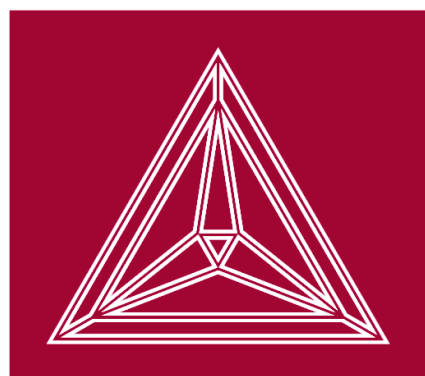
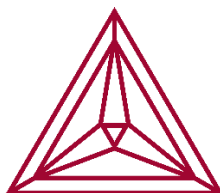




**Thermo-Calc
Software**

Data Optimisation User Guide for Thermo-Calc



© 1995-2014 Foundation of Computational Thermodynamics
Stockholm, Sweden

Data Optimization User Guide

Contents

1	Introduction.....	5
1.1	Data optimisation.....	5
1.1.1	The least-squares method.....	6
1.1.2	The CALPHAD approach.....	6
1.1.3	References.....	6
1.2	Documentation for data optimisation in Thermo-Calc.....	7
1.2.1	This User Guide.....	7
1.3	Data optimisation in the Thermo-Calc software package.....	8
1.3.1	The PARROT module.....	8
1.3.2	The EDIT_EXPERIMENTS sub-module.....	8
1.3.3	Other modules.....	9
1.3.4	Files.....	9
1.3.5	Workspaces.....	10
1.3.6	Method for optimisation and simulation.....	11
1.4	Overview of data optimisation workflow.....	11
2	Creating the POP-file.....	13
2.1	Experimental Information.....	13
2.1.1	Information that is conflicting or missing.....	13
2.1.2	Information about invariant equilibria.....	14
2.1.3	Information about equilibria in higher-order systems.....	14
2.1.4	Information about heat capacities.....	15
2.2	POP-file syntax.....	16
2.2.1	Legal commands.....	16
2.3	Entering experimental information.....	17
2.3.1	Creating a new equilibrium.....	17
2.3.2	Setting equilibrium conditions and their uncertainty.....	17
2.3.3	Setting reference states.....	18
2.3.4	Entering the experimental value and its uncertainty.....	18
2.3.5	Entering many equilibria as a table.....	19
2.3.6	Setting site fraction start values.....	20
2.3.7	Comments about an equilibrium.....	21
2.3.8	Simultaneous use of binary and ternary experiments.....	21
2.4	Using stability conditions.....	21
2.5	Grouping equilibria with labels.....	22
2.6	Dividing the POP-file into several blocks.....	22
3	Creating the setup file.....	24

3.1	Defining the system.....	24
3.1.1	Creating additional composition sets of a phase	24
3.2	Entering the optimising variables.....	25
3.2.1	Entering optimising variables in parameter specifications	26
3.2.2	Entering interaction parameters.....	27
3.3	Initialising optimising variables in PARROT.....	27
3.4	Creating the workspace file	28
3.5	Compiling the POP-file	28
3.6	Returning control to the console	29
4	Optimising your system in PARROT	30
4.1	Changes that require POP-file recompilation.....	30
4.2	Workflow	30
4.3	Ensuring that all equilibria can be computed	31
4.4	Setting weights.....	32
4.5	Optimising and evaluating listed results.....	33
4.5.1	The critical set of experiments	33
4.6	Continuing the optimisation and resetting variables.....	34
4.7	Updating the set of optimising variables.....	35
4.7.1	Reducing the number of optimising variables.....	37
4.8	Plotting intermediate results	37
4.8.1	Superimposing experimental data from EXP-files.....	37
5	Finishing the optimisation.....	38
5.1	How to round off optimising variable values	38
5.2	Updating the database file.....	38
5.3	Creating a database file.....	39
5.4	Update the setup file and the POP-file.....	39
6	Optimising in alternate mode.....	40
6.1	Turning alternate mode on and off.....	40
6.2	Optimising in alternate mode	41
6.3	Preparing the POP-file for alternate mode	41
6.3.1	Examples of SET_ALTERNATE_CONDITION usage.....	41
7	Troubleshooting.....	43
7.1	Excluding and including the right equilibria	43
7.2	Conflicting data	43
7.3	Useful rules of thumb	43
8	Appendix 1: Experiments and experimental data.....	45
9	Appendix 2: Thermodynamic models.....	51
9.1	Gas	51
9.2	Compound Energy Formalism.....	51
9.3	Substitutional Liquid	52
9.4	Ionic Two-sublattice Liquid Model	52

9.5	Models for ordered phases.....	52
9.6	4SL model for FCC, HCP and BCC.....	53
9.7	2SL model for FCC, HCP and BCC.....	54
9.8	Magnetic Ordering.....	55
9.9	Molar volumes and high pressure.....	55
9.10	Excess Models.....	56

1 Introduction

This introductory chapter describes the basic idea behind data optimisation, contains brief descriptions of the modules and file types that you will mainly be working with while doing your optimisation, as well as rough outline of overarching workflow that you will typically follow when working with optimisation.

1.1 Data optimisation

In a thermodynamic database, each phase in a system is characterised using a mathematical model of its Gibbs energy. The Gibbs energy of a phase depends on various state variables. It can be defined individually even in a heterogeneous system with many stable phases since the properties of one phase are completely independent of the properties of the other phases in the system. In most alloy systems, the thermodynamic properties of a phase can be modelled by expressing how the Gibbs energy depends on temperature and composition. State variables such as pressure, volume or entropy can also serve as parameters in the Gibbs energy expression. The Gibbs energy of the whole system is the sum of the products of the Gibbs energy of each phase multiplied by the amount of that phase.

How the Gibbs energy of the phase varies with various state variables is determined by a mathematical model and how various adjustable parameters of the model is set. By optimising these parameters, you can calculate the thermodynamic properties of a system under various conditions (by calculating phase diagrams for example). The thermodynamic properties themselves are functions of temperature, pressure or composition. They include, among other properties, standard enthalpy of formation, entropy, heat capacity, molar volume, thermal expansivity, compressibility and Curie temperature for magnetic transformations.

Data optimisation is about adjusting the model parameters so that calculated equilibria fit well with experimental data. An important part of a data optimisation is therefore collecting and assessing available experimental and theoretical information about phase equilibria and thermochemical properties of a system.

When you optimise the parameters of the model of a phase, you put what is called optimising variables into the Gibbs energy expressions of those parameters. During the optimisation, the values of these variables are varied in order to find a fit between the calculated equilibria and the experimental data that you base the optimisation on. Once you have settled on a Gibbs energy expression for a certain parameter of a phase of the element, then you must stick to this expression in all future optimisations involving the element. If you do not do this, then you must re-optimize all previous systems that involve this element when the expression is changed.

The Thermo-Calc program allows you to optimise parameters that characterise not only binary systems, but also ternary system and systems of even higher orders. You can even optimise parameters that characterise systems of different orders at the same time.

1.1.1 The least-squares method

Data optimisation in Thermo-Calc is based on the least-squares method for fitting values calculated on the basis of a model with observed quantities. The Thermo-Calc program is accordingly trying to find the optimising variable values that lead the minimised sum of the squares of the differences between the calculated values and the observed quantities (that is, of the errors or residuals).

The least-squares method works best under the following conditions:

- The observed quantities have a Gaussian probability distribution.
- The observed quantities are only subject to random errors.
- The different observations (experiments) are uncorrelated
- The standard deviation of each observation can be estimated.
- The number of observations is large.
- The models used give precise predictions.

Of course, these conditions are usually not all met in a normal thermodynamic assessment. But even in non-ideal conditions, there is no known method that works better than the least-squares method.

1.1.2 The CALPHAD approach

The data optimisation functionality in Thermo-Calc works according to the CALPHAD approach to computational thermodynamics. This approach builds on the development of models that accurately represent thermodynamic properties for various phases. These models enable users to predict the thermodynamic properties of multicomponent systems, based data concerning binary and ternary subsystems. The predictions can take many factors into accounts, such as, for example, crystallography, type of bonding, order-disorder transitions and magnetic properties.

“CALPHAD” was originally an abbreviation of “CALculation of PHase Diagrams”, but as the approach has expanded in scope, it now stands for “computer coupling of phase diagrams and thermochemistry”.

1.1.3 References

For in-depth information about CALPHAD as well as data optimisation, see *Computational Thermodynamics – The Calphad Method* by Hans Leo Lukas, Suzana G. Fries and Bo Sundman (Cambridge University Press, 2007). CALPHAD was given its first general description in *Computer Calculations of Phase Diagrams* by Larry Kaufman and Harold Bernstein (Academic Press, 1970). The book contains some case studies of optimisations using Thermo-Calc.

Information about the concept of ‘lattice stability’ (which is important for the development of multicomponent thermodynamic databases) can also be found in the paper “Hume-Rothery and Calphad Thermodynamics” by Larry Kaufman, published in *Calphad and Alloy Thermodynamics*, edited by Patrice E.A. Turchi, Antonios Gonis, Robert Shull (The Minerals, Metals & Materials Society, 2002), pp. 3-19.

1.2 Documentation for data optimisation in Thermo-Calc

Apart from this document, you may find the following documentation helpful when learning to do data optimisation in Thermo-Calc:

Thermo-Calc Console Mode Command Reference

contains information about all the commands that are available in the Thermo-Calc Console, including PARROT and ED_EXP. Note that the commands that are listed under PARROT and ED_EXP are only commands that are specific to PARROT and/or ED_EXP. These modules also contain some commands that are also available in POLY. These commands are listed under POLY.

Online help in the Thermo-Calc Console Mode

contains information about modules, commands, and conditions. Use the `HELP` and `INFORMATION` commands to access this information.

Example 36 in *Thermo-Calc Console Examples Collection*

is a detailed example of assessment and optimisation in Thermo-Calc.

DATAPLOT User's Guide and Examples

describes the syntax and semantics of the DATAPLOT graphical language used in EXP-files.

Additional documentation and educational materials are available on Thermo-Calc Software's website.

1.2.1 This User Guide

Target group

This user guide is intended for users who are already familiar with using the Thermo-Calc console mode with its command-line interface. It is expected that the user knows thermodynamics well and is familiar with the theoretical foundations of computational thermodynamics.

Purpose

This user guide is intended to get you started with data optimisation using the Thermo-Calc program. It describes how, in generic terms, you can go from collecting experimental data that your optimisation will be based on, to updating your Thermo-Calc database file. Note that the data optimisation is a craft that requires skill, experience and judgement. This User Guide does not provide you with the information required to assess whether your results are "good enough" for your purposes, but merely teaches you the basics about how the tool you will use functions.

Content

Chapters 2 to 5 each describe steps within the typical data optimisation workflow: creating a so-called POP-file that contains the experimental data you want to use in the optimisation (chapter 2), creating a setup macro file (chapter 3), interactively doing the optimisation in the console (chapter 4), and finalising the optimisation by, for example, updating your database (chapter 5). Chapter 6 describes how you optimise using the so-called alternate mode, which is useful when all or most optimising variables have their start values set to zero or when you have no idea which start values to use.

The following conventions will be used in this user guide:

- Console command names are written in small upper case letters. For example, the PARROT module command for starting the actual optimisation is OPTIMIZE_VARIABLES.
- Examples of how you can use a command (directly in the console or in a macro file or POP-file) are written like this:

```
SET_OPTIMIZING_VARIABLE 1 0
```

1.3 Data optimisation in the Thermo-Calc software package

The actual performance of the optimisation of a system is primarily done in the PARROT module, and in its sub-module ED_EXP. You will also use the POLY and POST modules to calculate and plot property or phase diagrams based on your optimised variables. This allows you to visually assess how good a fit you have achieved between your calculated results and your experimental data.

The end result of a successful optimisation is typically an updated Thermo-Calc database file (with filename extension “*.TDB”) or a new database file.

1.3.1 The PARROT module

The PARROT module consists of a comprehensive subroutine package for data evaluation of thermodynamic model parameters from experimental data. The module has 99 variables that you can use for optimisation and the module can handle a thousand experimental measurements in an optimisation. However, there are limits on the simultaneous numbers of variables and experiments. These limits are listed at each optimization.

The PARROT module is part of the Thermo-Calc program. The module is not included in the software development kits TQ-Interace, TC-API or TC-Toolbox for MATLAB.

The experimental data is entered and edited in a sub-module of PARROT: the EDIT_EXPERIMENTS module.

1.3.2 The EDIT_EXPERIMENTS sub-module

The EDIT_EXPERIMENTS module, or ED_EXP for short, is used for entering, evaluating and editing experimental data. To enter the module, use the PARROT command EDIT_EXPERIMENTS (you cannot enter the module with the GO command).

Since ED_EXP uses the POLY module for doing equilibrium calculations of experimental data points, many POLY commands are also available in the ED_EXP module. However, the commands may work somewhat differently in the two modules. There are also unique ED_EXP commands which are not found in POLY.

When you execute ED_EXP commands, the PARROT workspace is modified. However, note that you must always first use READ_WORKSPACES in ED_EXP before doing anything in the module. The command loads the experimental data that you are going to work with during the optimisation. Furthermore, before leaving ED_EXP you must use SAVE_WORKSPACES or the results of your work in the ED_EXP module will be lost.

1.3.3 Other modules

All kinds of thermodynamic data, calculated equilibrium states or dynamic parameters are transferred back and forth between the PARROT and the GIBBS module as well between these and the POLY module. Whenever an optimisation run is performed, PARROT calls the GIBBS module for stored system definition data and model parameters.

In Thermo-Calc, the GIBBS module (the Gibbs Energy System) handles the models of the various phases that can form in a multicomponent system. It stores thermodynamic data and performs various Gibbs energy equilibrium calculations. It contains subroutines to analytically calculate the first and second partial derivatives of integral Gibbs energy with respect to any set of variables. Many thermodynamic models for various types of substances are implemented in the module. Some commands that are available in the PARROT module are also available in the GIBBS module.

In the GIBBS module, parameters of the Gibbs energy models are referred to as "TP-functions".

