Pandat™ 2024

User's Guide

![PANDAT™ SOFTWARE logo]

CompuTherm LLC

Contents

1 Pandat Basics

Pandat™ software is an integrated computational tool developed on the basis of the CALPHAD (CALculation of PHAse Diagram) approach for multi-component phase diagram calculation and materials property simulation. It has a robust thermodynamic calculation engine, a friendly graphical user interface, and a flexible post-calculation table editing function which allows user to plot variety types of diagrams. The software is designed to create a working environment that allows variety of calculation modules be integrated in the same workspace. It currently includes five modules: **PanPhaseDiagram** (phase diagram and thermodynamic property calculation), **PanEvolution/PanPrecipitation** (microstructure and precipitation simulation), **PanOptimizer** (property optimization), **PanDiffusion** (diffusion simulation), **PanSolidification** (solidification simulation) and **PanPhaseField** (phase field simulation). Other modules can be easily integrated into the workspace for extended applications. The architecture of Pandat™ software is schematically shown in the Figure 1.1.

In addition to the functionalities provided by Pandat™ as a stand-alone program being demonstrated in this manual, its calculation/optimization engine (**PanPhaseDiagram**, **PanOptimizer**, **PanEvolution/PanPrecipitation**, **PanDiffusion**, and **PanSolidification**) are built as shared libraries and enable their integration with broader applications in the field of Materials Science and Engineering. More detailed information about Pandat™ software can be found in the published papers [2002Che, 2003Che, 2009Cao].
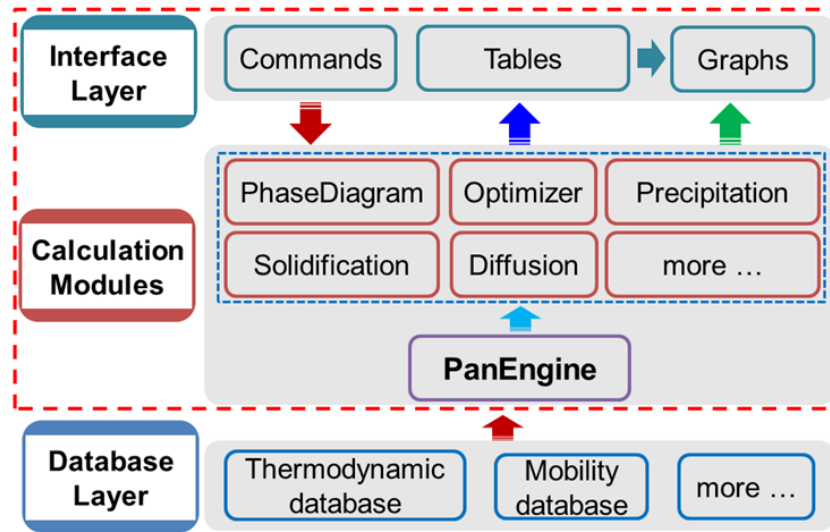
Figure 1.1: The architecture of Pandat™ software

1.1 Pandat GUI

Figure 1.2 shows the six components of the **PanGUI** (user interface of the Pandat™ workspace):Menus, Toolbar, Statusbar , Explorer Window , Property window and Main Display Window.
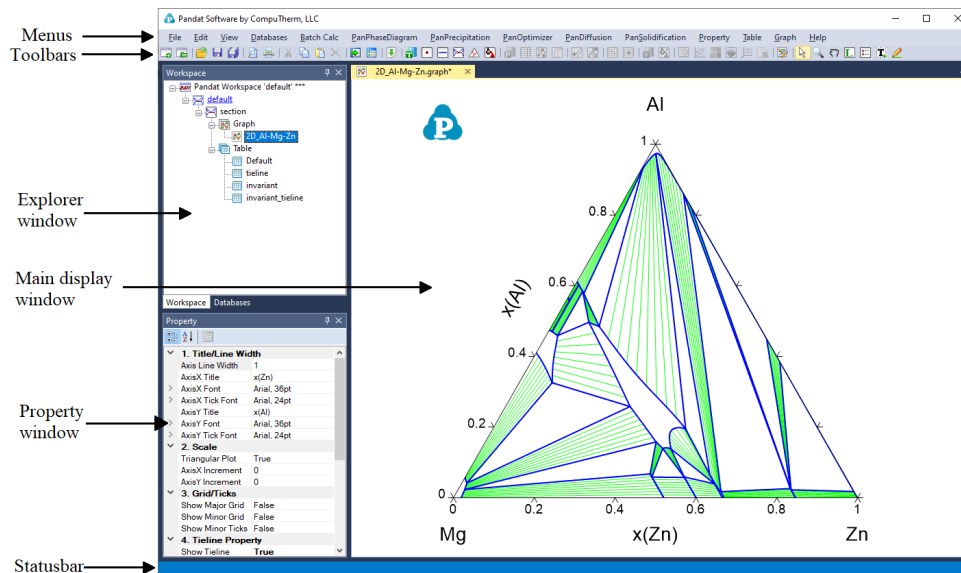
Figure 1.2: The components of the PanGUI

1.1.1 Menus

Pandat™ menus provide commands for performing operations on the active window and other general operations such as opening and saving a workspace file (*.pndx) or turning on the display of a toolbar. The menus are context sensitive; that is, they change as the context in Pandat™ main display window changes. This ensures that only Context-relevant menus and menu commands are displayed.

1.1.2 Toolbar

Pandat™ provides toolbar buttons for the frequently used menu commands. Like the menu commands, toolbar buttons are also context-sensitive. When a toolbar button is pointed at with the mouse, a "tool-tip" displays the button name.

1.1.3 Statusbar

The Pandat™ Status bar is located at the bottom of the workspace. If the cursor is on a graph, the coordinates are displayed in the status bar.

1.1.4 Explorer Window

The Explorer window lists the contents in the current workspace. The contents are displayed in a tree structure. By double clicking each node, the text, graph or table will be displayed in the Main Display Window of Pandat™. A node of a calculation result could be deleted by right click on the node and select "delete".

1.1.5 Property window

The property window is located below the explorer window. In Pandat™, object selection is carried out by single click. Based on the selected object, the properties of the plotted lines, text, graphs and tables will be shown and can be edited in this property window.

1.1.6 Main Display Window

The Main display window displays the text, graph or table which corresponds to each node in the Explorer window.

More than one graph or table can be opened and their titles are displayed on the top bar of the Main display window. User can display several plots together in the Main display window by moving them to different positions. The following Figure 1.3 and Figure 1.4 show how to move these windows. Select the window on the top bar, hold the left button and move the mouse toward the Main display window. A dialog showing relative positions will be displayed. Move the mouse to the selected position and release the left button of the mouse. Figure 1.3 shows moving the selected window to the right side and Figure 1.4 shows moving the selected window (the Property Window) to the center position.
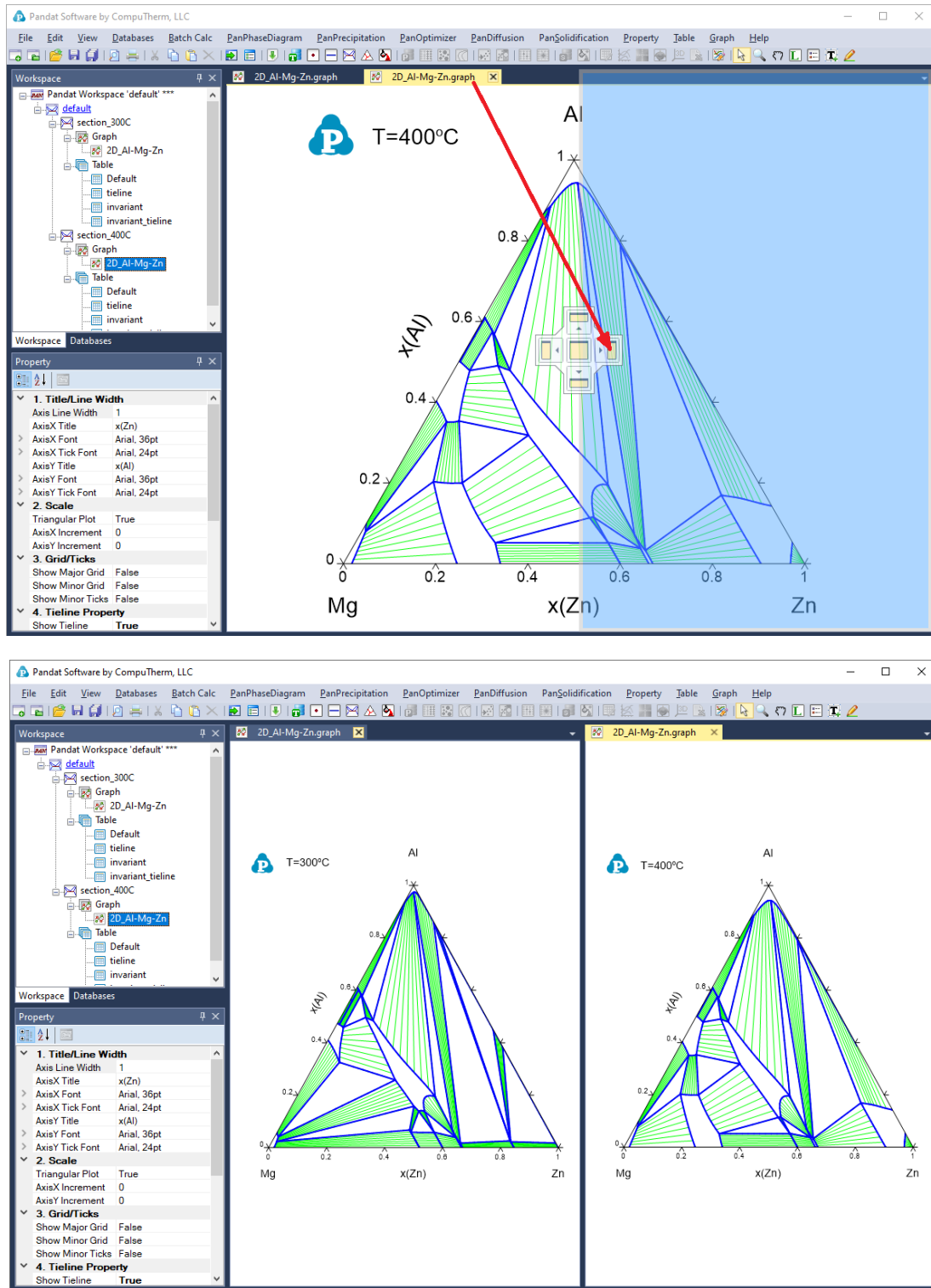
Figure 1.3: Rearrange the display window of Pandat™, before and after
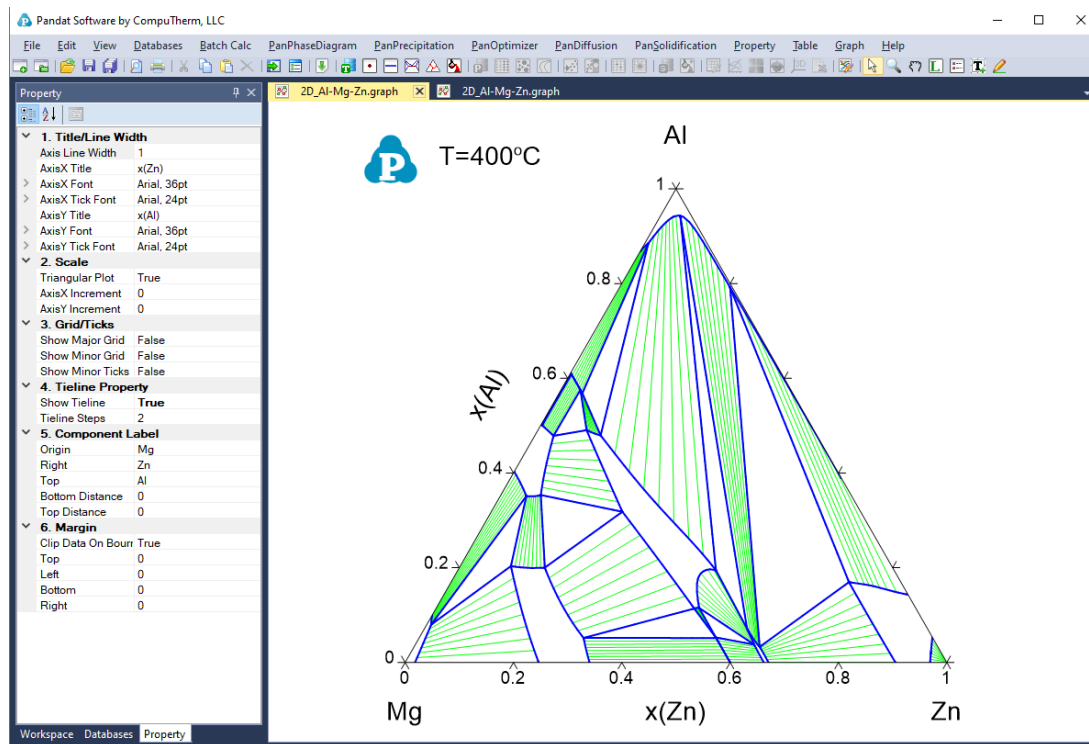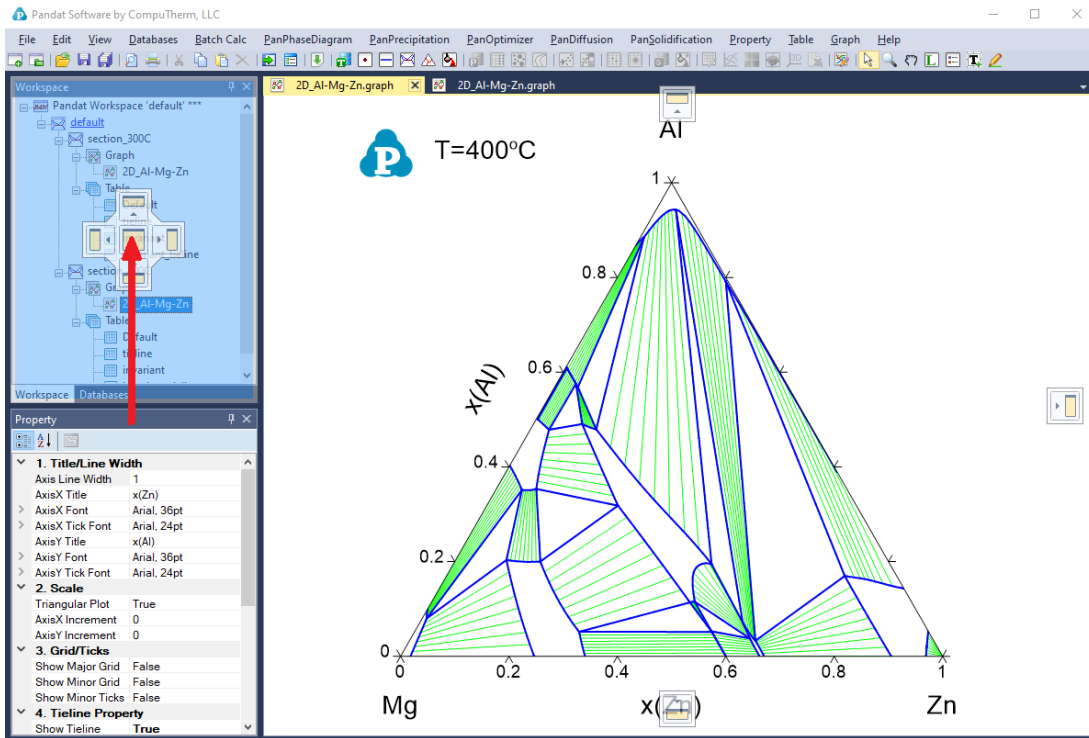
Figure 1.4: Rearrange the explorer and property windows of Pandat™, before and after

1.2 Console Mode

In addition to the most common `GUI` (graphical user interface) mode, Pandat™ can also run in the console mode without opening `GUI`. There are two ways to activate the **console mode**:

• double click a batch file (extension with `pbfx`) in a folder, or

• run through a windows `bat` file. The content of an example `bat` file is shown in Figure 1.5.
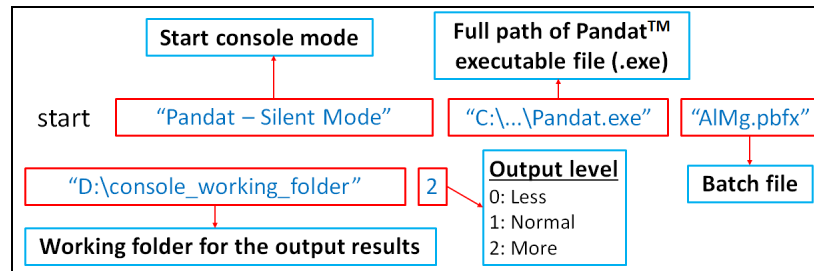
Figure 1.5: Example of the (`.bat`) file for console mode

The command is to start Pandat™ (with full path and titled as `Pandat - Silent Mode`) and run the batch file `AlMg.pbfx` (in the current folder if not in full path). The last two arguments are optional. If a working folder is given (`D:\2024 \console_working_ folder`), a Pandat™ workspace is created in the folder and all simulation results are saved under the workspace. The simulation progress is logged in the file `pandat.log`. The last argument is to control the output level with "1" the default value. The value of "2" is for more outputs and "0" for less. Please refer to Example 1.22 in the Pandat Example Book for more detail information.

2 Get started

Pandat™ starts with the following start page as shown in Figure 2.1. The user can open the start page at any time by clicking the icon ![icon] on the toolbar. On the start page, there are shortcuts which allow user to create a new workspace and open an existing workspace, link to CompuTherm's webpage (www.computherm.com) for recent updates and for comments and discussions from Pandat™ user. It also lists the most recent workspaces and files the user has created, so that user can reopen them easily.
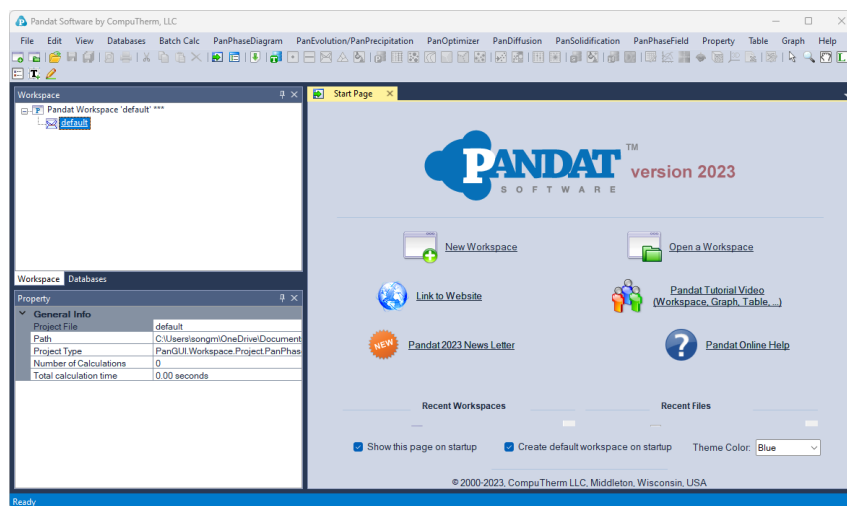
Figure 2.1: The start page of Pandat™ software

2.1 Pandat™ Learning resources

From the help menu, the user can have access to many resources on how to use Pandat™ software, Including Pandat™ online help and Pandat™ Tutorial Video. Click the "Open Pandat Example Book path" will open the Pandat example book folder, in which collected all the files required to run all examples in the example book.

```
"..\Program Files\CompuTherm LLC\Pandat 2024\Pandat 2024 Example
book"
```

The files mentioned in this manual are all saved in the example folder:

```
"..\Program   Files\CompuTherm   LLC\Pandat   2024\Pandat   2024
Examples"
```

In these two folders, the user can access templates for various file types, including tdb, pop, .kdb, sdb, pbfx etc.
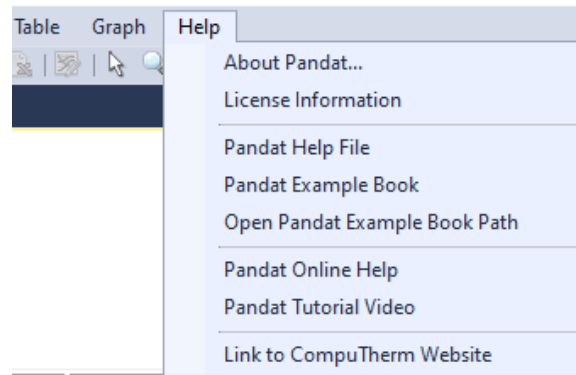
Figure 2.2: The Help menu of Pandat™ software

2.2 Workspace

The workspace provides a space for user to perform Pandat™ calculations and organize the calculated results. It must be created before any Pandat™ calculation is carried out. User can create a new workspace by clicking on the "`New Workspace`" link on the start page, or the icon ![icon] on the toolbar, or go through the menus (`File → Create a New Workspace`). A new window will pop out as shown in Figure 2.3. User can define the name of the workspace and select a working directory to put the workspace. For different types of simulation, user can then select a proper module from the six icons: Phase diagram calculation (![icon]), Optimization (![icon]), Precipitation simulation (![icon]), Phase Field simulation (![icon]), Diffusion simulation (![icon]) or Solidification simulation (![icon]). User can also give a "`Project Name`" for the calculations to be performed. User may choose to create a default workspace with default project name simply by clicking the "`Create`" button or double click the selected module.

After using Pandat™, user will be given an opportunity to save the workspace that the user has created. The user can open a saved workspace next time by clicking on the "`Open a Workspace`" link on the start page, or the icon on the toolbar, or going through the menus (`File → Open → Workspace`). For some most recent workspaces and files, the shortcuts listed on the start page allow the user to open them directly.
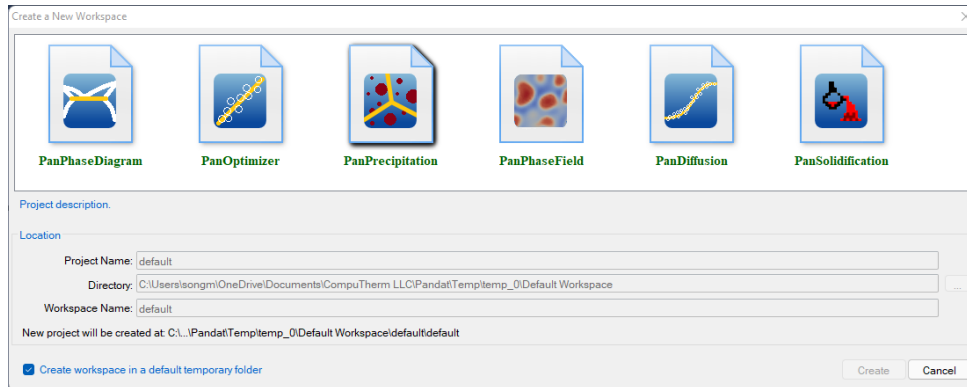
Figure 2.3: Create a new workspace dialog

In Pandat™, only one workspace is allowed. When creating a new workspace, the user will be asked if the current workspace needs to be saved. Think twice before clicking the "`Create`" button. The old workspace will be lost if it is not saved when a new workspace is created.

2.3 Project

In Pandat™, a workspace may contain several projects of different types. For example, a user creates a project for PanPhaseDiagram module which contains all calculations for phase diagrams. User can then create a new project of precipitation simulation in the same workspace using menus (`File → Add a New Project`). In this case, the workspace name and the working directory cannot be changed, but the user needs to give a new project name. The database file, table, graph and other data associated with one project can be viewed in individual tabs in the Display window.

When more than one project is created in a workspace, only one project will be activated at a time, and only those functions and toolbar icons associated with the activated project are available to the user at the time.

The name of the activated project will be highlighted as blue and be underlined. To select a project as the activated project, right click the mouse on the project name in the explorer window and choose "`Set as Active Project`" in the popup dialog as shown in Figure 2.4.
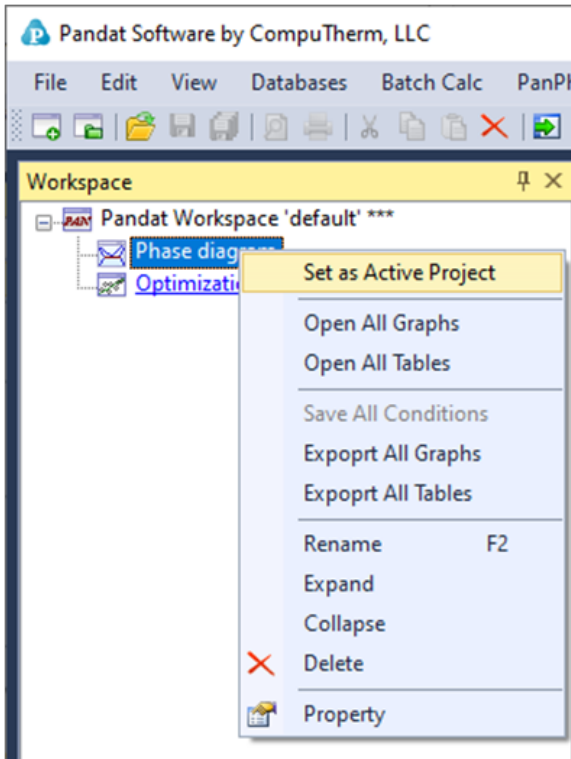
Figure 2.4: Set an active project
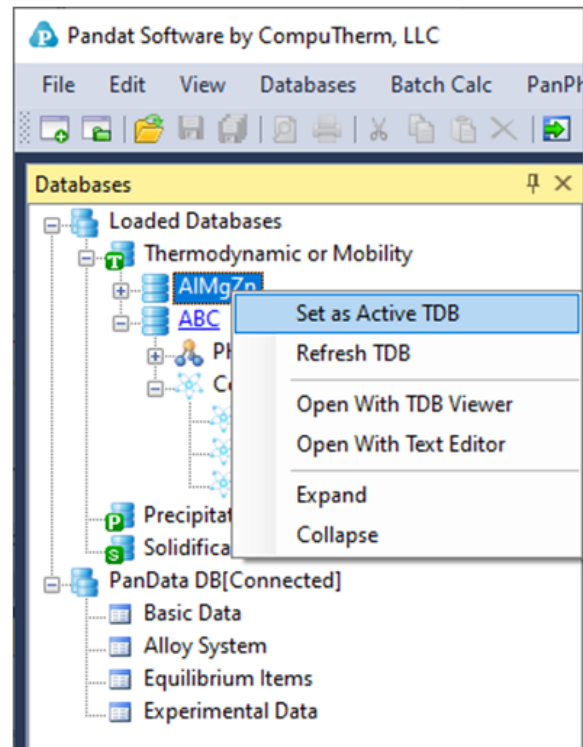
Figure 2.5: Set an active database

When switching between projects, the user may also need to switch between different databases so that the right database is used for the calculation. In the Databases view dialog in explorer window, all the loaded databases are listed and the name of currently activated database is also highlighted with blue and be underlined. To activate another database, right click the mouse on the database name and choose "`Set as Activate TDB`" as also shown in Figure 2.5 . Inside one project, the user may also load several databases and carry out different calculations of the same type. The user needs to make sure that the correct database is activated when performing a calculation.

2.4 Graph

In each project, the calculated results are presented in two different formats: Graph and Table. Graph is one of the most important parts in Pandat™ interface. A typical Pandat™ graph includes at least three elements: a set of X and Y coordinate axes, one or more data plots and associated text and drawing objects. Each graph can have one or more data plots and these data plots can be configured individually. Each data plot corresponds to a data set which can be either calculated results or experiment data.

The graph is plotted in the main display window with Pandat™ logo. The Property window defines the properties for all the elements in the graph in detail. When a typical element is selected, the property of this element will be displayed in the Property window and the user may modify the graph through the Property window.

2.4.1 Property

The overall property of a graph consists of six categories: `Title/Line Width`, `Scale`, `Grid/Ticks`, `Tieline Property`, `Component Label` and `Margin`, as shown in the red box area in Figure 2.6 All six categories will be shown in "Property" window when the whole graph is selected. The "`Title`" property defines the axis line width, the title, title font size and tick font size for both X axis and Y axis. The "`Scale`" property defines the minimum and maximum values, increment and Log scale status for both X and Y axises and a flag of "`Ternary Plot`". If this flag is set as "`True`", the figure is plotted as Gibbs triangle for a normal isothermal section and only the increments are also shown in the Scale property. If the flag is set as "`False`", a Cartesian coordinate figure will be plotted. The "`Grid/Ticks`" property defines whether to show grids on the graph or ticks on the axis. The "`Tieline`" Property defines whether to show the tie-lines and the density of the tie-lines. The "`Component label`" defines the labels on the origin, right corner and Top corner of the graph. The "`Margin`" property defines the position of the plot in the Main display window.

The other properties of the tie-lines, such as the color, the style, are defined by individual property window associated with these tie-lines. These properties can be modified when only the tie-lines on the Main Display window are selected and highlighted as shown in Figure 2.7. User can change the appearance of a set of lines belonging to the same group by selecting this line (or a group lines) only. The properties for such data plots, such as the line color, thickness, marker type, will show up in the Property window for user

to modify. The data points and lines selected will be highlighted in the graph while the others will be grey. As shown in Figure 2.8, the phase boundary lines are selected as a group of lines to be modified in this case. The properties of this group lines are defined in the "`Plot Property`" window as "`Blue, Solid, None marker`" lines. All these properties can be modified in the Property Window as framed by the red line in Figure 2.8. The "`Line Property`" defines the property of the line and the "`Marker Property`" defines the property of the point on the graph.

User can add texts and lines/arrows freely to the graph. The "Text Property" defines the position, content, font size, color and rotating angle of the text, as shown in the red box in Figure 2.9 when the text box is selected. The "Line Property" defines the color, width, start cap and end cap of the line as shown in the red box in Figure 2.10 when the line/arrow is selected.
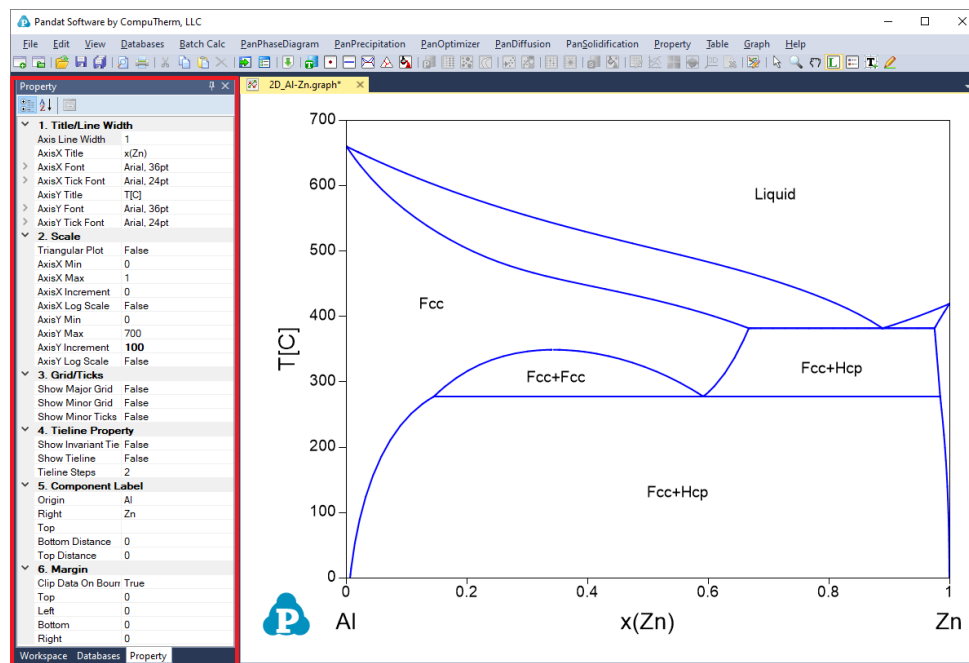
Figure 2.6: Graph property window
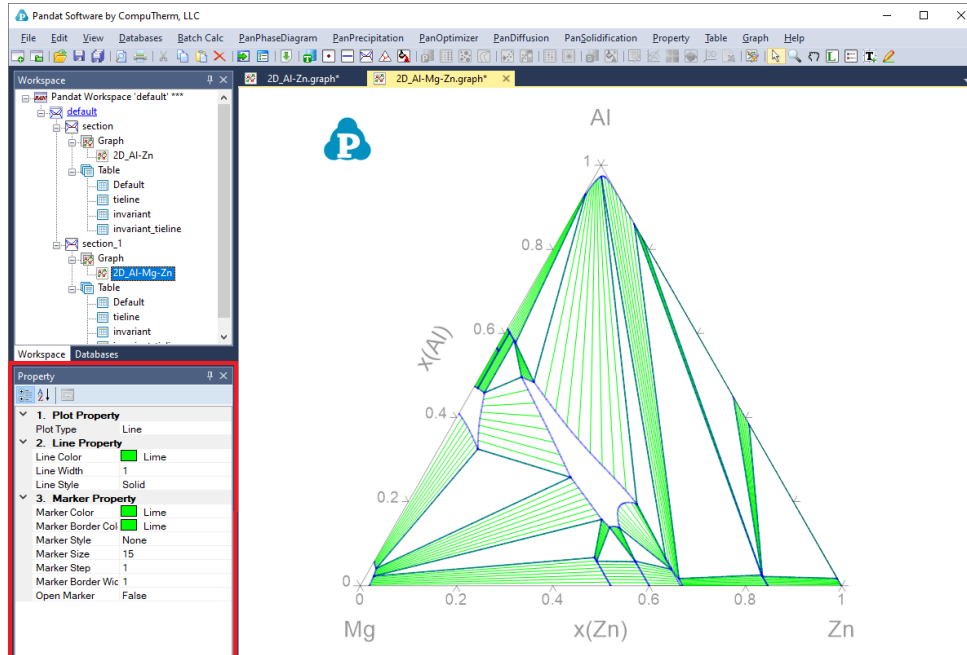
Figure 2.7: Tie-Line property window
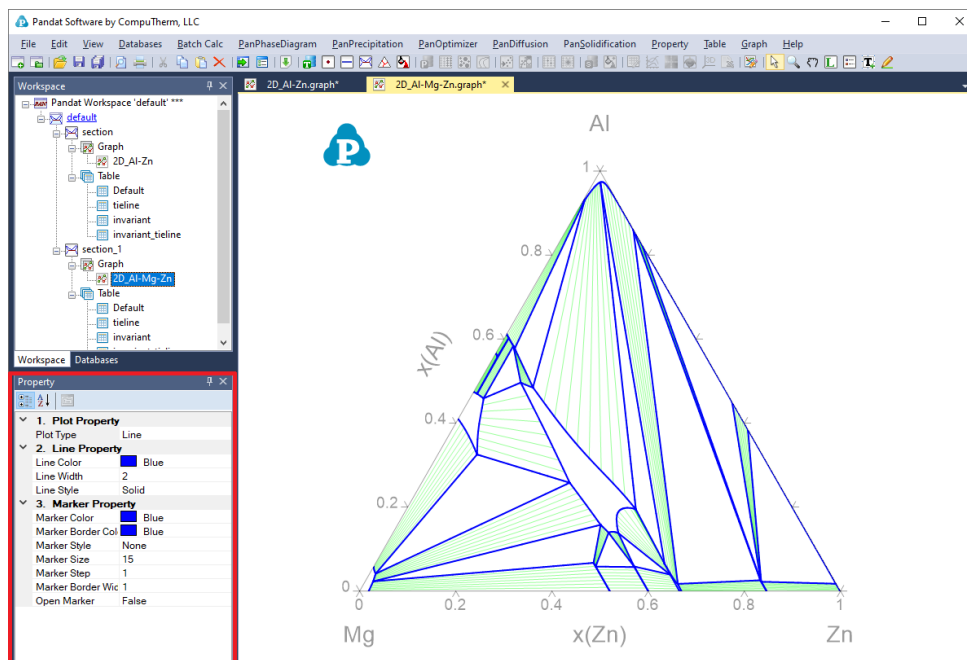
Figure 2.8: Plot property window
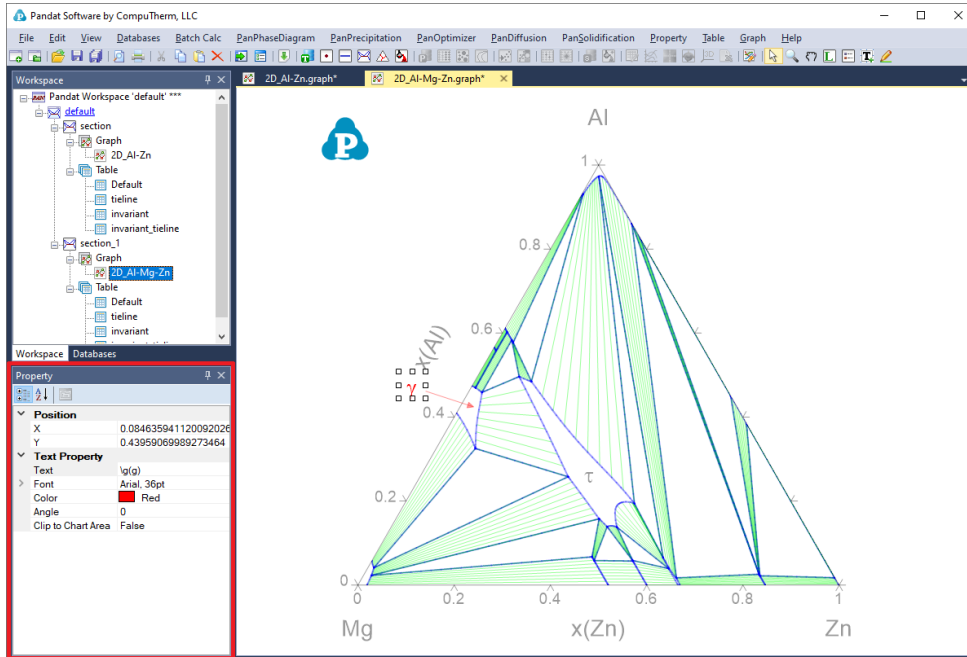
Figure 2.9: Text property window
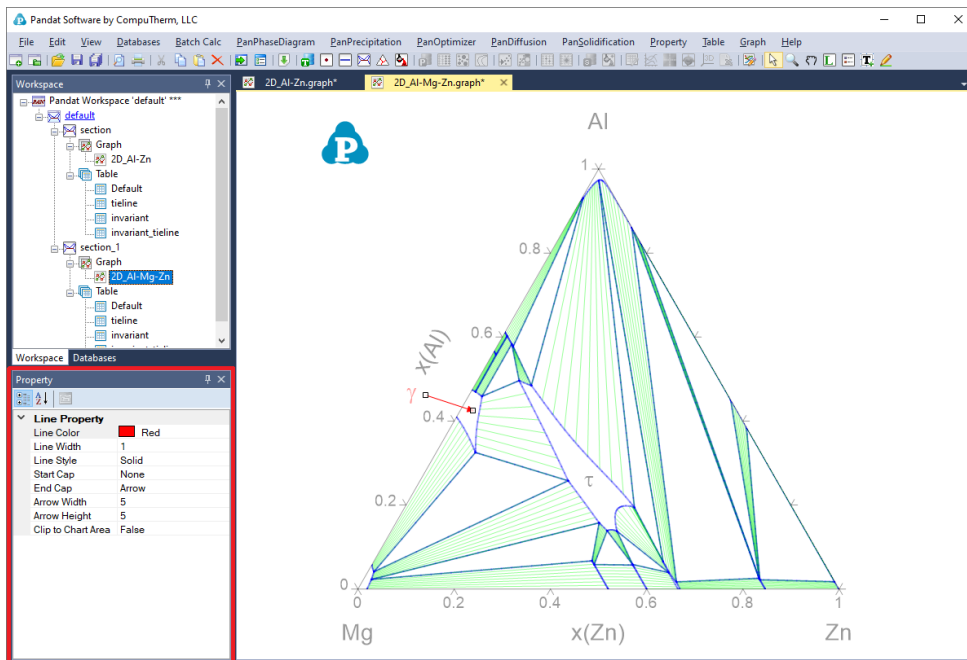
Figure 2.10: Line property window for added line

