295-311.

[12] W. Cao, S.-L. Chen, F. Zhang, K. Wu, Y. Yang, Y.A. Chang, R. Schmid-Fetzer, W.A. Oates, PANDAT software with PanEngine,

PanOptimizer and PanPrecipitation for multi-component phase diagram calculation and materials property simulation, Calphad 33 (2009) 328-342.

[13] J.-O. Andersson, T. Helander, L. Höglund, P. Shi, B. Sundman, Thermo-Calc & DICTRA, computational tools for materials science, Calphad 26 (2002) 273-312.

[14] OpenCalphad repository at http://www.github.com, visited 1 August 2016.

[15] H.L. Lukas, S.G. Fries and B. Sundman, *Computational Thermodynamics, the Calphad method*, Cambridge Univerity Press, 2007.

[16] M. Hillert, The compund energy formalism, J Alloys and Compounds, **320** (2001) 161-176.

[17] S. Hertzman, B. Sundman, A thermodynamic analysis of the Fe-Cr system, Calphad 6 (1982) 67-80.

[18] M. Hillert, B. Jansson, B. Sundman and J. Ågren, A two-sublattice model for molten solutions with different tendency of ionization, *Metall. Trans. A* **16A**, (1985) 261–266.

[19] M. Hillert, Some viepoints on the use of a computer for calculating phase diagrams, Physica **103B** (1981) 31-40.

[20] B. Jansson, Computer operated methods for equilibrium calculations and evaluation of thermochemical model parameters, PhD Thesis, MSE, KTH, Stockholm, Sweden 1984.

[21] H.L. Lukas, J. Weiss, E.-Th. Henig, Strategies for the calculation of phase diagrams, Calphad **6** (1982) 229-251

[22] B. Sundman, X.-G. Lu, H. Othani, The implementation of an algorithm to calculate thermodynamic equilibria for multi-component systems with non-ideal phases in a free software, Comp. Mat. Sci., **101** (2015) 127–137

[23] Valgrind, a suite of tools for debugging and profiling, http://valgrind.org, visited 1 August 2016.

[24] C.E. Campbell, U.R. Kattner, Z.-K. Liu, The development of phase-based property data using the CALPHAD method and infrastructure needs, Integr. Mater. Manuf. Innov. 3 (2014) 12.

[25] G. Eriksson, P. Spenser, H. Sippola, A proposal for a general thermodynamic calculation interface, in: A. Jokilaakso (ed). 2nd Colloquium on Process Simulations, pp 115-126, June 1995, HUT, Espoo, Finland.

[26] S. Petersen, K. Hack, The thermodynamic library ChemApp and its applications , Int J Mat Res **10** (2007) 935–945

[27] Thermo-Calc Software AB, TQ-interface Programmers guide at http://www.thermocalc.com/support/documentation, visited 1 August 2016.

[28] Java$^{\mathrm{TM}}$Native Interface, http://docs.oracle.com/javase/7/docs/technotes/guides/jni

[29] R. Le Tellier, L. Saas, F. Payot, Phenomenological analyses of corium propagation in LWRs: the PROCOR software platform, in: Proceeedings of the 7th European Review Meeting on Severe Accident Research ERMSAR-2015, Marseille, France, 2015.

[30] C. Cardon, R. Le Tellier, M. Plapp, Modelling of liquid phase segregation in the Uranium-Oxygen binary system, Calphad 52 (2016) 47-56.

[31] Thermo-Calc Software AB, Database manager guide at http://www.thermocalc.com/support/documentation, visited 1 August 2016.

[32] I. Ansara, B. Sundman, The Scientific Group Thermodata Europe (SGTE), in: P.S. Glaser (ed) *Computer Handling and Dissemination of Data*, Elsevier Science Publishers, CODATA, (1987) 154-158.