PARROT calls the POLY module for equilibrium calculations. The POLY module is used for calculating and storing complex heterogeneous equilibria.

1.3.4 Files

This section describes the most important files that you will be working with when optimising.

POP-file	The POP-file (Parrot OPTimisation file) is the file that contains all the experimental data that you use for optimising Gibbs energy values for your system. The POP-file is a plain text file that contains ED_EXP commands. By default, the file has the filename extension "*.POP" in Thermo-Calc.
Setup file	The setup file is a Thermo-Calc macro file (*.TCM). This file will typically contain commands that define your system, open the PARROT workspace (and associated GIBBS and POLY workspaces) and set optimising variables.

Other macro files	It will often be useful to have other macro files than the setup file at hand while you are doing your optimisation. For example, it is convenient to have a macro file that automatically plots phase diagrams or other diagrams based on the latest values of your optimising variables. An optimisation process typically involves many optimisation cycles, so this operation needs to be done many times during a typical optimisation.
Workspace files	The PARROT module has its own workspace with dynamic memories, similar to the workspaces of POLY and GIBBS. It is stored and updated in a PARROT workspace file with the filename extension “*.PAR”. The workspace file is a binary file whose format is hardware dependent. This means that the format is unique for each type of CPU. Hence, a workspace file saved on a computer with one type of CPU cannot be used on a computer with another type of CPU.
EXP-file	The EXP-file is a plain text file with information in the form of DATAPLOT Graphical Language commands. These commands can specify some or all of the data points in your POP-file. This allows you to plot these data points on top of your plotted optimisation results (with the POST commands APPEND_EXPERIMENTAL_DATA or QUICK_EXPERIMENTAL_PLOT). You can then visually assess the fit between these results and the experimental data. The PARROT workspace is not influenced in any way by the creation or use of EXP-files.

1.3.5 Workspaces

When writing the setup file, you will put in a `CREATE_NEW_STORE_FILE` command which creates a PARROT workspace file. Alternatively, select an existing PARROT workspace file with `SET_STORE_FILE`. The workspace file will automatically be updated and saved with your latest optimisation results (after each use of the `OPTIMIZE_VARIABLES` command). You can also explicitly instruct Thermo-Calc to save the workspace using `SAVE_PARROT_WORKSPACES`. If you want to get rid of your latest changes and return to the state of your workspace when it was last saved, use `READ_PARROT_WORKSPACES`.

When the PARROT workspace is saved, parts of the POLY and GIBBS workspaces are also saved to the PARROT workspace file, along with the PARROT workspace itself. Using the POLY command `SAVE_WORKSPACES` will simply create a new POLY3 file, nothing will be saved to the PARROT workspace file.

If you do any POLY calculations based on data in the current workspace file, then a “*.POLY3” file will be created, which will contain a copy of the current set of parameters. If you then continue on your optimisation and then read the old “*.POLY3” file, then your new set of parameters will be overwritten with the old set. It is therefore recommended that you never read a “*.POLY3” or “*.GES5” file while you are doing an optimisation.

1.3.6 Method for optimisation and simulation

The PARROT module typically uses ordinary POLY minimisation for equilibrium calculations. The global minimisation technique that is used in POLY cannot be used because it automatically creates new composition sets which corrupt the PARROT workspace. Furthermore, the equilibrium definitions that you use as input for your optimisation must each be attributed to a specific phase. Some of these definitions may specify some local or partial equilibrium state rather than a global minimum. However, it is possible to use the global minimisation technique for some equilibrium calculations by using the ED_EXP command `ADVANCED_OPTION TOGGLE_ALTERNATE`.

Optimisation in PARROT is performed on the basis of a ‘maximum likelihood’ principle. According to this principle, the best fit between various calculated results and all the input of experimental data is found where the sum of the square of the weighted residuals is at its minimum.

Typically, the results of an optimisation is considered to be better the fewer optimising variables that are needed to get the same level of fit between computed results and experimental data. If you can get almost the same fit using eight instead of twelve parameters, then this should be considered to be a significant improvement. When more parameters are used, the values of the individual parameters tend to become unrealistic. This is particularly true when it comes to temperature-dependent parameters. However, relatively small differences in the number of parameters are often insignificant. For example, it will typically not matter whether you used 24 or 25 parameters to reach a certain degree of fit.

It is often difficult to compare the quality of different optimisations based on the number of optimising variables that have been optimised. Since you may rely more or less heavily on different pieces of available experimental data (and do so with good reason), it is possible that two different users could end up with very different numbers of parameters for the same system. For example, suppose you optimise the Fe-Al system to incorporate it into an Al database, while someone else optimises the same system to incorporate it into an Fe database. In this situation, the two of you would probably make different judgements about which experimental information your calculated results must have a good fit with.

1.4 Overview of data optimisation workflow

When you carry out an optimisation using Thermo-Calc, you will typically follow a workflow that is roughly as outlined below.

- 1 Collect experimental data about your system from various sources such as journal articles and reports.
- 2 Create a POP-file in which you enter the experimental data you have collected.

- 3 Create a setup macro file in which you define your system and the variables you want to optimise. A useful general procedure is to first find a minimum set of variables that allows you to calculate most of the experiments.
- 4 Run the setup macro file.
- 5 Compile your POP-file. This file will often contain some errors, so you will typically have to correct errors and recompile the file several times before the compilation can be completed without errors.
- 6 In the ED_EXP module, check that the experiments in your POP-file reach equilibrium with reasonable results. If any experiments do not reach equilibrium with reasonable results, then try changing the starting values on equilibrium conditions (such as composition) to see whether the equilibrium can be computed after all. If this does not work, then you can temporarily exclude the experiment from the optimisation. At a later stage in the optimisation, when the optimising variables have different values, you can check whether the experiments can be computed with reasonable results.
- 7 Back in the PARROT module, run the first optimisation cycle and evaluate the feedback in the console. You can also plot diagrams in the POLY module that allow you to visually inspect the fit between calculated optimisation results and the experimental data. If you are not satisfied with the fit, then enter the ED_EXP module again and adjust the weights of the experiments in a way that is likely to improve the fit and run another optimisation cycle. Reiterate until you are satisfied with the fit.
- 8 Once you get the optimisation stable and smooth with the minimum set of variables chosen in step 3 above, try using different sets of variables to see whether improvements are possible.
- 9 When you are satisfied with the fit between calculated results and experimental data with your final set of optimising variables, update your setup file and POP-file. In the setup file, enter the calculated optimising variable values as the variables' start values. In the POP-file, enter the final weights of the experiments. With these files updated, you can easily regenerate the parameter values of your optimised system. Having the files updated also makes it easier to optimise the system again in light of new data or new theoretical models.
- 10 Finally, update the database with your calculated results or create a new database with information about the system that you have optimised.

2 Creating the POP-file

Before starting an optimisation in Thermo-Calc, you must collect relevant experimental data about your system that is available in the research literature. This data is then entered in the POP-file. The experimental data points and measurements are entered in this plain text file in the form of ED_EXP commands.

The syntax of the POP-file is in principle independent of the models used for describing the phases in the system.

2.1 Experimental Information

An optimisation of thermodynamic model parameters should be based on a range of reliable experimental results that is as wide as possible, as well as on empirical or semi-empirical correlations and theoretical considerations. During an optimisation in Thermo-Calc, many kinds of thermochemical and thermophysical data can be mixed and used together.

The information needed for an optimisation represent measurements of thermodynamic parameters in the system at equilibrium. These can be measurements of any of the thermodynamic quantities that can be set as conditions in Thermo-Calc. The information may represent measurements of activities or enthalpies in single-phase regions for example, or solubilities or transformation temperatures in a multi-phase region.

For each equilibrium that is used in the optimisation, at least one quantity must have been measured and you also need to know the conditions that must be set to compute the system's equilibrium state. For example, consider a binary system in a single-phase region. Suppose that you have measured the temperature, pressure, composition and the chemical potential. Three of these quantities are necessary to specify the equilibrium state and the forth can be used as experimental information to model the phase.

If you have a two-phase region and know the temperature and pressure conditions, as well as which the stable phases are, then you could use the compositions of one or both phases as experimental data. The former information about the conditions is sufficient to determine the equilibrium. The experimental information can then be used to optimise the parameters that are used to characterise and model the system.

For examples how to enter various kinds of experimental data in the POP-file, see Appendix 1: Experiments and experimental data.

2.1.1 Information that is conflicting or missing

You are likely to sometimes have inconsistent or conflicting experimental information. In principle, all available experimental data should still be entered into the POP-file, unless you have good reasons to exclude some information. For example, if you have reason to believe that the samples used in some experiments were not pure, then these experiments could be excluded from the POP-file.

For some systems, you will have very little experimental data to go on. When this is the case, do not try to fit the available data uncritically with high accuracy without considering possible errors. You can look for data in systems with similar elements to get an idea about what the reasonable estimates of data that is lacking could be. Or you can use calculations from first principles or semi-empirical methods to get reasonable estimates that can be used as experimental data.

2.1.2 Information about invariant equilibria

The most valuable experimental information for an optimisation is information about stable invariant equilibria. An invariant equilibrium is such that neither pressure, temperature nor composition can be changed without there being a change in the set of phases present in the system. It is recommended that all information about invariant equilibria for a system is included in the POP-file. These equilibria need not all be based on explicit experimental measurements of thermodynamic properties.

Reasonable estimates from available experimental data are often useful to include in the POP-file. Such estimates, which should be based on the available experimental data, will help Thermo-Calc find a good set of start values on the optimisation variables. However, at the end of the assessment, these estimated equilibria should be excluded and you should complete the optimisation based only on the experimental information that is based on measurements.

With some experience from phase diagram evaluation, it is possible to make reasonable estimates of metastable invariant equilibria. Such estimates will help reduce the number of phases that are assessed simultaneously during the optimisation. For example, you may assume that a certain intermediate phase does not form in a system. Then extrapolate the liquidus curves below the stable three-phase equilibria and finally, estimate temperatures and compositions of metastable three-phase equilibria between two other phases and the liquid.

Another useful technique is to extrapolate a liquidus line from a peritectic equilibrium to estimate the congruent melting temperature of a compound. This estimate may be more useful than the information about the peritectic equilibrium itself as the equilibrium involves only two phases.

2.1.3 Information about equilibria in higher-order systems

When you optimise a binary system, the available experimental information can often be described equally well by very different sets of parameters. Information about ternary and higher order systems that include the components of the binary system are then often useful for optimising the binary system (the same is true of information about quaternary system with respect to optimising ternary systems). Often, it is only when you extrapolate the optimised parameters to a higher-order system that you can judge which of the sets of model parameters that is best. Sometimes information from several ternary systems will be required for making a reliable judgement about which set of parameters that best describes a particular binary system.

2.1.4 Information about heat capacities

If you want to optimise your system based on data about heat capacities, then the corresponding parameters should first be optimised separately before you perform the general optimisation of your system. The parameters should be kept fixed during the general optimisation. This is because it is normally not possible to optimise heat capacity data together with other data in a meaningful way. If you do this, then it is very likely that the parameters related to the heat capacity will be badly determined.

2.2 POP-file syntax

It is recommended that you specify an experimental equilibrium as close as possible to actual experimental conditions. Ideally, you will have information about the set of stable phases, temperature, pressure and some or all compositions for the equilibrium.

In addition, it is recommended that you keep the POP-file well-organised and extensively commented. This is especially important if someone else will be reassessing the system in the future as new experimental information becomes available.

If you have too many equilibria defined in your file, or if you define equilibria with different sets of components in the same file, then you must divide the POP-file in “blocks” (using the `FLUSH_BUFFER` command). You will know whether you have defined too many equilibria for one block when you compile the file. A too large number of equilibria leads to an error message when you compile the file.

If you use the `DEFINE_COMPONENTS` command, then it must be the first command in the POP-file (or if you have several blocks, `DEFINE_COMPONENTS` can also be put as the first command after `FLUSH_BUFFER`). This is because the command automatically reinitiates the current workspace, so the effect of any commands placed before it will not be saved.

The last command in the POP-file should always be `SAVE_WORKSPACES`.

2.2.1 Legal commands

The following commands can be used in a POP-file. For information about all these commands, see *Thermo-Calc Console Mode Command Reference*.

- `ADVANCED_OPTIONS`
- `CHANGE_STATUS`
- `COMMENT`
- `CREATE_NEW_EQUILIBRIUM`
- `DEFINE_COMPONENTS`
- `ENTER_SYMBOL`
- `EVALUATE_FUNCTION`
- `EXPERIMENT`
- `EXPORT`
- `FLUSH_BUFFER`
- `IMPORT`
- `LABEL_DATA`
- `SAVE_WORKSPACE`
- `SET_ALL_START_VALUES`
- `SET_ALTERNATE_CONDITION`
- `SET_CONDITION`
- `SET_NUMERICAL_LIMITS`
- `SET_REFERENCE_STATE`
- `SET_START_VALUE`
- `SET_WEIGHT`

- TABLE_END
- TABLE_HEAD
- TABLE_VALUES

2.3 Entering experimental information

A POP-file in Thermo-Calc (*.POP) will mainly consist of descriptions of equilibria, each equilibrium describing a data point that is based on an experiment from the existing literature on your system. An equilibrium is specified using ED_EXP commands.

The following is an example of a specification of an equilibrium with a single stable phase in the Au-Cu system. Equilibria with a single stable phase are often from experiments where enthalpies of mixing or chemical potentials have been measured.

```
CREATE_NEW_EQUILIBRIUM 1 1
CHANGE_STATUS PHASE LIQUID=FIX 1
SET_CONDITION T=1379 P=1E5 X(LIQUID,AU)=0.0563
SET_REFERENCE_STATE AU LIQ * 1E5
SET_REFERENCE_STATE CU LIQ * 1E5
EXPERIMENT HMR=-1520:200
COMMENT Measuremnt by Topor and Kleppa, Met trans 1984
```

The subsections that follow explain what the commands in the above specification do.