2.4.2 Export Graphs

The user can output the Pandat graph to other popular formats such as `emf`, `bmp`, `jpg`, `png`, `gif` and `tif`. The command is located on the menus: "**Graph → Export**", or right click the mouse on the graph and choose "**Export**" from the popup menu.

2.4.3 Icons for Graph on Toolbar

"**Edit Plots for the Current Graph**" button ![icon], allow the user to modify the plots, such as add data plots with mouse drag and drop to set up *x* and *y* of the new plot as shown in Figure 2.11, delete data plots using button ![icon], exchange *x* and *y* of the plots using button ![icon]. The available columns can be imported from a file or from Clipboard if the check box in front is selected.
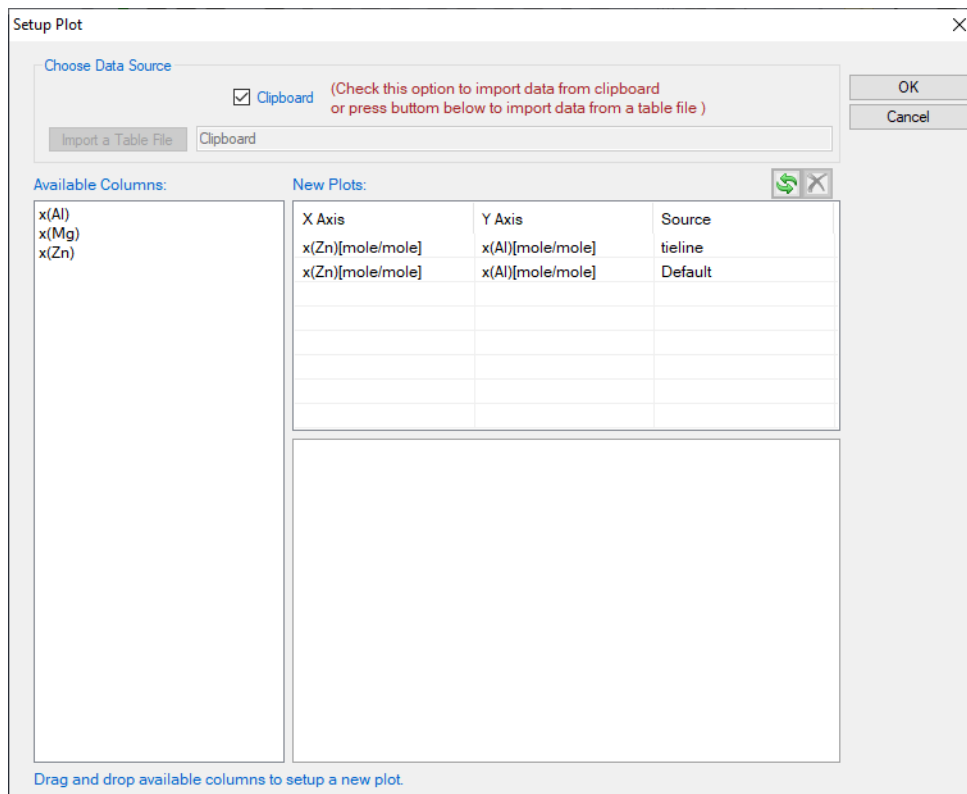
Figure 2.11: Set up data for X and Y for a plot

The table file to be imported can be Pandat™ table format or `ASCII` text format. For a general Microsoft Excel table, the user can copy the selected columns in the `Excel` file and

check the Clipboard option, and then the column names will show up in the "**Available Columns**" dialog for the user to select, which is also shown in Figure 2.11. User can also copy the data from the table columns in the Pandat™ calculation results.

- **Select Objects** button 　, allow the user to select drawing elements in a graph such as line, arrow or text.

- **Zoom Mode** button 　, use the Zoom mode to enlarge a small part of a graph. Hold the left button of the mouse and move the mouse to select a rectangular area on the graph to enlarge. The graph will zoom in to the selected area when the left button of the mouse is released. Double click this button will bring the zoom image back to the whole diagram again.

- **Pan Mode** button 　, when this mode is on, put the cursor on the plot and roll the mouse wheel to enlarge or shrink the graph, keeping the current center of the graph unchanged. Hold the left button of the mouse on the graph and move the mouse, the user can move the whole graph.

- **Label Phase Region** button 　, label the graph with text. Pandat™ can not only label the normal T-x or x-x phase diagrams from calculations, but also label those user generated graphs using the table data from calculation, i.e., G-x and μ-x diagrams. The user can modify the labeled text like normal text. If the user holds the **<Ctrl>** key first and then click the mouse on the diagram with the 　 function, the program will do a point calculation at the composition and temperature where the cursor locates. A new window will pop out to show the calculated result in detail as shown in Figure 2.12. The user can choose **Label** after reading or **Cancel** to close this window.

Point Result   — □ ×

| Label |
| Cancel |

A1Mg_Gamma + Hcp + T_A1MgZn

T		P	
K	°C	Pa	atm
573.15	300	100000	0.986923

Component	x	w	μ(J/mol)
A1	0.182467	0.169894	-26722.3
Mg	0.715659	0.600233	-21540.1
Zn	0.101873	0.229874	-43780.9

G(J/mol)	H(J/mol)	S(J/K·mol)	Cp(J/K·mol)
-24751.4	5588.84	52.936	29.0797

There are 3 stable phases

f	fw	G (J/mol)	H (J/mol)	DF (J/mol)
0.324334	0.319261	-25038.8	5502.61	0

Component	x	w	μ(J/mol)
A1	0.325173	0.307576	-26722.3
Mg	0.593284	0.505501	-21540.1
Zn	0.0815429	0.186923	-43780.9

A1Mg_Gamma

Sublattice Size	Species	y
10	Mg	1

Figure 2.12: Point calculation result for the combination of the "Ctrl" key and Labeling function

- **Add Legend for Graph** button ⊟ , add legend to the graph. The user can modify the legend using the Property window, as shown in Figure 2.13(a). Double click on the inserted legend will open a new Text Editor window as shown in Figure 2.13(b), and the user can input complex text, such as symbol, superscript and subscript in this windows.

(b)

(a)
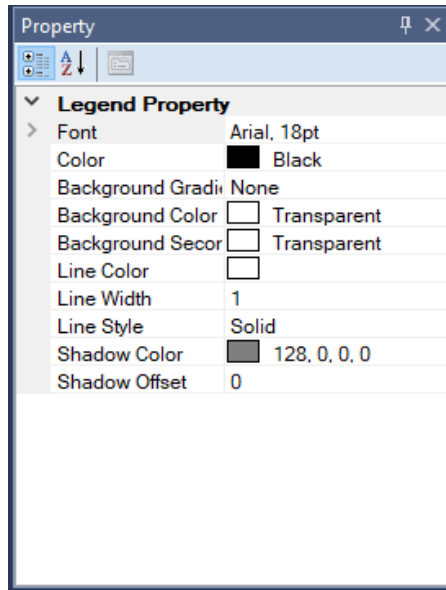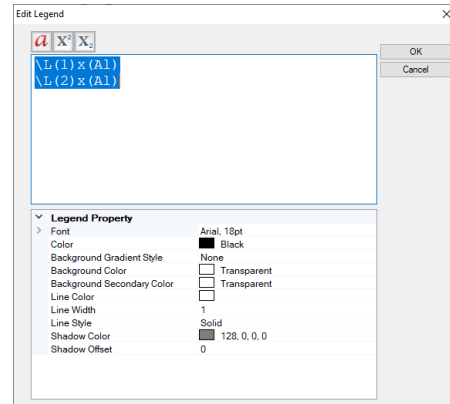
Figure 2.13: (a) Legend property window and (b) Legend editor window

- **Add Text or Label** button 🔲, add text to the graph. Change the text content, size, color and rotating angle in the **Property** window. Double click on the text also open the **Text Editor** window which allows user to input complex text in it as shown in Figure 2.14
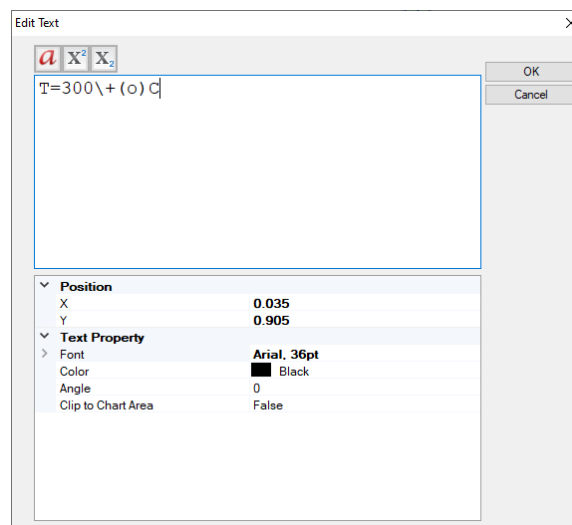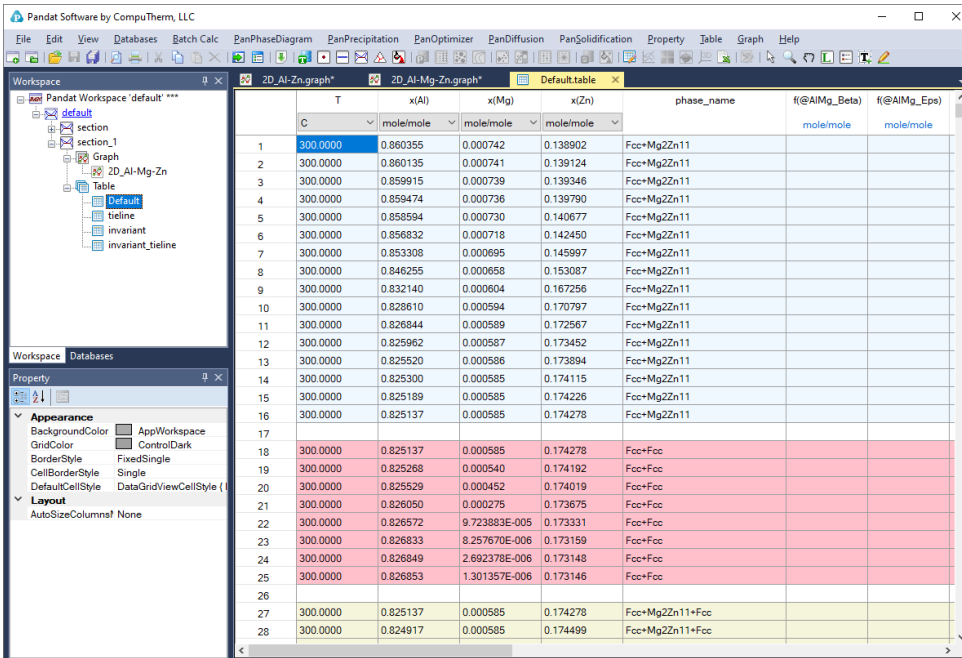
Figure 2.14: Text editor window

- **Add Line** button 🖊 , add line with or without arrow to the graph. Change the line or arrow width and color in the `Property` window. The default line has one arrow at end of the line. User can set the start cap and end cap in the `Line Property` window as shown in Figure 2.10.

2.5 Table

The menu of Pandat™ `Table` includes: `Add a New Table`, `Import Table from File`, `Create Graph`, and `Export to Excel`. A typical table is shown as Figure 2.15. Each column of data is associated with a corresponding unit. The user can change the unit of each column and observe the instant change of the table values

Figure 2.15: Table view of the calculation results in Pandat™

2.5.1 Icons for Table on Toolbar

- **Add a New Table** button 📝 , click this button or right click the mouse on the `Table` node in the `Explorer window` and select "`Add a New Table`", a window of `Table Editor` as shown in Figure 2.16 will pop out. User can create a new table by dragging and dropping the available properties in the left dialog to the right dialog as shown by the red arrow in Figure 2.16, and then click "`OK`" to generate a new table

with the selected properties.

Figure 2.16: Table Editor

- **Import Table from File** User can import a customized table from an existing file. This can be done by choosing the command located on the menus: "`Table → Import Table from File`", or right click the mouse on the `Table` node and choose "`Import Table from File`" from the popup menu as shown in Figure 2.17. Pandat™ can read data files with ".`dat`" and ".`txt`" as default file extension names. Figure 2.18 shows a typical data file for importing into Pandat™. Different columns are separated by Tabs. The first row defines the data name for each column. The second row defines the unit for the data in each column and the following rows define the data values. If the column names are the same as those in the default table, the second row (the unit row) can be blank and Pandat™ will use the same units as those in the default table.

Figure 2.17: Import a table from an existing file

Figure 2.18: A typical data file for importing

- Create Graph button ![icon], create a new graph for the selected properties. Use the `<Ctrl>` key and the left button of the mouse to select multiple columns in the table as shown in Figure 2.19. The first selected column is set as the abscissa (x-axis), and the other columns are set to be the ordinate (y-axis). After click on the button ![icon], a graph will then be generated in the Pandat™ main display window as shown in Figure 2.20.
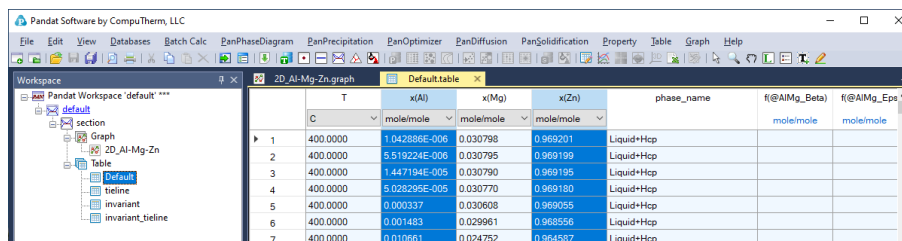
Figure 2.19: Select data for creating a new graph
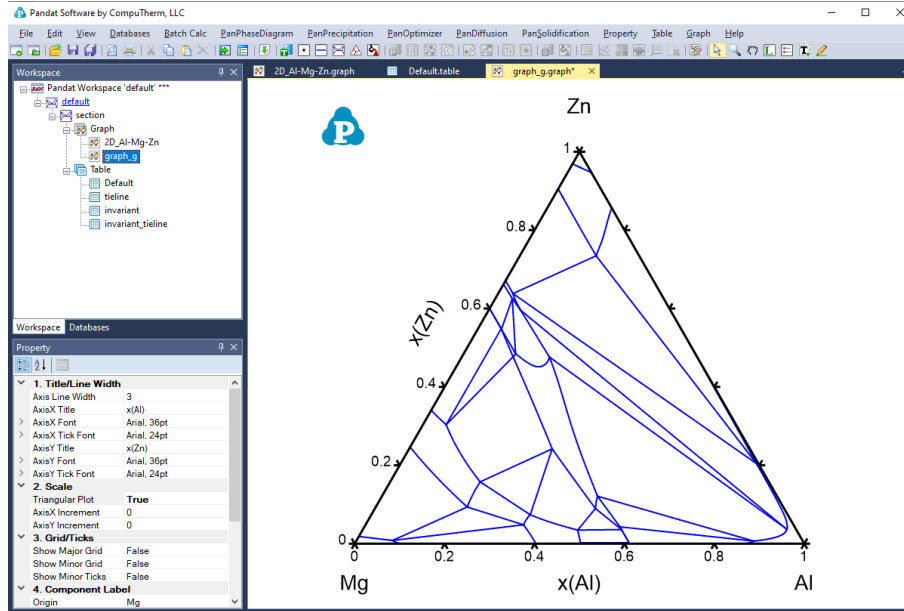
Figure 2.20: A new graph created from selected columns in the table

- **Create Color Map** button ![icon](icon), create a new color map graph for the selected properties. Use the `<Ctrl>` key and the left button of the mouse to select multiple columns in the table as shown in Figure 2.21. The first selected column `x(Mg)` is set as the abscissa (x-axis), the second column `x(Zn)` is set as the ordinate (y-axis), the value of the third column `T` will be displayed as color map in a 2D plot. After click on the ![icon](icon) button, a color map graph will then be generated in the Pandat™ main display window as shown in Figure 2.22, where the liquidus temperatures are represented by different colors.

Figure 2.21: Select data to create a new color map graph
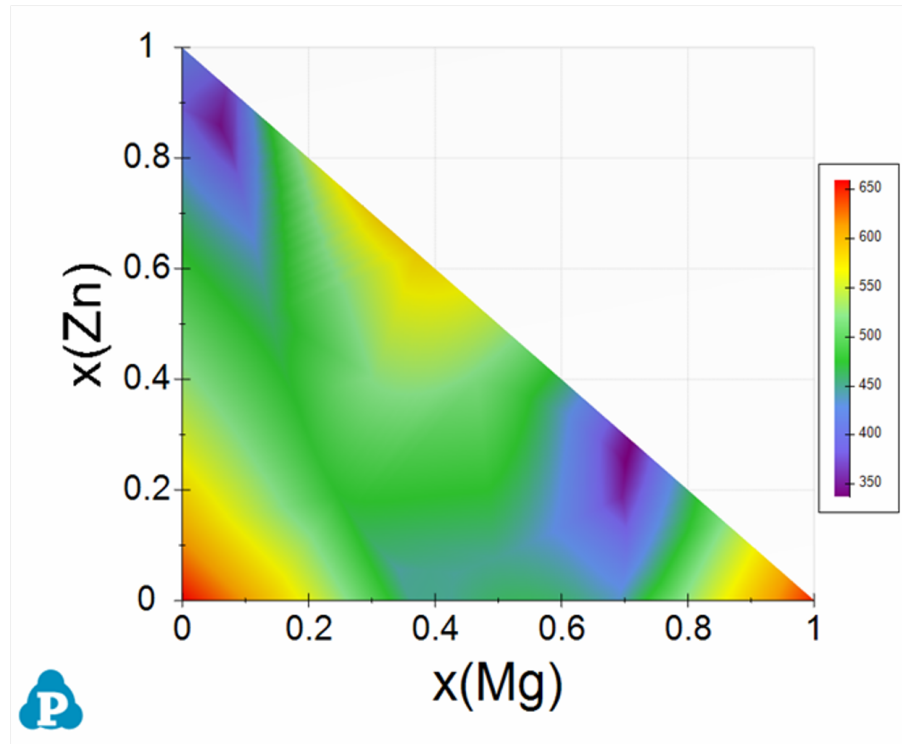
Figure 2.22: A new color map graph created from selected columns in the isotherm table from the liquidus projection calculation

- **Create 3D Surface Graph** button 🌐, create a new 3D surface graph for the selected properties. Use the `<Ctrl>` key and the left button of the mouse to select multiple columns in the table as shown in Figure 2.21. The first selected column is set as the x-axis, the second column is set as the y-axis, and the third column is set to be the z-axis for the new plot. After click on the button 🌐, a 3D liquidus surface graph will then be generated in the Pandat™ main display window as shown in Figure 2.23.
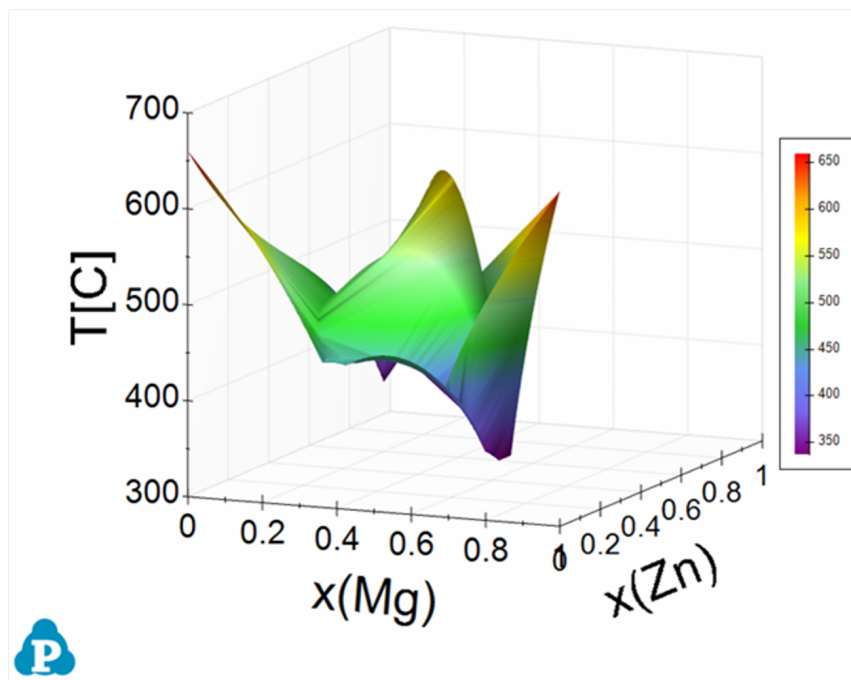
Figure 2.23: A new 3D liquidus surface graph created from selected columns in the
isotherm table from the liquidus projection calculation

- **Create 3D Graph** button 3D, create a new 3D graph for the selected properties. Use
the `<Ctrl>` key and the left button of the mouse to select multiple columns in the
table as shown in Figure 2.21. The first selected column is set as the x-axis, the
second column is set as the y-axis, and the third column is set to be the z-axis for the
new plot. After click on the button 3D, a 3D graph will then be generated in the
Pandat™ main display window as shown in Figure 2.24 representing the monovariant
lines of the liquidus projection.
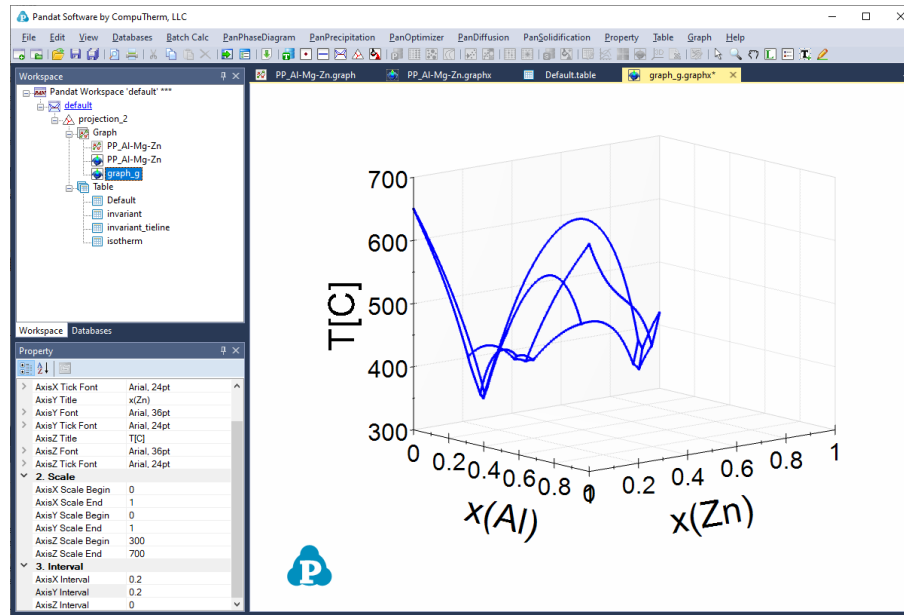
Figure 2.24: A new 3D graph created from selected columns in the table

- User can add more plots to the 3D graph. First click on the table containing the data, i.e. isotherm data in Figure 2.25, and the property window will show the column names in this table. Then select the column "x(Al)" as x-axis in the property window, drag and drop it to the graph; then hold the **<Ctrl>** key and select the second column "x(Zn)" as y-axis, drag and drop it to the graph; and then hold the **<Shift>** key, select the third column "T", drag and drop it to the graph. The new plot of the data will be added to the original 3D graph shown as Figure 2.26.
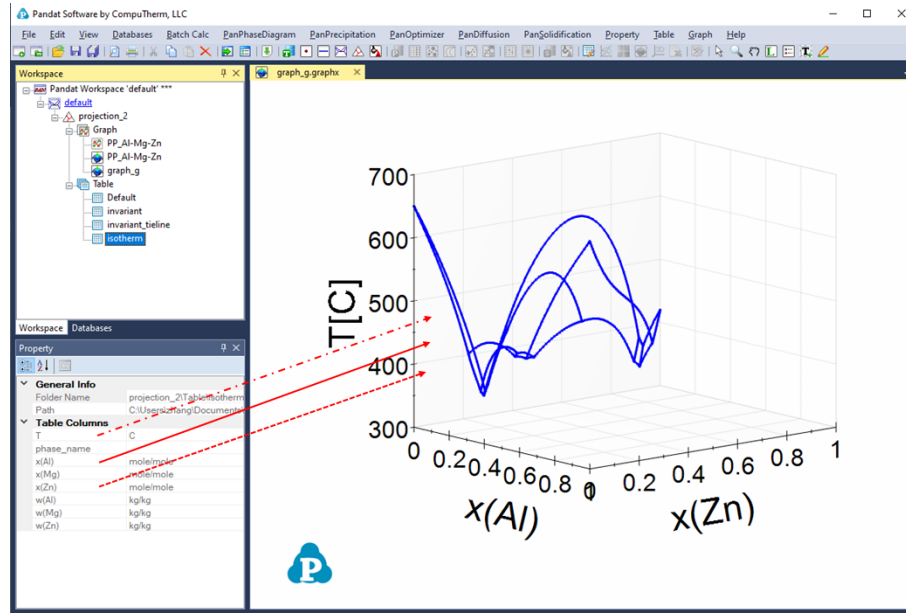
Figure 2.25: Add a plot to a 3D graph using selected columns in the table as follows: (1) Solid arrow: drag and drop, (2) Dash arrow: drag and drop with the `<Ctrl>` key held, (3) Dash dot arrow: drag and drop with the `<Shift>` key held
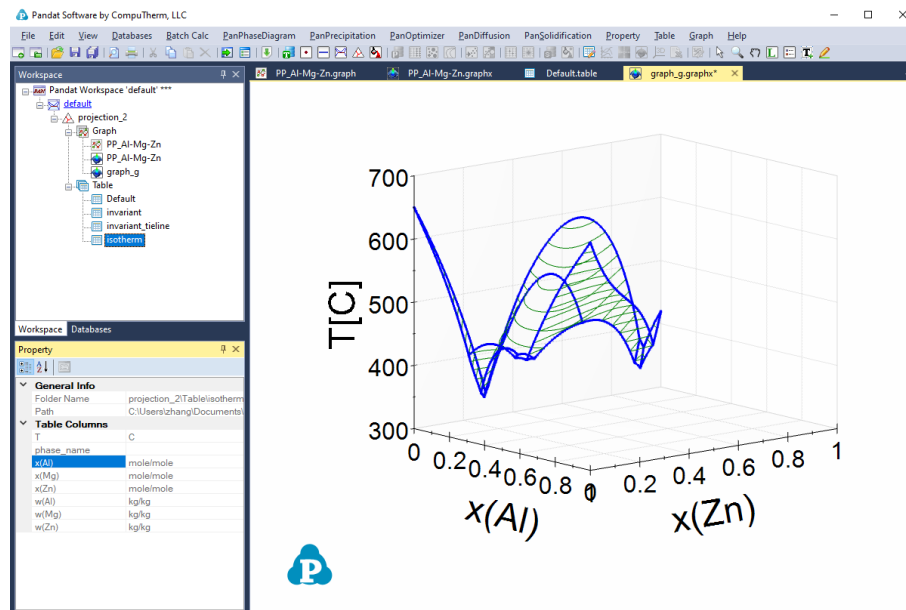
Figure 2.26: 3D graph with multiple plots

- **Export to Excel** button ▦ allow user to export the table data directly to a Microsoft Excel file, when the table is open in the main display windows to activate this function. Excel must be pre-installed in the user's computer.

- **Export to a Text File**: From the menu "`Table → Export to a Text File`", user can export the table data to `.dat` or `.txt` file.

2.5.2 Table Format Syntax

Pandat™ table column lists a series of property string switch that can be used to extract the corresponding properties from the calculation results. A property string can be simply a property name (e.g. `T, P, G, H,` and `S`) or an expression including property name and special symbols (e.g. `mu(Mg@Fcc)`). Generally, a property string has the following format:

`Z(component@phase:reference_phase[component])`

or in a simple form as

`Z(*@*:ref_ph[*])`

where Z is the property name and "`*`" represents a wild card which can be phases, components, or species. The first "`*`" after "`(`" is the name of a selected component or species. If "`*`" is used, it means all the components or species in the system. The second "`*`" represents the name of the selected phase, which must follow "`@`". Again, if "`*`" is used, it means all the phases. For example, `x(*@*)` means to list the composition of every element in every phase. The colon "`:`" is used for defining reference states. The reference phase name "`ref_ph`" must be given explicitly and the wild card "`*`" cannot be used for the names of reference phases. It should be pointed out that different element can have different reference state. For example, `a(*@*:Fcc[Al],Hcp[Zn])` means that `Al` uses `fcc` as its reference state and `Zn` uses `hcp` as its reference state. Similarly, `a(*@*:Fcc[*],Hcp[Zn])` means `Hcp` is selected as the reference state of `Zn`, while `Fcc` is selected for all the other components in the system.

If the property string has the following format:

`Z(@*:ref_ph[*])`

i.e., no component or species is specified, and the first "`*`" after "`(`" is missing, it defines the property of a phase or phases. For example, `H(@Liquid)` means the enthalpy of the liquid phase, and `H(@*)` lists enthalpies of all the phases in the system. If the reference state is not specified, it means to refer to the default reference state defined in the database which

is usually the standard element reference state. When the reference state is selected, `H` `(@Liquid:Fcc[Al],Hcp[Zn])` represents the mixing enthalpy of liquid phase referring to `Fcc Al` and `Hcp Zn` at the same temperature.

If the property string has the following format:

```
Z(:ref_ph[*])
```

i.e., both component and phase names are missing; it represents the property of the system. For example, `G(:Fcc[*])` represents the Gibbs energy of the system (equilibrium phases) referring to `Fcc` phase.

It is worth to pointing out that the wild card "`*`" is very convenient to extract properties for all the components and all the phases, especially in multicomponent, multiphase system. It should also point out that if no reference states are specified, it refers to the default reference state used in the database. Table 11.4 in Section 11.2 lists the syntaxes in the Table format.

2.5.3 Table Column Operations

When creating a new table in Pandat™, user can select the properties listed in the left column and drag them to the right column as shown in Figure 2.16 in Section 2.5.1 . Pandat™ allows user perform algebraic calculations and simple logic operations on the properties and create a customized table.

