Some important facts about Open Calphad in general and this test release in particular.

Bo Sundman, June 7, 2015

http://www.opencalphad.org or

opncalpad repoitory at http://www.github.com

bo.sundman@gmail.com

Please be aware that this software is a test version. Your feedback about problems and errors is important to make it better.

OC cannot replace your favourite thermodynamic software today or tomorrow but the main advantage is that you have access to the source code and can (with some efforts) add or fix things yourself that you are missing in your favourite software.

There is a review paper describing OC:

OpenCalphad - a free thermodynamic software, Bo Sundman, Ursula R Kattner, Mauro Palumbo and Suzana G Fries (Open Access) Integrating Materials and Manufacturing Innovation (2015) **4:**1 DOI 10.1186/s40192-014-0029-1

and a special paper about the minimization algorithm:

The implementation of an algorithm to calculate thermodynamic equilibria for multi-component systems with non-ideal phases in a free software, Bo Sundman, Xiao-Gang Lu and Hiroshi Ohtani, Computer Materials Science, **101** (2015) 127-137.

1 General information

- The code is written in the new Fortran standard and requires a compiler like GNU Fortran 4.8 or similar.

- The last release of the source code can be found as a zip file on http::://www.opencalphad.org.

  The last development version can be obtained as a zip file at the opencalphad repository on the http://www.github.com. On that site there is also a wiki for information, discussion and questions.

- Users with Windows can compile and link the program using the file "linkmake.txt" which you should rename to "linkmake.cmd" in order to compile and link an executable program from the source code.

  If you are using a UNIX system there is a Makefile to compile and link the program.

  In both cases the program will be called "ocj" where j is the version number.

- For the graphics you must download and install the GNUPLOT software. That is freely available from the web.

  Make sure your PATH includes the directory with the GNUPLOT program. If you do not know how to set your PATH ask an expert.

- There are several directories provided in the zip file, the source code is in the directories "minimizer, models, numlib, stepmapplot, userif" and "utilities".

- The documentation of the source code is in the directory "documentation" in two PDF files "gtp3" and "minpack5" for the model and minimizer package. There is also a very rudimentary user guide "ochelp". More informal guides are "news-OC2" and "OC3-commands".

- The user guide is also available in the file "ochelp.hlp" in LaTeX format used for the on line help. The user guide has not been updated since version 1.

- The "macro" directory has examples for a variety of calculations. There are subdirectories created for testing different versions but as the command interface changes those provided with this version will normally not work with previous versions.

- The TQ3lib directory has examples for using the software interface for Fortran and C/C++.

- Contributions of new and improved source code are welcome. You can do this using the github repository. Contact Bo Sundman if you want to know more.

- The command line interface has a "VAX/VMS" flavour which reflects the age of the developer. It means the commands are "verbs" like *set, list, calculate, enter* etc. After the verb several objects are usually possible like *set condition/status* etc. There is some redundancy so the same effect can sometimes be achieved by different combinations of verbs and objects.

  Each command and subcommand can be abbreviated.

2 Some short hints to the user interface

See also news-OC2 and OC3-commands. The information in these (as well as this) file is not always up-to-date. You are welcome to inform me about errors an inconsitencies. If you do not like the current user interface you are welcome to implement your own.

2.1 Entering and manipulating thermodynamic data

In this and the following sections you can find a brief and unstructured summary of the commands that may be useful when trying to get aquantied to OC.

- Thermodynamic data can be read from an unencrypted TDB file *read tdb "filename"* or entered interactively, see the *enter* commands. You can specify which elements you want from the TDB file but not select the phases. Phases you do not want you can suspend from the calculations with the command *set status phase ... = suspend.*

  If you read thermodynamic from a file and already have data in the program you have to confirm that you want to overwrite them. You can remove all the thermodynamic data in the program by the command *new YES.*

- There are some exception from how data are entered into OC comparared to Thermo-Calc, most important perhaps the "partitioning" of a phase into an ordered and disordered part. OC has implemeneted this in a different way and at present it is not always possible for OC to read such partitioned data from a TDB file without editing.

- You can save the thermodynamic data and results from a calculation on an unformatted file *save UNFORM "filename"* and this can be read back into the program *read UNFORM "filename"*. This is fragile. The DIRECT (random access) format is not yet implemented.

- You can save the thermodynamic data using the TDB format by a command *list data TDB filename.* The TDB format used by OC can with some exceptions be read also by other thermodynamic software.