2.3.1 Creating a new equilibrium

The first command in an equilibrium specification is always CREATE_NEW_EQUILIBRIUM. This command takes two integers as arguments. The first integer should specify a unique identifier that can be used later to refer to the equilibrium in question. The second integer is an initialisation code 0, 1 or 2:

- Code 1: This means that all components are entered but all phases are suspended. This code will be appropriate in most cases.
- Code 0: This means that all components and phases in the equilibrium are suspended initially. This code is needed if you are using experimental data about systems of different orders simultaneously, such as data about both binary and ternary systems for example.
- Code 2: This means that all components and phases are initially entered.

For example, the first command in the example above creates an equilibrium data point with the unique identifier 1. All components of the system are entered but all phases are suspended:

```
CREATE_NEW_EQUILIBRIUM 1 1
```

2.3.2 Setting equilibrium conditions and their uncertainty

All the equilibrium-related commands that are placed in between two CREATE_NEW_EQUILIBRIUM commands are interpreted as specifying the conditions of the equilibrium created by the first command. The values of these conditions should be based on the conditions specified in the experiment that you base the equilibrium definition on.

In the example, the `CHANGE_STATUS` command specifies that the equilibrium is a single-phase equilibrium with the liquid phase.

```
CHANGE_STATUS PHASE LIQUID=FIX 1
```

Furthermore, the temperature is 1379 K, the pressure is 1 bar (1E5 Pa) and the mole fraction of Au is 0.0563.

```
SET_CONDITION T=1379 P=1E5 X(LIQUID,AU)=0.0563
```

Sometimes you have reason to think that some information about an equilibrium condition may not be accurate. If this is the case, then you specify the degree of uncertainty for a condition with a colon after the condition quantity, directly followed by either an absolute value or a percentage. For example, if you have reason to doubt the reliability of the Au mole fraction measurement, then you could write:

```
SET_CONDITION T=1379 P=1E5 X(LIQUID,AU)=0.0563:10%
```

This indicates that you think the Au mole fraction is between 0.0507 and 0.0619. PARROT will calculate the equilibrium twice, once for 0.0507 and once for 0.0619, and then compute the change of each experimental value between these two equilibria. This change will then be used to modify the uncertainty specified for the experimental value (see section 2.3.4).

The possibility of specifying uncertainties on conditions may be useful when you have experimental data about a ternary system. In a binary system, a tie-line is determined by the two phases, the temperature and the pressure. Measurements of the compositions of both phases can then be used as experimental data. To specify the tie-line in a ternary two-phase equilibrium one of the four compositions must also be set as a condition. If the measurement of this composition has the same uncertainty as the other compositions, then you can assign this same degree of uncertainty both to the composition selected as a condition and the compositions selected as experimental data. Alternatively, the sample (overall) composition could be used as a condition and all four phase compositions could be used as experiment values. These phases must have the status “entered” (not fixed) since the relative amount of each is unknown.

2.3.3 Setting reference states

If you want to specify a non-default reference state for one or several of your system components, then you do this with the `SET_REFERENCE_STATE` command. In the following example, the reference state for the liquid phase is set to current temperature (“*”) and a pressure of 1E5 Pa.

```
SET_REFERENCE_STATE AU LIQ * 1E5  
SET_REFERENCE_STATE CU LIQ * 1E5
```

Note that in order for these reference states to be taken into account when the measurement of the enthalpy per mole of the system is entered with `EXPERIMENT`, the R-suffix must be used.

2.3.4 Entering the experimental value and its uncertainty

The `EXPERIMENT` command is used to specify the quantity that the calculated results should be fitted to. When specifying this quantity, you both specify the measured quantity itself and your estimation of the uncertainty of this value.

In the example, the EXPERIMENT command specifies the enthalpy per mole of the system, where this value is specified with respect to non-default reference states that you have set in the equilibrium specification:

```
EXPERIMENT HMR=-1520:200
```

The value before the colon is the quantity of the variable. The value after the colon is an assessment of the uncertainty of this quantity. A higher value means a greater uncertainty. In the example, the uncertainty has been specified with an absolute value but it could also be specified in percent of the quantity:

```
EXPERIMENT HMR=-1520:13%
```

Several experiments can be specified after the EXPERIMENT command. Besides assigning values to conditions, it is also possible to specify that a certain quantity is greater than or less than a certain value. For instance, the following command would specify that the activity of the C-component is less than 0.01 and the mass fraction of the BCC phase of Ag is greater than 0.05:

```
EXPERIMENT ACR(C)<0.01:0.001 W(BCC,AG)>0.05:10%
```

Note that an experimental quantity can typically also be treated as an equilibrium condition, and vice versa. Which measured quantities that you treat as conditions and which quantities you treat as experimental data should be based on the accuracy of the different measurements. In most cases, the quantity that is based on the experimental technique with the lowest accuracy should be used as the experimental value.

2.3.5 Entering many equilibria as a table

When one thermodynamic variable (heat capacity, say) has been measured as a function of another (temperature, say), then experimental data is often presented as a table. By using the TABLE_HEAD, TABLE_VALUES and TABLE_END commands, you can enter such experimental information into the POP-file in the form of a table. These three commands should always be used together and must be entered in the right order.

The following block of commands exemplifies how a table can be entered into a POP-file:

```
TABLE_HEAD 1
CREATE_NEW_EXPERIMENT @@ 1
CHANGE_STATUS PHASE TETR=FIX 0
CHANGE_STATUS PHASE ION=ENT 1
SET_CONDITION P=101325 N=1 ACR(O2,GAS)=.21 X(ALO3/2)=@1
EXPERIMENT T=@2:5
TABLE_VALUE
0.4608      2476
0.5714      2380
0.6682      2276
0.7496      2176
0.5038      2426
0.6313      2249
```

TABLE_END

The TABLE_HEAD command is followed by a number. This number identifies the table as a whole, but also uniquely identifies the first equilibrium that is included in the table. This numeric identifier is incremented by one for each additional equilibrium in the table. Each row in the table represents one equilibrium. Thus, the equilibrium defined by the first row in the table above (on the line immediately below TABLE_VALUE) is uniquely identified with the number 1, the equilibrium defined by the second row by number 2, and so on until the last equilibrium on the sixth row, which is identified by the number 6.

Since the unique identifiers of the equilibria in the table are automatically generated from the unique numerical identifier that you give to the table, the unique numerical identifier that is normally given to the CREATE_NEW_EQUILIBRIUM command is replaced by two at signs ("@@").

At least one of the equilibria conditions specified with the SET_CONDITION command must be a condition whose value is set by the numbers in one of the table columns. To assign the numbers in a column to a condition, use the syntax SET_CONDITION <CONDITION>=@<COLUMN NUMBER>. For instance, in the example above, the mole fraction of X(ALO3/2) is for each equilibrium given by the number in the first column of the table.

At least one of the measured variables that are entered with the experiment command must be a variable whose value is set by the numbers in a table column. To assign the numbers in a column to a measured variable, use the syntax EXPERIMENT <VARIABLE>=@<COLUMN NUMBER>. In the example above, the measured temperature for each equilibrium is given by the number in the second table column.

The actual table should be entered between TABLE_VALUES and TABLE_END. Separate the columns with tabs or any number of spaces. The columns do not have to contain numbers, but can also contain other types of values, such as phase names for example. If a column contains text with spaces or special characters, then the text must be surrounded by double quotes ("ABC DEF&ghi", for example). The table may have columns that are not used, but each row must have exactly the same number of columns.

2.3.6 Setting site fraction start values

In some systems, you have to set non-default composition start values on some experiments in order to get them to converge properly during optimisation. This may be important if your system has a miscibility gap for example, in which case several equilibria with different compositions exist at certain temperatures. In such cases, you have to make sure it is specified in the POP-file which side of the miscibility gap the equilibrium is on. The calculated equilibrium could otherwise end up on the "wrong" side, resulting in a large error.

In the POP-file, you can set the site fraction of a constituent on an equilibrium with SET_START_VALUE. For example, the following command sets the composition start value of VA constituent of the FCC_A1 phase to 0.99:

```
SET_START_VALUE Y(FCC_A1,VA#2)=0.99
```

Setting the mole fraction for a component in a phase is not as reliable as setting the site fraction.

Additional equilibria that are entered after the last SET_START_VALUE command will automatically be given a similar start value if the following command appears previously in the POP-file:

```
SET_ALL_START_VALUES Y
```

If start values are not set automatically for phase constituents, then you may have to set the composition start value for each equilibrium (or table of equilibria) separately.

2.3.7 Comments about an equilibrium

The COMMENT command inserts a comment about the equilibrium that is saved to the PARROT workspace file when the POP-file is compiled. The comment must be entered on one line.

Comments can also be entered immediately after a dollar sign (“\$”), but these are not saved to the PARROT workspace. Instead, these comments will simply be ignored by Thermo-Calc when the POP-file is compiled.

2.3.8 Simultaneous use of binary and ternary experiments

You can use experimental information about binary and ternary systems (as well as systems of even higher order) in the same POP-file and optimisation. To do this, you must inform the PARROT module that not all components of the ternary system (or system of even higher order) should be considered. This allows you, for example, to use experimental information about a binary equilibrium that exists within a ternary system.

Use CHANGE_STATUS with the keyword COMPONENT and which components that should be entered. You should also give the initialization code 0 (rather than 1) to the CREATE_NEW_EQUILIBRIUM command. This code indicates that all components must be entered.

For example, the following set of ED_EXP commands characterises a binary (A-B) three-phase equilibrium (FCC-BCC-LIQ) in a ternary system (A-B-C). It can be used in a POP-file that otherwise only contain experimental information about the ternary system (A-B-C):

```
CREATE_NEW_EQUILIBRIUM 1 0
CHANGE_STATUS COMPONENT A B = ENTERED
CHANGE_STATUS PHASE FCC BCC LIQ=FIX 1
SET_CONDITION P=1E5
EXPERIMENT T=1177:10
COMMENT from A-B
```

2.4 Using stability conditions

Early in the optimisation when parameter values are not so good it is often useful to set the driving force for precipitation of a phase (per mole of components) to make sure that it appears where it should. The “DGM” condition is very useful for making sure that phases appear where they should. It can be removed when the optimisation becomes stable.

The DGM condition is also useful for suppressing phases that appear where they should not appear. Do this by setting the value and the uncertainty to something reasonable. For example, the driving force for precipitation of the BCC phase could be set as follows:

```
DGM(BCC) < -0.1:0.1
```

This will produce an error even before BCC becomes stable and the weight of the experiment in question can be adjusted as needed. If you instead specify that the driving force for the BCC phase should be below 0 ($\text{DGM}(\text{BCC}) < 0.1\text{E-}4$ for example), then the optimisation will converge extremely slowly (if at all). With such a sharp error condition, calculating the equilibrium will be like finding the minimum point on a lawn that gently slopes towards a rock wall. The optimisation will keep bouncing into the wall.

Another useful stability condition is the phase stability function, abbreviated “QF”. This can be used to specify that a phase is outside the miscibility gap for a solution phase. If $\text{QF}(\text{phase})$ is negative, then the phase is inside the miscibility gap; if $\text{QF}(\text{phase})$ is positive, then it is outside the miscibility gap.

2.5 Grouping equilibria with labels

If you have several equilibria describing experiments that you want to be able to treat collectively, you can give all of the equilibria one and the same label.

Use LABEL to give an equilibrium a label. The label must start with the letter “A” and can only be up to four characters long. For example, you can insert the following command in several equilibria specifications:

```
LABEL ALH
```

During the optimisation, you could use this label to, for example, set all the equilibria that have it to the same weight in the ED_EXP module.

2.6 Dividing the POP-file into several blocks

For two different reasons, you may have to divide your POP-file into different blocks:

- PARROT uses a buffer for storing compiled experimental data. If there is not enough memory in this buffer for storing all the experiments from the POP-file, then the POP-file can be divided into several “blocks”. When PARROT encounters the end of a POP-file block, it saves the experimental data that has been compiled to PARROT workspace, clears out the memory buffer and then proceeds to read the next block in the POP-file. This goes on until the experimental data from all the blocks have been saved to the PARROT workspace.
- If you want to put equilibria with different sets of components in one and the same POP-file, then you must divide the file up into blocks. Each block should only contain equilibria that all have the same set of components.

To divide the equilibria in the POP-file into different blocks, enter the FLUSH_BUFFER command between two equilibria. This command marks the beginning of new block. When the command is encountered during the compilation of the POP-file, the compilation of the current block will be terminated, the equilibria saved to the PARROT workspace file and a new block will be initiated. That the buffer is reinitiated means that all functions and constants that were entered in the previous block must be entered again in new block in order to be used.

When editing your experimental data in the ED_EXP module, you can select which block to load and edit using `READ_WORKSPACES <BLOCK NUMBER>`. The block before the first instance of `FLUSH_BUFFER` is block number 1, the block after the first instance and before the second instance is block number 2, and so on.

3 Creating the setup file

The second step is to create the setup file. This is an ordinary Thermo-Calc macro file (*.TCM). Typically, your setup file should contain the following:

- A system definition.
- A section where you enter model parameters and optimising variables.
- A GO PARROT command.
- A section where initial values are assigned to the optimising variables for the first optimisation cycle.
- A CREATE_NEW_STORE_FILE <FILENAME.PAR> which creates the PARROT workspace file onto which the results of your optimisation will be continuously saved and updated. By default this workspace file will have the same filename as the setup file, but with the "*.PAR" filename extension.
- A COMPILE_EXPERIMENTS <FILENAME.POP> command. Since you will often encounter syntax errors when you first try to compile a POP-file, it may be useful to execute this command from the console rather than as part of the setup macro file.
- Normally, you will also have a SET_INTERACTIVE at the end of the setup file. This command returns control over Thermo-Calc to the console.

It is possible to interactively execute all these commands one at a time in the console. Indeed, it may be instructive to enter the commands that would normally go into the setup file interactively while using Thermo-Calc's on-line help while you are learning to use the PARROT module.

3.1 Defining the system

If you have an existing database with information about your system, then you can write the system definition part of your setup file in two different ways.