Table 11.1 and Table 11.2 in Section 11.1 list the mathematical algebraic functions available for table column expressions with examples. Nested functions are also allowed. Table 11.3 in Section 11.1 lists the logical expressions that can be used to extract a set of specific data from Pandat™ calculation results. These expressions can be applied to default table or other types of tables obtained from the calculation results. For example, user needs the first melting temperatures after a section calculation. A constraint can been easily set as "`f(@Liquid)=0`". Multiple constraints can be realized by setting one constraint in a row in the `Table Editor` shown in Figure 2.16 in Section 2.5.1 .

Here an example is given to demonstrate how to use the logical expression to obtain useful information from the calculated results. Figure 2.27 shows the calculated Al-Mg binary phase diagram. After the calculation, a table is generated with the constraint of `f`

(`@Liquid)=0`, as seen in Figure 2.28. The new generated table is shown in Figure 2.29, which is a subset of the phase boundaries of Al-Mg where liquid phase exists and has phase fraction of zero, the so-call "**first melting**" boundary. This set of boundaries can be plotted on the Al-Mg phase diagram following the approach in Section 2.4 . The first melting line is shown in red as in Figure 2.30. More examples will be presented in Section 3.3.8
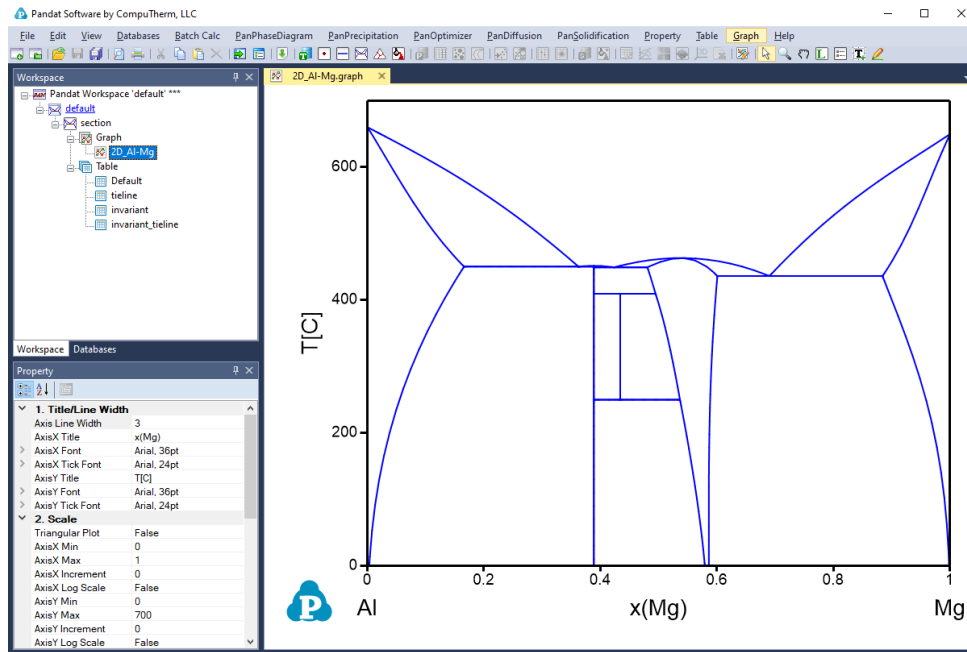
Figure 2.27: Calculated Al-Mg binary phase diagram
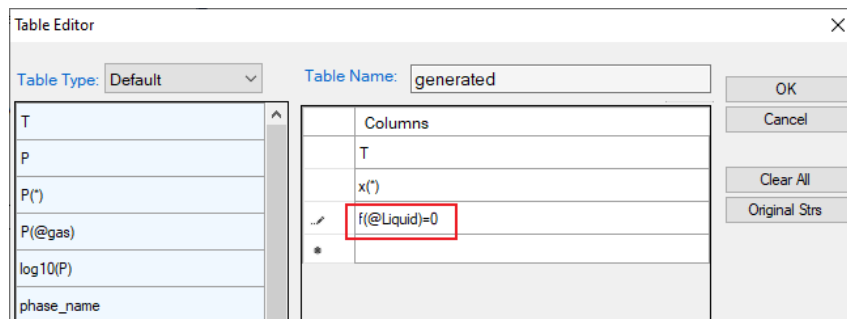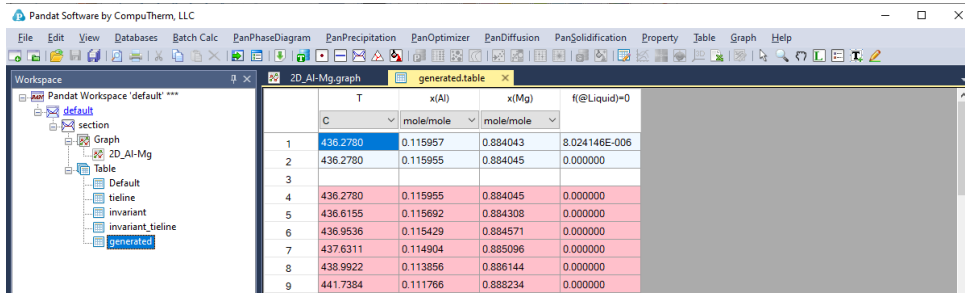
Figure 2.28: Generating a new table with a constraint of `f(@Liquid)=0`

Figure 2.29: The generated table with the constraint of `f(@Liquid)=0`
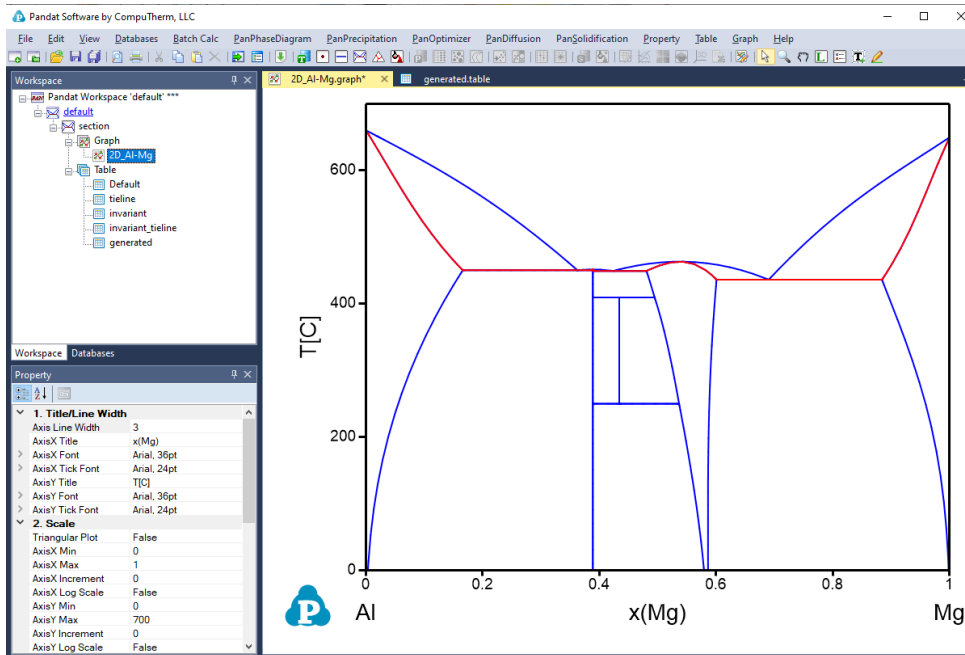
Figure 2.30: The first melting boundary merged on the Al-Mg phase diagram

3 PanPhaseDiagram

3.1 Thermodynamic Database

PanPhaseDiagram is for phase diagram and thermodynamic property calculation. Thermodynamic database is the prerequisite to fulfill such calculations. A thermodynamic database represents a set of self-consistent Gibbs-energy functions with optimized thermodynamic-model parameters for all the phases in a system. The advantage of CALPHAD method is that the separately-measured phase diagrams and thermodynamic properties can be represented by a unique "thermodynamic description" of the materials system in question. More importantly, on the basis of the known descriptions of the constituent lower-order systems, the thermodynamic description for a higher-order system can be obtained via an extrapolation method [1989Cho]. This description enables us to calculate phase diagrams and thermodynamic properties of multi-component systems that are experimentally unavailable.

In the following, thermodynamic models used to describe the disordered phase, ordered intermetallic phase, and stoichiometric phase are presented. The equations are given for a binary system, and they can be extrapolated to a multi-component system using geometric models [1975Mug, 1989Cho].

The Gibbs energy of a binary disordered solution phase can be written as:

$$G_m^\phi = \sum_{i=A,B} x_i G_i^{\phi,\circ} + RT \sum_{i=A,B} x_i \ln x_i + x_A x_B \sum_\nu L_\nu (x_A - x_B)^\nu \qquad (3.1)$$

where the first, second and third terms on the right hand of the equation represent, respectively, the reference states, the entropy of ideal mixing, and the excess Gibbs energy of mixing. Here x_i is the mole fraction of a component i, $G_i^{\phi,\circ}$ is the Gibbs energy of a pure component i, with ϕ structure, R is the gas constant, T is the absolute temperature, L_ν is the interaction coefficient in the polynomial series of the power ν. When $\nu = 0$, it is a regular

solution model, and when $\nu = 0$ and 1, it is a sub-regular solution model. Eq. 3.1 can be interpolated into a multi-component system using geometric models, such as the Muggianu model [1975Mug].

For a multi-component solution phase with c components, the following equation is used:

$$G^\phi = \sum_{i=1}^{c} x_i G_i^{\phi,\circ} + RT \sum_{i=1}^{c} x_i \ln x_i + {}^E G^{bin,\phi} \\ + {}^E G^{tern,\phi} + \dots$$

(3.2)

All excess contributions originating from all the binary interactions or ternary interactions are:

$$ {}^E G^{bin,\phi} = \sum_{i=1}^{c-1} \sum_{j>i}^{c} x_i x_j \sum_{\nu=0}^{n} L_{ij}^{\nu,\phi} (x_i - x_j)^\nu $$

(3.3)

$$ {}^E G^{tern,\phi} = \sum_{i=1}^{c-2} \sum_{j>i}^{c-1} \sum_{k>j}^{c} x_i x_j x_k \{ L_{ijk}^{0,\phi} (x_i + \delta_{ijk}) \\ + L_{ijk}^{1,\phi} (x_j + \delta_{ijk}) + L_{ijk}^{2,\phi} (x_j + \delta_{ijk}) \} $$

(3.4)

where

$$ \delta_{ijk} = (1 - x_i - x_j - x_k)/3 $$

(3.5)

For ternary systems, $x_i + x_j + x_k = 1$ and $\delta_{ijk} = 0$. But for a quaternary or higher system, $\delta_{ijk} \neq 0$. In a ternary system, if all the three L parameters are identical,

$$ L_{ijk}^{0,\phi} = L_{ijk}^{1,\phi} = L_{ijk}^{2,\phi} = L_{ijk}^{\phi} $$

(3.6)

Then

$$ {}^E G^{tern,\phi} = \sum_{i=1}^{c-2} \sum_{j>i}^{c-1} \sum_{k>j}^{c} x_i x_j x_k L_{ijk}^{\phi} $$

(3.7)

In Pandat™, if only one ternary interaction parameter $L_{ijk}^{0,\phi}$, is provided, Pandat™ will treat three ternary interaction parameters as the symmetrical case, i.e. all three parameters have the same value. Users can refer to [2005Jan] for more information about ternary interaction parameters.

An ordered intermetallic phase is described by a variety of sublattice models, such as the compound energy formalism [1979Ans, 1988Ans] and the bond energy model [1992Oat, 1995Che]. In these models, the Gibbs energy is a function of the sublattice species concentrations and temperature. The molar Gibbs energy of a binary intermetallic phase, described by a two-sublattice compound energy formalism, $(A, B)_p : (A, B)_q$, can be written as:

$$
\begin{aligned}
G_m^\phi =& \sum_{i=A,B} \sum_{j=A,B} y_i^I y_j^{II} G_{i:j}^\phi + RT \left(\frac{p}{p+q} \sum_{i=A,B} y_i^I \ln y_i^I + \frac{q}{p+q} \sum_{i=A,B} y_i^{II} \ln y_i^{II} \right) \\
&+ \sum_{j=A,B} y_A^I y_B^I y_j^{II} \sum_\nu (y_A^I - y_B^I)^\nu L_{A,B:j}^\nu + \sum_{i=A,B} y_i^I y_A^{II} y_B^{II} \sum_\nu (y_A^{II} - y_B^{II})^\nu L_{i:A,B}^\nu \\
&+ y_A^I y_B^I y_A^{II} y_B^{II} L_{A,B:A,B}
\end{aligned}
\tag{3.8}
$$

where y_i^I and y_i^{II} are the species concentrations of a component, i, in the first and second sublattices, respectively. The first term on the right hand of the equation represents the reference state with the mechanical mixture of the stable or hypothetical compounds: A_{p+q}, $A_p B_q$, $B_p A_q$, and B_{p+q}. $G_{i,j}^\phi$ is the Gibbs energy of the stoichiometric compound, $i_p j_q$, with a ϕ structure. The value of $G_{i,j}^\phi$ can be obtained experimentally if $i_p j_q$ is a stable compound; or it can be obtained by *ab initio* calculation if $i_p j_q$ is a hypothetical compound. Sometimes, $G_{i,j}^\phi$ are treated as model parameters to be obtained by optimization using the experimental data related to this phase. The second term is the ideal mixing Gibbs energy, which corresponds to the random mixing of species on the first and second sublattices. The last three terms are the excess Gibbs energies of mixing. The L parameters in these terms are model parameters whose values are obtained using the experimental phase equilibrium data and thermodynamic property data. These parameters can be temperature dependent. In this equation, a comma is used to separate species in the same sublattice, whilst a colon is used to separate species belonging to different sublattices. The compound energy

formalism can be applied to phases in a multi-component system by considering the interactions from all the constituent binaries. Additional ternary and higher-order interaction terms may also be added to the excess Gibbs energy term.

For a reciprocal system with two sublattices, sometimes the interactions among the two species on each sublattice are considered, $(A, B)_p(C, D)_q$. The interaction parameter is expressed as [2007Luk]

$$L_{AB:CD} = L^\circ_{AB:CD} + (y^I_A - y^I_B)L^1_{AB:CD} + (y^{II}_C - y^{II}_D)L^2_{AB:CD} \qquad (3.9)$$

Pandat™ treats the term $(y^I_A - y^I_B)L^1_{AB:CD}$ as the interaction on the first sublattice and the term $(y^{II}_C - y^{II}_D)L^2_{AB:CD}$ as the interaction on the second sublattice.Pandat™ does not use the alphabetical order of species.

The Gibbs energy of a binary stoichiometric compound A_pB_q, G^ϕ_m, is described as a function of temperature only:

$$G^\phi_m = \sum_i x_i G^{\phi,\circ}_i + \Delta_f G(A_pB_q) \qquad (3.10)$$

where x_i is the mole fraction of component i , and $G^{\phi,\circ}_i$ represents the Gibbs energy of component i with ϕ structure, $\Delta_f G(A_pB_q)$, which is a function of temperature, represents the Gibbs energy of formation of the stoichiometric compound. If $\Delta_f G(A_pB_q)$ is a linear function of temperature:

$$\Delta_f G(A_pB_q) = \Delta_f H(A_pB_q) - T\Delta_f S(A_pB_q) \qquad (3.11)$$

Then, $\Delta_f H(A_pB_q)$ and $\Delta_f S(A_pB_q)$ are the enthalpy and entropy of formation of the stoichiometric compound, respectively. Eq. 3.11 can be readily extended to a multi-component stoichiometric compound phase.

The strategy of building a multi-component thermodynamic database starts with deriving the Gibbs energy of each phase in the constituent binaries. There are C^2_n constituent binaries in an n-component alloy system, where

$$C^i_n = \frac{n!}{i!(n-i)!} \qquad (3.12)$$

To develop a reliable database, the Gibbs energies of these binaries must be developed in a self-consistent manner and be compatible with each other. Three binaries form a ternary, and a preliminary thermodynamic description can be obtained by combining the three constituent binaries using geometric models, such as the Muggianu model [1975Mug]. In some cases, a ternary database developed in this way can describe a ternary system fairly well, while in most cases, ternary interaction parameters are necessary to better describe the ternary system. If a new phase appears in the ternary, which is not in any of the constituent binaries, a thermodynamic model is selected for this ternary phase, and its model parameters are optimized using the experimental information for this ternary phase. There are a total of C_n^3 ternary systems in an n-component system. After thermodynamic descriptions of all C_n^3 ternaries for an n-component system are established,

the model parameters are simply used to describe quaternary and higher-order systems using an extrapolation approach. High-order interaction parameters are usually not necessary because although interactions between binary components are strong, in ternary systems they become weaker, and in higher-ordered systems they become negligibly weak [1997Kat, 2004Cha]. It is worth noting that the term, "thermodynamic database" or simply "database", is usually used in the industrial community instead of "thermodynamic description", particularly for multi-component systems.

At CompuTherm, multi-component databases have been developed for a variety of alloys, such as for Al-alloys, Co-alloys, Cu-alloys, Fe-alloys, Mg-alloys, Mo-alloys, Nb-alloys, Ni-alloys, Ti-alloys, TiAl-alloys, high entropy alloys, and solder alloys. Information for these databases is available in the Databases Manuals from CompuTherm LLC.

3.2 Thermodynamic Calculations

3.2.1 Load Database

Pandat™ can load both `TDB` and `PDB`(Encrypted `TDB`) database files. This command can be accessed through the toolbar icon ![icon]. User can also load a database file using the Menus "`Database → Load TDB or PDB (Encrypted TDB)`" as shown in Figure 3.1. Select a database file from hard drive and click open button, or just double click the database file. user can store the databases in any directory in the computer.
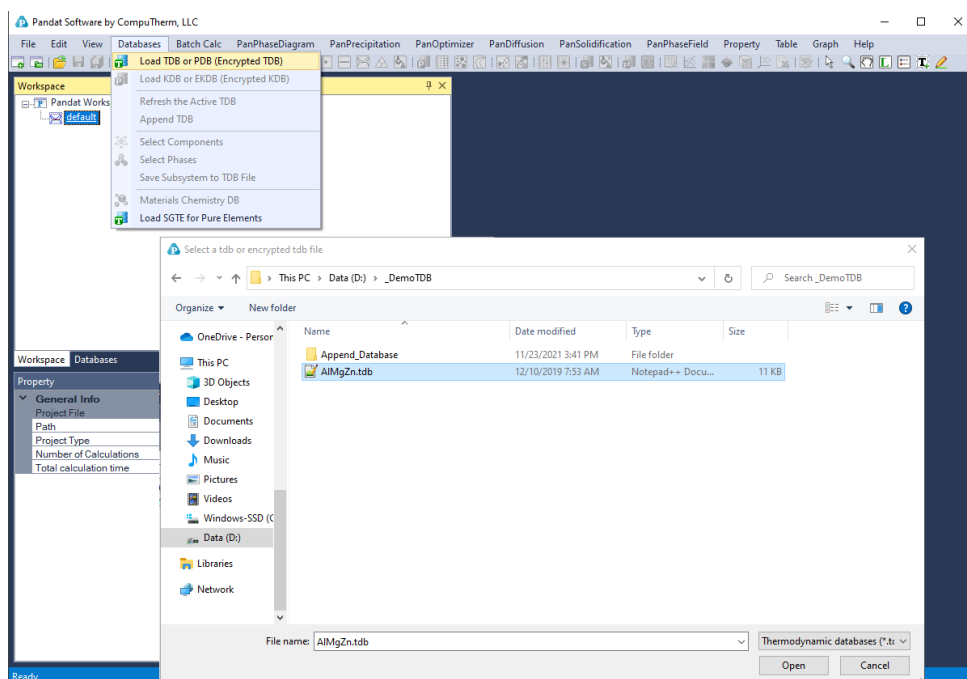
Figure 3.1: Load a database from hard drive through Menu

When a database is successfully loaded, a popup window shown in Figure 3.2 will ask the user to select the components for subsequent calculations. To add a component to the list of selected components, click on the component on the list of `Available Components` on the left column, and click on the ![icon] button to send them to the `Selected Components` on the right column. To select several components at one time, hold the `Ctrl` key and use mouse to select multiple components. To remove a component from selected components list, use mouse to select it, and click the ![icon] button.

When you want to change components during the calculations, or if the database has already loaded in the system, this window does not popup after load the tdb or pdb, click the menu "`Databases → Select Components`", this window will popup for component selection.
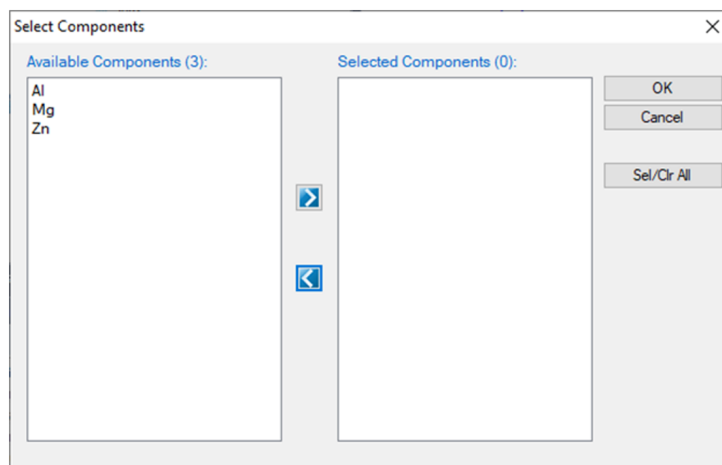
Figure 3.2: Select components for further calculation

3.2.2 TDB viewer

While loading a database file, Pandat™ checks the self-consistency of the database. A message will be displayed if the database is not internally consistent such as duplicated definitions of a function and missing definition of components or species in a sublattice. There are two types of messages: `Error` and `Warning`, as shown in Figure 3.3. The warning message can be ignored but the errors must be fixed, otherwise the database cannot be loaded. It is recommended to correct both error and warning before performing calculation. If you have difficulties loading a particular `TDB` or `PDB` file, please contact CompuTherm, LLC, and we will help resolve the problem.
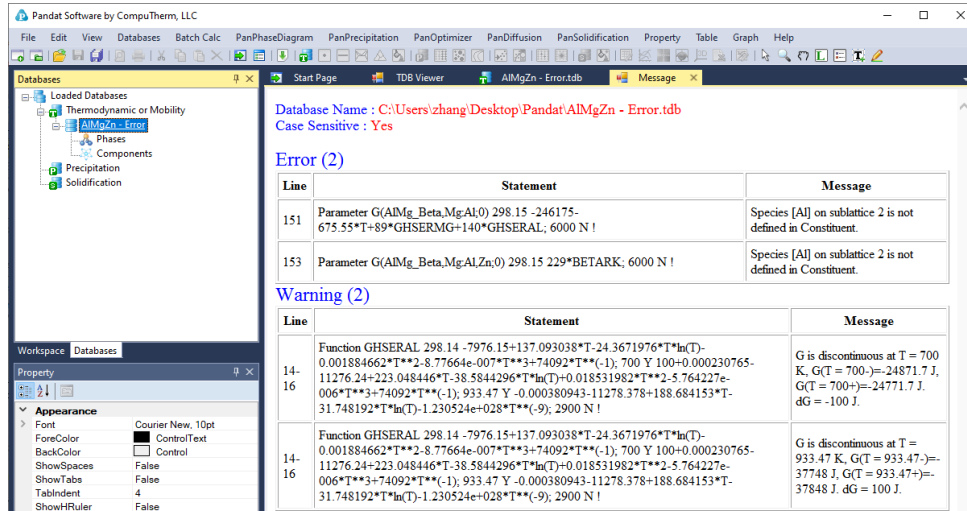
Figure 3.3: A typical message when loading a database

The successfully loaded database will be automatically summarized in the TDB Viewer mode in the main Display window as shown in Figure 3.4. For a TDB format database, the summary includes elements, type definitions, phases with name and thermodynamic model, model parameters and defined functions. For a PDB format database, the model parameters and functions will not be displayed. The user can open a TDB format database with the Text Editor mode by right click of the mouse on the database name in the explorer window. The Text Editor is actually a built-in notepad for text files. User can use the Text Editor directly within Pandat™ workspace to edit the TDB database, or he/she can always use his/her favorite text editor to do it.
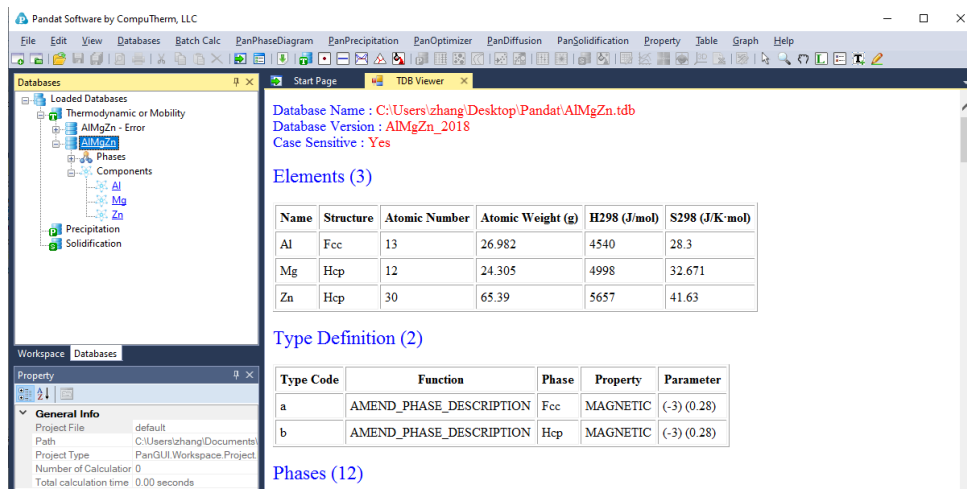
Figure 3.4: The TDB viewer mode for database summary

Multiple databases can be loaded into the same workspace but only one is activated. The current calculation will use the activated database, which is highlighted in the explorer window at Database view. The user may set the inactive database to be active with right click of the mouse as shown in Figure 3.5.
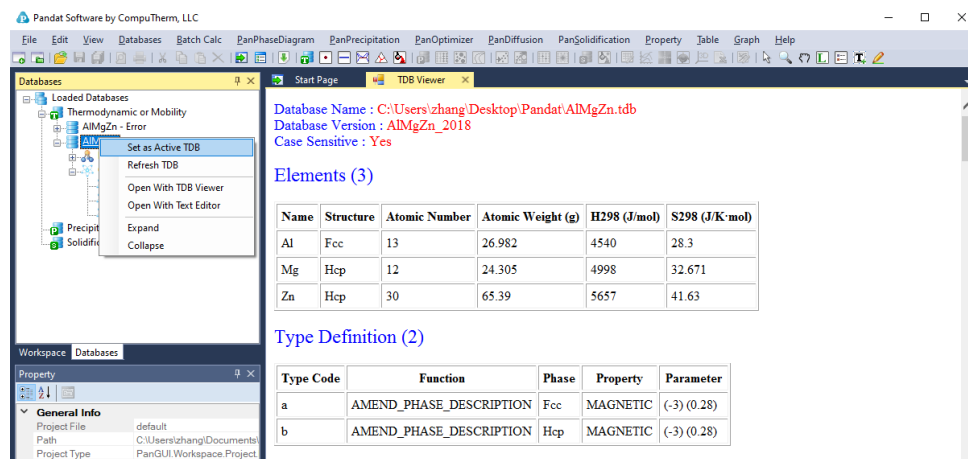
Figure 3.5: Set an inactive database to be active

3.2.3 Options

To configure the calculation options, go through the menus (`View` → `Options`), or click on the icon 📋 on the toolbar. A pop-up dialog box allows the user to change the units, and output formats of default Table and Graph.

- **Units:** This dialog box allows the user to choose the units to be used in the calculation. Use the left button of the mouse to select a proper unit for each property and click "`OK`" to complete the unit setting as shown in Figure 3.6.
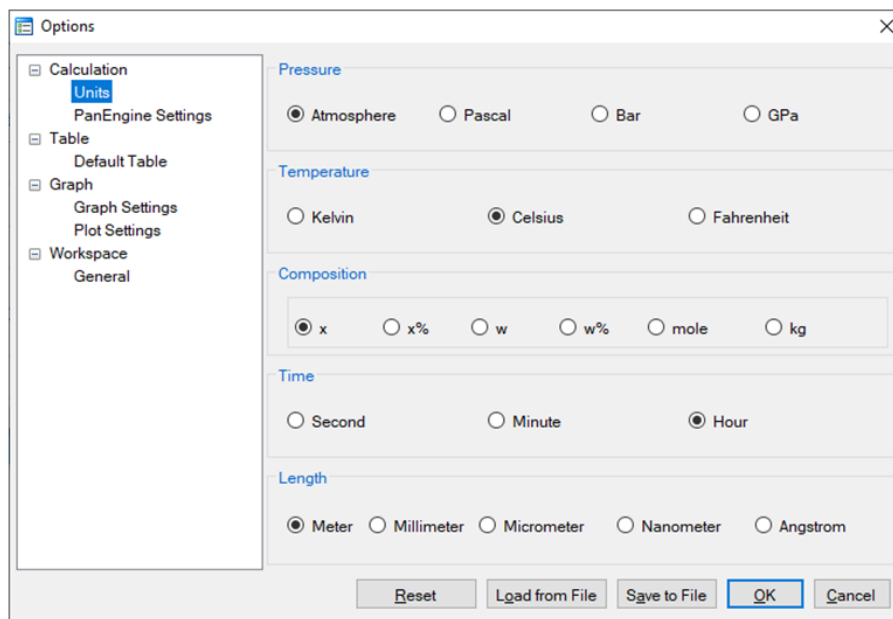
Figure 3.6: Set units for a calculation in Options

- **PanEngine Settings**: This dialog box allows the user to change the PanEngine Parameter and `TDB` Parsing parameters as shown in Figure 3.7 When "`Extensive Search`" is checked, PanEngine will perform a more thorough search of the global equilibrium state in case the Normal Search missed it. For majority of cases `Extensive Search` is not necessary and will give the same answer as that of Normal Search. The calculation will take longer time when Extensive Search is chosen. `Extensive Search` is recommended only when the user found metastable equilibrium in a particular calculation.
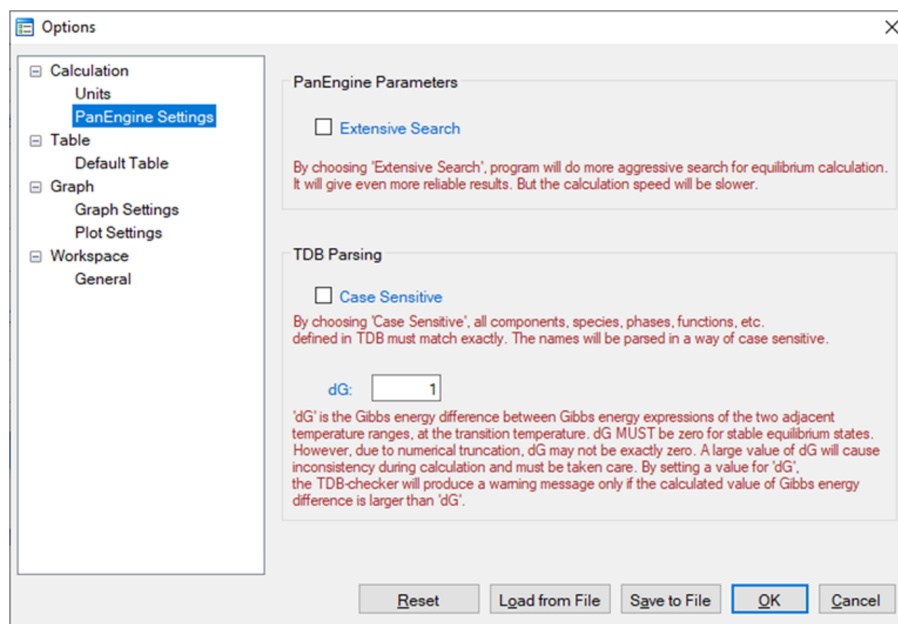
Figure 3.7: PanEngine Settings in Options

If the "`Case Sensitive`" box is checked, all the components, species, phases, functions, etc. defined in the `TDB` will be parsed in a way of case sensitive and should match exactly, in other words, the uppercase letters are treated as different from the corresponding lowercase letters.

The Gibbs energy of a pure component with a certain crystal structure is usually segmental function of temperature and is represented by different Gibbs energy expression at different temperature range. `dG` represents the Gibbs energy difference obtained from the two Gibbs energy expressions at adjacent temperature ranges at the transition temperature. Ideally, `dG` is zero if the two Gibbs energy expressions give the exact same value at the transition temperature. However, due to numerical truncation, `dG` is usually not zero. The default setting of `dG` in PanEngine is 1.0 J/mole atom. A warning message will be sent to `TDB` viewer if `dG` is greater than 1.0. A large `dG` will cause calculation fail, therefore should be fixed in the `TDB` file. User can set this value to a bigger or smaller value according to his/her tolerance.