- You can list the thermodynamic data on the screen with the *list data* command. You can list on a file by using the switch *list /output=filename ....* If you want to write a TDB file use the command *list data TDB.*

- You can enter thermodynamic data from a macro file as well as commands for calculations. In order to document errors or problems please send a complete macro file with data and commands reproducing the error.

  You can generate a macro file interactivly by setting a log file *set log "filename"* and then (after some editing) use the log file as a macro with the extention OCM. The log file has the extention LOG.

2.2 Setting conditions

- To calculate you must set conditions on the external variables like temperature, T, pressure, P etc. The number of conditions must be equal to the number of components plus 2 (Gibbs phase rule).

- Setting conditions is very similar to the Thermo-Calc software. Each condition is set separately by the command is *set cond "state variable" = "value"*. The safest set of conditions for calculating an equilibrium, i.e. which has most chance to converge, is to set values on T, P and N(element), i.e. the total amount of each element. The table at the end gives a list of available state variable symbols and their meaning.

- It is also possible to set conditions on chemical potentials, activities and that a phase is stable (fix). See the macro file examples how this is done. Note that some commands are fragile and they may also change between releases of the test versions of the OC software, depending on new ideas and suggestions by users.

- The intention is that you should be able to combine any set of conditions to calculate the equilibrium, i.e. you should be able to combine conditions on mole fractions, mass fractions, fix phases, chemical potentials, enthalpy, volume etc. In a future release the plan is to allow expressions of conditions, not just a single state variable.

- For FactSage users there is a command *set input_amount* which allows the user to set the overall composition by specifying the amount of different species in the system.

2.3   Calculation

- The command *calculate equilibrium*, which can be abbreviated *c e*, tries to calculate the equilibrium. As the minimizer needs a guess of stable phases and their start constitution, the OC software tries first to invoke a global grid minimizer to find such a guess. The grid minimizer requires that the conditions are T, P and overall composition (N, B, x or w).

- If you want to provide yourself a guess of the set of stable phases and consitutions yourself you can use the *amend const phase "name" "amount" "constitution"* $\cdots$ followed by the command *calculate no_global*, abbreviated c n.

- The grid minimizer that calculates start points for the general minimizer is powerful but a bit primitive. If you have ideas how to improve it you are welcome to provide advice or code.

- For this test version you should always check the results with you favourite thermodynamic software. Please give documented feedback on any differences.

- If the calculation does not converge directly try to use the command *"c n"* two or more times to continue to iterate from the set of phases you have. The *"c e"* command runs the grid minimiser and, therefore, will normally give the same result each time.

- You can also try to change the number of iterations and convergece criteria with the command

  set num "max iterations" "convergce criteria". Default values of these are 500 and 10^{-6}.

- If everyting fails try to simplify the conditions for a first calculation. The simplest are values of T, P and N(component). When that has converged change the conditions one by one to those you are interested in and for each change calculate without the grid minimizer, i.e. using *"c n"*. Calculations at temperatures and compositions where

the system is single phase have a higher chance of success. The algorithm to change the set of stable phases is fragile and is still being fine tuned.

- Phases with miscibility gaps can be a problem. The grid minimizer should detect such and automatically add new composition sets for phases that can be stable with two different compositions. Composition sets created automatically have the suffix AUTO.

- The user can also add composition sets manually with the command *amend phase "name" comp_set* . You can the add a prefix and suffix (max 4 letters) to the phase name (for example the cubic carbide treated as a composition set to the FCC phase can have the prefix MC so the full name is MC_FCC.

  Composition sets have a number after the phase name and the letter #, like LIQUID#2. The user can use both pre- and suffixes or set number to specify the composition set. A phase can have up to 9 composition sets.

- When entering a composition set manually you are asked for a default constitution. After a calculation the software will try to match a calculated constitution of a phase with several composition sets to the set with the closest default constitution.

- To set the default constitution of the first compostion set use *set phase "name" default_constitution* ⋯ .

- There are some more option to calculate things:

  - *calculate all* will be used when OC can do assessments,
  - *calculate global_gridmin* will only use the gridminimizer and not the final iterative routine,
  - *calculate phase* calculates the Gibbs energy and possibly derivatives for a specified phase and composition at the current T and P.
  - *calculate symbol* will calculate the value of one or more symbols.
  - *calculate tp_funs* will calculate the value and the first and second derivatives with respect to T and P for all tp functions (used for the model parameters).
  - *calculate transition "phase" "condition"* calculates the value of the "condition" when the phase is just stable, for example the melting T if the "phase" is LIQUID.