[33] Q. Chen, B. Sundman, Compuation of partial equilibrium solidification with complete interstitial and negligible substitutional solute back diffusion. Mater. Trans., **43** (2002) 551–559

[34] J.A. Sarreal and G.J. Abbaschian, The effect of Solidification rate on microsegreation, Met. Trans. A 17A (1986) 2063–2073

[35] J. Tiaden, B. Nestler, H.J. Diepers, I. Steinbach, The multiphase-field model with an integrated concept for modelling solute diffusion. Physica D, 115 (1998) 73-86.

[36] S.G. Kim, W.T. Kim, T. Suzuki, Phase-field model for binary alloys. Physical Review E, 60 (1999) 7186.

[37] I. Steinbach, L. Zhang, M. Plapp, Phase-field model with finite interface dissipation. Acta Mater., 60 (2012) 2689-2701.

[38] L. Zhang, I. Steinbach, Phase-field model with finite interface dissipation: Extension to multi-component multi-phase alloys. Acta Mater., 60 (2012) 2702-2710.

[39] L.S. Darken, Diffusion of Carbon in Austenite with a Discontinuity in Composition, Trans. AIME, 180 (1949) 430-438.

[40] MatCalc thermodynamic and kinetic databases under Open Database Licenses, thermodynamic mc_fe 2.057 and kinetic mc_fe 2.008 at http://matcalc.tuwien.ac.at/index.php/databases/open-databases, visited 1 August 2016.

[41] R. Le Tellier, L. Saas, S. Bajard, Transient stratification modelling of a corium pool in a LWR vessel lower head, Nuclear Engineering and Design 287 (2015) 68-77.

Appendix

Available state variables and their representation in OC.

All of them can be used to extract calculated values and many of them also to set conditions. For example N is the total amount in moles, N(O) is the total amount of oxygen in moles, N(GAS,O) is the amount of moles of O in the gas phase, NM(GAS,O) or X(GAS,O) is the mole fraction of O in the gas. The total number of moles of components in gas is NP(GAS).

The heat capacity is not in this list but can be calculated as a "dot derivative, H.T" , as explained in [22].

Symbol	Index 1	Index 2	Normalizing suffix	Meaning
		Intensive properties		
T	-	-	-	Temperature
P	-	-	-	Pressure
MU	component	-/phase	-	Chemical potential
AC	component	-/phase	-	Activity
LNAC	component	-/phase	-	LN(activity)
		Extensive and normallized properties		
U	-/phase#set	-	-	Internal energy for system
UM	-/phase#set	-	M	Internal energy per mole atoms
UW	-/phase#set	-	W	Internal energy per mass
UV	-/phase#set	-	V	Internal energy per m^3
UF	phase#set	-	F	Internal energy/mole formula unit
Sz	-/phase#set	-	-/M/W/V/F	Entropy
Vz	-/phase#set	-	-/M/W/V/F	Volume
Hz	-/phase#set	-	-/M/W/V/F	Enthalpy
Az	-/phase#set	-	-/M/W/V/F	Helmholtz energy
Gz	-/phase#set	-	-/M/W/V/F	Gibbs energy
NPz	phase#set	-	-/M/W/V/F	Moles of phase
BPz	phase#set	-	-/M/W/V/F	Mass of phase
Qz	phase#set	-	-/M/W/V/F	Stability of phase
DGz	phase#set	-	-/M/W/V/F	Driving force of phase
Nz	-/phase#set/comp	-/comp	-/M/W/V/F	Moles of component
X	phase#set/comp	-/comp	-/%	Mole fraction/per cent
Bz	-/phase#set/comp	-/comp	-/M/W/V/F	Mass of component
W	phase#set/comp	-/comp	-/%	Mass fraction/per cent
Y	phase#set	const#subl	-	Constituent fraction
		Some model parameter identifiers		
TC	phase#set	-	-	Curie temperature
BMAG	phase#set	-	-	Aver. Bohr magneton number
MQ&X	phase#set	constituent	-	Mobility of X
THET	phase#set	-	-	Debye temperature