- **Table:** This dialog box allows a user to customize the default output table with the properties the user needs. As shown in Figure 3.8, user first needs to choose the "`Calculation  Type  to  Configure`" which includes point, line, section, solidification and so on. Once the Calculation Type is selected, all the possible

properties for the chosen Calculation Type will be listed in the "`Choose Table Columns`". The user can check the box in front of each item to select the column to be shown in the default output table. All the selected properties will be listed in the "`Choose Y Axis Properties`". In this same dialog, user can also define the X axis and Y axis of the default graph that is automatically shown when a calculation finishes.
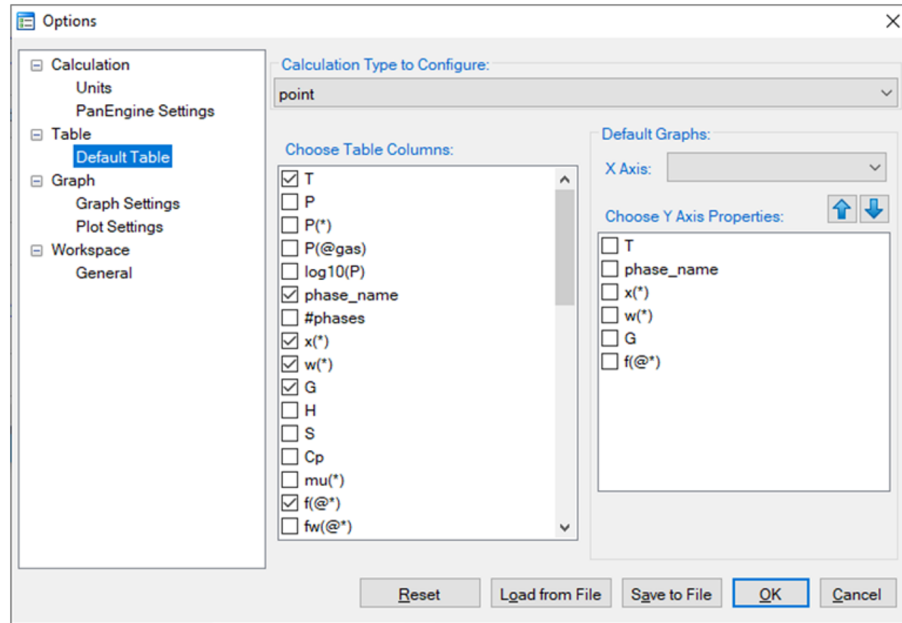
Figure 3.8: Set default table output and default dataset for graph output

- **Graph**: This dialog box allows the user to set up the appearance of the output graph. As shown in Figure 3.9, user can set the font and size of the texts, color and width of the lines, color, shape and size of the symbols, and logo size and positions for the default graph. User can also set the color and width of the lines, color, shape and size of the symbols for different plots in one graph as shown in Figure 3.10
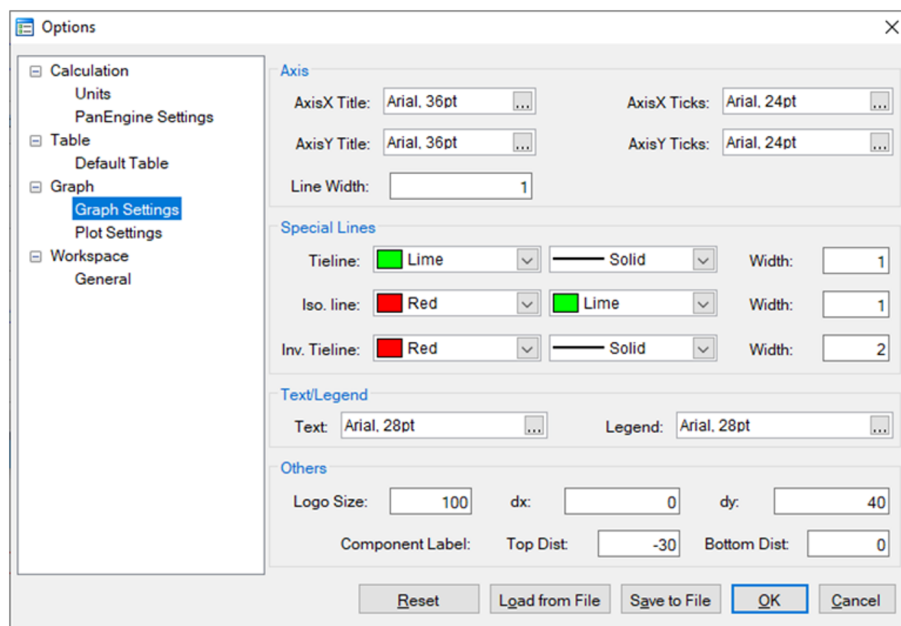
Figure 3.9: Set the appearance parameters for graph
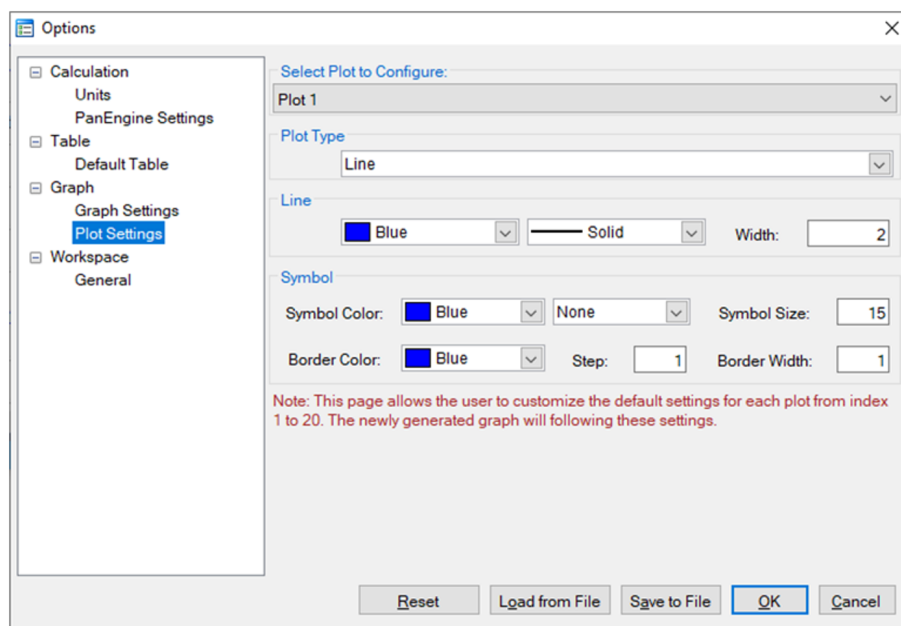
Figure 3.10: Set the appearance parameters for each plot in the graph

- **Workspace**: This dialog box allows the user to set up the default working directory for Pandat™ workspace as shown in Figure 3.11.
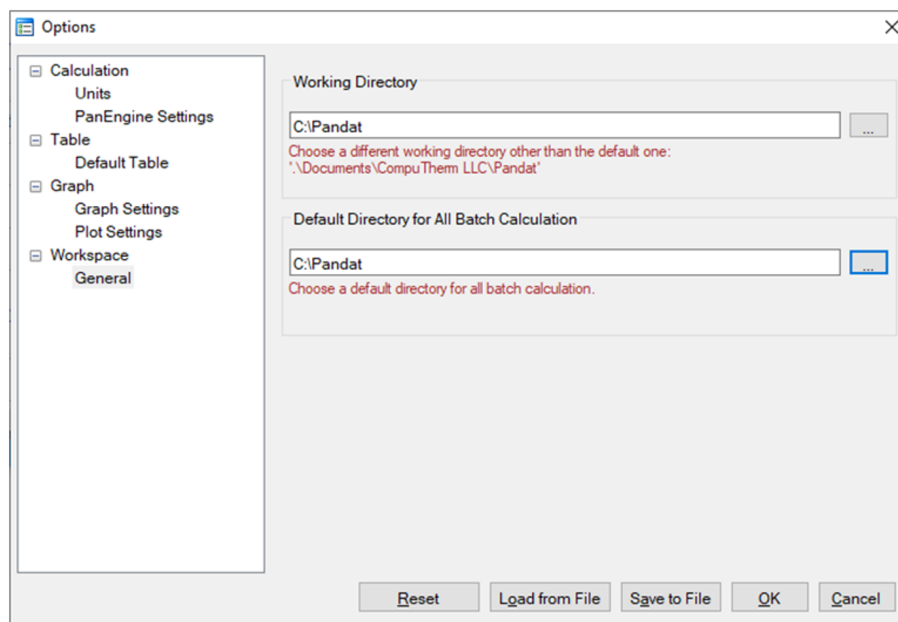
Figure 3.11: set up the default working directory for Pandat workspace

It should be pointed out that "`Options`" gives user the opportunity to change the `PanEngine Setting` and customize the output of Default Table and Graph. However, user do not have to do anything if not necessary. By default, the most popular setting has been selected for each type of calculation.

3.3 Tutorial

In general, user should follow four steps to carry out the calculations in PanPhaseDiagram module:

- **Create Workspace and Project:** Create a PanPhaseDiagram module project in an existing or a new workspace.

- **Load Thermodynamic Database:** Load thermodynamic database (.tdb or .pdb file). The Al-Mg-Zn system is used for these tutorials. The `AlMgZn.tdb` is in the "PanPhaseDiagram" directory under "Pandat 2024examples" folder.

- **Set Calculation Conditions:** Set the calculation conditions from the pop up condition setting dialog based on different types of calculations.

- **Post Calculation Operation**:Edit tables and graphs from the calculated results.

3.3.1 Point Calculation (0D)

This function allows the user to calculate the stable phase equilibrium at a single point in a multicomponent system

Go to **PanPhaseDiagram** on the menu bar and select `Point Calculation` or click the icon [·] on the tool bar. The dialog box for `Point Calculation` as shown in Figure 3.12(a) allows the user to set up the calculation conditions: composition and temperature. User can either input the conditions manually or use the `Load Condition` function to load from an existing batch file. User can always save the current calculation condition to a batch file by clicking the `Save Condition` button, or save the alloy composition on hard disc by clicking `Save Chemistry` button. User can load the alloy composition for the future calculation using the function `Load Chemistry` following Figure 3.12(b).
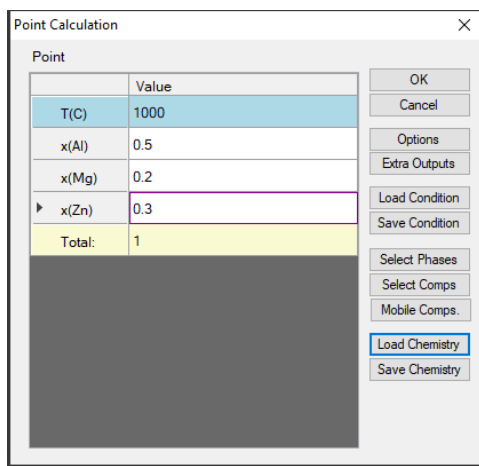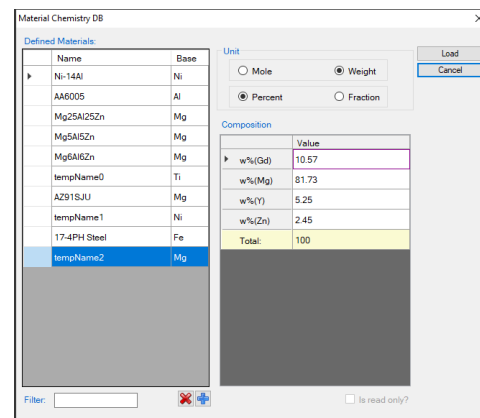
(a) (b)

Figure 3.12: Set alloy composition and temperature for point calculation

The functions of other buttons as shown in Figure 3.12(a) are given below. User can access the `Options` window again by clicking `Options` and make changes for the units. Please refer to the previous section for information on the `Options` window. The `Extra Outputs` button allows the user to modify the default output table and graph and add more tables and graphs as needed. A pop out window as shown in Figure 3.13(a) provides the access to the table and graph for output. User may add or delete output table/graph using the buttons [+] or [✖] , or modify the selected table/graph by clicking the "`edit`..." button. Please refer to Section 2.5.3 for information on Table Edit. The operations for editing the

table are the same as those described in the previous section as shown Figure 3.13(b).The Graph dialog is slightly different from that described in the previous section. As shown in Figure 3.13 (c), data source for the graph can only be the tables generated from the calculations and there is a check box for the triangle plot.
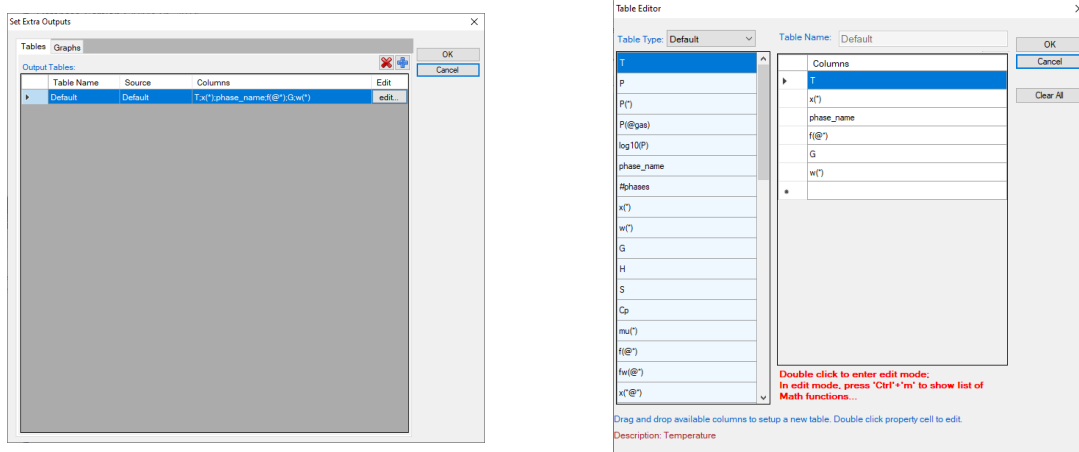
(a)

(b)
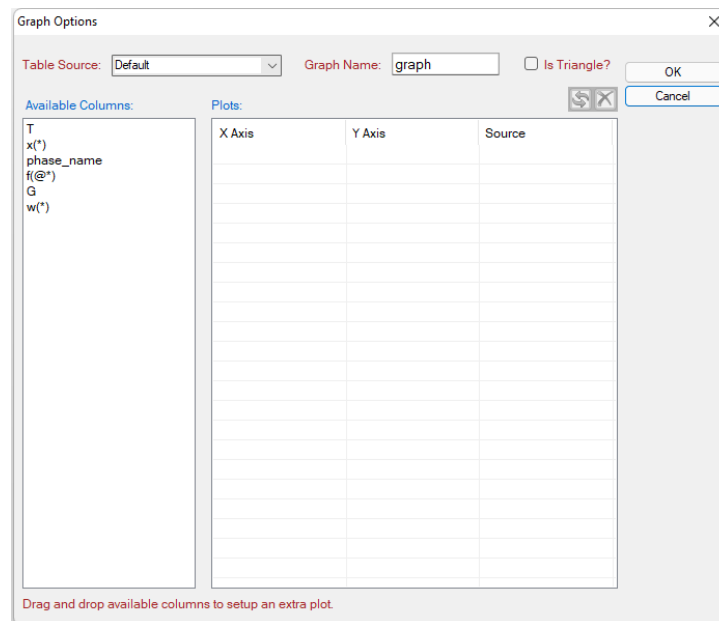
(c)

Figure 3.13: Set extra output table and graph dialog

The `Select Phases` button in Figure 3.12(a) leads the user to a dialog which allows the user to select or deselect phases to be involved in the calculation as shown in Figure 3.14. The default setting is that all the phases in the system are selected and listed in the

`Entered Phases` column, which means all the phases are involved in the equilibrium calculation. User can select the phases that will not participate in the equilibrium calculation by sending them to the `Dormant Phases` or `Suspended Phases` column using mouse to drag and drop or using the ◀ button. To select several phases at one time, hold the "`Ctrl`" key and use mouse to select multiple phases. The `Suspended Phases` means that the phases in that column are excluded in any calculations. The `Dormant Phases` means that the phases in that column are excluded from the equilibrium calculation but will be included for the property calculations, such as driving force, after the equilibrium calculation is finished. The `Dormant Phases` option mainly works for point calculation and line calculation. The `Suspended Phases` option is usually used for metastable phase equilibrium calculations.
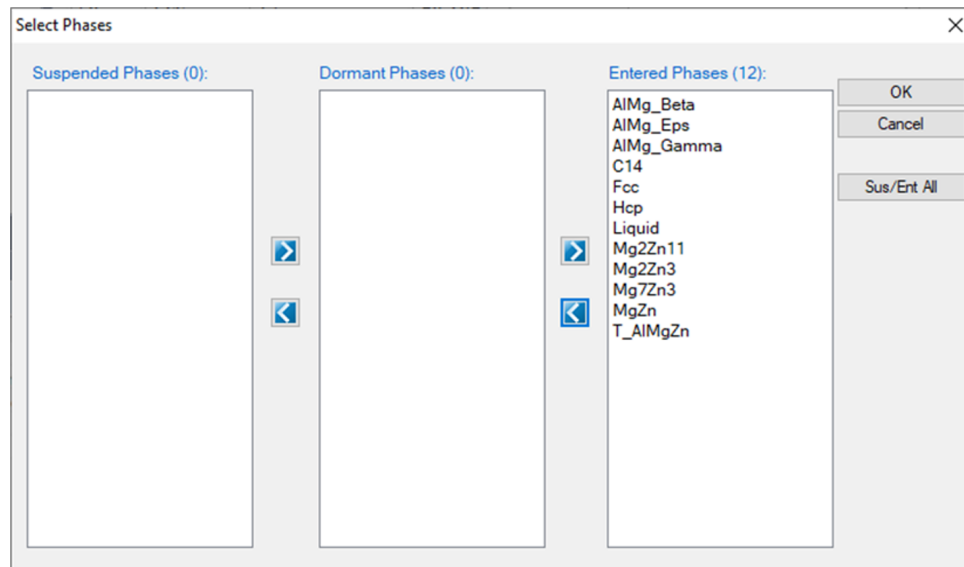
Figure 3.14: Select Phases dialog

The `Select Comps` button in Figure 3.12(a) allows user to make last minute change on the component selection. Please refer to the previous section for information on the `Select Components` window.

After the calculation is completed, the calculated results are displayed in the Pandat™ main display window as shown in Figure 3.15. The listed properties include thermodynamic properties at this point, such as Gibbs energy, enthalpy, entropy and heat capacity. It also lists the stable phases and the fraction of each phase.

Liquid

T		P	
K	°C	Pa	atm
1273.15	1000	101325	1

Component	x	w	μ(J/mol)	H(J/mol)	S(J/K·mol)
Al	0.5	0.355316	-69753.7	41481.3	87.3698
Mg	0.2	0.128025	-89785.3	25793.6	90.7819
Zn	0.3	0.516658	-95576.5	37298.7	104.367

n(mol)	G(J/mol)	H(J/mol)	S(J/K·mol)	Cp(J/K·mol)
1	-81506.8	37089	93.1515	31.7387

Vm(cm^3/mol)	density(g/cm^3)
10.541	3.60202

There is 1 stable phase

Component	x	w	μ(J/mol)	H(J/mol)	S(J/K·mol)
Al	0.5	0.355316	-69753.7	41481.3	87.3698
Mg	0.2	0.128025	-89785.3	25793.6	90.7819
Zn	0.3	0.516658	-95576.5	37298.7	104.367

f	fw	n(mol)	G(J/mol)	H(J/mol)	Cp(J/K·mol)	G_ex(J/mol)	H_ex(J/mol)	HSN
1	1	1	-81506.8	37089	31.7387	185.724	-1270.52	57165.6

	eVal	eVec	
		Al	Zn
Liquid	137.403	-0.774654	0.632385
	416.043	-0.632385	-0.774654

Figure 3.15: Results of the point calculation

3.3.2 Line Calculation (1D)

This function allows the user to perform a series of point calculations along a line in a multicomponent system.

Go to **PanPhaseDiagram** on the menu bar and select **Line Calculation** or click the — button on the tool bar. The **Line Calculation** (1D) dialog box allows the user to set the start and end points of the line, and the number of steps to calculate along the line as shown in Figure 3.16. User can also access the **Options**, **Extra Outputs**, **Load Condition**, **Save Condition**, **Select Phases**, and **Select Comps** the same way as those in point calculation. Note that the line calculation set up in Figure 3.16 is along the line of fixing alloy chemistry with varying temperature.

Figure 3.16: Set calculation conditions for a line calculation

The calculated results are stored in both graph and table. The default graph plots the variation of fraction of each phase with respect to temperature, as shown in Figure 3.17. User can modify the graph through the **Property** window by modifying the title, scale and add legend to get a better view as shown in Figure 3.18. In the **Explorer** window, user can double click the "**Default**" under the "**Table**" tree to view the default table. User can select any columns to create new plots. Furthermore, select the "**Table**" and right click the mouse, user has the option to **Add a New Table** to generate new tables and then plot other figures as needed.
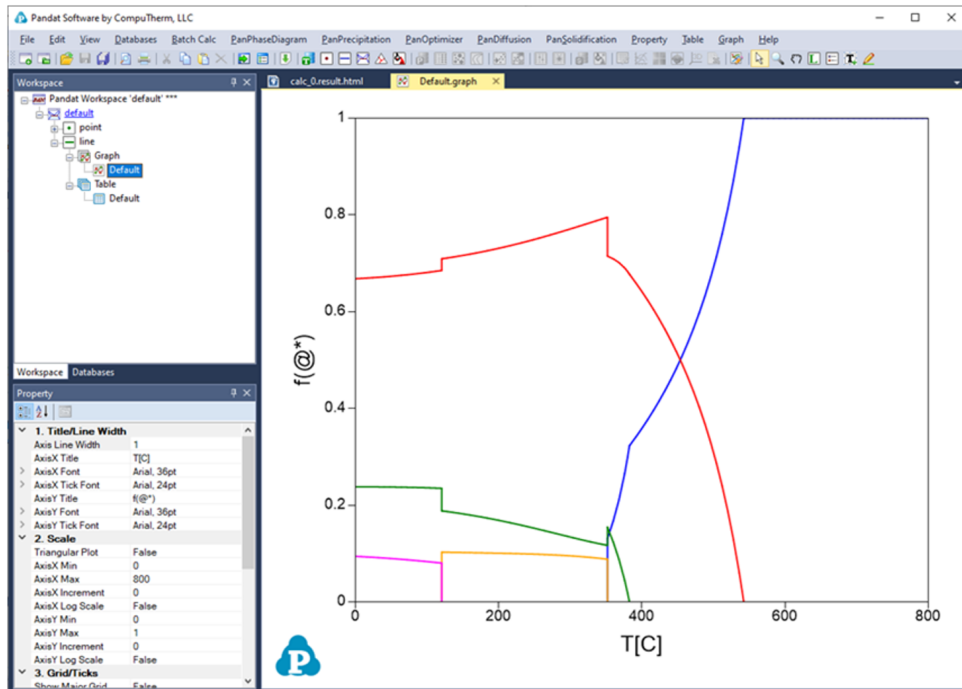
Figure 3.17: Default graph view of the line calculation result
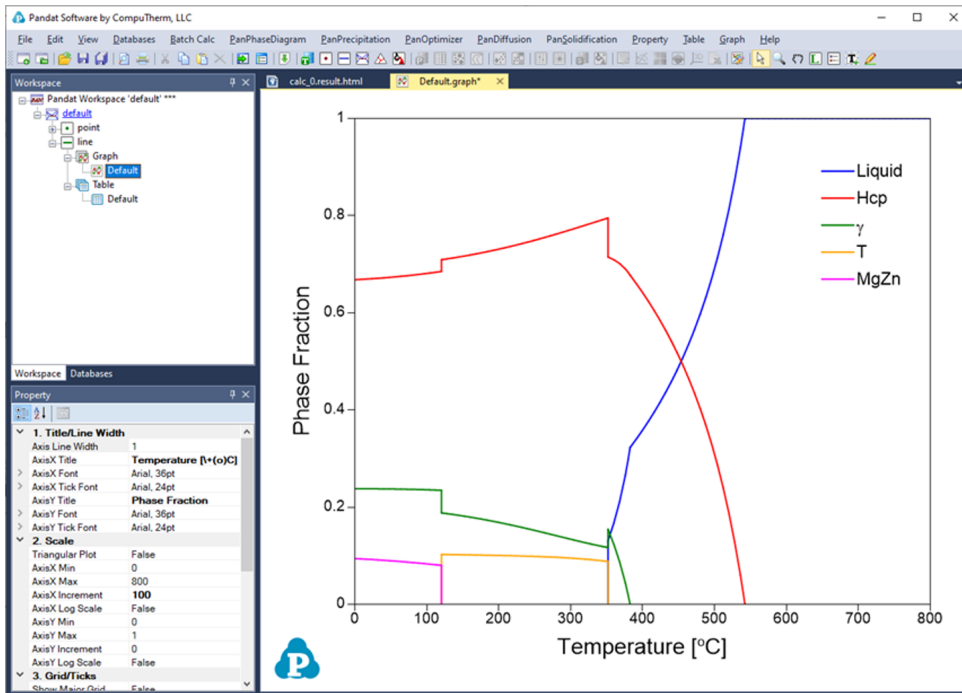
Figure 3.18: Modify the graph with a better view

It should be pointed out that user can certainly choose to fix the temperature and vary the alloy composition for a line calculation.

3.3.3 Section Calculation (2D)

This function allows user to calculate any two-dimensional section of a multicomponent system. Three non-collinear points in the calculation space are required to define a 2D section. Common 2D section diagrams are isotherms and isopleths.

Go to **PanPhaseDiagram** on the menu bar and select **Section Calculation** or click the ✉ button on the tool bar. The **Section (2D) Calculation** dialog box allows the user to set up the calculation conditions in terms of composition and temperature. Figure 3.19 shows the two most common 2D calculations for a ternary system: (a) Isotherm which is a horizontal section that fixes the temperature and (b) Isopleth which is a vertical section that uses the temperature as the Y axis.

(a)   (b)
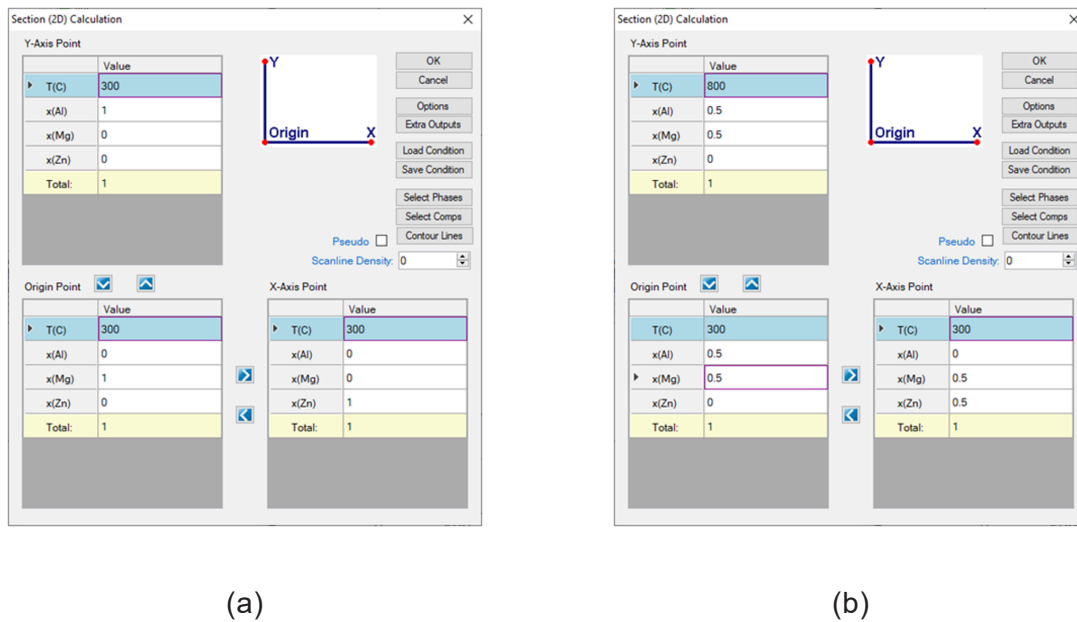
Figure 3.19: Set calculation conditions for section calculation

The composition at each point should be self-consistent. For example, `x(Al)+x(Mg)+x(Zn)=1` for the `Al-Mg-Zn` ternary system. It is not necessary to have a correlation between the `Y-Axis` point, the `Origin` point and the `X-Axis` point. User can also access the **Options, Extra Outputs, Load Condition, Save Condition, Select Phases**, and **Select Comps** windows through this dialog box. The Contour Lines option allows the user to add special lines to the output results, such as T_c curve and T_0 curve as shown in Figure 3.20.
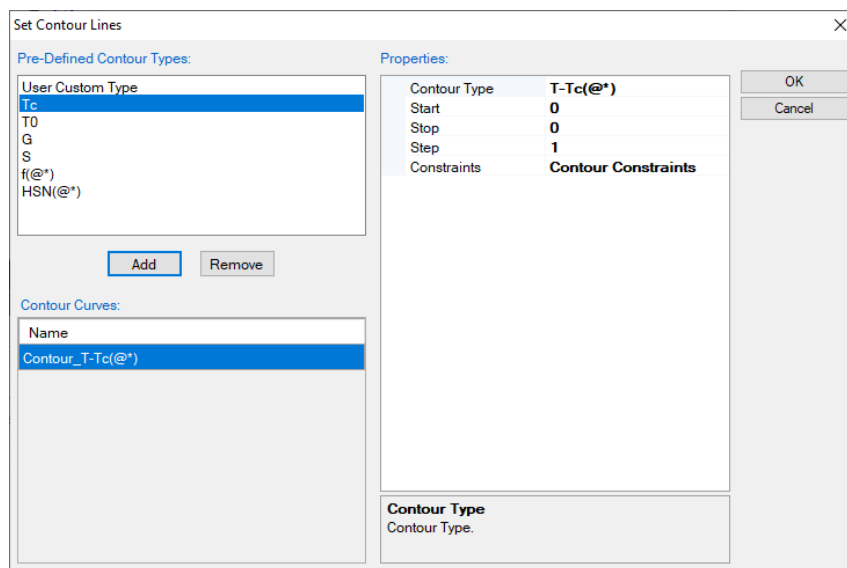
Figure 3.20: Contour Lines dialog

The setting on Figure 3.19(a) defines a calculation of an isothermal section for the Al-Mg-Zn ternary system at 300°C. The setting on Figure 3.19(b) defines a vertical section calculation from the middle of the Al-Mg binary to the middle of the Mg-Zn binary in the temperature range of 300-800°C. The results from a section calculation are displayed in two types of format in the PandatTM main window: Graph and Table. Figure 3.21 and Figure 3.22 show the graph view of the calculation results for the settings in Figure 3.19(a) and (b), respectively. Figure 3.23 shows a table view of the isotherm calculation result. User can switch between Graph view and Table View by double clicking on the graph name or table name in the Pandat™ Explorer window. Other operations on **Graph** and **Table**, such as labeling and adding a legend, have been discussed in detail in Section 2.4   and Section 2.5 .
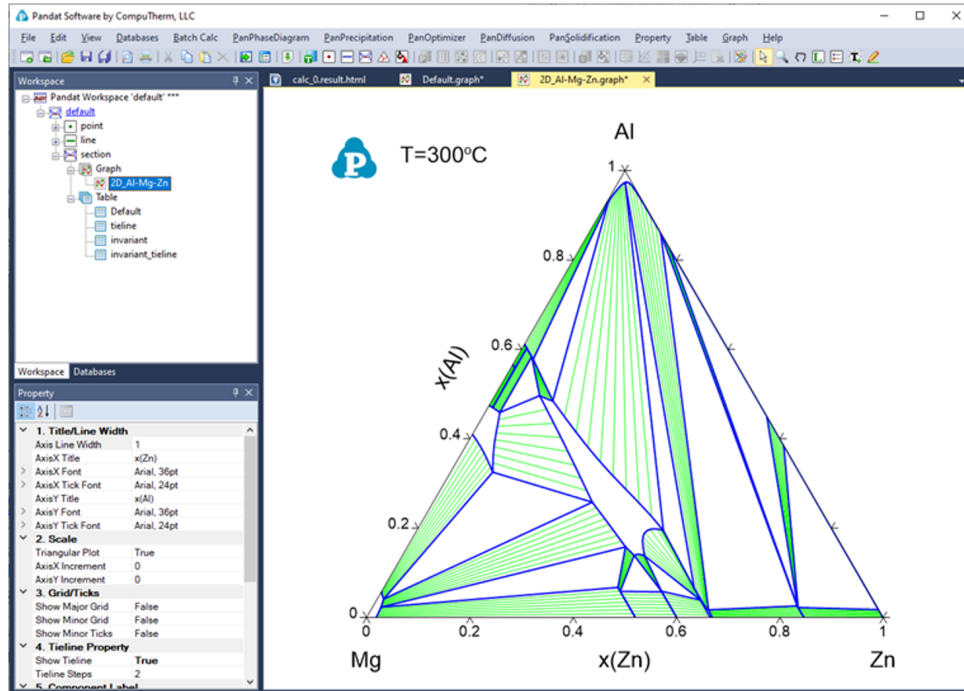
Figure 3.21: Isothermal section of the Al-Mg-Zn ternary system at 300°C
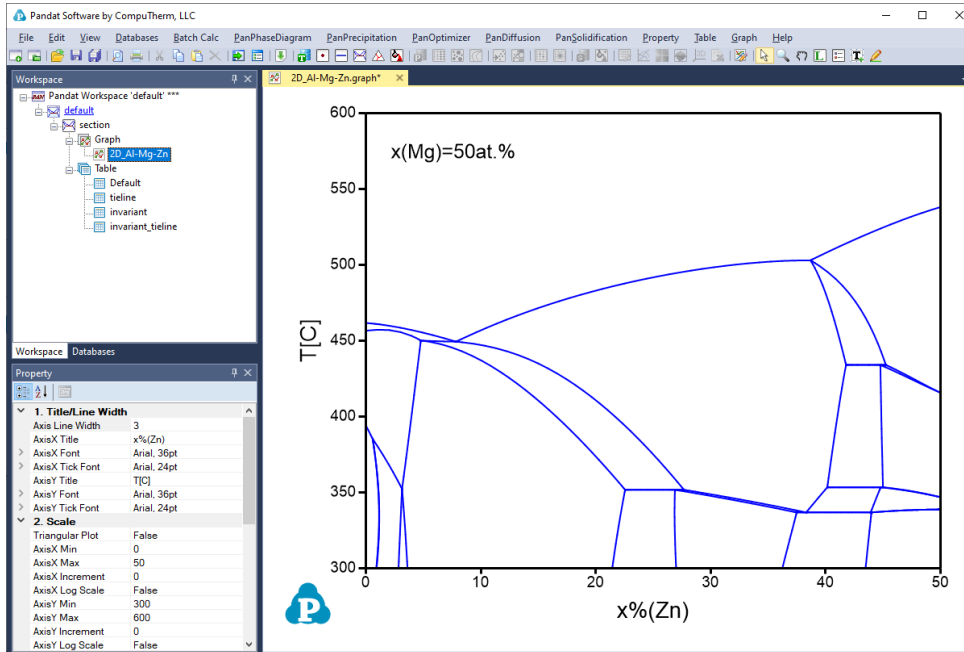
Figure 3.22: Isopleth of the Al-Mg-Zn ternary system at 50 at.% of Mg

Figure 3.23: Table view for the 300°C isotherm calculation results

3.3.4 Pseudo Binary Section

Sometimes a ternary isoplethal section is so special that all the tie-lines are within this section to form a pseudo binary section. To calculate this type of section, a special algorithm is designed in Pandat™ and a special keyword "`pseudo`" is required. In the folder where Pandat™ software is installed:

    `Pandat 2024 Examples/PanPhaseDiagram/Pseudo_Binary/,`

There is an example A-B-C system with "`ABC.tdb`". After load the database file, click on "`Section Calculation`"; the normal section calculation conditions can be set as shown in Figure 3.24 for AB-AC isoplethal section calculation. The advanced Pseudo section calculation function is introduced in Pandat™. Once the "`Pseudo`" box is checked, the select species dialog will pop out as shown in Figure 3.25 for users to select or define the species for Pseudo section calculations. For example, users can define A1B and A1C as two species for the AB-AC Pseudo binary section in A-B-C system. After the two species are defined, the "`Section Calculation`" interface changes to using the defined species for conditions setting as shown in Figure 3.26. The calculated pseudo binary section is shown in Figure 3.27.