2.4   Setting the status of phases: entered, fixed, dormant, suspend

To set the status of a phase to be fixed is treated as a condition.

- By default all phases have the status ENTERED. You can suspend phases you do not want by *set status phase "name"* ⋯ = *"suspend"*.

- To specify that a phase must be stable you can set its status to fix, the command is *set status phase "name" = "fix" "amount"*.

- The dormant status means the phase cannot be stable but its driving force is calculated. If this is positive the phase wants to be stable and the calculated equilibrium is metastable.

2.5   Listing many things

Most listing are written on the screen but when output obviously should be written on a file the program will ask for a file name. If you want the output from other listings to be written on a file you can use the switch */output=filename* or */append=filename* directly after the command. For example *list /out=result r,,,*. The extension of the output file will be DAT. The output file will be reset to the screen after each command.

   There is also a SAVE command mainly intended for saving on a file. For example *save tdb filename* will write all the thermodynamic data in the OC TDB format on the file with the extention TDB. This can be read back later as a TDB file.

- Listing results after a calculation is done by list result "option". This is the default list and can be abbreviated l,,. This list will always include the current conditions and some global results and one line for each component. The options are at present:

    - 1 for list of stable phases with amount and composition in mole fractions.
    - 2 for list of stable phases with amount, composition in mole fractions and constitution.
    - 3 same as 1?
    - 4 for list of stable phases with amount and composition in mass fractions.
    - 5 same as 4?
    - 6 for list of stable phases with amount and composition in mass fractions and the constitution
    - 7 for list of all phases with amount and composition in mass fractions.
    - 8 for list of all phases with amount and composition in mass fractions and constitution.
    - 9 for list of all phases with amount and composition in mole fractions and constitution.

- The *list short A* gives one line for each component, species and phase. In this list you may note the use of the "phase tuple" to indicate composition sets. A phase tuple has two values, one is the phase number, the other the composition set number. The first composition set of a phase has the same tuple number as the phase number. As

composition sets can be added and deleted the tuple number of a higher composition set may change.

- The *list short P* gives one line for each phase, separated in stable, entered and dormant, listed in order of increasing instability (driving force).

- The *list bibliography* list all or a single bibliographic reference. Bibliograpic references are entered when entering parameters. The *amend biblio* allows interactiving amending of the bibliographic item.

- The *list data* lists all thermodynamic data with bibliographic references. Note this output is not the TDB format.

- The *list equilibria* list all entered equilibria (not any results). By default there is only one equilibrium but during step and map each node point is entered as an equilibria. The user can also enter equilibria manually.

- The *list model_param_id* list all symbols for which one can enter parameters. By default the G is Gibbs enrgy, TC the combined Curie and Neel temperature etc.

- The *list phase* has several options to list various data for a phase.

- The *list state_variables* lists individual values of state variables. Terminate with an empty line.

- The *list symbols* list the symbols and their expressions. There two predefined symbols, R and RT. The user can enter any number of additional symbols as functions of state variables or other symbols. To calculate (and list) the value of a symbol use *c sym*.

- The *list tp_symb* lists the functions entered for the model parameters. To calculate their current values use *c tp*.

2.6   Calculating and plotting diagrams

This part is unfinished and very rudimentary.

- You can calulate property diagram (one independent axis variable) or phase diagrams (two or more idependent axis variables). The command to set an independent axis is

  set axis 1 "state variable" "min value" "max value" "increment". The state variable used must be a condition. Axis 1 and 2 are implemented.

- With one axis the *step* command will step from the current equilibrium to the max and min value.

- With two axis the command *map* will follow "zero phase fraction" (ZPF) lines within the limits of the axis by replacing one of the axis consition with a phase fix with zero amount. In this way phase diagrams for any number of components can be calculated. While following such a line other phases may become stable or disappear and this generates a node point where several lines meet.

- Plotting of diagrams relies on GNUPLOT, a free software you can install from the web. The command *plot* asks for two axis variables and generates pairs of values from the resluts calculated by step or map commands.

  There are several options to the plot command that are demonstrated by the macro files.

- Macro files are easily created by using a log file while running OC interactivly. On the log file all input (as well as default values) are written. Then you can edit this and finish it with the *set interactive* command to give command back to the keyboard.