- If you have an unencrypted user database, then you can define your system by entering DATA module commands directly in the setup file. Use the DATA commands that you normally use to define a system when doing equilibrium calculations in Thermo-Calc Console Mode.
- Define your system by entering DATA module commands directly in the console, then generate a macro file with the GIBBS command LIST_DATA <NAME OF SETUP FILE>.TCM P. This command will save all the data that has been loaded in the GIBBS workspace as a macro file. This file can then be modified as you see fit into an optimisation setup file.

3.1.1 Creating additional composition sets of a phase

If you need several composition sets of a phase, then these should be created with the GIBBS command AMEND_PHASE_DESCRIPTION <PHASE NAME> COMPOSITION_SETS <COMPOSITION SET NUMBER> in the setup file.

Note that if you add composition sets of a phase after the POP-file has been compiled, the data structure in the workspace will be corrupted and you have to recompile the POP-file.

3.2 Entering the optimising variables

When you have retrieved the data about your system for the setup file, you must manually enter the model parameters that you want to optimise. This is done using the GIBBS command `ENTER_PARAMETER`, but when you specify the Gibbs energy expression for the parameter, you enter variables that can be given various values into the expression. It is the values of these variables (and thus, the Gibbs energy of the parameter) that you will be optimising.

How you in detail specify the optimising variables in your setup file will depend on which phase you are trying to optimise and what thermodynamic model that is used in the optimisation. See

Appendix 2: Thermodynamic models for a range of examples.

3.2.1 Entering optimising variables in parameter specifications

There are ninety-nine pre-defined variables in the PARROT module which you can use in your optimisation. These variables are referred to as “V1”, “V2”, “V3”, ... “V98” and “V99”. When the GIBBS workspace is initialised, all optimising variables are set to a fixed numerical value of zero. When you do an optimisation, there will always be some variable for which you want to evaluate the best value, but you can assign fixed values to optimising variables too. It is often useful to enter all the variables that you can conceivably want to optimise, and then you can interactively select and change which variable to focus at various stages of the optimisation.

The setup file excerpts that follow exemplify how optimising variables are entered. However, before the optimising variables are introduced in this example, the following command enters an ionic liquid solution phase (a phase for which the user has some experimental information):

```
ENTER_PHASE IONIC_LIQ Y,  
AL+3,ZR+4 ; AL02-1,O-2 ; N N
```

This ENTER_PHASE command enters a phase called IONIC_LIQ into the GIBBS workspace. This phase is of the type ionic liquid solution, which is specified by type code “Y”. Such a phase is modelled by the Ionic Two Sublattice Liquid Model. The next two arguments specify the two sublattices, in this case the cations AL+3,ZR+4 and the anions AL02-1,O-2. Components that interact in a sublattice are separated by a comma. The first “N” at the end means that no additional constituents will be added to the phase. The second “N” means that a list of possible parameters for the phase should not be shown in the console.

There will be model parameters that describe the phase which you do not want to optimise, in which case the Gibbs energy expression of the parameter will not contain any optimising variables. For example, the following three parameters of the IONIC_LIQ phase do not contain any optimising variables:

```
ENTER_PARAMETER G(IONIC_LIQ,AL+3:AL02-1;0) 298.15  
+2*GAL2O3_L; 6000 N  
  
ENTER_PARAMETER G(IONIC_LIQ,AL+3:O-2;0) 298.15  
+GAL2O3_L+400000; 6000 N  
  
ENTER_PARAMETER G(IONIC_LIQ,ZR+4:O-2;0) 298.15 +2*GZR02L;  
6000 N
```

These parameters define the Gibbs energy expressions for three constituents in the constituent array, namely AL+3:AL02-1, AL+3:O-2 and ZR+4:O-2. The names that occur in these expressions—that is, GAL2O3_L and GZR02L—are the names of complex functions of temperature that are entered in the GIBBS workspace with the ENTER_SYMBOL command.

The numbers that frame the Gibbs energy expressions specify the temperature range within which the expression should be applied. In the above, the expressions are applied in a temperature range from 298.15 K to 6000 K.

In this example, suppose that you want to optimise the model parameter for the ZR+4:AL₂O₃-1 combination, then you could enter the variables V1 and V2 in the Gibbs energy expression for that parameter in the following way:

```
ENTER_PARAMETER G(IONIC_LIQ,ZR+4:AL2O3-1;0) 298.15 +GZRO2L
+2*GAL2O3_L +V1+V2*T; 6000 N
```

You will normally have different optimising variables in different parameters, but in some cases you may want several parameters to vary together, in which case you can use the same variable in the Gibbs energy expressions of several parameters.

3.2.2 Entering interaction parameters

For some systems, you have to consider interaction parameters in order to take excess Gibbs energy into account and correctly model the phase. By default, the Thermo-Calc program will use a Redlich-Kister expression for the excess Gibbs energy. The coefficient is entered after the constituent array when you use the ENTER_PARAMETER command. In the last section, all the parameters were entered with a coefficient of zero ("0"). To take account of the excess Gibbs energy, you might enter the following:

```
ENTER_PARAMETER L(IONIC_LIQ,ZR+4:AL2O3-1,O-2;0) 298.15
+V11+V12*T; 6000 N
```

```
ENTER_PARAMETER L(IONIC_LIQ,ZR+4:AL2O3-1,O-2;1) 298.15
+V13+V14*T; 6000 N
```

```
ENTER_PARAMETER L(IONIC_LIQ,ZR+4:AL2O3-1,O-2;2) 298.15
+V15+V16*T; 6000 N
```

Parameters with interaction coefficients are referred to as "interaction parameters". The regular parameter has a coefficient of 0. The subregular parameter has a coefficient of 1. It is possible to interaction coefficients up to a value of 9, but you are advised not to use more than three coefficients (that is, with coefficient values of 0, 1 and 2).

3.3 Initialising optimising variables in PARROT

Variables that have been entered into Gibbs energy expressions for model parameters in the setup file has to be initialised and given start values before they are optimised.

You have to estimate what start values to give to the optimising variables based on your knowledge of the system or of relevantly similar systems. There are no general rules for what start values your optimising variables should have. If you are not able to make any informed estimation about what start values to use, then you may not be able to find a good fit between your experimental data and your calculated results. In this case, it is often best to start the optimisation in the so-called alternate mode. In alternate mode, the PARROT module will help you find some reasonable start values for your optimising variables. For more information about optimising in alternate mode, see section 6.

Even if you are starting your optimisation in alternate mode, you still have to initialise the optimising variables though (you can all give them a start value of 0).

To initialise an optimising variable and to give it a start value, use either `SET_OPTIMIZING_VARIABLE` or `SET_SCALED_VARIABLE`. Using `SET_OPTIMIZING_VARIABLE` results in a scaling factor equal to the current value of the parameter unless the current value is zero, in which case the scaling factor will be 1000. This is typically appropriate for enthalpic parameters, but not necessarily for other parameters. The scaling factor is the factor by which the optimising variable value will be varied during the optimisation.

For example, if you want to optimise the variables V1, V2, V11, V13 and V17, then you could enter the following in the setup file:

```
SET_OPTIMIZING_VARIABLE 1 +43000
SET_OPTIMIZING_VARIABLE 2 -35.2
SET_OPTIMIZING_VARIABLE 11 +72000
SET_OPTIMIZING_VARIABLE 13 +27000
SET_OPTIMIZING_VARIABLE 17 -100000
```

The value given to each optimising variable (+43000 for variable V1 for example) will also be that variable's initial scaling factor. If you want to set the initial scaling factor of a variable to a value that is lower or higher than its start value, then use `SET_SCALED_VARIABLE <START VALUE><SCALING FACTOR><MIN VALUE><MAX VALUE>`.

3.4 Creating the workspace file

In the setup file, after the system definition and the entered optimising variables, use the PARROT command `CREATE_NEW_STORE_FILE` to create and save a PARROT workspace file for the system defined in the setup file. For example, the following command will save all current data on system definitions (elements, species, components, phases), symbol definitions (constants, variables, functions, tables) and parameters which are stored in the workspace used by GIBBS, POLY and PARROT into a file called "AL2O3-ZRO2.PAR":

```
CREATE_NEW_STORE_FILE AL2O3-ZRO2
```

Note that the experimental information in the POP-file is not saved to this PARROT workspace file.

3.5 Compiling the POP-file

At the end of the setup file, use the command `COMPILE_EXPERIMENTS` followed by the name of your POP-file to compile the data in that file and save it to the current PARROT workspace file. Note that you must first have opened or created a new PARROT workspace file before you compile your POP-file. The other argument to the command can typically be given default values with comma signs (","):

```
COMPILE_EXPERIMENTS AL2O3-ZRO2.POP,, ,
```

If Thermo-Calc encounters errors in the syntax of the POP-file, then error messages will be presented in the console. Normally, the compilation will also be stopped when syntax errors are encountered. If this happens, then you must correct the errors and recompile the file. It will often be useful to have two console windows open when you do this. That way, you can use one console for editing the experimental data and another console for compiling and reading any further error messages.

3.6 Returning control to the console

Put the command `SET_INTERACTIVE` at the end of the setup file if you want to return control to the console when the commands in the setup file have all been executed.

4 Optimising your system in PARROT

When you have created a POP-file, a setup macro file and (optionally) EXP-files, you should start doing optimisation runs interactively in the console. This process can be divided into a number of steps (such as running optimisation, evaluating optimisation results, modifying weights and models and adding new information), but usually, you will have to go through many optimisation cycles before you achieve a satisfactory fit between calculation results and experimental data.

Typically, it is not obvious when an optimisation is finished. You must exercise your judgement about when the fit you have achieved is good enough.

Note that it is possible to interactively change almost everything in the initial setup macro file and in the POP-file. For example, you can add more parameters to be optimised and modify or add more experimental information.

4.1 Changes that require POP-file recompilation

Some changes that you do interactively in the PARROT module or the ED_EXP sub-module will destroy the data structure in the workspace. For example, if you add more composition sets to a phase, then the number of phases actually change and the links break between the experimental data and the thermodynamic models. Note that if the global minimisation technique is used in POLY, then new composition sets that corrupts the PARROT workspace may be created automatically. If this happens, then you have to compile the POP-file again.

If you recompile the POP-file, then the experimental equilibria will be calculated with the default start values.

4.2 Workflow

Normally, the workflow during an optimisation is the following:

- 1 In the PARROT module, run the setup macro file using `MACRO_FILE_OPEN`.
- 2 Go to ED_EXP with `EDIT_EXPERIMENTS`, use `READ_WORKSPACES` to load your experimental data and then execute `COMPUTE_ALL_EQUILIBRIA`.
- 3 If an equilibrium cannot be computed, then this either means that the start values (site fractions, for example) were inappropriate or that the equilibrium cannot be calculated with the current set of parameters. If the latter, then the equilibrium may have to be excluded from the optimisation. However, you should make an effort to exclude the possibility that the reason is just that you do not have the appropriate start values for the equilibrium.

- 4 When each equilibrium is either successfully computed or excluded from the calculation, go back to the PARROT module and run the OPTIMIZE_VARIABLES command. Give the number of optimisation iterations you want the Thermo-Calc program to carry out as an argument to the command. This number specifies how many different sets of values for the optimising variables that PARROT will try to fit with the experimental data. It is recommended that you run "0" iterations first to see the discrepancy between the experimental data and the initial calculated results. The current workspace file is automatically updated with the last set of optimised variables and calculated results.
- 5 List and evaluate the result of the optimisation using LIST_RESULT. Typically, what will be of most interest in the results are the relative standard deviation of the optimising variables and the error values in the list of calculated experimental data results. If the relative standard deviation for an optimising variable is very large, this means that the variable doesn't have any effect on the computed results and that the variable should not be used (you can make PARROT ignore the variable by setting its value to "0" with SET_FIXED_VARIABLE). If the relative standard deviation is close to zero for a variable, then this might indicate that you need to use more variables. When it comes to errors in the calculated equilibrium results, these can be decreased by lowering the weight that PARROT gives the experiment in the calculation (the more weight, the more the experiment will affect the calculated results).
- 6 Use OPTIMIZE_VARIABLES or CONTINUE_OPTIMIZATION to carry out a new optimisation cycle. Go back to step 5 above and continue to do optimisation cycles until you judge that there is a good enough fit between your calculated results and your experimental data. It is recommended that you do not only rely on feedback from LIST_RESULT but also frequently plot various diagrams and compare them visually to some of your experimental data.

4.3 Ensuring that all equilibria can be computed

When you have run the setup macro file, you should start by trying to compute all the equilibria that were created when your POP-file was compiled. To do this, you must first go to the ED_EXP submodule, and load the experimental data from the current PARROT workspace file with the READ_WORKSPACES <BLOCK #> command (normally, the block number will be "1").

When you have loaded all equilibria, you should try to compute them all using the ED_EXP command COMPUTE_ALL_EQUILIBRA. A list of calculated equilibrium results will be shown in the console, but if the Thermo-Calc program is not able to reach an equilibrium for an experiment, then the program will abort the calculations and show an error message.

The following list shows an example output where the first two experiments in the POP-file have been successfully computed, but an equilibrium is not reached for the third experiment. No other equilibria are computed, since the calculations are aborted when the error occurs.

Eq	Lab	Iter	Weight	Temp	Exp	Fix	phases	or	comments
1	A1	22	1.	2575.3		IONIC_	FLUORI	TETR	

```
2 A1      18      1.      2156.9      IONIC_ CORUND TETR
```

```
*** ERROR 1614 IN QTHISS
```

```
*** CONDITIONS CAN NOT BE FULLFILLED
```

When this happens, try to find out why the equilibrium calculation fail. Are the condition start values that you have given the equilibrium inappropriate for example? If you have a similar equilibrium that has already been successfully computed, then you can select the equilibrium that does not work, use the ED_EXP command SET_ALL_START_VALUES Y and accept all the suggestions for the site fractions. Alternatively, use TRANSFER_START_VALUES <EQUILIBRIUM NUMBER> to make the currently selected equilibrium inherit the start values of the indicated equilibrium. Run COMPUTE_EQUILIBRIUM again to see if the equilibrium can be computed with the “inherited” start value.