Note: The species defined here is used as two end members for Pseudo section calculation. They are different from the species defined in the database. The species name must separate with a number, for example, A1B.
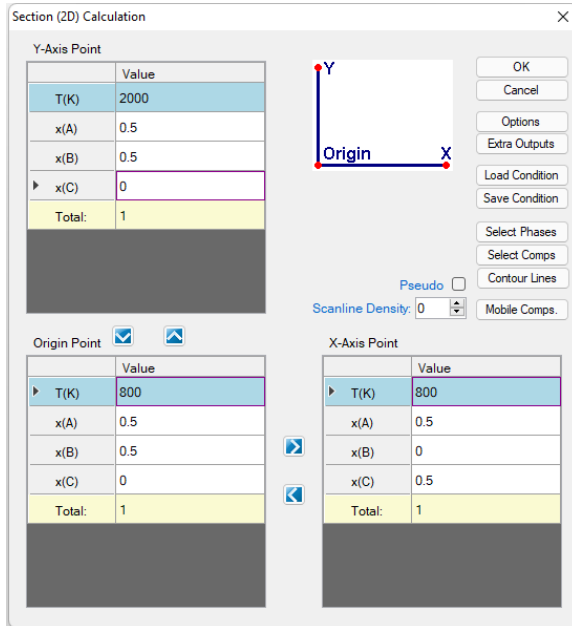
Figure 3.24: Set calculation condition for a normal AB-BC section based on elements
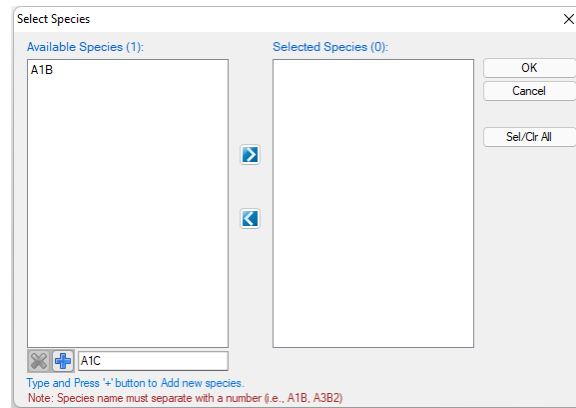
Figure 3.25: Define A1B and A1C as species for a Pseudo section calculation
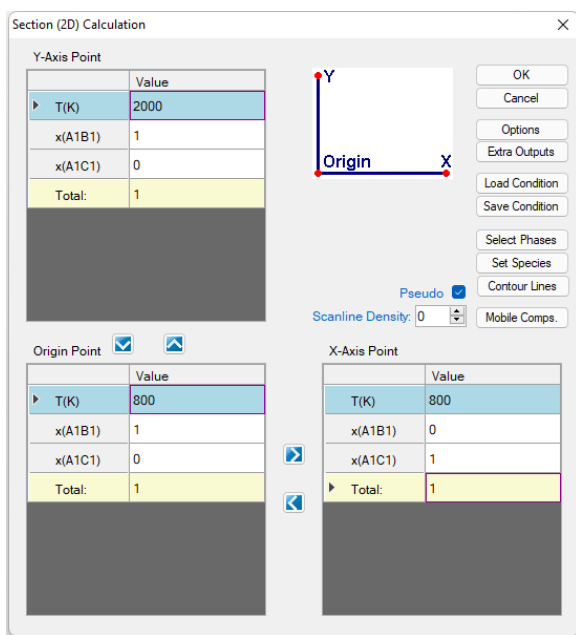
Figure 3.26: Set calculation condition for an AB-BC pseudo-binary section based on defined species
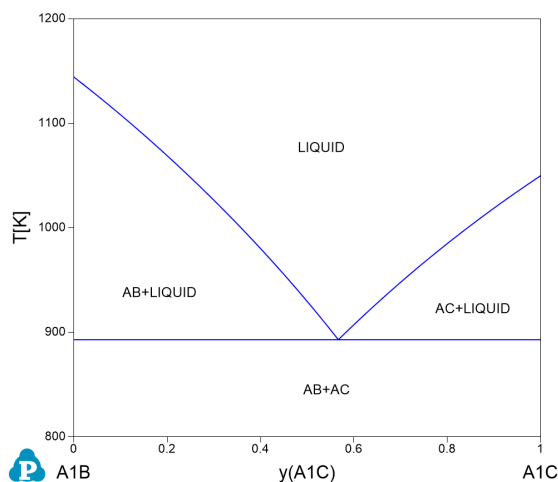
Figure 3.27: A calculated pseudo-binary section in ternary A-B-C system

3.3.5 Contour Diagram

Contour diagram shows how a property changes in a 2D or 3D diagram [2015Che]. The most commonly used contour diagram is the ternary isothermal lines superimposed on a liquidus projection. Another example is the activity contour diagram. Pandat™ extends contour diagram to many other properties including thermodynamic properties, physical properties and any combination of them. A few examples of useful contour diagrams are given in the following sections. Please keep in mind that Pandat™ can plot many other contour diagrams beyond these examples.

3.3.5.1 Activity Contour Diagram

Let's use the contour diagram of Mg activity in Al-Mg-Zn as the first example. Figure 3.28(a) shows the input window for calculating the isothermal section of Al-Mg-Zn at 500°C. Click on the "Contour Lines" button, a new window "Set Contour Lines" will pop out, as shown in Figure 3.28(b). Click on "Add" button and change the "Contour Type" to "a

(Mg:Hcp)", which means that the contour diagram is for the activity of Mg with reference state of Hcp. There are three initials related to which values of the contour lines to be calculated: "Start", "Stop" and "Step". In this example, "Start" and "Stop" can be left as empty and Pandat™ will search for all possible values. User can set values for "Start" and "Stop" for specific range of the contour lines. However, the value of "Step" must be given. Another condition "Constraints" will be explained later. Figure 3.28(b) sets the initial condition to calculate the contour lines for the activity of Mg referring to Mg in Hcp phase with the step size of 0.1. Click "OK" in this window and also in the "Section (2D) Calculation" window to proceed the calculation. The calculated isothermal diagram with the contour lines of Mg activity is shown in Figure 3.29 after labeling.

If user want to calculate the contour lines of the Mg activity only in the liquid phase, it can be achieved by setting a constraint in the contour diagram input condition. In Figure 3.28(b), click on "Contour Constraints", a small button ⊡ will show up. Click on it, A window of "Set Contour Constraints" will pop out as shown in Figure 3.30. Add "f (@Liquid)=1" as the constraint in this window and then click "OK" to return to "Set Contour Lines" window as in Figure 3.31. This will set a constraint on the contour line calculation that only the contour lines satisfying the constraint "f(@Liquid)=1" will be calculated. Figure 3.32 is the calculated phase diagram with the contour diagram of a(Mg) only in the liquid phase region in the isothermal section of Al-Mg-Zn at 500°C.

(a)

(b)
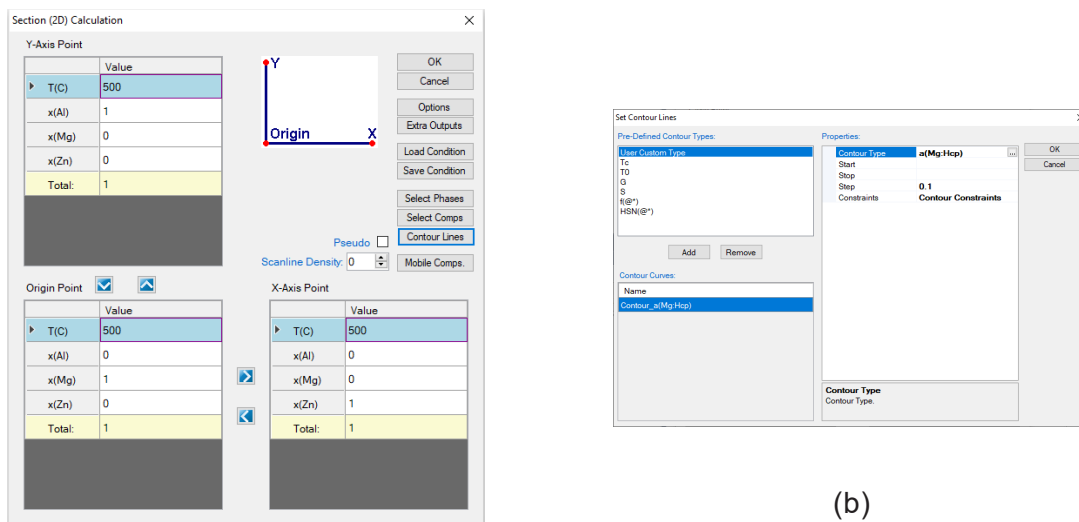
Figure 3.28: Input windows for contour diagram of activity of Mg at 500°C: (a) 2D Diagram input window; (b)Contour Diagram input window.
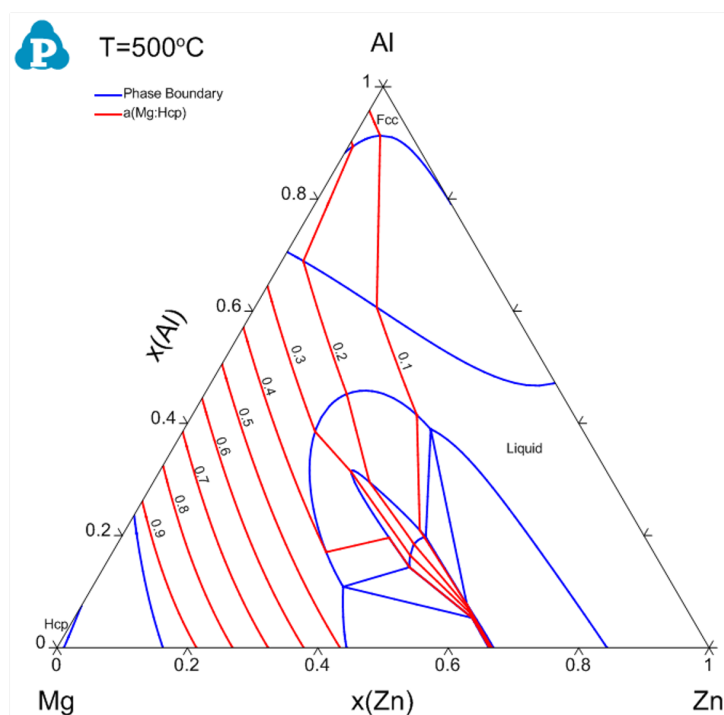
Figure 3.29: Contour diagram of Mg activity in the isothermal section of Al-Mg-Zn at 500°C
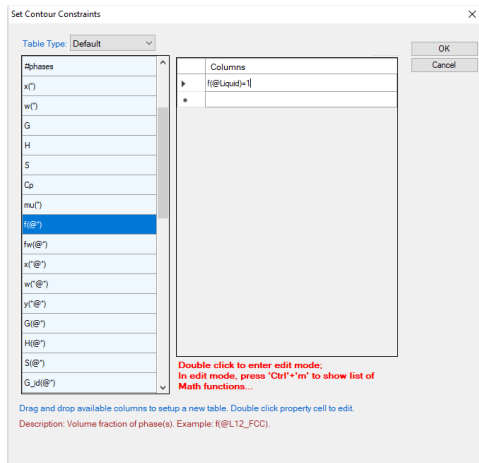
Figure 3.30: Set contour constraints
window
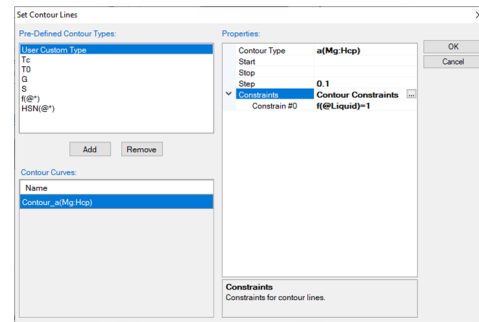
Figure 3.31: Set contour constraints
window
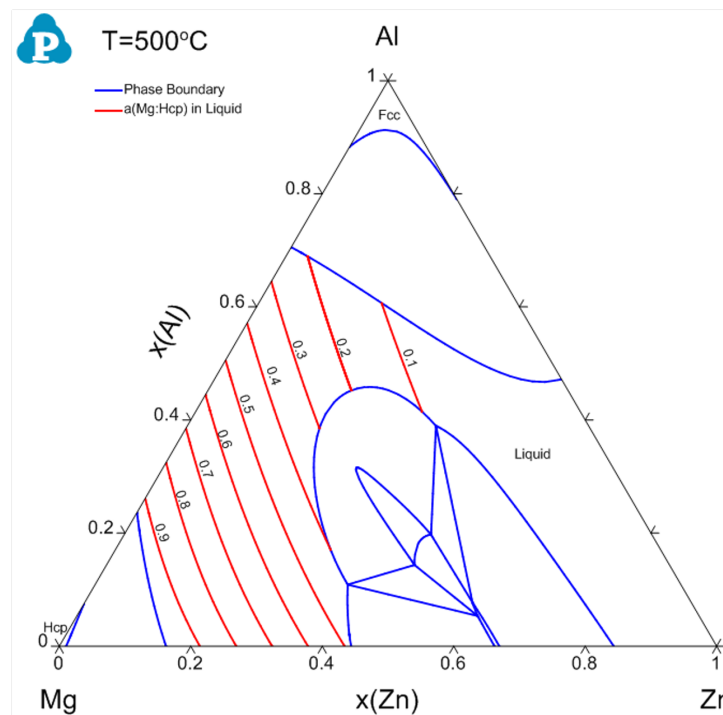
Figure 3.32: Contour diagram of Mg activity only in liquid phase in the isothermal section of
Al-Mg-Zn at 500°C

3.3.5.2 Driving Force Contour Diagram

Contour diagram is also useful in viewing the stability of a phase comparing to the
equilibrium state. Figure 3.34 shows the driving force contour lines of the `Hcp` phase in the
Al-Mg-Zn system at 500°C. The contour type is defined as `DF(@|Hcp)` as shown in Figure

3.33 (see Table 11.4 in Section 11.2 for the definition of `DF`). Since `Hcp` phase may not be stable in all the phase regions in this isothermal section, in between "`@`" and "`Hcp`" a vertical bar symbol "`|`" is added to specify that the phase `Hcp` is in the "`entered`" status. From Figure 3.34, we see that `Hcp` phase has less driving force to be stable in the central region of the compositional triangle.
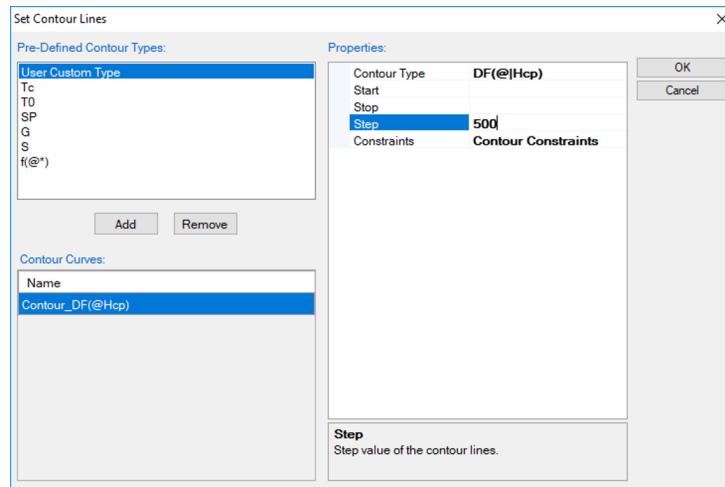
Figure 3.33: Set driving force contour
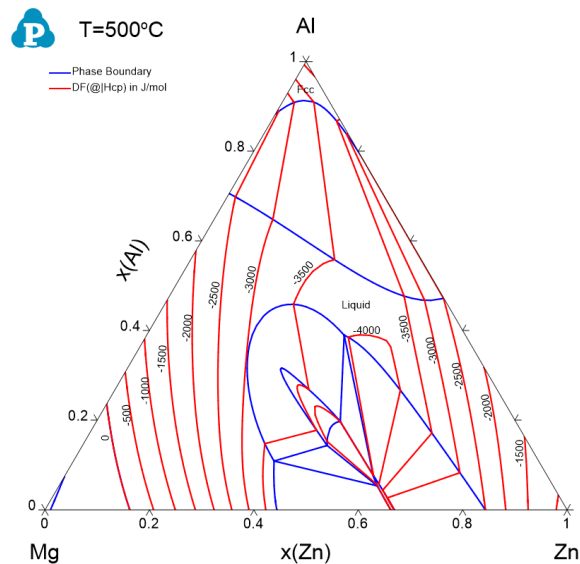
Figure 3.34: Contour diagram of the driving force of the Hcp phase relative to the equilibrium phases in Al-Mg-Zn at 500°C

3.3.5.3 Partial Molar Property Diagram

Partial molar properties such as partial molar enthalpy and entropy of a component can also be calculated as a contour diagram along with the phase diagram. The red curves in Figure 3.35 and Figure 3.36 are the calculated contour curves of the partial molar enthalpy and entropy of Zn in the liquid phase for the Al-Zn system. The constraints in both calculations are `f(@Liuid)>0` since the calculation is for the stable liquid phase only.
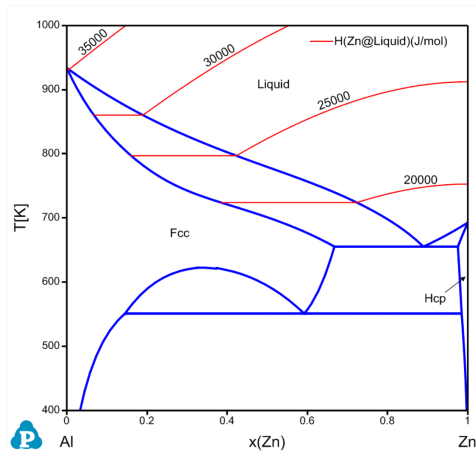
Figure 3.35: Contour diagram of the partial molar enthalpy of Zn in the liquid phase of Al-Zn system
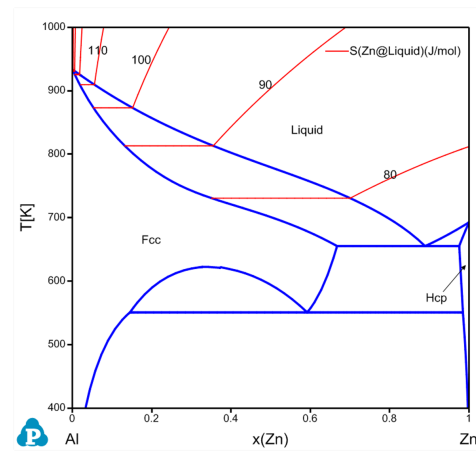
Figure 3.36: Contour diagram of the partial molar entropy of Zn in the liquid phase of Al-Zn system

3.3.5.4 Excess Molar Property Diagram

Sometimes we have interest in the excess properties, such as excess Gibbs free energy, excess enthalpy and excess entropy of a phase. These properties are available from the property definitions of `G_ex`, `H_ex` and `S_ex` in a contour diagram. Figure 3.37, Figure 3.38 and Figure 3.39 show the calculated contour curves for the excess properties of liquid in the Mg-Zn system.
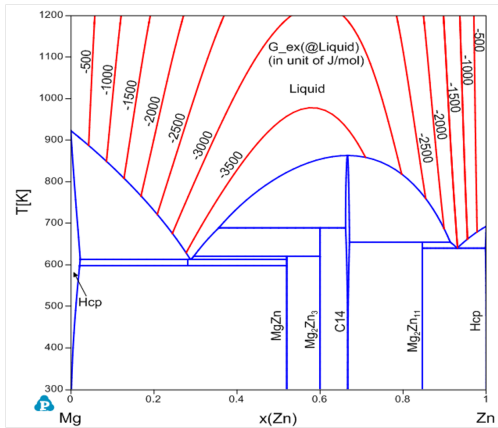
Figure 3.37: Contour diagram of the excess Gibbs free energy in the liquid phase of Mg-Zn system
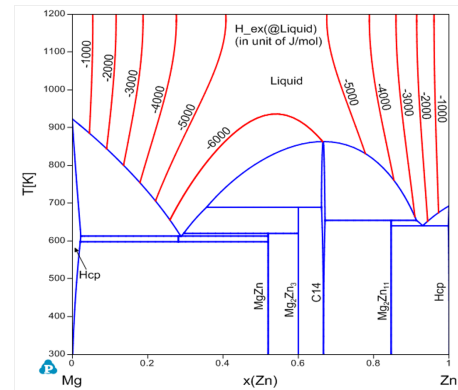
Figure 3.38: Contour diagram of the excess enthalpy in the liquid phase of Mg-Zn system
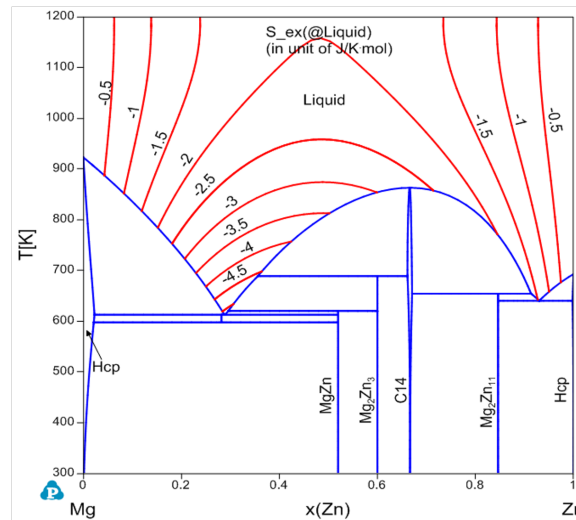
Figure 3.39: Contour diagram of the excess entropy in the liquid phase of Mg-Zn system

3.3.5.5 User Defined Contour Diagram

Contour diagram can be any type of customized properties defined by the user using a mathematical expression. Contour diagram can therefore be used to plot some special property lines. For example, we want to calculate Curie temperature `Tc` curves in the Fe-Cr system. A `Tc` curve can be considered as a special contour line when `T` equals to `Tc`, which means the constraint: `T-Tc=0` is set on the `Tc` curve. Figure 3.40 is the input window for calculating the `Tc` contour curves. Since the constraint is `T-Tc(@*)=0`, the "`Start`" and

"`Stop`" values are both set to be zero, and "`Step`" value is ignored. `@*` means in every phase. Figure 3.41 is the calculated Fe-Cr phase diagram with the `Tc` curves of the Bcc phase.
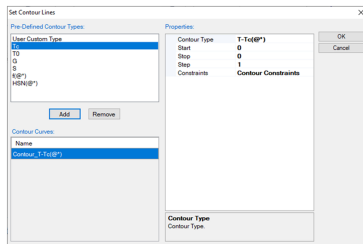
Figure 3.40: Input windows for `Tc` curves in Fe-Cr system
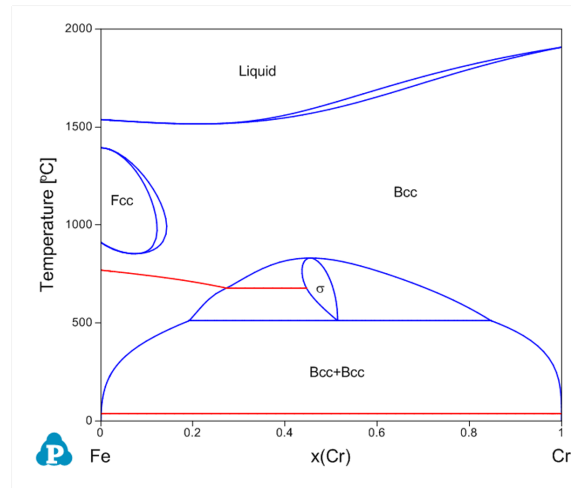
Figure 3.41: Phase diagram of Fe-Cr with Curie temperature curves (in red color) of Bcc phase

A second order transition curve could also be calculated as a contour curve. Fe-Al binary system [2009Sun] is given as an example here. Figure 3.42 is the calculated Fe-Al binary phase diagram with `Tc` curves and the second order transition curves. The `Tc` curves can be calculated the same way as that shown in this section for the Fe-Cr system.
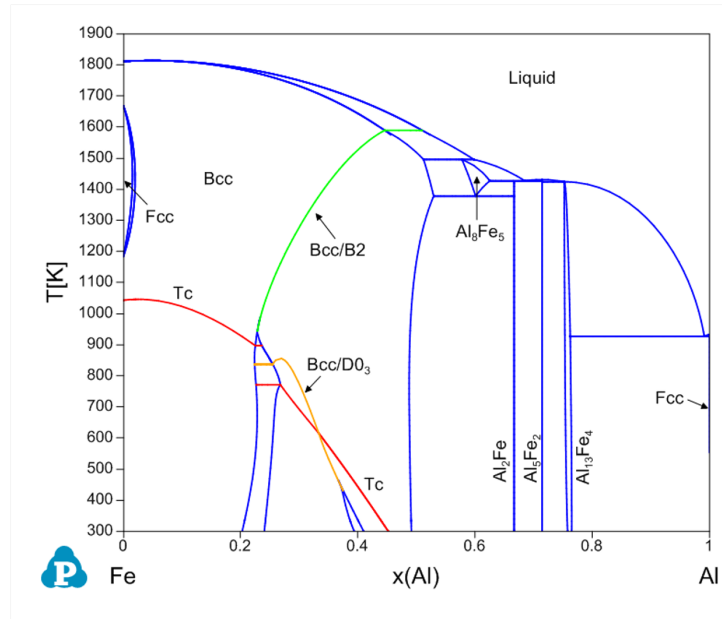
Figure 3.42: Phase Diagram of Fe-Al with the `Tc` curves and the 2nd order transition curves between Bcc and `B2` and between `B2` and `D03`

The calculation of the second order transition needs special contour property definition. For the Bcc/B2, the contour property is defined as

```
abs(y(Fe#2@BCC_4SL)-y(Fe#3@BCC_4SL))
```

which represents the absolute value of the difference between the site fractions of Fe on the 2nd and 3rd sublattices. The "`start`" and "`stop`" values are set to be "`0.05`", which avoids the numerical difficulty in calculation and gives a very good approximation of the order/disorder transition curves.

The contour property for the second order transition between `B2` and `D03` is defined as

```
abs(y(Fe#3@BCC_4SL)-y(Fe#4@BCC_4SL))
```

with the "start" and "stop" values of "0.05". A constraint is also added in this contour calculation to make sure that the first and second sublattices have the same site fractions,

```
abs(y(Fe#1@BCC_4SL)-y(Fe#2@BCC_4SL))<0.001
```

Pandat™ batch file is written in the language of XML (Extensible Markup Language). The less-than "<" and great-than ">" characters are reserved as the XML special characters. In an XML file, the less-than "<" and great-than ">" symbols are written as "`<`" and "`>`". Above constraint is written in a Pandat batch file as

```
abs(y(Fe#1@BCC_4SL)-y(Fe#2@BCC_4SL))&lt;0.001
```

3.3.5.6 Partial Pressure Contour Diagram

The following example shows how to calculate the contour lines for the partial pressure of the gas phase in the Ti-N system [1996Zen]. Figure 3.43 (a) is the input condition window and Figure 3.43 (b) is the calculated phase diagram of Ti-N with the contour lines of `log(P(N2@gas))` (pressure unit is Pa), which is the common logarithm of the partial pressure of N2 in gas. Figure 3.43 (c) is another input condition window and Figure 3.43 (d) is the calculated phase diagram of Ti-N with the contour lines of `log(P(@gas))`, which is the common logarithm of the total pressure of gas with the gas species `N`, `N2`, `N3` and `Ti`.

More examples can be found in the Pandat™ example folder: `/PanPhaseDiagram/Contour/`
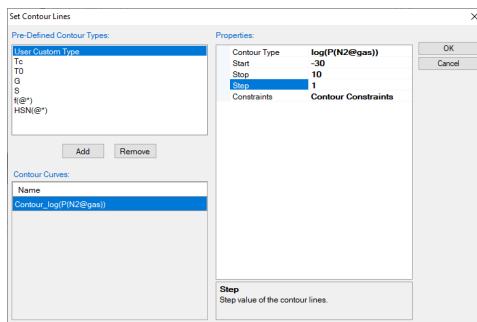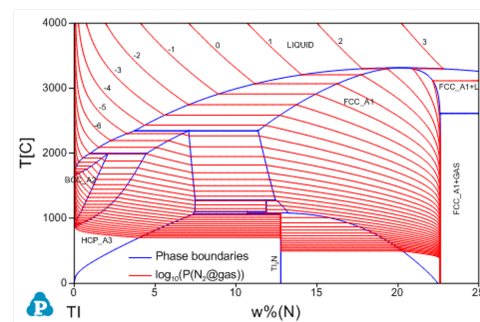
(a) Input condition   (b) log(P(N2@gas)) contour
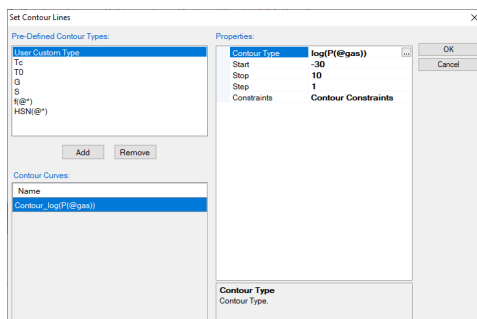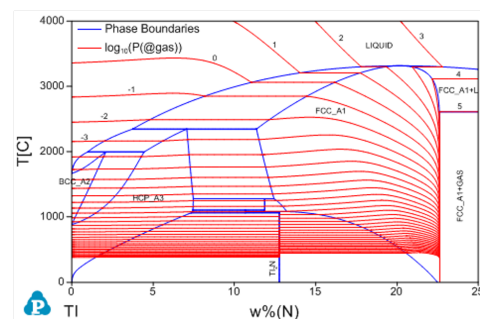
(c) Input condition   (d) log(P(@gas)) contour

Figure 3.43: Contour diagrams of partial pressure of gas in the Ti-N system

3.3.6 Phase Projection

This function permits the calculation of a phase projection diagram for a system with two or more components. PanEngine automatically searches the univariant phase projection lines.

Go to **PanPhaseDiagram** on the menu bar and select **Phase Projection** or Click the ⚠ button on the tool bar. The **Phase Projection** dialog box shown in Figure 3.44 allows the user to calculate projection of any phase in the system. If "Liquid" is selected as Target Phase, liquidus projection for the system is calculated. Isothermal curves involving the selected phase in a ternary system can also be calculated by checking the box **Calculate Isotherms**. The density of isothermal curves depends on the scale of **Temperature Interval**. A larger value for the temperature interval leads to fewer isothermal curves but with faster speed of calculation. The default **Compositional Range** is full range for each component. User can define the default output graph by selecting the X and Y axis. User can also access the **Options**, **Extra Outputs**, **Load Condition**, **Save Condition**, **Select Phases**, and **Select Comps** windows through this dialog box. If the **Show Results for Subsystems** box is checked, the output files will include the results for the subsystems together with the multicomponent system. Be aware that the output table will be huge if the box is checked for a calculation with a higher order multi-component system. User can choose the specific phase for projection calculation in the **Target Phase** dialog, such as Fcc phase. If "*" is selected, the projection for all the phases will be calculated. If the **Show 3D Diagram box** is checked, there will be an extra 3D diagram showing the calculation results as well as the original 2D diagram. For the 3D diagram, user can rotate the diagram by holding the left button of mouse and move it around.
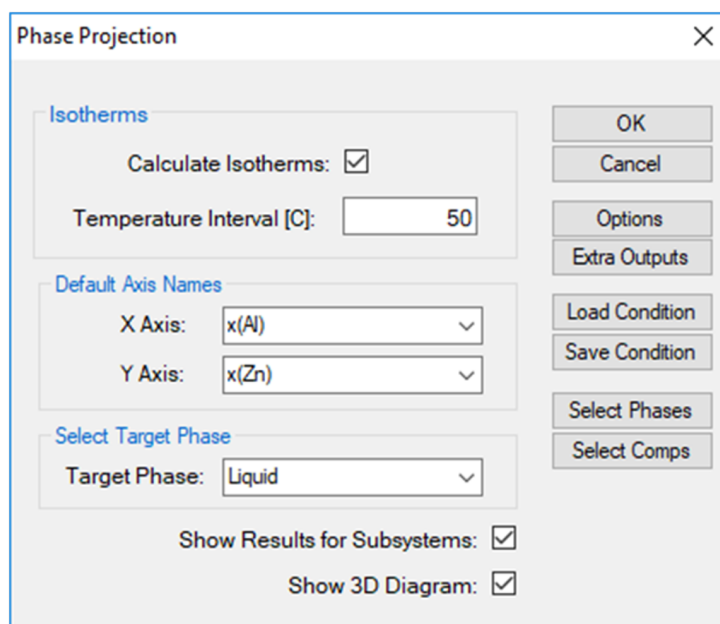
Figure 3.44: Phase projection calculation setting dialog

The results from a liquidus projection calculation are displayed in two types of format in the PandatTM main window: Graph and Table. Figure 3.45 to Figure 3.48 show the 2D and 3D graph view, default table and isotherm table for the liquidus projection calculation results, respectively. user can switch between Graph view to Table View by double clicking on the graph name or the table name in the PandatTM Workspace window. Extensive operations on Graph and Table, such as labeling and adding legend, have been discussed in detail in Section 2.4 and Section 2.5 .
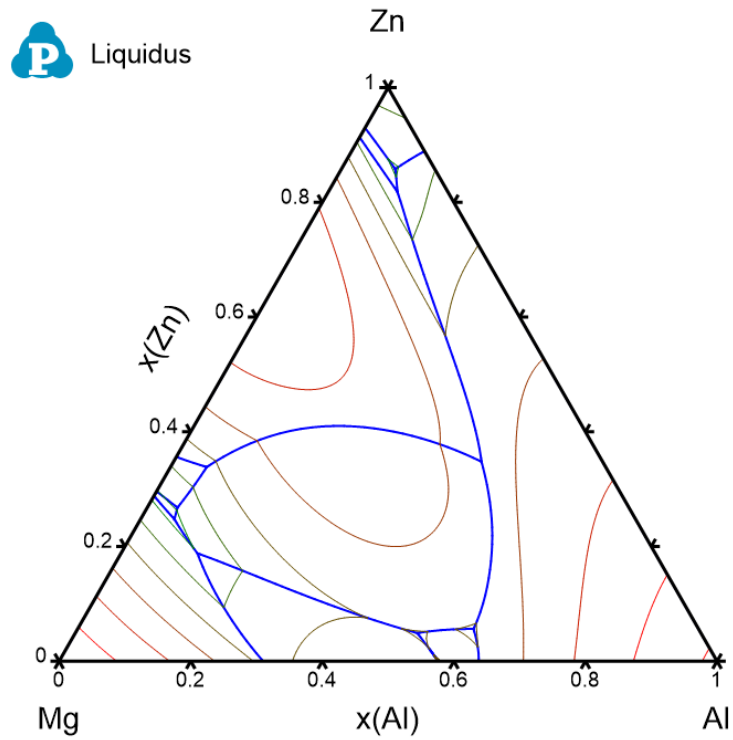
Figure 3.45: Liquidus projection in 2D of the Al-Mg-Zn ternary system
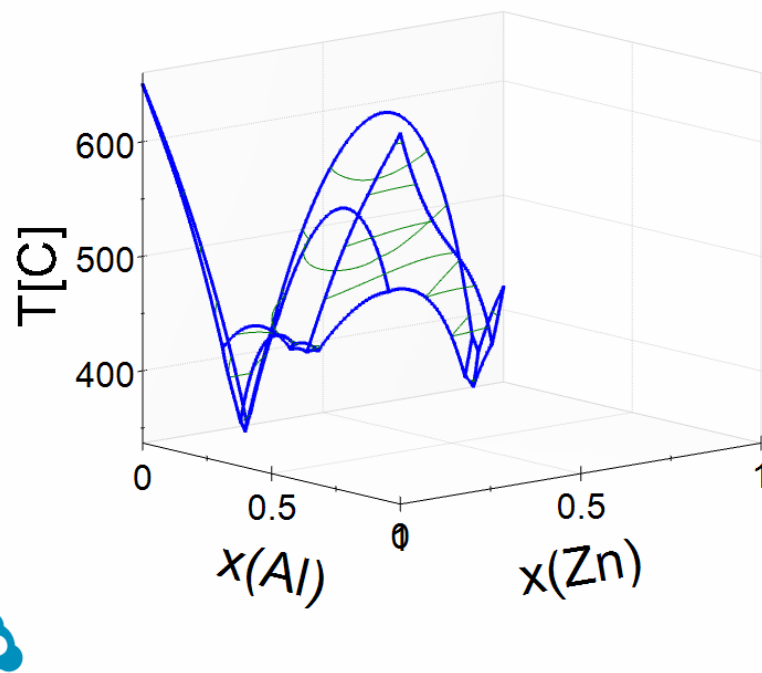
Figure 3.46: Liquidus projection in 3D of the Al-Mg-Zn ternary system

Figure 3.47: Phase projection calculation setting dialog

Figure 3.48: Phase projection calculation setting dialog

3.3.7 Solidification Simulation

Two simple models, equilibrium solidification and Scheil solidification, have been integrated into Pandat™ for calculating solidification pathways. The equilibrium model, also called the lever rule model, assumes that complete diffusion occurs in both liquid and solid phases,

and the compositions of solid and liquid always follow the phase boundaries defined by the equilibrium phase diagram. The fractions of liquid and solid can be calculated through the lever rule. In contrast, Scheil solidification assumes that no diffusion occurs in the solid phases, that the composition of liquid phase is uniform (infinite diffusivity in liquid) and that local equilibrium at the solid-liquid interface is always maintained. The `Solidification Simulation` function calculates the solidification path of an alloy using either the `lever rule` or `Scheil` model as being decided by the user.