  When you report problems or errors always sent a macro file that reporoduces the error. Please try to find the simplest case that contains the error.

3   A summary of state variables.

The state variables are very important for setting conditions and listings. Those recognized by OC are given in table 1.

The state variables in the user interface have their common symbols, T for temperature, P for pressure, N for the total amount of moles, "N(element)" for the amount of moles of a component, "X(element)" for the mole fraction "MU(element)" for the chemical potential, "AC(element)" for the activity. The symbol B is used for the total mass (copied from the Thermo-Calc software), "B(element)" for the mass of an element and "W(element)" for the mass fraction. There are many more state variables like H, G etc, see the table, but not all of them can be used as conditions.

4   Phase status

You can specify that a phases should be stable by the command *set status phase* $\cdots$. For example to calculate the melting point of an alloy after specifying the composition and making a calculation at fixed T and P, you can give the commands *set cond T=none; set status liquid=fix 0; c n*. (The commands must be given on separate lines). You can also use the command *calculate transition* $\cdots$.

Table 1: A very preliminary table with the state variables and their internal representation. Some model parameter properties are also included.

Symbol	Id	Index		Normalizing	Meaning
		1	2	suffix	
Intensive properties					
T	1	-	-	-	Temperature
P	2	-	-	-	Pressure
MU	3	component	-/phase	-	Chemical potential
AC	4	component	-/phase	-	Activity
LNAC	5	component	-/phase	-	LN(activity)
Extensive and normallized properties					
U	10	-/phase#set	-	-	Internal energy for system
UM	11	-/phase#set	-	M	Internal energy per mole
UW	12	-/phase#set	-	W	Internal energy per mass
UV	13	-/phase#set	-	V	Internal energy per m^3
UF	14	phase#set	-	F	Internal energy per formula unit
Sz	2z	-/phase#set	-	-	entropy
Vz	3z	-/phase#set	-	-	volume
Hz	4z	-/phase#set	-	-	enthalpy
Az	5z	-/phase#set	-	-	Helmholtz energy
Gz	6z	-/phase#set	-	-	Gibbs energy
NPz	7z	phase#set	-	-	Moles of phase
BPz	8z	phase#set	-	-	Mass of phase
Qz	9z	phase#set	-	-	Stability of phase
DGz	10z	phase#set	-	-	Driving force of phase
Nz	11z	-/phase#set/comp	-/comp	-	Moles of component
X	111	phase#set/comp	-/comp	0	Mole fraction
X%	111	phase#set/comp	-/comp	100	Mole per cent
Bz	12z	-/phase#set/comp	-/comp	-	Mass of component
W	122	phase#set/comp	-/comp	0	Mass per cent
W%	122	phase#set/comp	-/comp	100	Mass per cent
Y	130	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 X	-	Mobility
THET	-	phase#set	-	-	Debye temperature

5 Manipulating the source code

The OC software is provided with a GNU license which means that you have the source code and can use it and modify it as you wish as long as you do not try to make money of

it. If you want to include the OC software in a commercial program you must contact the copyright owners.

There is a fairly extensive documentation of the source code in the directory "documentationupdate" and if you look at the code itself there are some comments there too. I have spent a lot of effort to make the datastructures general and flexible to handle multicomponent and multiphase systems. But there was quite a lot of redundancy introduced during the development that eventually will be removed. The set of subroutines is less structured and one problem has been that this code was my first attempt to use the new Fortran standard so there are probably many things that can be made simpler. You are welcome to point out where this can be done. The hope is to have a new release before the end of 2015 with an improved version of this part of the code and some more thermodynamic models and facilities for conditions and generating diagrams will be available.

As I have understood the data structures (TYPE) in the new Fortran standard is more or less identical to those used in C++ so it should not be too difficult to combine code written in these languages.

6   Application software library, TQ

There is a software intrface following the TQ standard to make it possible to use OC from application programs. The subroutines in this interface can be found in the directory "TQ3lib" together with a few test program. To compile and run these the OClib.a file and some of the "mod" files must be copied to this directory, there is a readme file that should explain how to use these.

There is a Fortran version of the TQ library and also an isoC version that can be called from C++/C routines. The isoC binding makes it possible to access data directly from the datastructures defined inside the OC software. There are some test programs for this also.

Use only subroutines in the TQ library to access the OC software, do not call directly subroutines inside the OC code as they may not be available or have different functionallity in a future release. If you miss some routines please sent a message or discuss this on the github wiki.

Have fun and help make OC useful!