If you cannot find an explanation of why the equilibrium cannot be computed, then it may be that the experiment is faulty. If so, then this experiment should be excluded from the optimisation by being given a weight of 0 (see the next section). You should then run COMPUTE_EQUILIBRIUM again to make sure that all equilibria can be computed. However, experiments that fail to reach equilibria could succeed in doing so if the optimising variables have other values. It is therefore sometimes worth trying to include experiments later in the optimisation process (when the optimising variables are relatively close to their final values), even if they had to be excluded at its start. In some cases, it might also be appropriate to change the error tolerance for in an experiment (using the ED_EXP command EXPERIMENT).

You must save the parrot workspace with SAVE_WORKSPACES before you exit the ED_EXP module.

4.4 Setting weights

To change the weight of an experiment, first enter the ED_EXP module from PARROT using EDIT_EXPERIMENTS. Then use the SET_WEIGHT command to set the weight for a specific equilibrium (equilibrium number “3” for example), for a range of equilibria (the equilibria “10-19”) or all the equilibria that have a certain label (such as all the equilibria with the label “A1”). To save the new weight(s), use SAVE_WORKSPACES.

By default, all the experimental information in your POP-file is treated equally: each experiment has a weight of “1”. However, experiments that you excluded because they did not reach equilibrium will now have a weight of “0”. The higher the weight of an experiment is, the more will Thermo-Calc try to ensure that the computed result fits the condition measured in that experiment.

The contribution of an experiment’s error to the sum of errors is multiplied by the square of the weight set for that experiment. In other words, if you want to cut the error of an experiment with weight 1.0 by half, then you should use a weight of 0.7. This will make the experiment’s error 0.49. Setting the weight to 1.4 will instead make the error twice as large (1.96).

Note that if you use READ_WORKSPACES after you have changed weights on the experiments, your changes will be lost unless you have saved those changes. In the ED_EXP module, you can get a list of all the experiments in the console if you first read the experimental data from the workspace file (with READ_WORKSPACES) and then use LIST_ALL_EQUILIBRIA.

4.5 Optimising and evaluating listed results

When you have ensured that all the experimental data points can reach equilibrium, go back to the PARROT module and use OPTIMIZE_VARIABLES <NUMBER OF ITERATIONS> with “0” iterations. Present the result of this initial optimisation by using LIST_RESULT.

When evaluating the optimisation results, the most important information is found in the list of experiments at the end of the listed results. The rightmost column shows the difference between the value of the experimental data and the value computed by Thermo-Calc in the last optimisation cycle. The following shows part of list of experiments in the output of a LIST_RESULT command:

106 ACR(B)=0.34	0.3127	2.89E-02	-2.7282E-02	-0.9444
107 ACR(B)=0.23	0.2085	2.89E-02	-2.1522E-02	-0.7450
108 ACR(B)=0.12	0.1042	2.89E-02	-1.5761E-02	-0.5455
110 HMR(LIQUID)=-1964	3.6380E-12	5.00E+02	1964.	3.928
111 HMR(LIQUID)=-3500	0.000	5.00E+02	3500.	7.000 *
112 HMR(LIQUID)=-4588	0.000	5.00E+02	4588.	9.176 *
113 HMR(LIQUID)=-5239	7.2760E-12	5.00E+02	5239.	10.48 *
114 HMR(LIQUID)=-5454	-3.6380E-12	5.00E+02	5454.	10.91 *
115 HMR(LIQUID)=-5233	1.8190E-12	5.00E+02	5233.	10.47 *
116 HMR(LIQUID)=-4575	-1.8190E-12	5.00E+02	4575.	9.150 *
117 HMR(LIQUID)=-3481	-1.8190E-12	5.00E+02	3481.	6.962 *
118 HMR(LIQUID)=-1950	0.000	5.00E+02	1950.	3.900

The experiments for which the optimisation has not found a good fit are marked with an asterisk (“*”) or a hash sign (“#”) in the rightmost column. An asterisk in this column indicates that the experiment has an error that is larger than the error tolerance and a hash sign indicates that the error is much larger. The error tolerance will depend on the degree of accuracy you have set for that particular piece of experimental information. The error values themselves are presented in the next to rightmost column.

Note that if you are optimising in “alternate mode”, then you usually do not need to change the weights for the experiments. Instead, you can simply run OPTIMIZE_VARIABLES again, specifying, say, thirty iterations, evaluate the results again, and continue in that way until you are satisfied with the results. See section 6 for more information about alternate mode.

4.5.1 The critical set of experiments

The set of weighted experiments that you end up with as a result of your optimisation is referred to as the “critical set” of experiments. When you work to determine this set and the weights of the experiments, the following factors should be reflected in both the selection of experiments and the weights of those experiments:

- The reliability of the experimental technique.
- The extent of agreement/disagreement between independent measurements of the same quantity.

- The extent of agreement/disagreement between data obtained with different experimental methods.

4.6 Continuing the optimisation and resetting variables

To continue the optimisation after you have used OPTIMIZE_VARIABLES, you can either use OPTIMIZE_VARIABLES <NUMBER OF ITERATIONS> again or CONTINUE_OPTIMIZATION <NUMBER OF ITERATIONS>.

If you use CONTINUE_OPTIMIZATION, then PARROT continues the optimisation using the same Hessian matrix. If the optimisation is going well but you run out of iterations, then continuing with the same Hessian matrix may be useful.

As before, use LIST_RESULT to inspect the fit between the calculated and the experimental values for your experiments. If necessary, adjust the weights of the experiments again in the ED_EXP module. You may even discover that you have to exclude additional experiments from the optimisation.

Besides looking at the rightmost column in the list of experiments (that is produced when you use LIST_RESULT), it is also useful to look at the “sum of squares” of the errors of all the variables. This is shown in the output of OPTIMIZE_VARIABLES and CONTINUE_OPTIMIZATION. This sum should be as low as possible.

The following screenshot shows part of the output that you get when using LIST_RESULT:

```
== OPTIMIZING VARIABLES ==

AVAILABLE VARIABLES ARE V1 TO V00

VAR.   VALUE           START VALUE      SCALING FACTOR    REL.STAND.DEV
V1      6.94641846E+05    4.30000000E+04    4.30000000E+04    1.42299558E+00
V2     -3.32619842E+02   -3.52000000E+01   -3.52000000E+01   7.91369573E-01
V11     8.37422071E+04    7.20000000E+04    7.20000000E+04    2.02212692E-01
V13     2.65253341E+04    2.70000000E+04    2.70000000E+04    1.45201828E-01
V17    -3.66637428E+05   -1.00000000E+05   -1.00000000E+05    8.04279315E-01

NUMBER OF OPTIMIZING VARIABLES :    5
ALL OTHER VARIABLES ARE FIX WITH THE VALUE ZERO
THE SUM OF SQUARES HAS CHANGED FROM    3.50196879E+05 TO    3.49096720E+05
DEGREES OF FREEDOM    14.  REDUCED SUM OF SQUARES    2.49354800E+04
```

Here you can see that the sum of squares have decreased slightly. (The final sum of squares that you get after an optimisation run is also presented directly in the output from OPTIMIZE_VARIABLES and CONTINUE_OPTIMIZATION.) As long as the sum of squares figure is decreasing, the optimisation is working as it should.

When your calculated results have been improving for a while, that is, when the sum of squares has been decreasing, and you are confident that you are making definitive progress, then you should use the PARROT command RESCALE_VARIABLES to reset the starting values of the optimising variables to their current values. When you do this, the start values and scaling factors of the optimising variables (shown in the table's third column in the preceding screen shot), are all set to the current values (shown in the table's second column).

If your optimisation is not going well, then you may want to reset the values of your optimisation variables to their current start values. To do this, use RECOVER_VARIABLES.

After rescaling the variables, continue the optimisation using OPTIMIZE_VARIABLE again. Cycle through calls of the commands OPTIMIZE_VARIABLES/CONTINUE_OPTIMIZATION, CONTINUE_OPTIMIZATION, RESCALE_VARIABLES and OPTIMIZE_VARIABLES again until your set of optimising variables does not seem to improve anymore. Even when it seems to you that you have reached this point, it is recommended that you use OPTIMIZE_VARIABLES/CONTINUE_OPTIMIZATION a few extra times to make sure that the variable values cannot be further improved, that is, that the sum of squares of the errors of all the variables cannot be reduced further. If the final solution is repeatedly calculated after the same number of iterations and reaching the same result, then it is generally reasonable to accept the current variable values.

4.7 Updating the set of optimising variables

Besides evaluating the fit between the calculated results and the experimental data, you should also ensure that you have the right number of optimising variables. If you have too many or too few optimising variables, then you may not be able to use your model to accurately extrapolate thermodynamic properties in a wide range of temperature, pressure and composition conditions.

With more optimisation parameters, the sum of errors usually decrease, but the parameters also tend to become less precisely determined. A measure of this precision is the Relative Standard Deviation (RSD) of each optimising variable. However, the RSD is only significant if you have used RESCALE_VARIABLES followed by an optimisation which converges and the values of the variables do not change much. The RSD tells you how much the parameter can be changed in either a positive or in a negative direction without changing the reduced sum of errors with more than one unit. A large RSD value thus means that the parameter has not been determined well.

== OPTIMIZING VARIABLES ==

AVAILABLE VARIABLES ARE V1 TO V00

VAR.	VALUE	START VALUE	SCALING FACTOR	REL.STAND.DEV
V1	2.01820948E+04	2.01874462E+04	2.01874462E+04	2.60519742E-02
V2	-2.90936164E+01	-2.90969472E+01	-2.90969472E+01	1.41576516E-02
V11	-2.18127453E+04			
V12	1.55559524E+01			
V15	2.36701148E+04	2.37258719E+04	2.37258719E+04	1.00528931E-01
V16	-7.56541897E+00	-7.64417807E+00	-7.64417807E+00	3.17218507E-01
V17	3.00342429E+03	3.02274918E+03	3.02274918E+03	2.60675550E-01
V19	2.20133190E+04	2.14534347E+04	2.14534347E+04	6.92923849E-01
V20	-6.72498276E+00	-6.31334932E+00	-6.31334932E+00	1.79242248E+00

NUMBER OF OPTIMIZING VARIABLES : 7

ALL OTHER VARIABLES ARE FIX WITH THE VALUE ZERO

THE SUM OF SQUARES HAS CHANGED FROM 1.51360340E-01 TO 1.44420287E-01

DEGREES OF FREEDOM 18. REDUCED SUM OF SQUARES 8.02334928E-03

If such a significant RSD has a value larger than 1, then this typically means that you are using too many optimising variables. However, the RSD values also depend on the weighting of the experiments, so it is sometimes possible to reduce the RSD by changing weights.

The RSD should not be too low either. If one or more of them is very close to 0, then this suggests that you are trying to optimise the system with too few optimising variables. You can add optimising variables interactively using ENTER_PARAMETER in GIBBS. However, it is recommended that you add a sufficient number of variables in the setup file from the start so that do not need to do this. You can fix variables to a value of 0 until you discover that they are needed.

Besides the RSD, you should also keep an eye on the variable values themselves (shown in the "VALUE" column, next to the leftmost column in the table shown in the preceding screenshot). If the value of a variable is in the order of 1E5 or higher and this variable has been put in the temperature-independent part of an interaction parameter's Gibbs energy expression, then you may not have the right weights on your experiments or you are using too many optimising variables. This is also typically the case if a variable with a value in the order of 10 or more (per mole atoms) is in the temperature-dependent part of the Gibbs energy expression of an interaction parameter. Such a high parameter value may lead to inverted miscibility gaps or re-stabilisation of a phase at high temperature. It is important to keep an eye on this since the temperature-independent and the temperature-dependent interaction parameter will often vary together. If the temperature-dependent interaction parameter cannot be reliably determined, you may have to fix its value.

Note that even if you have found a good enough fit between calculated and experimental values after an optimisation run, a parameter value may suddenly start to change by several orders of magnitude when you use OPTIMIZE_VARIABLES is used again. If this happens, then careful reconsideration of the weighting of the experiments of the set of optimising variables is required. However, when the optimisation has converged and the parameter values remain stable after repeated use of OPTIMIZE_VARIABLES, this should usually not happen.

4.7.1 Reducing the number of optimising variables

If you have one or more optimising variables with RSD larger than 1, then you should remove one or more of the variables by setting them to a value of 0, or alternately, set them to a reliably estimated value (estimated based on, for example, semi-empirical methods).

To fix an optimising variable to certain value, use the PARROT command `SET_FIX_VARIABLE <VARIABLE NUMBER> <VALUE>`.

4.8 Plotting intermediate results

It is typically not sufficient to look at the plain text output of `LIST_RESULT` to reliably judge how the optimisation is progressing. Plotting a phase or property diagram of the system you are optimising using the `POLY` and `POST` modules is often very useful.

It is recommended that you create a macro file for plotting the diagram since you probably want to repeatedly plot the diagram as you progress with the optimisation. This macro file can then be called directly from the `PARROT` module with the `MACRO_FILE_OPEN` command.

4.8.1 Superimposing experimental data from EXP-files

If you create one or several `EXP`-files based on the experiments in your `POP`-file, you can visually compare the fit your plot and the experimental data you are optimising the system against. An `EXP`-file contains data points specified in the `DATAPLOT` format. For more information about `DATAPLOT`, see the *DATAPLOT User's Guide and Examples*. You could make an `EXP`-file for each type of data that you are using in the optimisation.

Use the `POST` command `APPEND_EXPERIMENTAL_DATA` to superimpose the data points in an `EXP`-file on a phase or property diagram that you have plotted. For example, to put the experimental data points from a file entitled "`DP.EXP`" onto a plotted diagram, you could enter the following:

```
APPEND_EXPERIMENTAL_DATA Y DP.EXP 0 1
```

Such a command call tells Thermo-Calc to superimpose the data points in dataset 1 of `DP.EXP` onto the plot in the Console Results window, but not to impose any information about axis scaling, labels and other information which is normally found in the prologue of the `DP.EXP` file.