Go to `PanPhaseDiagram` on the menu bar and select `Solidification Simulation` or click the button on the tool bar. The dialog box as shown in Figure 3.49 allows the user to set up the alloy composition for the solidification simulation. User can choose the simulation under either `Non-Equilibrium (Scheil)` or `Equilibrium (Lever Rule)` condition. There are two check boxes at the bottom of the input dialog: `Start simulation from liquidus surface`, and `End when no more liquid`. If both boxes were checked, the solidification simulation will be carried out in the temperature range when solid start to form from liquid (liquidus surface) to the point when liquid just completely disappear. Pandat™ software will find these two points automatically. If one or both of the check boxes are not checked, user will need to define the start or/and the end temperature for the solidification simulation proceeds.

Figure 3.49: Set calculation conditions for solidification simulation

The results are displayed in either Graph View or Table View. Figure 3.50 and Figure 3.51 show the graph view and table view for the solidification simulation results with the calculation condition shown in Figure 3.49. Extensive operations on Graph and Table, such as labeling and adding legend, have been discussed in detail in Section 2.4 and Section 2.5 .
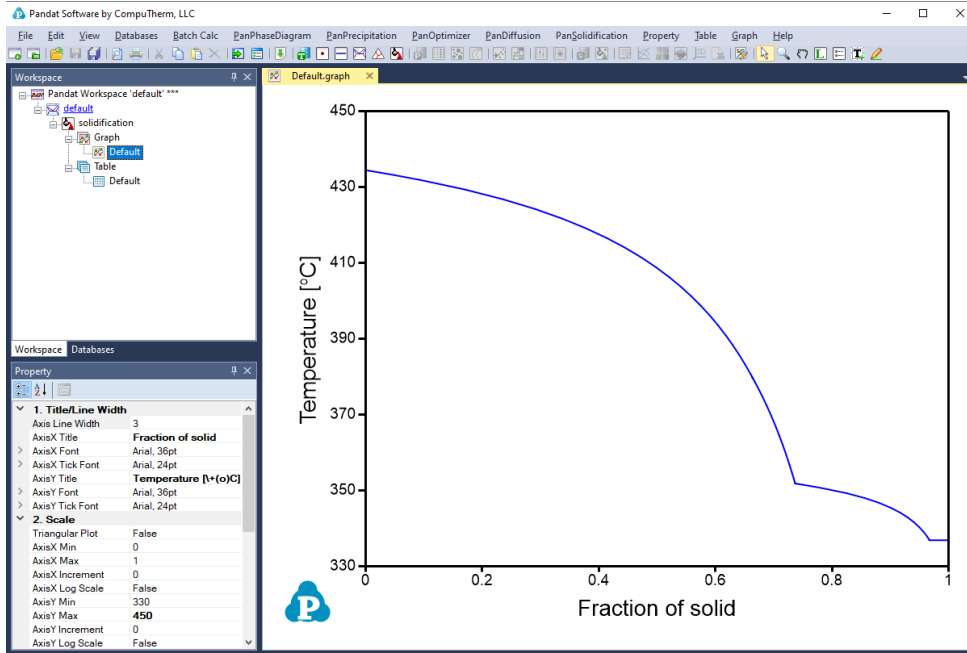
Figure 3.50: Graph view of the solidification simulation result

Figure 3.51: Table view of the solidification simulation result

PanSolidification module considers the back diffusion in the solid phase during solidification. Detailed description on **PanSolidification** module is in Chapter 7 PanSolidification.

3.3.8 Table Column Functions

Examples are given in this section show how to use the table column property names and functions to obtain more information from the calculated results.

3.3.8.1 Activity and Activity Coefficient

The activity of a component j, a_j, is defined by

$$\mu_j = \mu_j^\circ + RT \ln a_j \qquad (3.13)$$

where μ_j is the chemical potential of the component j in equilibrium state and μ_j° is the chemical potential of this component at its reference state. For example, the activity of component `Al` in the `liquid Al-Mg` system referring the `Al` in `Fcc` phase is calculated by

$$a_{Al} = e^{\frac{\mu_{Al}^{liquid} - \mu_{Al}^{\circ,fcc}}{RT}} \qquad (3.14)$$

The corresponding Table column name is `a(Al@Liquid:Fcc [Al])`, or `a (Al@Liquid:Fcc)`.

If a reference state is not specified in Table column name, the default reference state in the database is taken as the reference state. For example, `a(Al@Liquid)` is calculated by

$$a_{Al} = e^{\frac{\mu_{Al}^{liquid}}{RT}} \qquad (3.15)$$

Activity coefficient is defined as

$$\gamma_j = \frac{a_j}{x_j} \qquad (3.16)$$

which is available from Table by defining Table column name similar to that of activity. Table column name for the activity coefficient of `Al` in `liquid` is `r(Al@Liquid:Fcc [Al])`, or `r(Al@Liquid:Fcc)`.

Figure 3.52 to Figure 3.55 show the screen images for creating a table of activity and activity coefficient from a line calculation result at 1000K in the Al-Mg-Zn ternary system. The two end points are at `x(Mg)=0.2,x(Al)=0.8` and `x(Mg)=0.2,x(Zn)=0.8`. The

liquid phase is the stable phase under this condition. The setting in Figure 3.52 is to choose `Fcc Al, Fcc Mg,` and `Hcp Zn` as the reference state for the activities and activity coefficients.
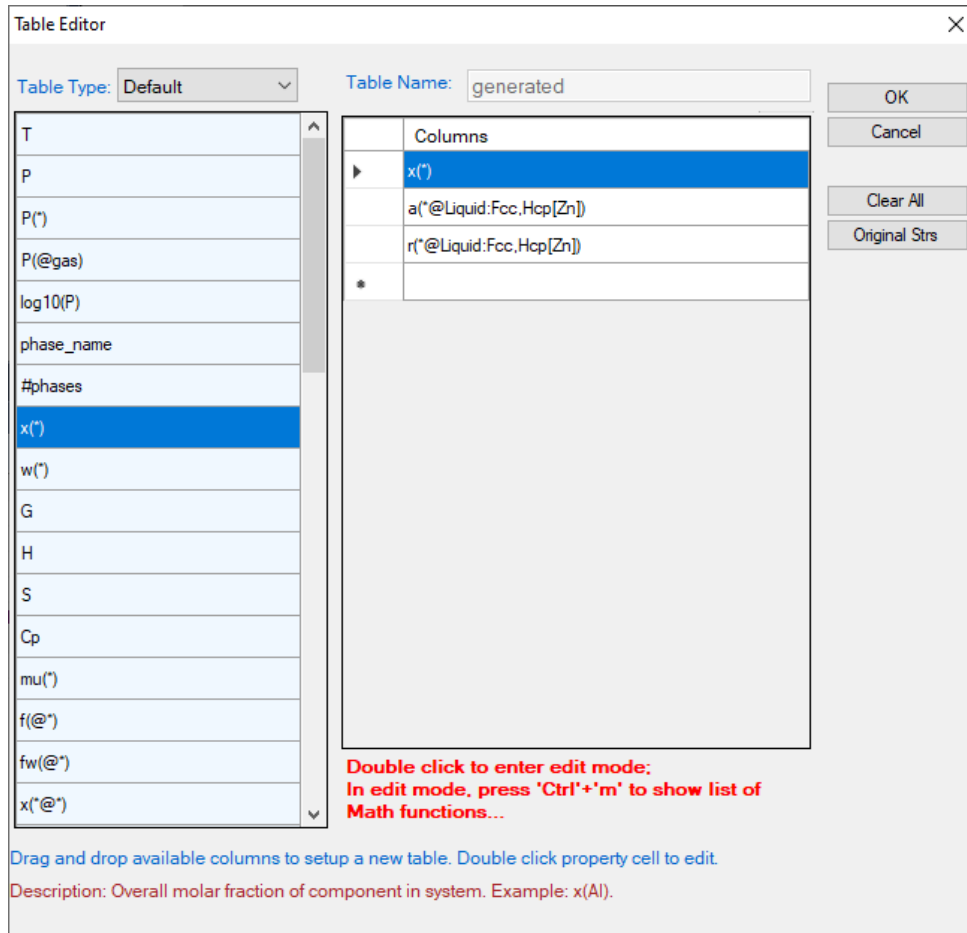
Figure 3.52: Table editor for creating activity and activity coefficient table

Figure 3.53: Table of activity and activity coefficient
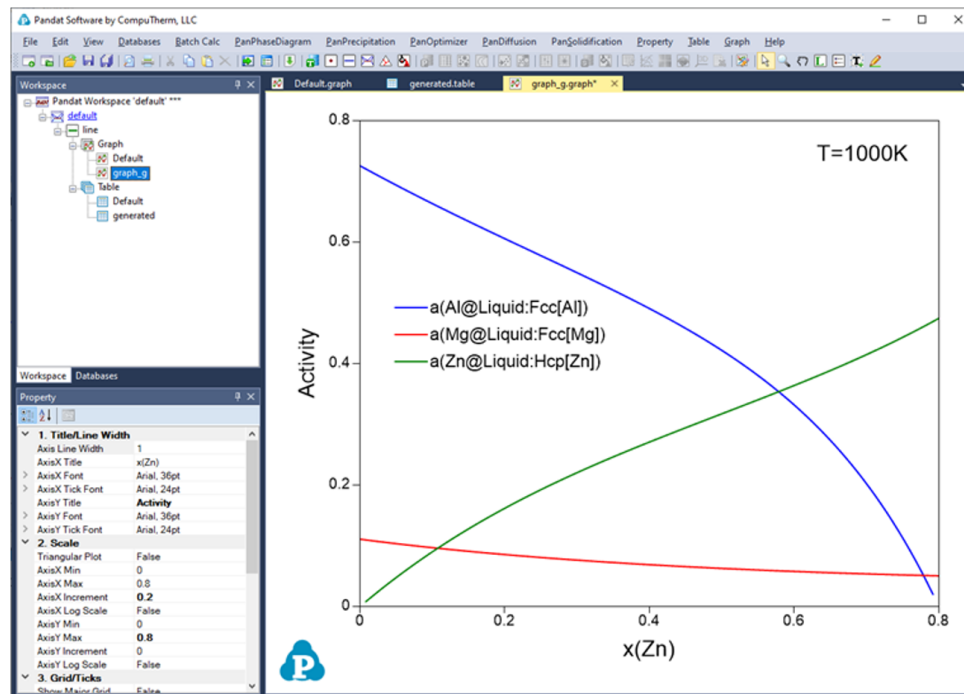
Figure 3.54: Graph of activity vs. x(Zn)
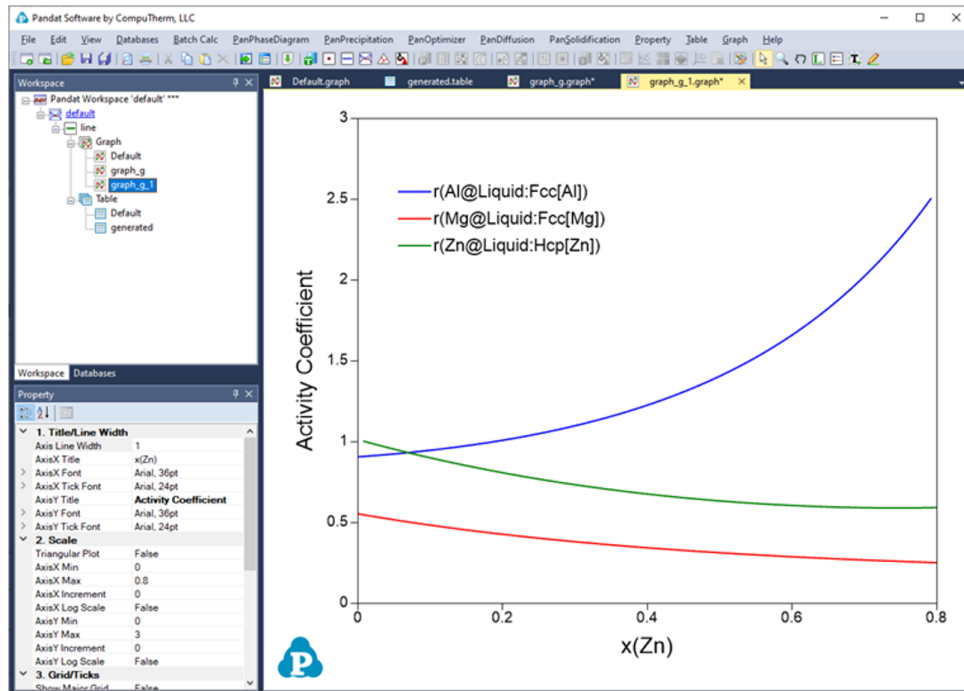
Figure 3.55: Graph of activity coefficient vs. x(Zn)

3.3.8.2 Tielines

A special column property name "`tieline`" is introduced to create a table of tielines from the calculated results. It can be combined with other column constraints to select a special set of tielines.

Figure 3.56 is a 3D ternary phase diagram calculated from the database file "`ABC.tdb`" which is provided in the Pandat™Examples folder. This system has three binary eutectic reactions. To show the invariant tielines in these binaries, set the "`Show Invariant Tieline`" in the Property window as "`true`", then the invariant lines will be plotted on it as shown in Figure 3.57.
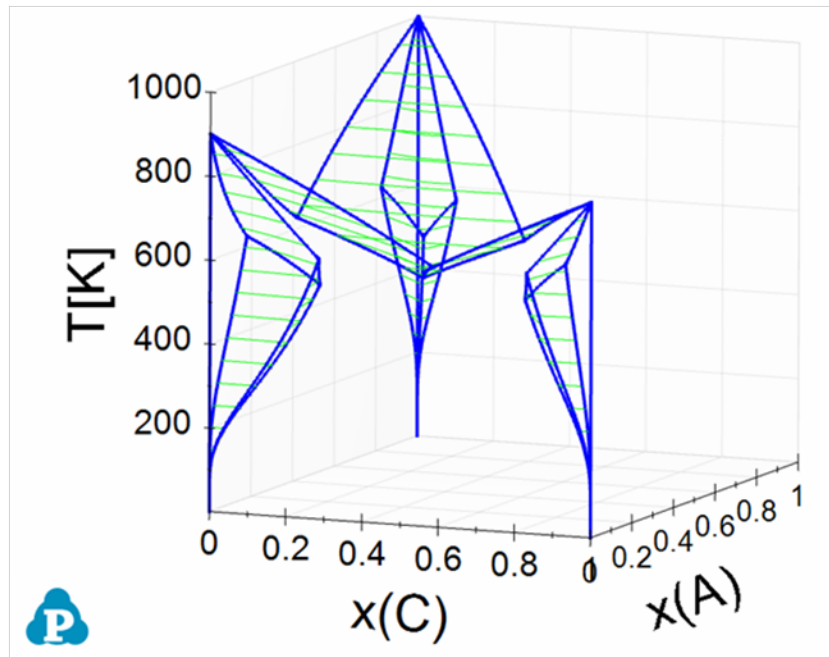
Figure 3.56: 3D phase projection of a simple ternary system with eutectic reaction in every binary system
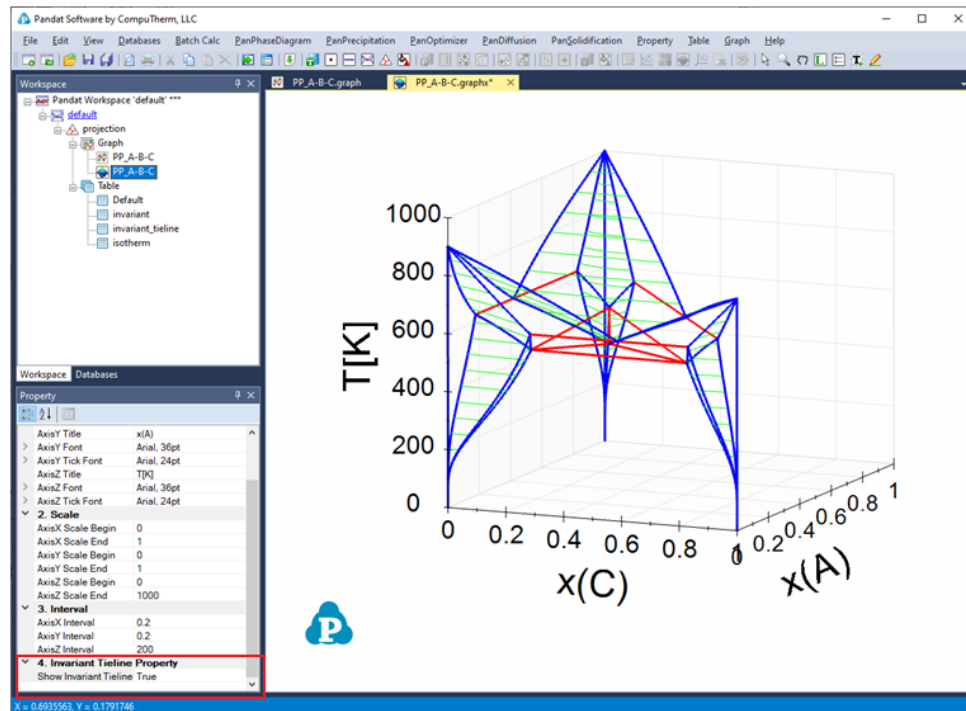
Figure 3.57: 3D phase projection with three binary eutectic and one ternary eutectic tielines shown as red

The ternary univariant lines (`L+Bcc+Fcc`, `L+Bcc+Hcp`, and `L+Fcc+Hcp`, red color) connects these reactions can be highlighted as follows. First, create a new table as shown in Figure 3.58. The purpose of the new table is to extract the boundary line(s) that connect Liquid, and the other two solid phases, and the fraction of Liquid on this boundary line is 1. After the table is created, select this new table in the Explorer window, and drag `x(C)` from the `Property window` to the `Main Display window`, and drop it as the `x-axis`; press "`Ctrl`" and then drag `x(A)` from the `Property window` to the `Main Display window`, and drop it as the `y-axis`; press "`Shift`" and then drag `T` from the `Property window` to the `Main Display window`, and drop it as the `z-axis`. This line showing the gradually change from the peritectic reaction to the eutectic reaction is then highlighted (green color), as shown in Figure 3.59.
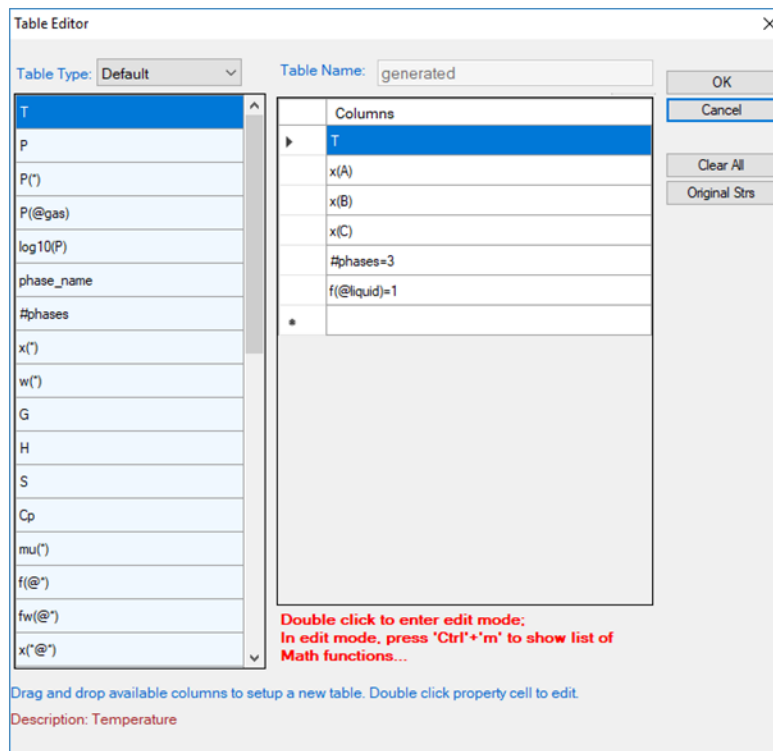
Figure 3.58: Create a new table to extract the ternary univariant lines
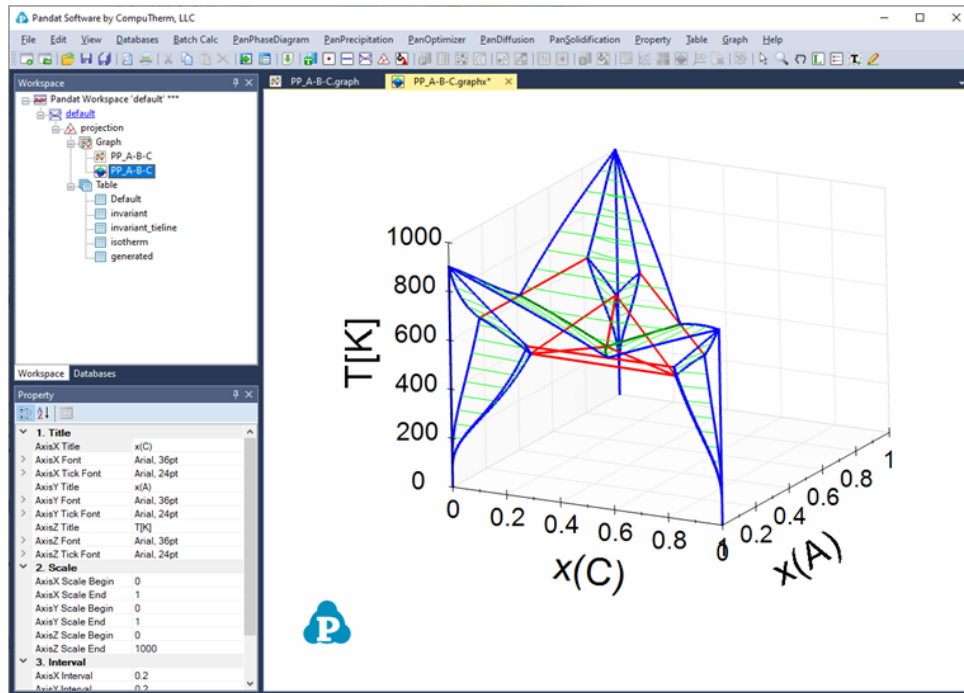
Figure 3.59: The ternary univariant lines (dark green lines) connecting the invariant

reactions in the binaries

Figure 3.60 shows how to create 4 new tables to plot the 3D diagram in Figure 3.61, which shows the ternary three-phase equilibrium tie-triangle volume. The 4 tables include a set of the tie-triangles by setting its density as "tieline=5" and the phase lines for the liquid, Bcc, Fcc phases by setting the fraction of the corresponding phase to be 1.
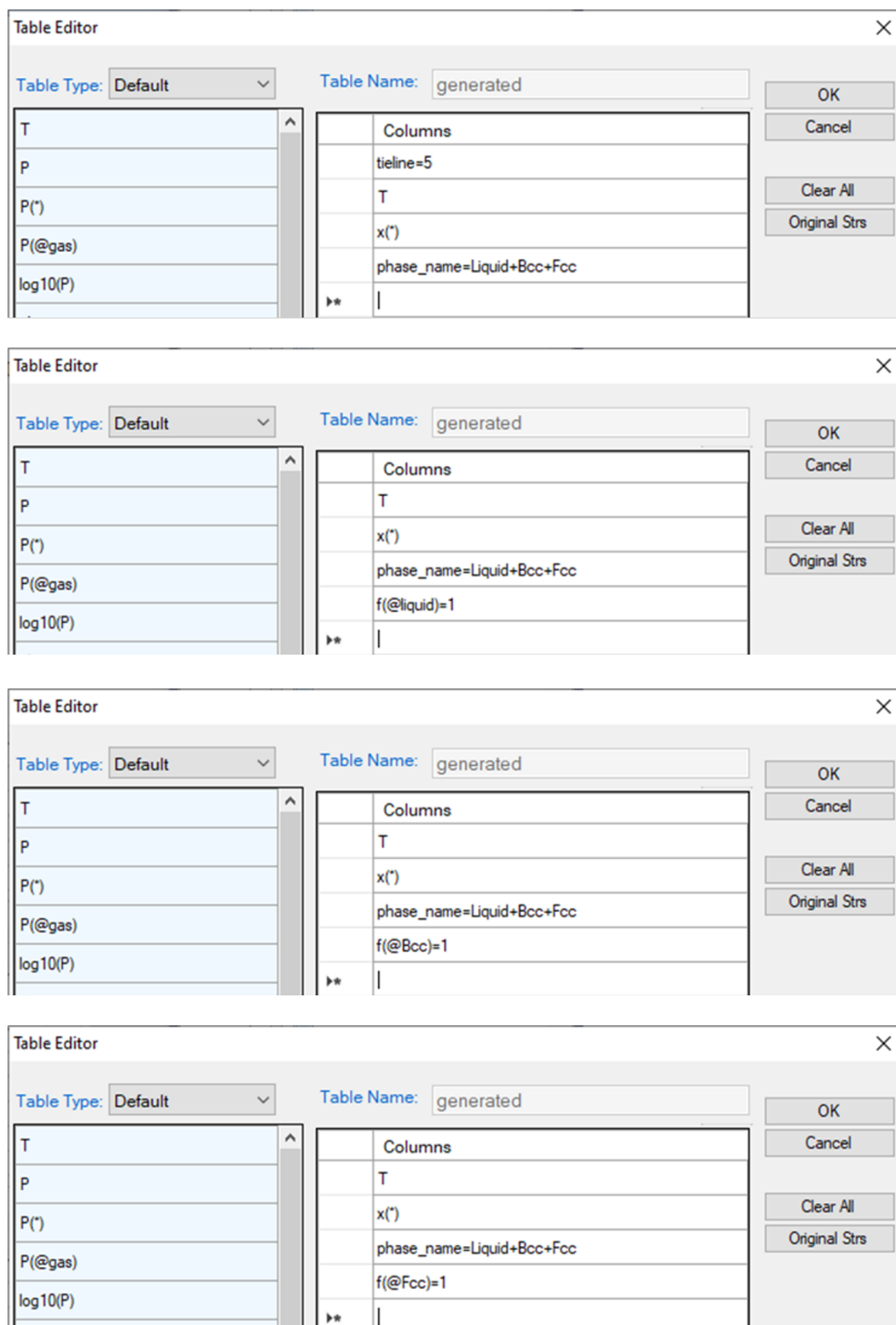
Figure 3.60: Create tables to extract tie-triangles and the three edges of the three-phase volume
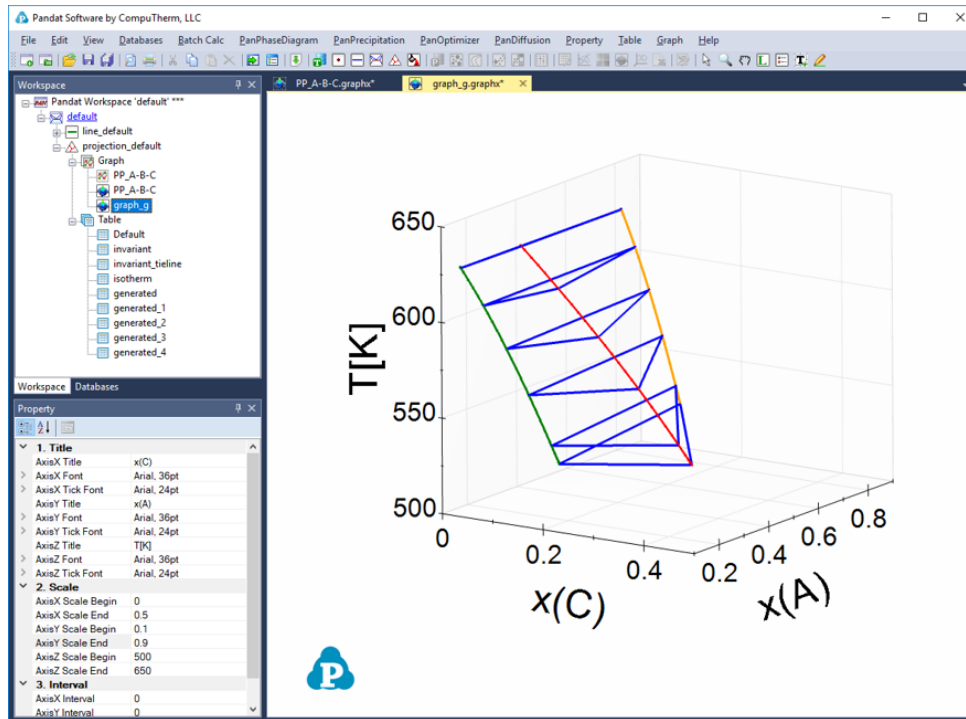
Figure 3.61: 3D diagram of the three-phase volume

3.3.8.3 Numerical Derivative

The derivative $\frac{dY}{dZ}$ can be calculated numerically from the two columns of Y and Z. The operator for numerical derivative is "//", double slashes. The numerical derivative of $\frac{dY}{dZ}$ is written in the form of "$Y//Z$" as the column name. Only one numerical derivative operator is allowed in one column. In other words, user cannot type in "$Y//Z//X$".The derivative "$Y//Z$" will be parsed into three columns in the new table: "Y", "Z" and "$Y//Z$", which makes it easy for the user to view the original data set "Y", "Z" and choose to plot either $\frac{dY}{dZ}$ vs Z or $\frac{dY}{dZ}$ vs Y.

The example given here is to find the "effective heat capacity" of the system during solidification ($H\_tot//T$). A system of Al-Mg-Zn is chosen and the composition is shown in Figure 3.62. After Scheil simulation is done, a new table is created with the definition of the column names shown in Figure 3.63. Figure 3.64 is the newly generated table. Select the columns "T" and "$H\_tot//T$", we have the $\frac{dH_{tot}}{dT}$ vs T diagram as in Figure

3.65. This diagram shows the effective heat capacity change during the solidification by Scheil simulation. There are two peaks in Figure 3.65, which represent roughly the phase transformation from liquid to `Hcp+γ` and that from liquid to `Hcp + τ`, respectively.
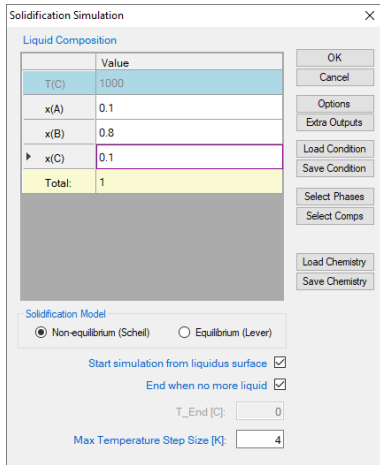
Figure 3.62: Solidification simulation condition
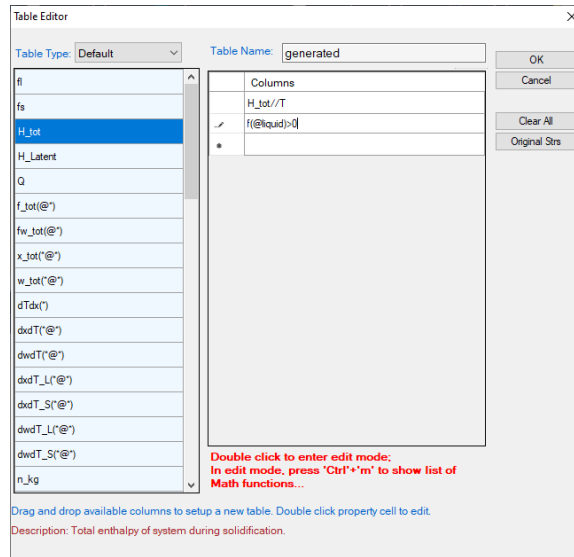
Figure 3.63: Create a table for the numerical derivative of H_tot w.r.t. T.