The `POST` command `QUICK_EXPERIMENTAL_PLOT` can also be used to superimpose data points on a plotted phase or property diagram.

5 Finishing the optimisation

When to stop optimising is a matter of judgement. It will rarely be the case that you finish optimising a system with the feeling that it cannot be improved.

The final step is to update your files with the final results. Besides updating your database file (or alternatively, creating a new database file), it is advised that you also update your setup and POP-files with the final weights and optimising variable values.

5.1 How to round off optimising variable values

When you save the optimising variable values, it is important that you round off the values correctly. When you have a metallic system, it is usually best to keep whatever number of decimal digits that you need to make less than one J/mol difference at 1000 K. When you have an aqueous system or a system that you are optimising based on data from very different temperature ranges, then this is not always the only appropriate guideline. However, rounding off the values in such a way that it gives a difference that is larger than one J/mol may lead to differences in the phase diagram.

Another approach you can take when rounding off variable values is by progressively fixing the value of more and more of the variables. This allows you to round-off all but one of the variables values so that each only has a few significant digits. Variables whose values have been rounded off in this way are easier to handle compared with those that are rounded off so that they continue to have many significant (non-zero) digits.

To round off the variable values according to this second approach, first use `SET_FIX_VARIABLE` to set the variable(s) with the highest RSD to a rounded off value (such as, say, 0.4). When you then re-optimize using `OPTIMIZE_VARIABLES`, the sum of errors will change. However, you should get almost exactly the same sum of errors as before after you rescale the variable values using `RESCALE_VARIABLES`. (If the sum of errors is different, then this means that the variable with the highest RSD was not rounded off in a good way.) If you successfully round off the first variable value, then continue to round off the optimising variable which now has the highest RSD. Continue doing this until you have fixed all but one of the optimising variables to their rounded-off values. The final sum of errors that you get after having fixed the variable values should not deviate significantly from the initial sum of errors you had before starting to round-off the variable values.

5.2 Updating the database file

To update the database file (with filename extension "TDB") open it in a text editor and enter or update each of the parameters that you have optimised.

For example, suppose you have the following command in your setup file:

```
ENTER_PARAMETER L(IONIC_LIQ,AL+3,ZR+4:O-2;0) 298.15  
+V17; 6000 N
```

If you then end up with a value for optimising variable V17 of -100000, then you should enter the following under the IONIC_LIQ phase in the database file:

```
PARAM L(IONIC_LIQ,AL+3,ZR+4:O-2;0) 298.15 -100000; 6000 N  
REF !
```

Note that you will often need to round-off the optimising variable value in some way. For more information about how to round-off the optimising variable values, see section 5.1.

When you have updated or entered the parameters that you optimised, save the database file.

5.3 Creating a database file

In some circumstances you may want to save all the information about your system into a new database file. To do this, go to the GIBBS module and use LIST_DATA <FILENAME> N. This will create new user database file that contains all the information about the system that is in the workspace.

The N-argument tells the LIST_DATA command to save the output into database file rather than some other type of file, such as a macro file for example. Note that the output database file that you get from LIST_DATA have to undergo some manual editing before it can be used as proper database file.

5.4 Update the setup file and the POP-file

It may be useful for you to be able to recreate your optimisation from scratch. It is therefore recommended that you do the following when you are done with the optimisation:

- Update your POP-file so that it contains the weights of your critical set of experiments. Use SET_WEIGHT to set the weights directly in the POP-file. You may also have to use SET_START_VALUE for some conditions in the POP-file. Otherwise, you may not be able to compute the equilibria with COMPUTE_ALL_EQUILIBRIA.
- Copy the final values of your optimising variables and set them as start values of the variables in the setup macro file (with SET_FIX_VARIABLE). Here, it is important that you round off the values of the optimising variables correctly. For more information about rounding-off these values, see section 5.1.

When you have done this, you can simply run the setup macro file to recreate the final result of the optimisation.

6 Optimising in alternate mode

When you have one or more multi-phase equilibria (that is, with the fixed phase-status) in your POP-file and the optimising values lack appropriate starting values (such as the default value of 0 for example), then Thermo-Calc may not be able to compute the equilibria for the experiments. In these cases, a “common tangent” cannot be found, that is, a state where the chemical potentials for all components are the same in all phases cannot be found. A common tangent may not exist or it be found at the wrong composition or temperature given the initial set of model parameters.

If you are not able to estimate reasonable starting values for the optimising variables, then this problem can be overcome by initially running your optimisation in “alternate mode”. The alternate mode in PARROT module should help you find reasonable starting values for the optimising variables. Note that you are only supposed to use alternate mode in the beginning of the optimisation. Once you have achieved a reasonable fit between calculated results and experimental data, turn off alternate mode and calculate all equilibria again in the normal mode.

In alternate mode, PARROT modifies multi-phase equilibria so that the chemical potentials for each stable phases is calculated separately. For example, consider the following experimental equilibrium:

```
CREATE_NEW_EQUILIBRIUM 1 1
CHANGE_STATUS PHASE LIQ FCC=FIX 1
SET_CONDITION T=1000 P=1E5
EXPERIMENT X(LIQ,B)=.2:.01 X(FCC,B)=.1
```

When PARROT optimises in alternate mode, the computation of this equilibrium will be done by way of the following two calculations:

```
CHANGE_STATUS PHASE LIQ=FIX 1
SET_CONDITION T=1000 P=1E5 X(LIQ,B)=.2
CHANGE_STATUS PHASE FCC=FIX 1
SET_CONDITION T=1000 P=1E5 X(FCC,B)=.1
```

The difference between the chemical potentials in every pair of phases is calculated and PARROT tries to adjust the parameters of the phases so that these differences are minimised. In other words, PARROT tries to make the chemical potentials equal in all phases. For PARROT to be able to do this, some additional information about the experimental multi-phase equilibria is required, at least for all invariant equilibria.

6.1 Turning alternate mode on and off

To turn on or off alternate mode, use the PARROT command
`SET_ALTERNATE_MODE <Y/N>`.

6.2 Optimising in alternate mode

When you are optimising in the alternate mode, you usually do not need to enter the ED_EXP module and manually change the weights of experiments that the computed results have not been fitted well with. Instead, after using OPTIMIZE_VARIABLES initially with no iterations, and listing the result with LIST_RESULT, simply run OPTIMIZE_VARIABLES again specifying, say, thirty iterations. Continue to optimise until your optimisation seems to be going in the right direction, then set the optimising variables' starting values to their current values with use RESCALE_VARIABLES before you turn alternate mode off with SET_ALTERNATE_MODE N.

6.3 Preparing the POP-file for alternate mode

Information in your POP-file that concerns equilibria with "ENTERED" or "DORMANT" phases is ignored when you optimise in alternate mode.

To use the alternate mode, ensure that there is enough information about each phase in a given equilibrium. In addition to the temperature and pressure of the system, you should specify the composition of all phases in a given equilibrium. This is done with SET_CONDITION, EXPERIMENT or SET_ALTERNATE_CONDITION.

If both compositions of a binary tie-line have been measured, then you could specify this in the POP-file in the following way:

```
CREATE_NEW_EQUILIBRIUM 1 1
CHANGE_STATUS PHASE FCC BCC=FIX 1
SET_CONDITION P=1E5 T=1000
EXPERIMENT X(BCC,B)=.2:.01 X(FCC,B)=.3:.01
```

This equilibrium could be calculated with alternate mode without any modification. The compositions specified with the EXPERIMENT command will be used as condition when the thermodynamic properties of each phase are calculated.

6.3.1 Examples of SET_ALTERNATE_CONDITION usage

The following sections demonstrate how SET_ALTERNATE_CONDITION is used to prepare different kinds of equilibria for optimisation in alternate mode.

Specifying the composition of one side of a binary tie-line

If only one side of the tie-line has been measured, then you must provide an estimate of the composition of the other phase. You can make such an estimate with SET_ALTERNATE_CONDITION. This command will have no effect unless alternate mode is turned on. The equilibrium described in the last previous paragraph could thus be specified in the following way:

```
CREATE_NEW_EQUILIBRIUM 1 1
CHANGE_STATUS PHASE FCC BCC=FIX 1
SET_CONDITION P=1E5 T=1000
EXPERIMENT X(BCC,B)=.2:.01
SET_ALTERNATE_CONDITION X(FCC,B)=.3
```

When the alternate mode is turned on, the composition given by the SET_ALTERNATE_CONDITION command will be used for the FCC phase. The properties of the BCC phase is calculated using the composition specified by EXPERIMENT as a condition.

Specifying the compositions of a three-phase equilibrium

If you have a three-phase equilibrium in your POP-file, then one of its compositions can be specified as experimentally determined, while two can be provided as alternate conditions:

```
CREATE_NEW_EQUILIBRIUM 1 1
CHANGE_STATUS PHASE FCC BCC LIQ=FIX 1
SET_CONDITION P=1E5
EXPERIMENT T=912:5 X(LIQ,B)=0.2:.02
SET_ALTERNATE_CONDITION X(FCC,B)=0.1 X(BCC,B)=.4
```

Specifying the composition of a stoichiometric phase

When you specify the composition of stoichiometric phase with SET_ALTERNATE_CONDITION, the value must be given with at least seven decimal digits, as in the following example:

```
CREATE_NEW_EQUILIBRIUM 1 1
CHANGE_STATUS PHASE LIQ A2B=FIX 1
SET_CONDITION P=1E5 X(LIQ,B)=0.2
EXPERIMENT T=992:5
SET_ALTERNATE_CONDITION X(A2B,B)=.6666667
```

More examples

The following table shows some additional examples of how to enter different kinds of experiments in the POP-file when you are using the alternate mode.

Example	ED_EXP commands
Two-phase equilibrium: The melting temperature of an Au-Cu alloy	<pre>CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE LIQUID FCC=FIX 1 SET_CONDITION X(FCC,CU)=0.14 P=1E5 EXPERIMENT T=970:2 SET_ALTERNATE_CONDITION X(LIQUID,CU)=0.16</pre>
Invariant equilibrium experiment: A three-phase equilibrium in a binary system	<pre>CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE FCC BCC LIQUID=FIX 1 SET_CONDITION P=1E5 EXPERIMENT T=912:5 SET_ALTERNATE_CONDITION X(FCC,B)=0.1 SET_ALTERNATE_CONDITION X(BCC,B)=0.4 SET_ALTERNATE_CONDITION X(LIQ,B)=0.2</pre>
Ternary system, with two compositions measured (both with uncertainty 0.02)	<pre>CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE FCC BCC=FIX 1 SET_CONDITION T=1273 P=1E5 SET_CONDITION X(FCC,B)=0.1:0.02 EXPERIMENT X(FCC,C)=0.12:.02 SET_ALTERNATE_CONDITION X(BCC,B)=0.17 SET_ALTERNATE_CONDITION X(BCC,C)=0.07</pre>

7 Troubleshooting

During an optimisation, you are likely to run into many different problems and challenges. It is impossible in a guide like this one to explicitly go through them and state how they should be handled. However, in this section, you are provided with a few general guidelines that will help you avoid some of the most common mistakes and problems.

7.1 Excluding and including the right equilibria

Use as few experiments as possible initially to get a reasonable overall fit. It is generally best to first focus on the invariant equilibria, as well as metastable states that can be estimated by excluding some phases. Crucially, any experiments for which the computed results are clearly faulty should be excluded from the final critical set, while all important invariant equilibria are calculated and included in this critical set.

If you cannot compute some invariant equilibria with intermediate phases, then it may be best to first exclude those phases from the optimisation and only optimise the liquid and the most important solution phases. After you have obtained reasonable results for these phases, you can enter the intermediate phases back in again. With the variables for the optimised liquid and solid phases set fixed, you can finally optimise the intermediate phases.

7.2 Conflicting data

It is important that you do not include conflicting sets of data at the same time during an optimisation. After all, if they are genuinely conflicting, then at least one of the sets must be incorrect. If you have conflicting sets of data, then only use one of these sets at a time together with the rest of the data. Often, the optimisation will make it clear which dataset coheres best with the other information that is available about the system.

Sometimes, conflicting data cannot be detected directly. For example, you may have activity data which is inconsistent with solubility data from the phase diagram. This would be indicated by large errors in the fit when both the activity data and the solubility data are included. To find these inconsistencies, let alone correct the errors, you may thus have to optimise with some datasets excluded.

7.3 Useful rules of thumb

The following rules of thumb are generally good to adhere to when you are planning and executing an optimisation:

- Do not use data from thermochemical tables of unknown origin.
- Use experimentally determined properties; avoid converted quantities.
- Carefully estimate the accuracy of the experiments.
- Correct systematic errors (by adjusting a temperature scale for example).
- Use “negative” information such as, for example, the information that a phase should not be stable in a composition or temperature region.

- With a hundred activity measurements in a system but only ten composition points from the phase diagram, it is often appropriate to decrease the weights on the activity experiments.
- If you initially exclude any intermediate phases to only optimise the liquid and the terminal or end-member phases for the pure components, then it is often useful to compute a metastable phase diagram with just these phases. In the metastable phase diagram, the metastable solubility lines should not have any strange kinks or turns. Such a diagram is often also useful to compute at later stages in the optimisation.
- When you have fitted the liquid phase and some solution phases reasonably well, then you can fix the optimising variables that characterise these phases. You can then go on and optimise the variables for the intermediate phases.
- Phases with miscibility gaps are always difficult to optimise. Try to keep control of the miscibility gaps by using real or estimated experimental information. (Note that it is not possible to calculate the top of a miscibility gap as a single equilibrium.)
- Phases with order/disorder transformation are often difficult to optimise. It may be difficult to find the right start values on the optimising variables. The TABULATION module can be useful sometimes. Properties for specified site fractions can be calculated in the TABULATION module. This is not possible in the POLY module since this module only calculates for equilibrium site fractions. The ordered state may also disappear during the optimisation of these phases. If this happens, you may be able to add an experiment that controls the state of order.
- Only use the alternate mode to find an initial set of model parameters that can then be made it possible to calculate the experimental equilibria in the normal mode.
- When you have achieved a satisfactory fit between calculated results and experimental data with an appropriate number of optimisation variables, perform a couple of final optimisation runs to make sure that your results are stable.
- Make sure that the values of the optimising variables keep within a reasonable range. If a variable starts to change several orders of magnitudes, then you must consider whether you have the right weights on the experiments and whether you are using too many optimising variables.

8 Appendix 1: Experiments and experimental data

This appendix contains examples of different kinds of equilibria in different kinds of systems. They are intended to help you figure out how to use various kinds of experimental data when creating your POP-file. The sections that follow contain examples how to enter various kinds of experimental data into a POP-file.

Phase diagram data

P-T-X phase diagram data for binary and ternary systems, or systems of even higher order can be used in the POP-file. This includes data of the following types:

- Data about the temperature- and composition-dependence of various properties (obtained from differential thermal analysis (DTA) measurements for example), including enthalpy, lattice parameter, dilatometric length, electronic conductivity and magnetic susceptibility;
- Data about reaction types (invariant/monovariant) and phase relations obtained from qualitative or quantitative metallographic studies;
- Data about phase amounts or compositions obtained from microprobe measurements (using X-ray spectroscopy or transmission electron microscopy (TEM));
- Data about the positions or directions of two-phase tie-lines and three-phase equilibria for ternary systems.

The following table shows a POP-file excerpts that demonstrates how some kinds of phase diagram data can be entered:

Example	ED_EXP commands
Solidus (melting temperature) of an Au-Cu alloy	<pre>CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE LIQUID FCC=FIX 1 SET_CONDITION X(FCC,CU)=0.14 P=1E5 EXPERIMENT T=970:2</pre>
Three-phase equilibrium in a binary A-B system	<pre>CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE FCC BCC LIQUID=FIX 1 SET_CONDITION P=1E5 EXPERIMENT T=912:5</pre>
Congruent transformation in a binary A-B system	<pre>CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE BCC LIQUID=FIX 1 SET_CONDITION P=1E5 X(BCC,B)-X(LIQ,B)=0 EXPERIMENT T=1213:10</pre>
Tie-line in a ternary A-B-C system	In the following, an uncertainty factor has been set not only for the experimental measurement, but also for the X(FCC,B) condition. This factor is 0.02. <pre>CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE FCC BCC=FIX 1 SET_CONDITION T=1273 P=1E5 X(FCC,B)=0.1:0.02 EXPERIMENT X(FCC,C)=0.12:.02</pre>

Data for individual compounds

You can use data about individual components in the POP-file. Such data includes activity, heat capacity, heat content, entropy, Gibbs energy of phase transformations, heat of transition and heat of melting.

Example	ED_EXP commands
Carbon activity data in the fcc phase	<pre> CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE FCC_A1=FIX 1 CHANGE_STATUS PHASE GRAPHITE=D SET_REFERENCE_STATE C GRAPHITE,,,, SET_CONDITION P=101325 T=1273 X(MN)=0.03 SET_CONDITION X(C)=0.03 EXPERIMENT ACR(C)=0.29:5%</pre>
Ln(activity coefficient for O)	<pre> CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE IONIC_LIQ=FIX 1 CHANGE_STATUS PHASE GAS=D SET_CONDITION X(O)=0.02 T=1523 P=101325 SET_REFERENCE_STATE O GAS,,,, ENTER_SYMBOL FUNCTION LNFO=LOG (ACR (O) /X (O)) ; EXPERIMENT LNFO=-1.5:0.01</pre>
Entropy at 298.15 K of Cu2O	<pre> CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE CU2O=FIX 1 CHANGE_STATUS PHASE CUO=FIX 0 SET_CONDITION P=101325 T=298.15 EXPERIMENT S=92.36:1</pre>
Gibbs energy of formation of NiAl2O4	<pre> CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE SPINEL=ENT 1 CHANGE_STATUS PHASE FCC O2GAS=DORM SET_REFERENCE_STATE NI FCC,,,, SET_REFERENCE_STATE AL FCC,,,, SET_REFERENCE_STATE O O2GAS,,,, SET_CONDITION P=101325 T=1000 N(NI)=1 N(AL)=2 SET_CONDITION N(O)=4 EXPERIMENT GM=-298911:5%</pre>
Heat capacity of MgFe2O4	<pre> CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE SPINEL=ENT 1 SET_CONDITION P=101325 N(Fe)=2 N(Mg)=1 N(O)=4 SET_CONDITION T=800 ENTER_SYMBOL FUNCTION CP=H.T; EXPERIMENT CP=207:5%</pre>
Heat content (H-H298) for MnFe2O4	<pre> CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE SPINEL=ENT 1 SET_CONDITION P=101325 T=298.15 N(MN)=1 N(Fe)=2 SET_CONDITION N(O)=4 ENTER_SYMBOL VAR H298=H; ENTER_SYMBOL FUN HMNFE2O4=H-H298; CREATE_NEW_EQUILIBRIUM 2 1 CHANGE_STATUS PHASE SPINEL=ENT 1 SET_CONDITION P=101325 N(MN)=1 N(Fe)=2 N(O)=4 SET_CONDITION T=400 EXPERIMENT HMNFE2O4=16610:5%</pre>

Single phase mixing enthalpies or partial enthalpies

You can use calorimetric data for mixing in the POP-file. This data could be the result of, for example, drop calorimetry or scanning calorimetry. This includes data about, for example, enthalpy of mixing of liquids, series of mixing-enthalpy and about partial enthalpy. Information about enthalpies of mixing can also be derived from theoretical principles using special quasi-random structures (SQSs).

Example	ED_EXP commands
Enthalpy of mixing in the liquid state of an Au-Cu alloy	<pre>CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE LIQUID=FIX 1 SET_CONDITION T=1379 P=1E5 SET_CONDITION X(LIQUID,AU)=0.0563 SET_REFERENCE_STATE AU LIQ * 1E5 SET_REFERENCE_STATE CU LIQ * 1E5 EXPERIMENT HMR=-1520:200</pre>

Enthalpies of formation and enthalpies of reactions

You can use calorimetric data for phase transformations in the POP-file. This data could be the result of, for example, direct-reaction calorimetry, solution calorimetry or combustion calorimetry. Information about enthalpies of formation can also be derived from theoretical principles.

Example	ED_EXP commands
Enthalpy of melting of the eutectic in the Al ₂ O ₃ -ZrO ₂ binary (J/g)	<pre>CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE CORUNDUM ZRO2_TETR=FIX 0 CHANGE_STATUS PHASE IONIC_LIQ=ENT 1 SET_CONDITION P=101325 N=1 LNAC(O)=-80 SET_REFERENCE_STATE ZRO2 TETR,,,,, SET_REFERENCE_STATE AL2O3 CORUND,,,,, EXPERIMENT HWR=1080:90</pre>
Enthalpy of formation of o-CaZrO ₃ from the component oxides	<pre>CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE ZRO2_MONO HALITE CAZRO3_O=ENT 1 SET_CONDITION T=298.15 P=101325 N(CAO)=.5 SET_CONDITION N(ZRO2)=.5 LNAC(O)=-20 SET_REFERENCE_STATE CAO HALITE,,,, SET_REFERENCE_STATE ZRO2 ZRO2_MONO,,,, EXPERIMENT HMR(CAZRO3_O)=-15960:5%</pre>

Chemical potentials via EMF measurements

You can use chemical potentials and activities/activity coefficients from electromagnetic field (EMF) measurements in the POP-file.

Example	ED_EXP commands
Emf data on Cu(s)/Cu ₂ O(s)	<pre>CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE FCC CU2O=FIX 1 CHANGE_STATUS PHASE O2GAS=DORM SET_CONDITION P=101325 T=1000 SET_REFERENCE_STATE O O2GAS,, 100000 EXPERIMENT MUR(O)=-95387:1000</pre>

Driving force for metastable phases

You can use chemical driving forces for metastable phases in the POP-file.

Example	ED_EXP commands
To prevent BCC_A2 to be stable	<pre>CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE FCC_A1 M7C3 M23C6=FIX 1 CHANGE_STATUS PHASE BCC_A2=D SET_CONDITION P=101325 T=1473 EXPERIMENT DGM(BCC)<-0.1:0.1</pre>
To promote CEMENTITE to be stable	<pre>CHANGE_STATUS PHASE BCC M7C3 GRAPHITE=FIX 1 CHANGE_STATUS PHASE CEMENTITE=DORMANT SET_CONDITION P=P0 T=973 EXPERIMENT DGM(CEM)>0.1:0.01</pre>

Pressure-Temperature-Volume EOS data

You can use Pressure-Temperature-Volume Equation of State (EOS) data in the POP-file. Such data includes, for example, molar volume, density, thermal expansion and bulk-modulus/compressibility.

Example	ED_EXP commands
Volume of liquid Pt	<pre>CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE LIQUID=ENT 1 SET_CONDITION P=101325 N=1 T=1873 EXPERIMENT VM=1.00758E-5:5%</pre>
Thermal expansion	<pre>CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE FCC_A1=ENT 1 SET_CONDITION P=101325 N=1 T=400 ENTER_SYMBOL FUNCTION ALPHA=VM.T/VM/3; EXPERIMENT ALPHA=2.35E-5:5%</pre>

Crystal structure, lattice parameters and site-occupancy

You use information about defects, ordering or site occupancy, as well as information about crystal structures in the form of lattice parameter values. This information may come from measurements made X-ray, neutron diffraction, Mössbauer spectroscopy or perturbed angular-correlation measurements.

Example	ED_EXP commands
Lattice parameter for fcc	<pre>CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE FCC_A1=ENT 1 SET_CONDITION P=101325 N=1 T=298.15 SET_CONDITION X(CR)=0.05 ENTER_SYMBOL FUNCTION LPFCC=((4*VM/6.02214179E23)**(1/3))*1E10; EXPERIMENT LPFCC=4.02:5%</pre>
Lattice parameter for bcc	<pre>CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE BCC_A2=ENT 1 SET_CONDITION P=101325 N=1 T=298.15 SET_CONDITION X(AL)=0.08 ENTER_SYMBOL FUNCTION LPBCC=((2*VM/6.02214179E23)**(1/3))*1E10; EXPERIMENT LPBCC=2.90:5%</pre>
Site-occupancy, degree of inversion for spinel	<pre>CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE SPINEL=ENT 1 SET_CONDITION P=101325 N(MG)=1 N(AL)=2 N(O)=4 SET_CONDITION T=1073 EXPERIMENT Y(SPINEL,AL+3#1)=0.31:5%</pre>

Magnetism and other atomistic properties

You can use data about atomistic properties such as magnetism and Curie temperatures in the POP-file.

Example	ED_EXP commands
Curie temperature of $\text{Mn}_x\text{Fe}_{3-x}\text{O}_4$, $x=1.26$	<pre>CREATE_NEW_EQUILIBRIUM 1 1 CHANGE_STATUS PHASE SPINEL=ENT 1 SET_CONDITION T=520 P=101325 N(O)=4 SET_CONDITION N(Fe)+N(MN)=3 N(MN)=1.26 EXPERIMENT TC(SPINEL)=520:5</pre>

Data about systems of different orders

You might want to use data about systems of different orders in the same optimisation, such as data about a binary equilibrium with components A and B in a ternary system with components A, B and C, together with data about equilibria in the binary system with components A and B. To do this, you must first suspend all components in the ternary system (A, B and C), and then set the two components you want binary information on (A and B) to status “ENTERED”.

Suspending all components is done by giving the initialization code 0 to the `CREATE_NEW_EQUILIBRIUM` command. The components about which you want to use binary information must then be entered again using the `CHANGE_STATUS` command.

Example	ED_EXP commands
A binary (A-B) three-phase equilibrium (FCC-BCC-LIQ) in a ternary system (A-B-C)	<pre>CREATE_NEW_EQUILIBRIUM 1 0 CHANGE_STATUS COMPONENT A B = ENTERED CHANGE_STATUS PHASE FCC BCC LIQ=FIX 1 SET_CONDITION P=1E5 EXPERIMENT T=1177:10</pre>

9 Appendix 2: Thermodynamic models

The most common models used in assessments are described briefly in the subsections that follow. Examples of how to write the phase definitions in the setup-file are also given for each of these models. In addition to the models described here, a large number of thermodynamic models for various phases in different states have been implemented.

9.1 Gas

The gaseous mixture phase is usually treated as a substitutional phase without sublattice. In the gaseous mixture phase, there are usually molecules formed, and the number of constituents of the gas phase is often much larger than the number of elements. These complex constituents have to be defined as species. The ideal gas model may handle a gas phase under low pressures (and low temperatures). This implies that the P-V-T relations and thermodynamic properties of pure gaseous species are calculated as for ideal gas, and that there is interaction between gaseous species in the mixture. In a defined system, there may only exist one gaseous mixture phase. Type code "G" is used for the gas phase:

```
ENTER_PHASE GAS G, 1 H,H1O1,H1O2,H2,H2O1,H2O2,O,O2,O3; N
N
```

9.2 Compound Energy Formalism

If the atoms are sufficiently different in size, electronegativity or charge, they may prefer different types of sites in the lattice of crystalline solids. In some cases, a solute atom may even occupy interstitial sites between the normal lattice sites. All such phenomena are treated in Thermo-Calc by the sublattice concept. The Compound Energy Formalism (CEF) has proved to be the most general formalism for many different types of solution phases. It is therefore by default applied to various solid solution phases in the Thermo-Calc software package. CEF can also take ionic constraints (charged cation/anion species) into account. For a comprehensive description on the CEF, please see Hillert (2001), Frisk and Selleby (2001), Sundman and Ågren (1981).