Figure 3.64: The table for the numerical derivative of H_tot w.r.t.T.
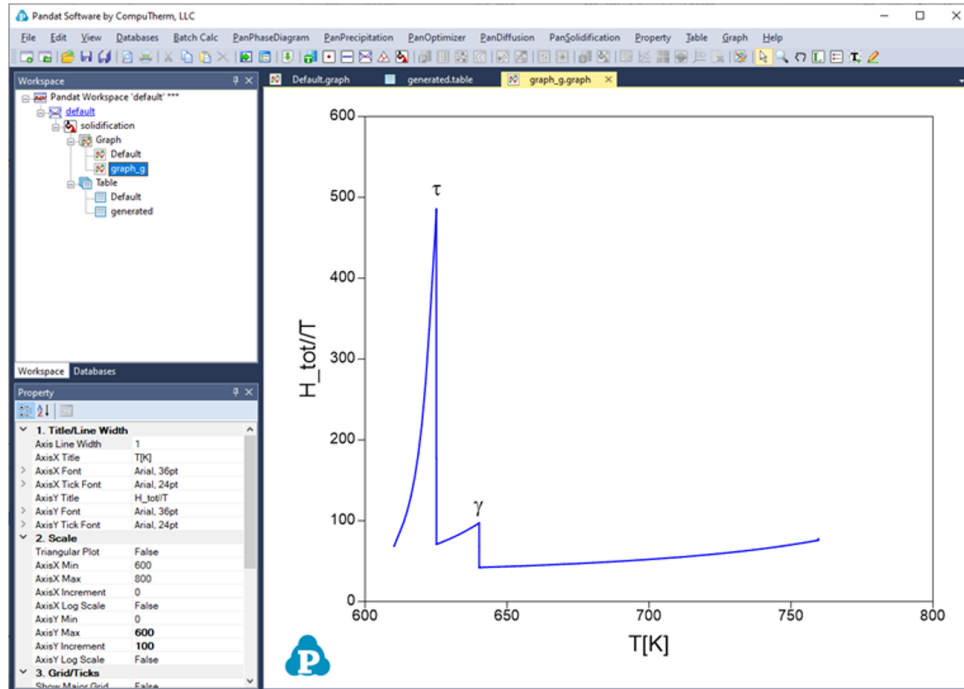
Figure 3.65: Plot of effective heat capacity change during the solidification

3.3.9 Append Database

On top of the original database (`*.tdb` or `*.pdb`) loaded from Pandat GUI, user can append a custom-made database (`*.tdb`) by select the **Append TDB** function from the **Databases** menu as shown in Figure 3.66.
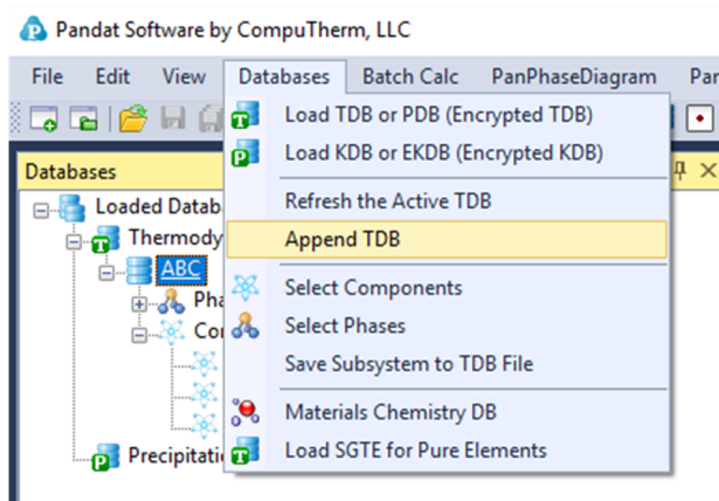
Figure 3.66: Append TDB function under the Databases menu

Using this function, user can (1) replace the value of an existing parameter, (2) add value to an existing parameter, (3) add new parameters to an existing phase, (4) add new phases to the original database, and (5) add user-defined properties. In the following, the hypothetical A-B system will be used as an example to explain the **Append TDB** function in detail:

3.3.9.1 Replace the Value of an Existing Parameter

In this example, we are going to replace the interaction parameter of the liquid phase: G (Liquid,A,B;0) within the original AB_original.tdb database. Load the AB_original.tdb from the **Databases** menu by selecting the **Load TDB or PDB** function. The interaction parameter of G(Liquid,A,B;0) described in the original database is expressed as:

    Parameter G(Liquid,A,B;0) 298 3000; 6000 N !

In the TDB Viewer, one can see the original value of G(Liquid,A,B;0)=3000.

Name	Property	x-Term	x-order	Parameter	T-limit (K)
	L	(A,B)	0	3000	6000

In the AB_replace parameter.tdb, only the interaction parameter for G (Liquid,A,B;0) is defined, but with a different value:

    Parameter G(Liquid,A,B;0) 298 -2000; 6000 N !

Load the AB_replace parameter.tdb via the **Append TDB** function, one can see from the TDB Viewer that the interaction parameter of G(Liquid,A,B;0) is replaced with the value (-2000) from the appended database.

Name	Property	x-Term	x-order	Parameter	T-limit (K)
Liquid	L	(A,B)	0	-2000	6000

The calculated phase diagrams using both original database and original + appended databases are shown in Figure 3.67.
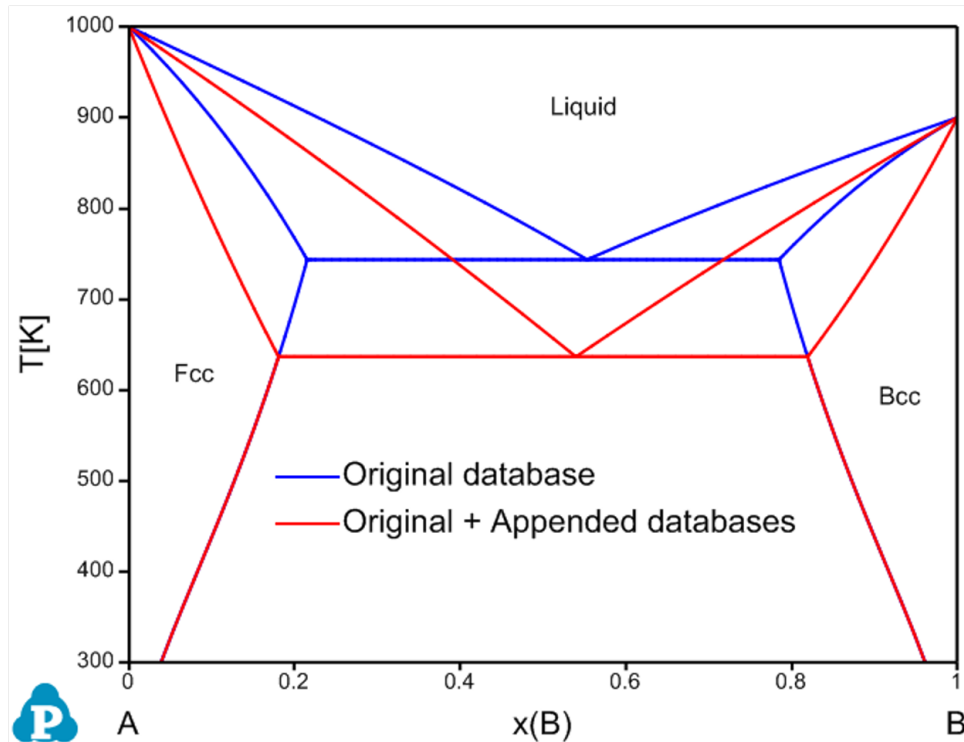
Figure 3.67: Calculated A-B phase diagram using both original database and original + appended databases with adjusted parameters

3.3.9.2 Add Value to an Existing Parameter

In this example, we are going to modify the interaction parameter of the liquid phase `G (Liquid,A,B;0)` within the original `AB_original.tdb` database by adding a value to it. As shown in the `AB_modify parameter.tdb`, the interaction parameter is expressed as `GG(Liquid,A,B;0)`, which means adding this assigned value to the original value rather than replacing it.

```
ParameterGG(Liquid,A,B;0) 298 -2000; 6000 N !
```

As shown in the above Section 3.3.9.1 , the original value of the `G(Liquid,A,B;0)` interaction parameter is +3000. When we append the `AB_modify parameter.tdb` to the original database, the value of -2000 will be added to the original value +3000 for the `G (Liquid,A,B;0)` interaction parameter. As shown in the TDB viewer below, one extra term "`GG`" is listed. For this case, the total value of the interaction parameter of `G (Liquid,A,B;0)` will be modified to be 3000 + (-2000) = +1000.

Name	Property	x-Term	x-order	Parameter	T-limit (K)
Liquid	L	(A,B)	0	3000	6000
	GG	(A,B)	0	-2000	6000

The calculated phase diagrams using both original database and `original + appended` databases are shown in Figure 3.68.
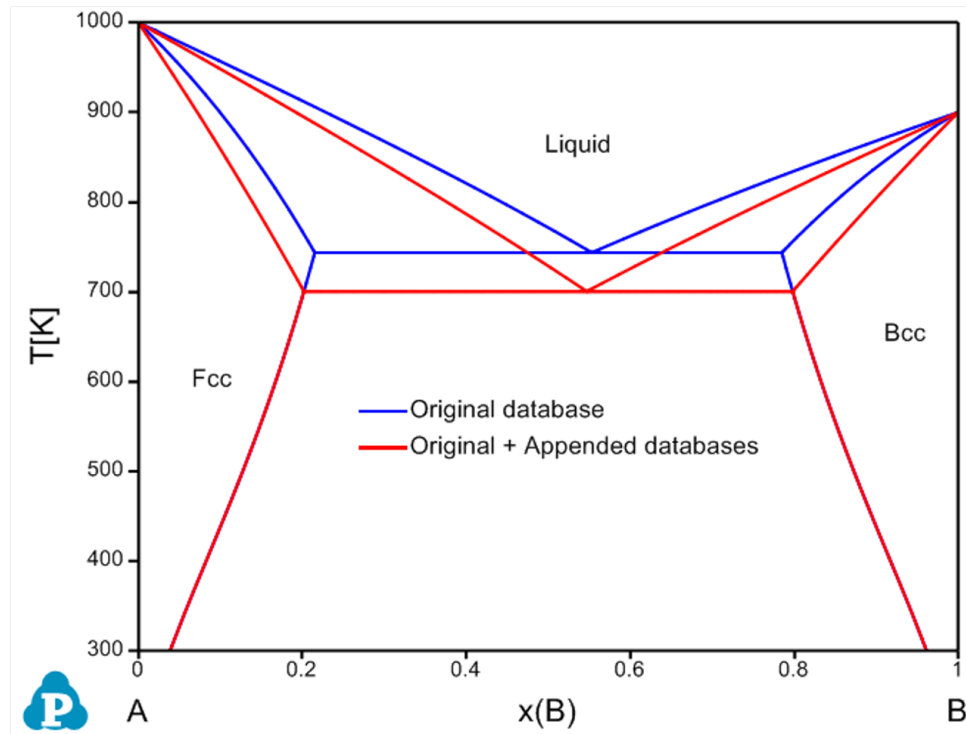
Figure 3.68: Calculated A-B phase diagram using both original database and original + appended databases with adjusted parameters

Note that, the `Append TDB` function allows user to append only one database to the original database. When user wants to append another database to the original database, the previously appended database will need to be removed first. Pandat™ will notify the user as shown in Figure 3.69 and user need to click `Yes` to confirm.
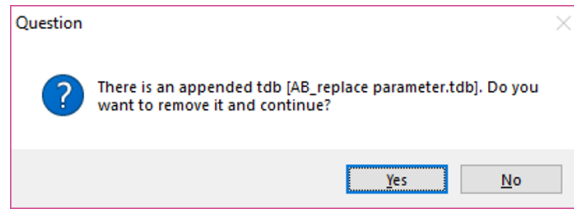
Figure 3.69: Append database confirmation

3.3.9.3 Add New Parameter to an Existing Phase

In addition to replace or modify the existing parameters for an existing phase in the original database, one can also add new parameters to this phase. In the `AB_Original.tdb` database, there is only one interaction parameter for the Liquid phase `G(Liquid,A,B;0)`. Using the **Append TDB** function, we can add more interaction parameters to the Liquid phase. As shown in the `AB_new parameter.tdb`, another interaction parameter for the liquid phase, `G(Liquid,A,B;1)`, is given as:

    ParameterG(Liquid,A,B;1) 298 -2000; 6000 N !

Load the original `AB_original.tdb` and then load the `AB_new parameter.tdb` via the **Append TDB** function. As shown in the TDB viewer, the new interaction parameter (x-order value is 1) is added.

Name	Property	x-Term	x-order	Parameter	T-limit (K)
Liquid	L	(A,B)	0	3000	6000
	L	(A,B)	1	-2000	6000

The calculated phase diagrams using both original database and original + appended databases are shown in Figure 3.70
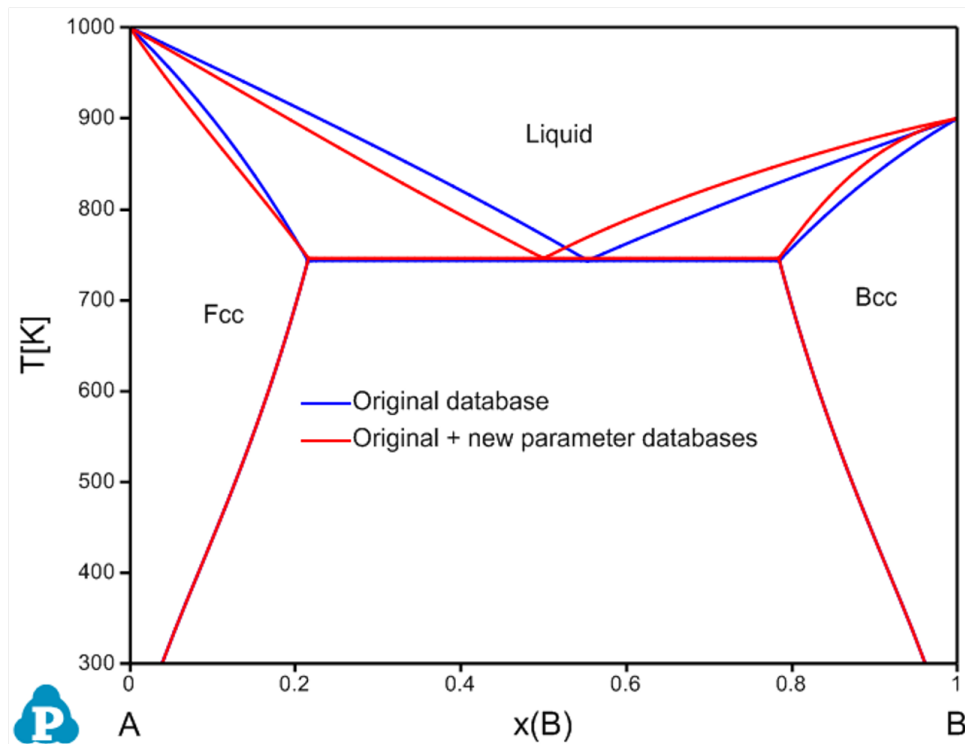
Figure 3.70: Calculated A-B phase diagram using both original database and original + appended databases with new parameters

3.3.9.4 Add New Phases to the Original Database

In addition to modify the parameters for existing phases within the original database, one can also add new phases via the **Append TDB** function. A new phase AB is introduced in the AB_new phase.tdb, which is described as:

```
Phase AB % 2 0.5 0.5 !
Constituent AB :A:B:!
    Parameter G(AB,A:B;0) 298.15 -10000+6*T; 6000 N !
```

Load the AB_original.tdb first and then load the AB_new phase.tdb via the **Append TDB** function. As shown in the TDB Viewer, a new AB phase is introduced.

Name	Property	x-Term	x-order	Parameter	T-limit (K)
AB	L0	(A)(B)	0	-10000+6*T	6000

The calculated phase diagrams using both original database and original + appended databases are shown in Figure 3.71.
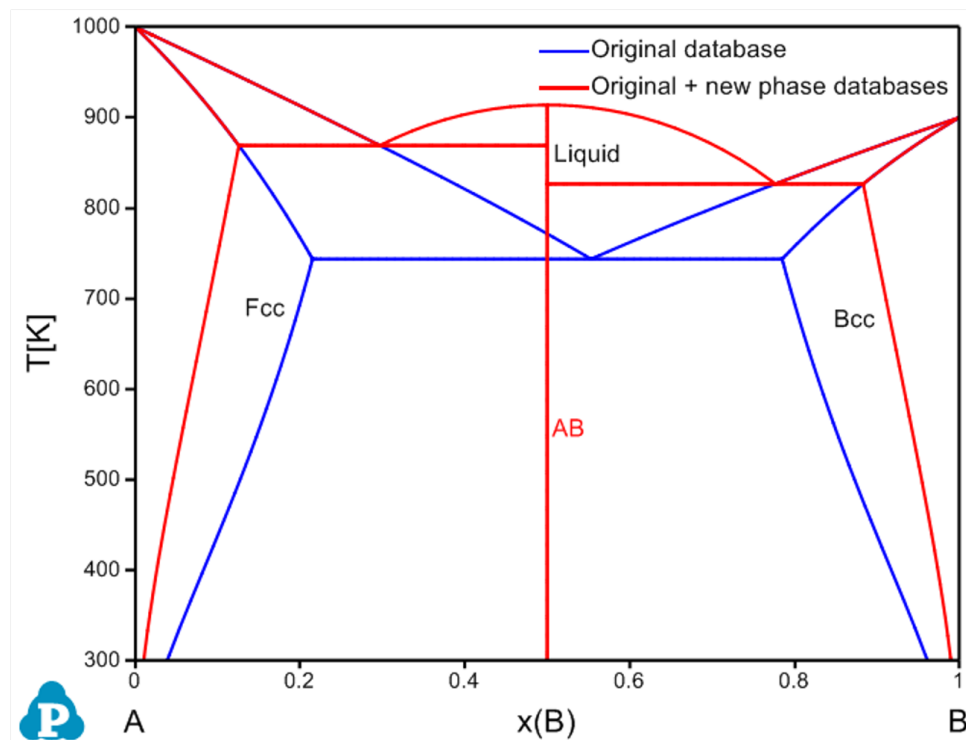
Figure 3.71: Calculated A-B phase diagram using both original database and original + appended databases with new phase

3.3.9.5 Add User-defined Properties to the Original Database

As is well known, the CALPHAD method has now been used to describe various types of phase properties in addition to thermodynamic properties. Mobility databases, molar volume databases and other thermo-physical property databases can be developed via similar route as that of developing a thermodynamic database. The **Append TDB** function allows a user to add user-defined properties to an original database. In this example, we add molar volume to `AB_Original.tdb` via the **Append TDB** function. The molar volume parameters of A and B within the `Bcc`, `Fcc`, and `Liquid` phases are described in the AB_ property.tdb database as listed below:

```
ParameterVm(Bcc,A;0)  298 +7.4e-6*exp(1e-6*T);  3000  N  !
ParameterVm(Bcc,B;0)  298 +8.4e-6*exp(1e-6*T);  3000  N  !
ParameterVm(Fcc,A;0)  298 +7.0e-6*exp(1e-6*T);  3000  N  !
ParameterVm(Fcc,B;0)  298 +8.0e-6*exp(1e-6*T);  3000  N  !
```

```
ParameterVm(Liquid,A;0) 298 +8.0e-6*exp(1e-6*T); 3000 N !
ParameterVm(Liquid,B;0) 298 +9.0e-6*exp(1e-6*T); 3000 N !
```

Load the original database `AB_original.tdb` and then append the `AB_property.tdb` database via the **Append TDB** function. The currently combined `AB_original + AB_property` database enables us to calculate molar volume of the A-B binary system in addition to the phase diagram. The molar volume contour lines are shown in Figure 3.72 (refer to Section 3.3.5 for calculating contour diagrams). Moreover, density and linear thermal expansion coefficient can also be calculated with this `AB_original + AB_property` database.

Please note that, various types of user-defined properties including but not limited to atomic mobility, molar volume, viscosity, surface tension can all be combined with the original thermodynamic database via the **Append TDB** function.
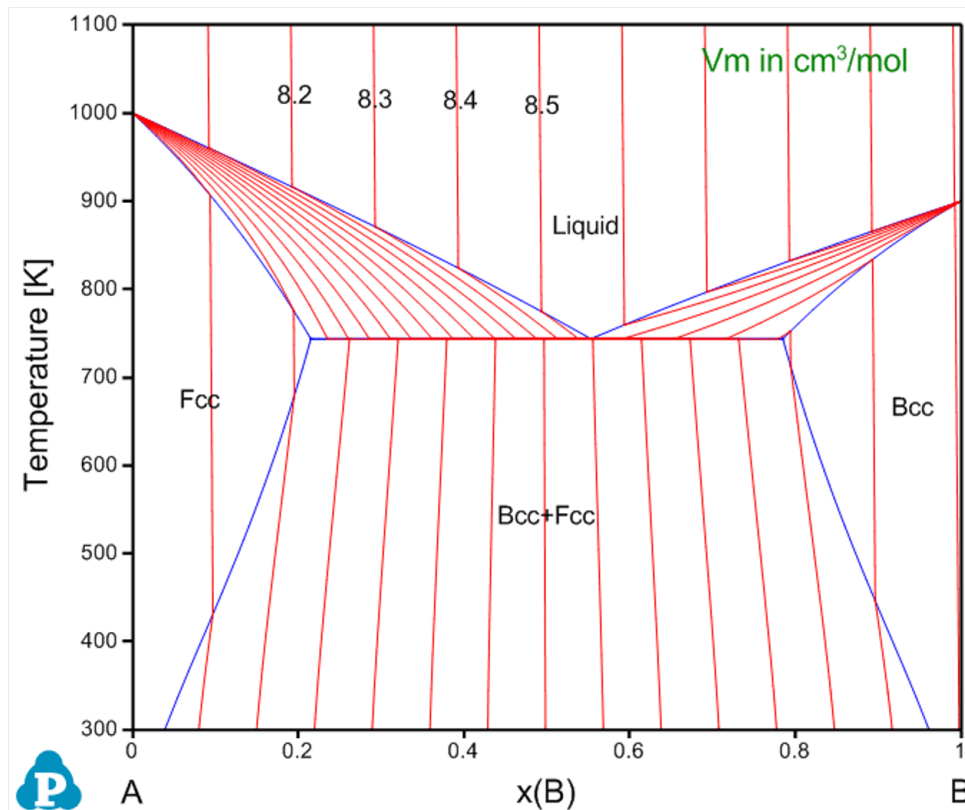
Figure 3.72: Calculated A-B phase diagram using both original database and original + appended databases with specific property

3.3.10 User-Defined Property

This section is an extension of the above Section 3.3.9 to develop and add user-defined-property database to the original database. Pandat™ allows user to define any property of a phase in a format similar to that of the Gibbs energy describing a disordered solution phase. Let *U* be the user-defined property and it is expressed as:

$$U = \sum_{i=1}^{c} x_i U_i^{\circ} + \sum_{i=1}^{c-1} \sum_{j=i+1}^{c} x_i x_j \sum_{k} (x_i - x_j)^k L_{ij}^k \quad (3.17)$$

where x_i is the molar fraction of component *i* and U_i^0 is the property of the pure component *i*, L_{ij}^k is the k$^{\text{th}}$ order interaction parameter between components *i* and *j*.

   User can also define special properties associated with the properties of phases in the original database. Any phase property available from Pandat Table can be used for user-defined property, such as `G, H, mu`, and `ThF`. However, the star symbol in a property, like `mu(*)`, cannot be used. User may refer to Chapter 9 Property for more properties in detail.

3.3.11 Advanced Features

In this section, we are going to cover some advanced features of the **PanPhaseDiagram** module.

3.3.11.1 Local Equilibrium

In default,Pandat™ always calculates global stable phase equilibria. Even some phases are suspended, the calculated phase equilibria are still global stable ones for those "`Entered`" phases. The current version of Pandat enables us to calculate a "`real`" local equilibrium. The Al-Zn binary system is used as an example to demonstrate how to calculate the local phase equilibrium between the `Fcc` and `Hcp` phases. Note that, there is not GUI for the local equilibrium function and user has to run it through the batch file (`.pbfx`).

   As shown below, a point with initial values needs to be defined in the batch file.

```
<point>
   <statespace>
         <T value="500"/>
         <P value="1"/>
```

```xml
            <n component="Al" value="0.5"/>
            <n component="Zn" value="0.5"/>
    </statespace>
    <initial_value>
            <mu species="Al" value="-16000" />
            <mu species="Zn" value="-22000" />
            <phase_point phase_name="Fcc">
                    <y species="Al" sublattice="1" value="0.9" />
                    <y species="Zn" sublattice="1" value="0.1" />
            </phase_point>
            <phase_point phase_name="Hcp">
            <y species="Al" sublattice="1" value="0.01" />
            <y species="Zn" sublattice="1" value="0.99" />
            </phase_point>
    </initial_value>
  </point>
```

In addition to the point with initial values, equilibrium type needs to be set as "`local`" (as shown below)

```xml
  <condition>
     <equilibrium_type type="local"/>
  </condition>
```

Figure 3.73 is the calculated stable `Al-Zn` binary phase diagram with the local-equilibrium between `Fcc` and `Hcp` phases.
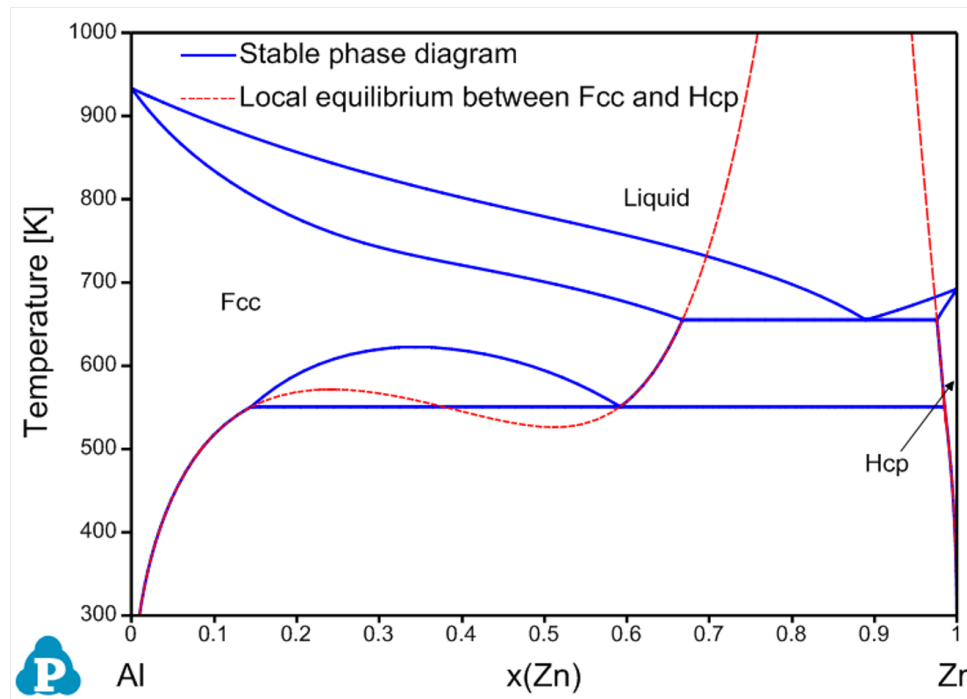
Figure 3.73: Calculated `Al-Zn` stable phase diagram with local-equilibrium between `Fcc` and `Hcp` phases

3.3.11.2 Para-Equilibrium

Pandat provides a function to calculate para-equilibrium. The detail description and definition of para-equilibrium can be found in the book of Professor Hillert [1998Hil].

In a normal phase equilibrium, all components are considered to be mobile, freely to move from one phase to another to reach the equilibrium state, in which the same component in different phases has identical chemical potential.

The concept of para-equilibrium applies to alloy systems where there is a large difference in mobility of the different components. Then some components can be treated as mobile and others as immobile. In a para-equilibrium, the ratio of molar fractions of immobile components in one phase is the same as that in other phases:

$$x_{i_1}^\alpha : x_{i_2}^\alpha = x_{i_1}^\beta : x_{i_2}^\beta \ (i_1 \text{ and } i_2 : \text{immobile components}) \tag{3.18}$$

However, the chemical potential of an immobile component in one phase is different from that in other phases. In rare cases an immobile component may have the same chemical potential in different phases. The same mobile component in different phases in para-equilibrium has the same chemical potential: $\mu_m^{\alpha} = \mu_m^{\beta}$ (m: a mobile component)

The calculation for a para-equilibrium follows the same procedure as the normal calculation except for setting the mobile components before calculation. In dialog window for all three calculations of Point (0D), Line (1D), and Section (2D), there is a button "**Mobile Comps.**" After clicking on this button, a new dialog window pops up for setting the mobile components. As shown below are the dialog windows for setting an isothermal section calculation for Fe-Ni-C system with C is mobile. The default state is that the all the components are immobile. When the element is set as "**mobile comps**", After click "**OK**", the mobile component will be shown as orange at the calculation condition setting interface, as shown in Figure 3.74. The calculated para-equilibrium isothermal section is shown in Figure 3.75.
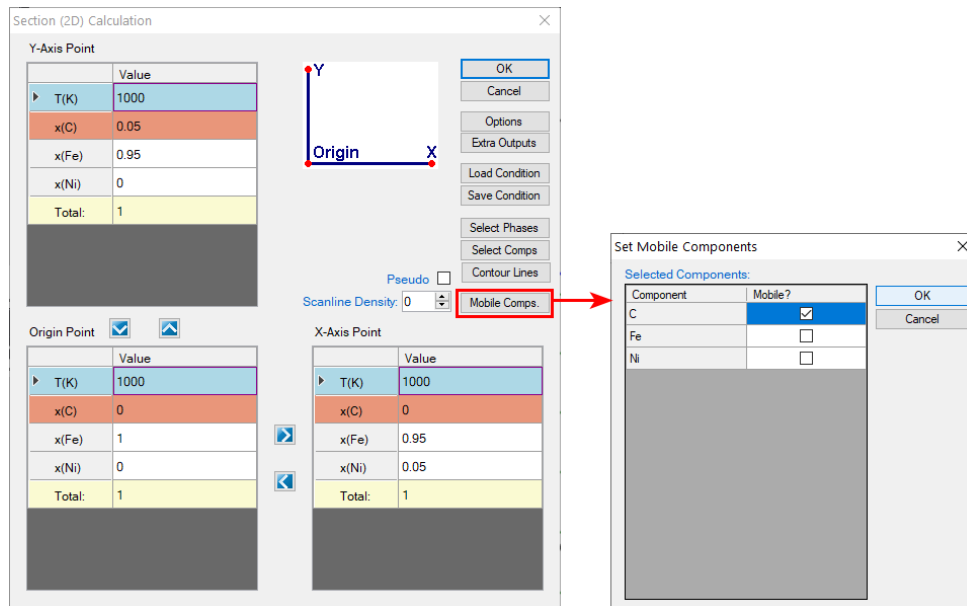
Figure 3.74: Set calculation condition for a para-equilibrium section calculation
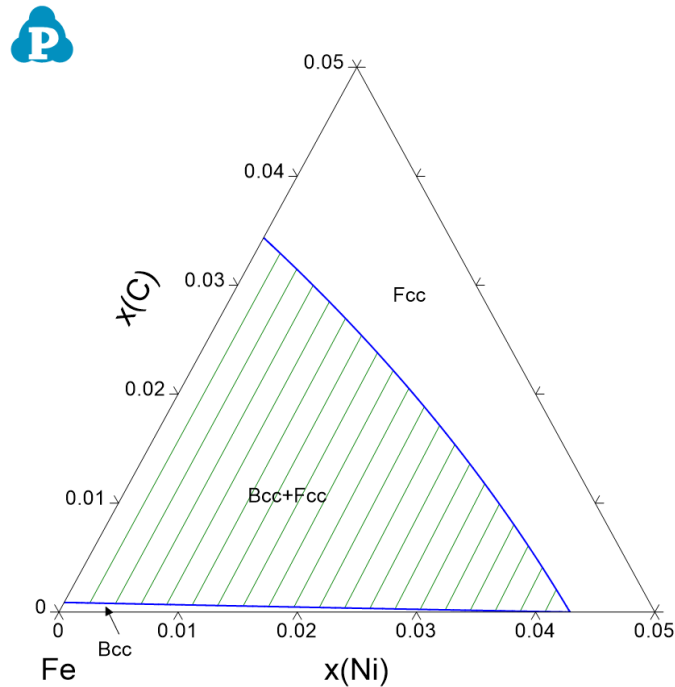
Figure 3.75: Set calculation condition for a para-equilibrium section calculation

3.3.11.3 Hessian Matrix of Gibbs Energy

Pandat™ can calculate the determinant of Hessian matrix of Gibbs energy of a phase and the eigenvalues and eigenvectors of the Hessian matrix.

Since there is one dependent molar fraction for the molar fraction variables $(x_1, x_2, \cdots; x_n)$, one of the components is selected as the dependent one. Without loss of generality, x_n is selected as the one, i.e., the last component is considered as the solvent. Then, the second derivatives of Gibbs free energy of a phase form the Hessian matrix, which is an $(n\text{-}1) \times (n\text{-}1)$ symmetrical matrix.

$$
HSN = \begin{pmatrix}
\dfrac{\partial^2 G}{\partial x_1^2} & \dfrac{\partial^2 G}{\partial x_1 \partial x_2} & \dfrac{\partial^2 G}{\partial x_1 \partial x_3} & \cdots & \dfrac{\partial^2 G}{\partial x_1 \partial x_{n-1}} \\[2ex]
\dfrac{\partial^2 G}{\partial x_2 \partial x_1} & \dfrac{\partial^2 G}{\partial x_2^2} & \dfrac{\partial^2 G}{\partial x_2 \partial x_3} & \cdots & \dfrac{\partial^2 G}{\partial x_2 \partial x_{n-1}} \\[2ex]
\dfrac{\partial^2 G}{\partial x_3 \partial x_1} & \dfrac{\partial^2 G}{\partial x_3 \partial x_2} & \dfrac{\partial^2 G}{\partial x_3^3} & \cdots & \dfrac{\partial^2 G}{\partial x_3 \partial x_{n-1}} \\[2ex]
\vdots & \vdots & \vdots & \ddots & \vdots \\[2ex]
\dfrac{\partial^2 G}{\partial x_{n-1} \partial x_1} & \dfrac{\partial^2 G}{\partial x_{n-1} \partial x_2} & \dfrac{\partial^2 G}{\partial x_{n-1} \partial x_3} & \cdots & \dfrac{\partial^2 G}{\partial x_{n-1}^2}
\end{pmatrix}
$$

Its determinant is given by

$$
|\mathbf{HSN}| =
\begin{vmatrix}
\dfrac{\partial^2 G}{\partial x_1^2} & \dfrac{\partial^2 G}{\partial x_1 \partial x_2} & \dfrac{\partial^2 G}{\partial x_1 \partial x_3} & \cdots & \dfrac{\partial^2 G}{\partial x_1 \partial x_{n-1}} \\[2ex]
\dfrac{\partial^2 G}{\partial x_2 \partial x_1} & \dfrac{\partial^2 G}{\partial x_2^2} & \dfrac{\partial^2 G}{\partial x_2 \partial x_3} & \cdots & \dfrac{\partial^2 G}{\partial x_2 \partial x_{n-1}} \\[2ex]
\dfrac{\partial^2 G}{\partial x_3 \partial x_1} & \dfrac{\partial^2 G}{\partial x_3 \partial x_2} & \dfrac{\partial^2 G}{\partial x_3^3} & \cdots & \dfrac{\partial^2 G}{\partial x_3 \partial x_{n-1}} \\[2ex]
\vdots & \vdots & \vdots & \ddots & \vdots \\[2ex]
\dfrac{\partial^2 G}{\partial x_{n-1} \partial x_1} & \dfrac{\partial^2 G}{\partial x_{n-1} \partial x_2} & \dfrac{\partial^2 G}{\partial x_{n-1} \partial x_3} & \cdots & \dfrac{\partial^2 G}{\partial x_{n-1}^2}
\end{vmatrix}
$$

The determinant of Hessian matrix for phase f is available from "`HSN(@f)`". The value of "`HSN(@f)`" is independent of the selection of the solvent component [2021Mor, 2022Che].