Since the CEF is used by default, no special notation is needed in the phase definition, except for ionic phases, where type code "I" is given. Examples of phase definitions in a setup-file using the CEF:

```
ENTER_PHASE SPINEL I, 4 1 2 2 4
FE+2,FE+3; FE+2,FE+3,VA; FE+2,VA; O-2; N N
ENTER_PHASE SIGMA, 3 10 4 16
AL,CO,CR,FE,NB; AL,CO,CR,FE,NB; AL,CO,CR,FE,NB; N N
ENTER_PHASE FCC_A1, 2 1 1
CR,FE,MN; C,N,VA; N N
ENTER_PHASE BCC_A2, 2 1 3
CR,FE,MN,VA; C,N,VA; N N
```

9.3 Substitutional Liquid

The liquid phase is normally (but not always) modelled as a substitutional phase with no sublattice, which is occupied by two or more constituents. The constituents are normally the elements, but molecules or associates can also be used. Type code "L" is used for the liquid phase:

```
ENTER_PHASE LIQUID L, 1
      AL,C,CR,FE,MO,NI,SI ;  N N
ENTER_PHASE LIQUID L, 1
      C,FE,FEO,FEO3/2,NI,NIO ;  N N
```

9.4 Ionic Two-sublattice Liquid Model

Within the framework of the CEF, the ionic two-sublattice liquid (I2SL) model was developed to be used when there is a tendency for ionization in the liquid, which happens in liquid oxides and sulphides for example. The same model can be used both for metallic and oxide melts. At low levels of oxygen, the model becomes equivalent to a substitutional solution model between metallic atoms. Two sublattices are assumed, one containing charged cations and one containing charged anions, neutrals and vacancies. The complication for an ionic liquid is that the numbers of sites on the cation sublattice and anion sublattice vary with the composition to preserve electroneutrality. For a comprehensive description on the I2SL model, see Hillert et al. (1985), Sundman (1991). In these references, the possible excess parameters are also discussed. Type code "Y" is used for the I2SL model. Examples of phase definitions in a setup-file:

```
ENTER_PHASE LIQUID Y,
      AL+3,CR+2,FE+2,NI+2; VA; N N
ENTER_PHASE IONIC_LIQ Y,
      CA+2,MG+2; O-2; N N
ENTER_PHASE IONIC_LIQ Y,
      AL+3,CA+2,FE+2,MG+2,SI+4;
      ALO2-1,O-2,SIO4-4,VA,FEO3/2,SIO2; N N
```

9.5 Models for ordered phases

Some solution phases have chemical order-disorder transformations: for example, the disordered FCC_A1 phase can transform to the ordered L1₂ or L1₀ structure where the atoms of different kinds occupy different sublattices. In some cases, it may be advantageous and easy to describe such ordered structures as completely different phases; but in some other cases, like the B2-ordered BCC structure in Al-Fe, the disordered and ordered structures must be described as one single phase because the transformation is of second order along a line in temperature and composition.

The ordering can easily be described by the CEF. For instance, an ordered FCC phase with only $L1_2$ or $L1_0$ ordering can be efficiently handled by the so-called Two Substitutional-Sublattice Ordering Model (2SL model), while an ordered FCC with both $L1_2$ and $L1_0$ ordering must be described with four sublattices using the so-called Four Substitutional-Sublattice Ordering Model (4SL model). Please note that in both these models, all the substitutional constituents must enter into these sublattices. In addition, you may also have one sublattice for interstitials.

Thermo-Calc supports a feature of splitting the parameters of a chemically ordered phase onto two different phase descriptions. This is very convenient. In this way, you can have one phase for all parameters describing the disordered state (the configuration independent state) and in the other phase only those parameters needed to describe the remaining part (the configuration dependent part), that is, the Gibbs energy contribution due to the ordering transformation. This has been implemented in such a way that the contribution from the configuration dependent part is zero when the solution phase is disordered. The optimization of the disordered phase is independent of the order-disorder transformation; the ordering energy is an add-on to the disordered part. This is particularly useful in higher order systems, as an ordered phase can dissolve several elements that have no particular contribution to the actual ordering. The phases Gibbs energy add together by giving this command in the setup-file: `AMEND_PHASE_DESCRIPTION <ORDERED PHASE> DISORDERED_PART <DISORDERED PHASE>`

Since PARROT does not automatically create new composition sets, it is necessary to create composition sets for both the ordered and disordered phases in the setup-file: `AMEND_PHASE_DESCRIPTION <PHASE> COMPOSITION_SETS <NEW HIGHEST SET NUMBER>,,,`

Example of phase definitions for the corresponding disordered and ordered parts in a setup-file:

```
ENTER_PHASE FCC_A1, 2 1 1
      AL,CR,FE,NI; VA; N N
ENTER_PHASE FCC_L12, 3 .75 .25 1
      AL,CR,FE,NI ; AL,CR,FE,NI; VA; N N
AMEND_PHASE_DESCRIPTION FCC_L12 DIS_PART FCC_A1
```

9.6 4SL model for FCC, HCP and BCC

Ordered FCC, HCP and BCC solution phases handled by the 4SL model requires four sublattices for substitutional ordering and can additionally have an interstitial sublattice. A 4SL model is especially useful for modelling different kinds of ordered phases that are based on the same disordered phase, such as $L1_0$ and $L1_2$ based on FCC_A1, and B2, D0₃, L2₁ and B32 based on BCC_A2 etc.

For ordered FCC or HCP phases, these four substitutional sublattices represent four corners of the regular tetrahedron on these lattices, all of which are the nearest neighbours. These corners are equivalent lattice points, thus all G parameters for each end-member with the same elements, but distributed on different sites, must be identical. It should be emphasized that the end-member energy here represents the ordering energy rather than the formation energy of the compound.

The constraints on the parameters in the 4SL model can be derived based on the symmetry of the lattice:

$$\begin{aligned} G_{A:B:B:B} &= G_{B:A:B:B} = G_{B:B:A:B} = G_{B:B:B:A} \\ G_{A:A:B:B} &= G_{A:B:A:B} = G_{A:B:B:A} = G_{B:A:A:B} = G_{B:A:B:A} = G_{B:B:A:A} \\ G_{B:A:A:A} &= G_{A:B:A:A} = G_{A:A:B:A} = G_{A:A:A:B} \\ L_{A,B:*,*} &= L_{*,A,B:*,*} = L_{*,*,A,B:*,*} = L_{*,*,*,A,B} \end{aligned}$$

The asterisk “*” means that the interaction parameter is independent of the occupation of that sublattice. For the disordered phase to be completely disordered, i.e. that all site fractions are equal on all four sublattices, all constraints must be correct.

For ordered BCC phases, the situation is a bit more complicated, as the four sublattice ordering phase represents an irregular tetrahedron with two pairs of sites that are next nearest neighbours. Thus, for an A-B binary solution phase, with A located on two sublattice sites and B on the other two, the end-member described by $G_{A:A:B:B}$ has four nearest neighbour bonds between A and B atoms, whereas the end-member described by $G_{A:B:A:B}$ has two nearest neighbour bonds between A and B atoms and two next nearest neighbour bonds. Many parameters thus have a relation:

$$\begin{aligned} G_{A:B:B:B} &= G_{B:A:B:B} = G_{B:B:A:B} = G_{B:B:B:A} \\ G_{B:A:A:A} &= G_{A:B:A:A} = G_{A:A:B:A} = G_{A:A:A:B} \\ G_{A:A:B:B} &= G_{B:B:A:A} \\ G_{A:B:A:B} &= G_{A:B:B:A} = G_{B:A:A:B} = G_{B:A:B:A} \\ L_{A,B:*,*} &= L_{*,A,B:*,*} = L_{*,*,A,B:*,*} = L_{*,*,*,A,B} \end{aligned}$$

The $G_{A:B:B:B}$ term represents D0₃, $G_{A:A:B:B}$ B2 and the $G_{A:B:A:B}$ B32 ordering. There are also two kinds of reciprocal interaction parameters:

$$\begin{aligned} L_{A,B:A,B:*,*} &0 \dots 9 \\ L_{A,B:*,A,B:*,*} &0 \dots 9 \end{aligned}$$

Kusoffsky et al. (2001) have shown the influence of different parameters available in the 4SL model applied to fcc ordering. The possibility to use the 4SL model to BCC ordering has been studied by Sundman et al. (2009) in the Al-Fe system.

9.7 2SL model for FCC, HCP and BCC

The reason for using a 2SL model instead of a 4SL model is that the calculations are faster, but on the other hand it can only model one kind of ordered phase(s). For instance, it is not possible to model FCC with both L1₀ and L2₁, or BCC with D0₃ and B2, etc.

In order to get the fully disordered phase to form, constraints are needed as for the 4SL model. For the symmetric phases, e.g. B2 and L1₀, the relation between parameters are:

$$\begin{aligned} G_{A:B} &= G_{B:A} \\ L_{A,B:*,*} &= L_{*,A,B} \end{aligned}$$

For a 2SL asymmetric model including, but not limited to, $L1_2$, more constraints are needed. This model has many complicated relations between the parameters. The relation between the parameters for the 2SL $L1_2$ model can be derived from a 4SL model. The conversion from 4SL model parameters to 2SL parameters for $L1_2$ can be found in Dupin et al. (2001) and for $L1_0$ in Yuan et al. (2012). Higher-order systems with an $L1_2$ phase modelled with two sublattices require a lot of ternary and some quaternary interaction parameters in order to make the disordered state stable. These parameters have been derived by eg Dupin (1995), also found in Kusoffsky et al. (2001).

Many publications exist on how to model order/disorder transformations using a two-sublattice model, see e.g. Dupin, Ansara (1999), De Keyser et al. (2009).

9.8 Magnetic Ordering

The magnetic contribution to the Gibbs energy is given by a model proposed by Inden (1975) and adapted by Hillert and Jarl (1978), and is described by:

$$G = RT \ln(\beta + 1) f(\tau)$$

where $\tau = T/T_c$. The values of the Curie temperature, T_c , and Bohr magneton number, β , at a certain composition are then used in an optimisation procedure to calculate the contribution to Gibbs energy due to magnetic ordering. Curie temperature and Bohr magneton number are given in the command ENTER_PARAMETER for phases that undergo a magnetic transition:

```
AMEND_PHASE_DESCRIPTION BCC_A2 MAGNETIC -1.0 0.4
ENTER_PARAMETER TC(BCC_A2,FE:VA;0) 298.15 +V1; 6000 N
ENTER_PARAMETER BMAGN(BCC_A2,FE:VA;0) 298.15 +V2; 6000 N
```

The arguments -1 and 0.4 specify the values of parameters in the magnetic model: -1 is the antiferromagnetic factor and 0.4 is the structure factor. Usually the antiferromagnetic factor = -1 for bcc and = -3 for fcc (and other non-bcc phases). The structure factor corresponds to the short range order fraction of the enthalpy. This factor is 0.4 for bcc and 0.28 for non-bcc phases.

9.9 Molar volumes and high pressure

Including volumes in a thermodynamic database enable calculations of volume changes, volume fraction, lattice mismatch in cubic structures and phase diagrams at increased pressure. Pressure independent volumes can be used up to about 1 GPa. At pressures above 1 GPa the pressure dependence of the volumes must be included by using a proper equation of state (EOS). An EOS has been developed by Lu et al. (2005), based on the EOS from Jacobs and Oonk (2000). Four parameters are used in the implementation of the model in Thermo-Calc:

V0: Volume at 1 bar and at reference temperature T_0
VA: Integrated thermal expansivity
VK: Isothermal compressibility at 1 bar
VC: Parameter to fit high pressure data

All parameters can be composition dependent in the same way as all other model parameters in Thermo-Calc. Experimental information about the temperature, pressure and composition dependence of the volume can be assessed using these model parameters. To evaluate V_0 , V_A and V_K experimental data at 1 bar only are needed. Examples on modelling molar volume in a setup-file:

```
ENTER_PARAMETER V0(BCC_A2,FE:VA;0) 298.15 +V1; 6000 N
ENTER_PARAMETER VA(BCC_A2,FE:VA;0) 298.15 +V2*T+V3*T**2;
6000 N
ENTER_PARAMETER VC(BCC_A2,FE:VA;0) 298.15 +V4+V5*T; 6000
N
ENTER_PARAMETER VK(BCC_A2,FE:VA;0) 298.15 +V6+V7*T; 6000
N
ENTER_PARAMETER V0(FCC_A1,CR,FE:VA;0) 298.15 +V8; 6000 N
ENTER_PARAMETER VA(FCC_A1,CR,FE:VA;0) 298.15 +V9*(T-
298.15); 6000 N
```

9.10 Excess Models

Unless specifically amended, the excess Gibbs energy terms are modelled by the Redlich-Kister-Muggianu Model. Other excess models for the Gibbs energy are also implemented in Thermo-Calc, such as e.g. Kohler and Toop-Kohler extrapolation models. More information on different excess models can be found in example 51 in the *Thermo-Calc Console Mode Examples Collection*.

Thermo-Calc is not restricted to binary interaction parameters. Ternary, quaternary and higher-order interaction parameters can also be used if there is enough information to evaluate these from experimental data. The ternary interaction parameters are identified by the degrees of the so-called “ternary L parameter” (i.e. 0, 1 and 2) in the command ENTER_PARAMETER. If the parameter is composition-independent, you only need a single interaction parameter with a degree of zero, i.e. only the 0L term. If there is a ternary composition-dependent interaction parameter, you must enter all three parameters (0,1 and 2). If such a ternary interaction parameter should be zero, then assign it a function which is zero. If the three interaction parameters have the same value, then that is identical to having a composition-independent ternary parameter.

```
ENTER_PARAMETER L(LIQUID,A,B,C;0) 298.15 +1000; 6000 N
ENTER_PARAMETER L(LIQUID,A,B,C;1) 298.15 +1000; 6000 N
ENTER_PARAMETER L(LIQUID,A,B,C;2) 298.15 +1000; 6000 N
```

is identical to

```
ENTER_PARAMETER L(LIQUID,A,B,C;0) 298.15 +1000; 6000 N
```

This also means that

```
ENTER_PARAMETER L(LIQUID,A,B,C;0) 298.15 +1000; 6000 N
ENTER_PARAMETER L(LIQUID,A,B,C;1) 298.15 +ZERO#; 6000 N
ENTER_PARAMETER L(LIQUID,A,B,C;2) 298.15 +ZERO#; 6000 N
```

is NOT identical to

```
ENTER_PARAMETER L(LIQUID,A,B,C;0) 298.15 +1000; 6000 N
```