The above Hessian matrix has (n-1) eigenvalues and each eigenvalue has a corresponding eigenvector. In general, both eigenvectors and eigenvalues depend on the choice of the solvent component. These eigenvalues can be used to identify the spinodal boundaries. At the spinodal boundary, one of the eigenvalues will be zero for every choice of the solvent element. The eigenvalues and their eigenvectors are available from "`eVal(#*@f)`" and "`eVec(*#*@f)`", respectively. The individual eigenvalues can be accessed in the Table as "`eVal(#1@f)`", "`eVal(#2@f)`", … ,"`eVal(#n-1@f)`". The eigenvalues are inherently sorted in the ascending order. Thus, "`eVal(#1@f)`" is the minimum eigenvalue and "`eVal(#n-1@f)`" is the maximum eigenvalue. Each eigenvalue has an eigenvector. For example, `eVal(#1@f)` has an eigenvector of (`eVec(C1#1@f)`, `eVec(C2#1@f)`,…, `eVec(Cn-1#1@f)`), where C_k is the name of the kth component. The keywords `HSN`, `eVal` and `eVec` are listed in Table 11.4 in Section 11.2 .

The "`solvent`" component can be defined while using "`Thermodynamic Property`" calculation, or defined in Pandat™ Batch file (`.pbfx`). For instance, "`<solvent name="Fe" />`" can be added to the batch file for choosing "`Fe`" as the solvent element. If "`solvent`" component is not defined, Pandat™ will use a special treatment (called orthogonalization [2022Kon]) that ensures both the eigenvalues and the eigenvectors are independent of the solvent component. This special treatment would be very helpful in analyzing spinodal decomposition in high- entropy alloys. Note that eigenvalues are generally not calculated during thermodynamic calculation for efficiency. But it can be

calculated by adding the flag "`<Hessian value="true" />`" to the batch file. Sample batch files for calculating eigenvalues/eigenvectors by choosing a solvent (`Eigenvalues_ solvent_ Fe.pbfx`) or without choosing a solvent element (`Eigenvalues_ nosolvent.pbfx`) can be found in the Examples folder (`\Pandat 2024 Examples\PanPhaseDiagram\Eigenvalues`).

There is 1 stable phase

Component	x	w	n(mol)	n(kg)	μ(J/mol)	H(J/mol)	S(J/K·mol)
Cr	0.6	0.576606	0.6	0.0311976	-35315.8	26228.2	63.242
Fe	0.2	0.206437	0.2	0.0111694	-50962	30620.1	83.833
Ni	0.2	0.216957	0.2	0.0117386	-49720.7	33422.1	85.4367

f	fw	fv	n(mol)	n(kg)	G(J/mol)	H(J/mol)	Cp(J/K·mol)	G_ex(J/mol)	H_ex(J/mol)	HSN
1	1	1	1	1	-41326	28545.3	32.7349	2948.75	5786.48	-1.26145e+09

eVal		eVec	
	Cr	Fe	Ni
Bcc -17665.4	0.758541	-0.117618	-0.640923
71407.9	0.30213	-0.807981	0.50585

Vm(cm^3/mol)	density(g/cm^3)
7.0753	7.64711

Sublattice Size	Species	y	dGdy
1	Cr	0.6	-23674.6
	Fe	0.2	-39320.8
	Ni	0.2	-38079.5
3	Va	1	0

Figure 3.76: Calculated eigenvalues/eigenvectors without choosing the solvent component in Fe-Cr-Ni ternary alloy (from the batch file "`Eigenvalues_ nosolvent.pbfx`").

There is 1 stable phase							
Component	**x**	**w**	**n(mol)**	**n(kg)**	**μ(J/mol)**	**H(J/mol)**	**S(J/K·mol)**
Cr	0.6	0.576606	0.6	0.0311976	-35315.8	26228.2	63.242
Fe	0.2	0.206437	0.2	0.0111694	-50962	30620.1	83.833
Ni	0.2	0.216957	0.2	0.0117386	-49720.7	33422.1	85.4367

f	fw	fv	n(mol)	n(kg)	G(J/mol)	H(J/mol)	Cp(J/K·mol)	G_ex(J/mol)	H_ex(J/mol)	HSN
1	1	1	1	1	-41326	28545.3	32.7349	2948.75	5786.48	-9.4609e+08

	eVal	eVec	
		Cr	**Ni**
Bcc	-8975.63	-0.772098	0.635503
	105406	-0.635503	-0.772098

Vm(cm^3/mol)	density(g/cm^3)
7.0753	7.64711

Sublattice Size	Species	y	dGdy
	Cr	0.6	-23674.6
1	Fe	0.2	-39320.8
	Ni	0.2	-38079.5
3	Va	1	0

Figure 3.77: Calculated eigenvalues/eigenvectors by choosing "Fe" as the solvent element in Fe-Cr-Ni ternary alloy (from the batch file "`Eigenvalues_solvent_Fe.pbfx`")

Calculation of Spinodal Curve Using the Eigenvalue of Hessian Matrix

One of the important application using the eigenvalues of Hessian matrix is to calculate the spinodal line of a spinodal decomposition phase. The `eVal(@*)` can be added as an condition in the batch file to find the spinodal line as contour line. Below is the example to calculate the spinodal line of the Bcc phase, and the calculated spinodal line is superimposed on the Fe-Ni-Cr isothermal section as shown in Figure 3.78.

```
<condition>
    <contour   name="Spinodal"   property="eVal (#1@Bcc)"   start="0"
stop="10" step="100" />
    <equilibrium_type type="individual" />
    <scanline_density value="1" />
    <Hessian value="true" />
</condition>
```
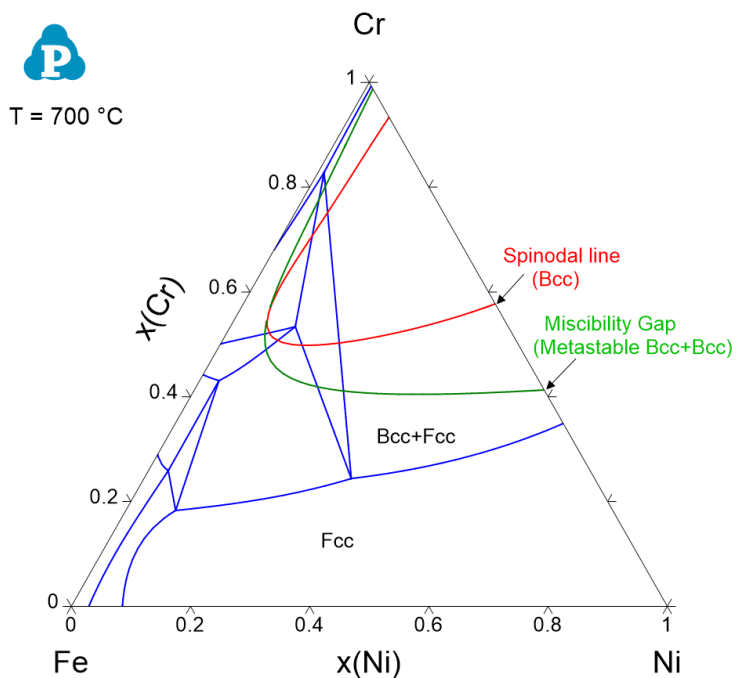
Figure 3.78: Calculated Fe-Cr-Ni isothermal section at 700 °C together with the metastable miscibility and spinodal lines (from the batch file "`Spinodal_boundary_on_section.pbfx`")

4   PanOptimizer

PanOptimizer is a module of Pandat™ software designed for optimizing thermodynamic, kinetic and thermo-physical model parameters using experimental data. The module allows user to develop their own databases or model parameters for customized applications. This module is also easy to master by students who want to learn thermodynamic modeling.

4.1 Features of PanOptimizer

Derived from the maximum likelihood principle, when the discrepancies between model-calculated and experimental values are assumed to be independent, identically distributed with a normal distribution function, a set of model parameters with the best fit to the given experimental data can be obtained by the least square method. In the real world the experimental data may come from different sub-populations for which an independent estimate of the error variance is available. In this case, a better estimate than the ordinary least squares (OLS) can be obtained using weighted least squares (WLS), also called generalized least squares (GLS), then the sum of the squares can be written as

$$\frac{1}{2} \sum_{j=1}^{m} w_j [y_j - \phi(x; T_j, c_j)]^2$$

(4.1)

The idea is to assign each observation a weight factor that reflects the uncertainty of the measurement. According to our previous experiences this method is efficient and reliable in the optimization of model parameters and develop variety of databases.

4.2 Functions

The major functions included in the PanOptimizer menu are shown in Figure 4.1.
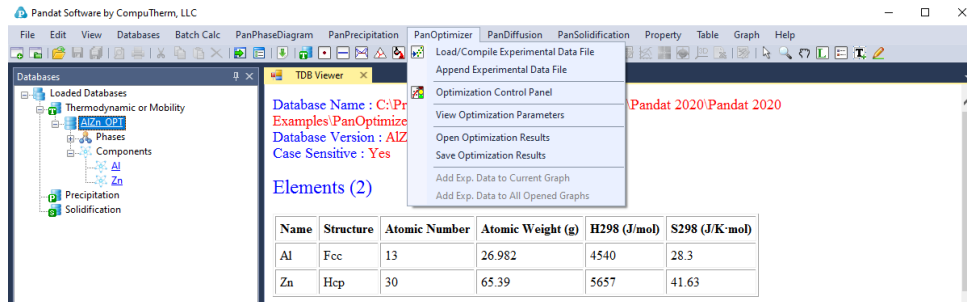
Figure 4.1: The PanOptimizer Menu

The shortcut of the three commonly used functions **Load/Compile Experimental File**, and **Optimization Control Panel** are also displayed in the Toolbar as shown in Figure 4.2.
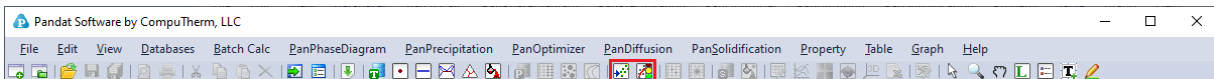
Figure 4.2: PanOptimizer toolbar

The functionality of each function is given below:

- **Load/Compile Experimental File:** Load/Compile experimental data file for model parameter optimization

- **Append Experimental File:** Append more experimental data from different data file

- **Optimization Control Panel:** Allow user to set boundaries for the model parameters to be optimized, select experimental data to be used, and assign weight to each set of experimental data

- **View Optimization Parameters:** View/Edit model parameters to be optimized

- **Open Optimization Results:** Open the saved intermediate optimization results including the model parameters and experimental data

- **Save Optimization Results:** Save the intermediate optimization results during the optimization process including the model parameters and experimental data

4.3 Optimization Procedures

In general, user should follow a few steps below to perform the model parameter optimization.

4.3.1 Step 1: Prepare Files

4.3.1.1 Prepare the Thermodynamic Database file (.TDB)

In order to do the optimization, user must first define the model parameters to be optimized in the thermodynamic database (`.tdb`). The model parameters to be optimized can be defined in Pandat™ workspace where all the built- in keywords are automatically highlighted. It can also be done through outside text editor such as "Notepad". Each model parameter that needs to be optimized is defined by the keyword "`Optimization`". The format of defining a model parameter is:

```
Optimization [parameter name][low bound][initial value][high bound] N
!
```

A definition sample for liquid phase in binary Al-Zn system is given as follows

```
$ Keyword Name Low Bound Init. Value High Bound
Optimization LIQ_AA 00; 60000 N !
Optimization LIQ_AAT -200; 20 N !
Optimization LIQ_BB -600000; 60000 N !
Optimization LIQ_BBT -200; 20 N !
Phase Liquid % 1 1 !
Constituent Liquid : Al, Zn : !
Optimization G(Liquid,Al;0) 298.15 G_Al_LIQUID; 6000 N !
Optimization G(Liquid,Zn;0) 298.15 G_Zn_LIQUID; 6000 N !
Optimization G(Liquid,Al,Zn;0) 298.15 LIQ_AA+LIQ_AAT*T; 6000 N !
Optimization G(Liquid,Al,Zn;1) 298.15 LIQ_BB+LIQ_BBT*T; 6000 N !
```

Two modes of optimization are allowed in the current version: one is "bounded" optimization and the other one is "no bound" optimization. Even though the low and high bounds will be in effect in the bounded optimization only, they are required for definition of model parameters in a `TDB` file. The model parameters can be named at user' choices. It is suggested that the name is related to the phase to be optimized.

4.3.1.2 Prepare the Experimental File (`.POP`)

user need to provide their own experimental data file for optimization of model parameters. The most widely accepted format for experimental data file in the CALPHAD society is a `POP` file. PanOptimizer accepts most of the keywords in the `POP` format and adds a few special keywords. In a `POP` file, a phase can have four statuses: ENTERED, FIXED, DORMANT, and SUSPEND. The first two statuses were used most frequently. When phases are in the ENTERED status, PanOptimizer does not require user to input any initial values for calculation since the truly stable phase equilibria will be found automatically in this case with the built-in global optimization algorithm. On the other hand, for those phases in FIXED or DORMANT status, the initial values should be provided by the user. A detail description on keywords and syntax of a POP file recognized by PanOptimizer is given in Section 11.4 . Example POP files are provided in the installation dictionary of Pandat™.

4.3.2 Step 2: Carry Out Optimization

Once above files have been prepared, user is ready to carry out optimization.

4.3.2.1 Load thermodynamic database file

First of all, the database file (`.TDB`) containing the phases to be modeled is loaded by click the ⬛ icon. This database file defines the model type and model parameters of each phase in the system. The model definition is consistent with the currently accepted format in the CALPHAD society, and the model parameters can be real numbers or variables to be optimized.

4.3.2.2 Load and compile experimental file

After loading a `TDB` file with defined model parameters to be optimized, user should then load and compile the experimental `POP` file. Go to PanOptimizer menu and choose Load/Compile Experimental File, user can then select the `POP` file already prepared as shown in Figure 4.3. User can also choose **Append Experimental File** to append new experimental data stored in a separate file to the currently opened `POP` file.
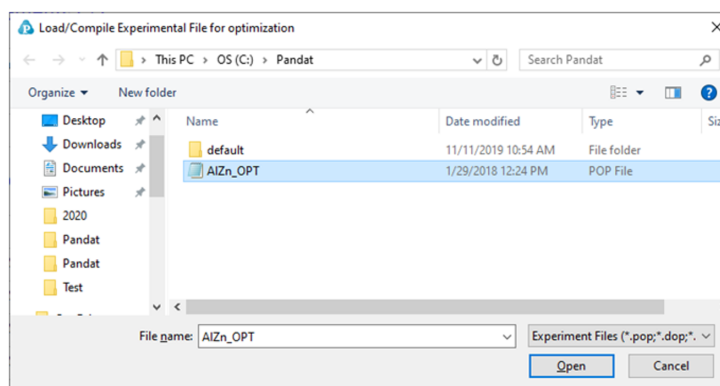
Figure 4.3: Dialog window of loading an experimental file

4.3.2.3 Perform Optimization

Once the `TDB` file and the experimental `POP` file are loaded, user is ready to do the optimization. In the current version of PanOptimizer, the optimization is controlled through the optimization control panel as shown in Figure 4.4. There are four control areas in the optimization control panel. They are: Histogram (A), Bound/Unbound Variables (B), Optimization (C), and Optimization Results (D).
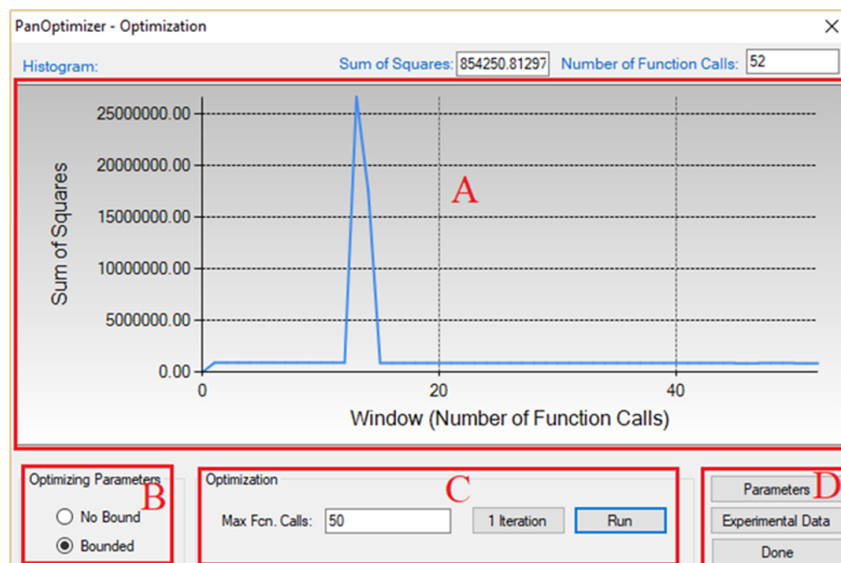
Figure 4.4: Optimization control pane

4.3.2.4 Histogram (A)

Histogram is for displaying and tracing the history of the discrepancy between model-calculated values and experimental data, which is characterized by the Sum of Squares displayed during the whole optimization procedure. The histogram plots the sum of squares vs. the number of function calls. The exact value of the sum of squares in the current step can be found at the up-right corner of this area.

4.3.2.5 Free Bound/Unbound Variables (B)

The user can choose the model parameters to be optimized in either the bounded or the unbounded mode. In the bounded mode, the low and high bounds defined in a `TDB` file will take into effects.

4.3.2.6 Optimization (C)

The goal of the optimization process is to obtain an optimal set of model parameters so that the model calculated results can best fit the given experimental measurements. The optimization process can be controlled by choosing "`1 Iteration`" or "`Run`". With "`1 Iteration`", the maximum number of function calls is set to be $2(N + 2)$, where N is the number of model parameters to be optimized. The user can click "`Run`" mode several times until the optimal solution is found or the designated maximum number of function calls is reached.

Here we take the binary Al-Zn system as an optimization example. The `TDB` and `POP` files are available at the installing directory of Pandat " `/Pandat_ Examples/PanOptimizer/`". In this example, there are totally 11 parameters to be optimized and all the initial values are set to be zero. The available experimental data include:

- mixing enthalpy of liquid phase at 953K

- invariant reaction at 655K: `Liquid → Fcc + Hcp`

- invariant reaction at 550K: `Fcc#2 → Fcc#1 + Hcp`

- tie-lines: `Fcc#1+Fcc#2`, `Liquid + Fcc`, `Liquid + Hcp` and `Hcp + Fcc`

Before the optimization, user may want to check the calculated results using the initial set of model parameters without optimization. Phase diagram calculation can be done through 2-D section calculation, and enthalpy calculation can be done through the 1-D line calculation. Figure 4.5 shows the calculated Al-Zn phase diagram and enthalpy of mixing for liquid phase at 953K along with the given experimental data, respectively. The parameters before optimization result in large discrepancies between the calculated values and the experimental measurements. Optimization of these model parameters (initially set as zero) is needed.
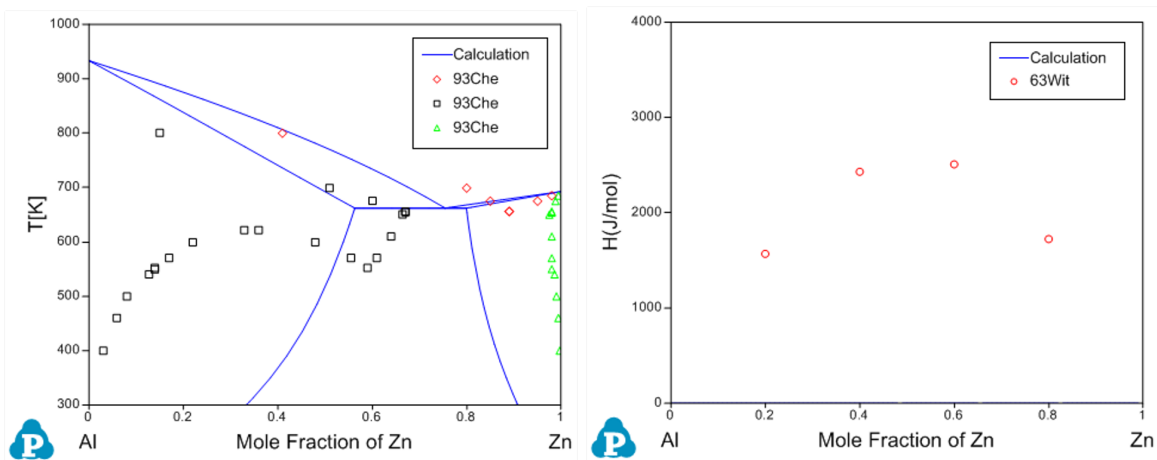
Figure 4.5: Comparison between calculated results and experimental data before the optimization

By clicking "Run" twice (two rounds), and each run performs 50 function calls, the sum of squares decreases from 854251 to 403 as shown in Figure 4.6(a). After another round of Run, the sum of squares decreases from 403 to 3.15 as shown in Figure 4.6(b). Now, we can do the real time calculation using the instantly obtained optimized parameters. The two comparisons in Figure 4.7 show excellent agreements between the calculated results and the measured data. By selecting the command from PanOptimizer menu: "`PanOptimizer → Add Exp. Data to the current Graph`", user can add the corresponding experimental data defined in the pop file directly to the calculated diagram for comparison.
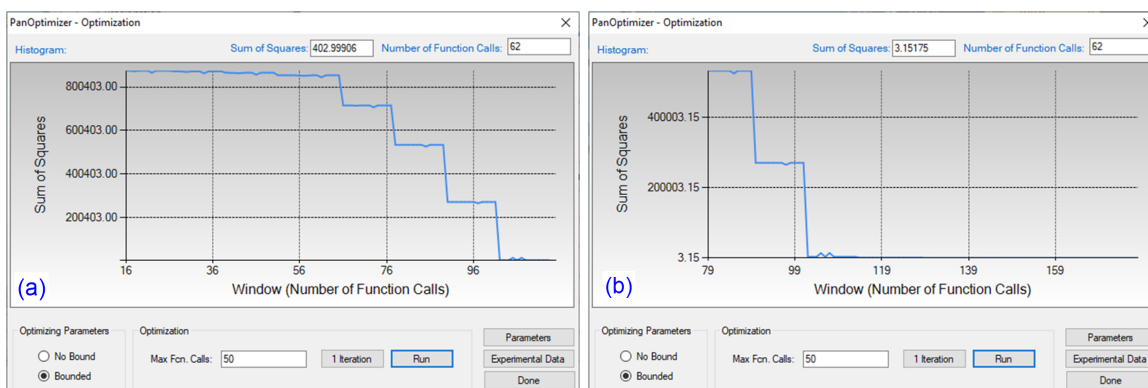
Figure 4.6: (a) Sum of squares after two rounds of optimization (b) Sum of squares after another two rounds of optimization
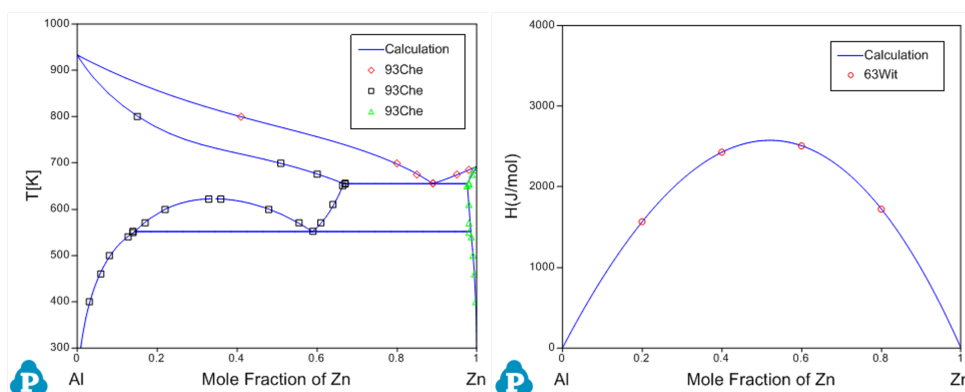
Figure 4.7: Comparison between the calculated results and the experimental data after optimization

4.3.2.7 Optimization Results (D)

During optimization, user can check model parameters through the `Parameters` button as shown in Figure 4.4. For each model parameter, user can change its low bound, upper bound and initial value in the dialog window as shown in Figure 4.8. User can "`Include`" or "`Exclude`" a certain parameter in the optimization through the "`check box`" in front of the parameter. If a set of optimized parameters is satisfactory, user can save this set of values as default ones through `Set Default` shown in Figure 4.8. Otherwise, user can reject this set of values and go back to previous default values through `Get Default`. User also can save the `TDB` file with the optimized model parameters through `Save TDB`. The standard deviation and relative standard deviation (RSD) of each parameter are computed during optimization. The parameter evolution during the last 100 iterations can be tracked

by clicking the parameter name in the table as indicated in Figure 4.8. It should be noted that any changes on model parameters made by user will take into effect only after the button `Apply` is clicked.

By clicking the "`Experimental Data`" button onFigure 4.6(b), a table which lists the calculated values together with the experimental data will pop out as shown in Figure 4.9. This table allows user to view how well the current optimized parameters describe each set of experimental data. If the discrepancy is found to be large and not satisfied for a certain set of data, a larger weight factor for this set of data can be given in the next round of optimization. Through the column of `Residual` shown in Figure 4.9, user can see different color after each comparison. The green means satisfactory results are achieved, while the red color means large deviation remaining. Again, user can `Include` or `Exclude` a certain experimental data set for the optimization through the check box in front of each experimental data `Name`. Also, more detailed experimental data information can be viewed by clicking the `Equilibrium ID` of the experimental data in the table.
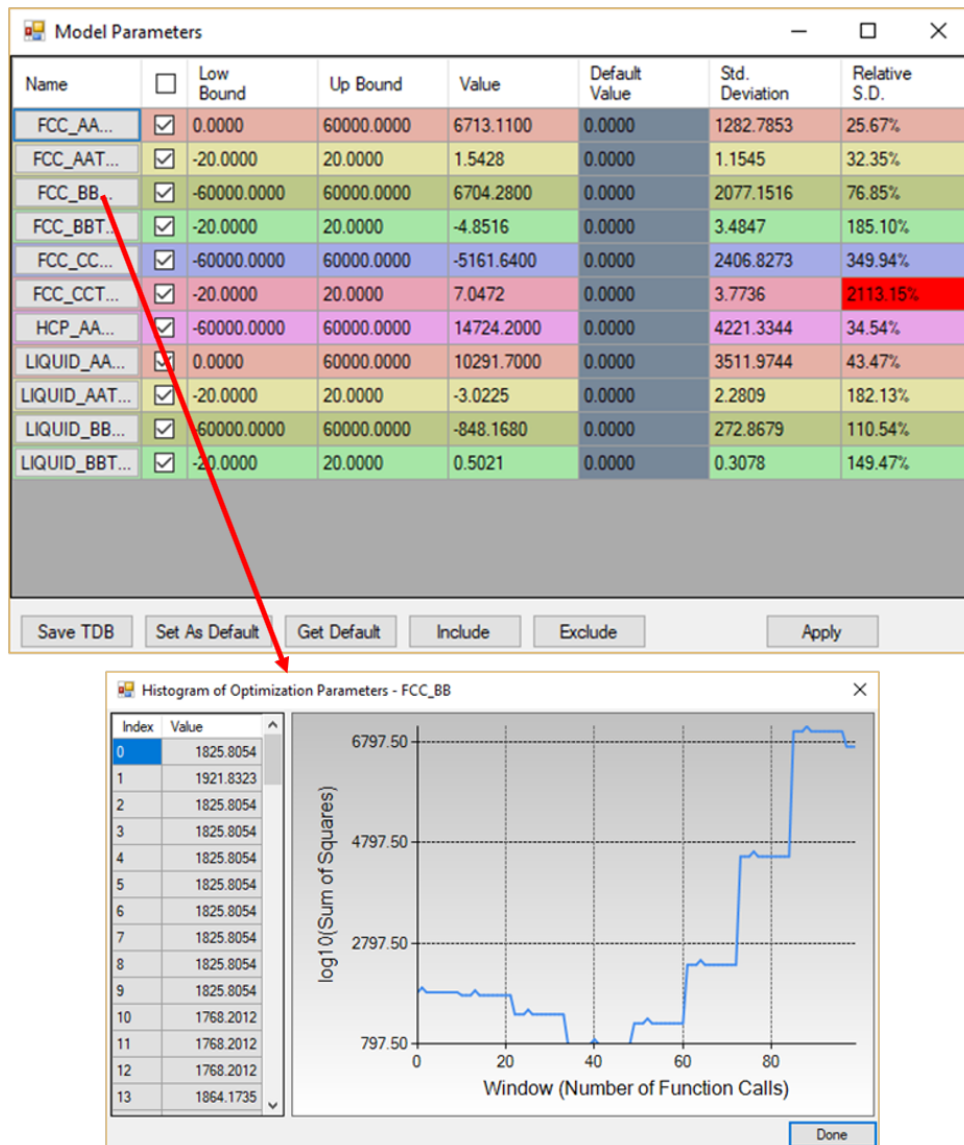
Figure 4.8: Dialog windows of Model Parameters and Histogram of Optimization Parameters
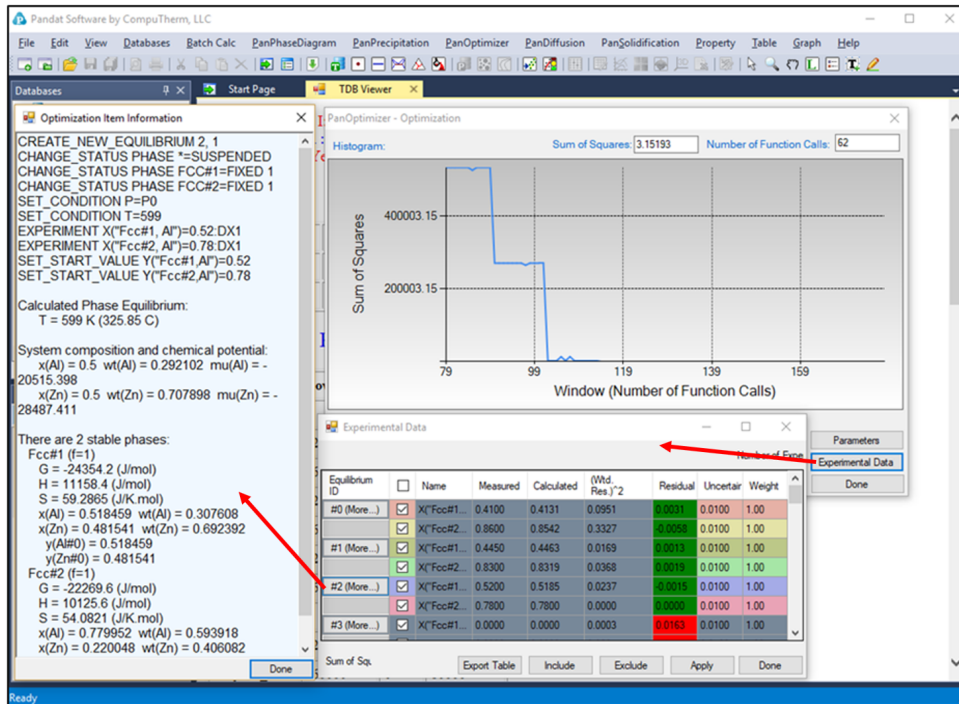
Figure 4.9: Dialog windows of Experimental Data and Optimization Item Information

4.3.3 Step 3: Save and Open Optimization Results

During optimization, the user may save and load optimization file through `Save Optimization Results` and `Open Optimization Results` (see Figure 4.10). The optimization results file has the extension name of POR that can only be read by PanOptimizer. Intermediate optimization results can be saved and restored through these two operations. These functions are very useful especially when a user is optimizing model parameters for a complicated system. The user can always go back to a certain middle stage and restart from there. This will save user's time by avoiding some repeated work.
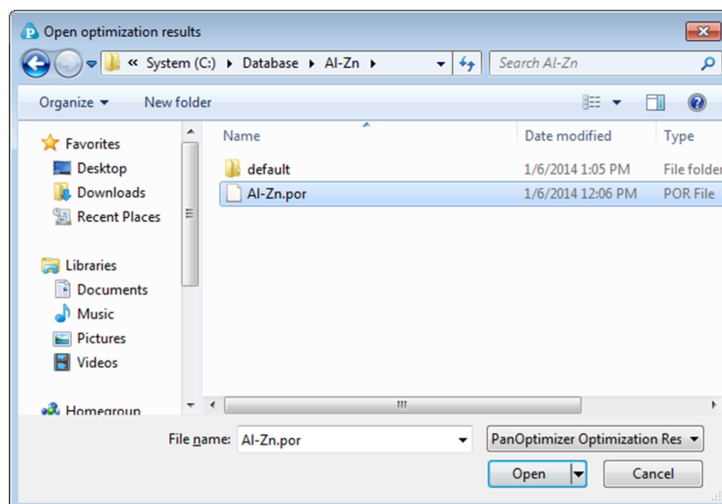
Figure 4.10: Dialog window of open the saved optimization results

4.4 User-Defined Property Optimization

The parameters for physical property, kinetic property, and user-defined property can also be optimized with respect to experimental data just like the parameters in a thermodynamic database. User can refer to Section 9.4 for the format to define user-defined property in a database. Similar to the optimization of thermodynamic model parameters, user needs to define the parameters to be optimized in the database and prepare a pop file with corresponding experimental data. An example for the parameter definition in the database is shown below:

```
$Thermal conductivity of pure element in the Fcc phase
PARAMETER  ThCond(Fcc,Al;0)  298.15   +311.72511- 0.09661*T- 13510.26/T;
3000 N !
PARAMETER  ThCond(Fcc,Cu;0)  298.15   +417.28065- 0.06598*T+944.065/T;
3000 N !
$Thermal  conductivity  interaction  parameters  for  Fcc  phase  to  be
optimized
OPTIMIZATION    Fcc_TC 0 0; 60000 N !
OPTIMIZATION Fcc_TCT -50 -24.6; 0 N !
OPTIMIZATION Fcc_TC_1 -60000 0; 0 N !
OPTIMIZATION Fcc_TCT_1 0 28; 50 N !
PARAMETER ThCond(Fcc,Al,Cu;0) 298.15 +Fcc_TC+Fcc_TCT*T; 3000 N !
PARAMETER ThCond(Fcc,Al,Cu;1) 298.15 +Fcc_TC_1+Fcc_TCT_1*T; 3000 N !
```

And an example pop file for the experimental data is shown below.

```
$Thermal Conductivity of the Fcc phase at 1000K
TABLE_HEAD
CREATE_NEW_EQUILIBRIUM @@,1
CHANGE_STATUS PHASE * = S
CHANGE_STATUS PHASE Fcc = FIX 1
SET_CONDITION T=1000, P = P0, X(Fcc,Cu) = @1
EXPERIMENT ThCond(Fcc)= @3 : @4
TABLE_VALUES
$ x(Cu) x(Al) TC DTC
0.05 0.95 194 50
0.1 0.9 295 50
0.2 0.8 750 50
0.8 0.2 3116 50
0.9 0.1 2122 50
0.95 0.05 1343 50
TABLE_END
```

The optimization procedure is the same as that for thermodynamic parameters. User may refer to the previous section for details and get the optimization result step by step.

5 PanEvolution/PanPrecipitation

The new PanEvolution module (an extension of the previous PanPrecipitation) enables the concurrent simulation of dislocation density, recrystallization, grain coarsening, in addition to the traditional precipitation simulation. The module is seamlessly integrated with the thermodynamic calculation engine, PanEngine, for the necessary thermodynamic input and mobility data. It is built as a shared library and integrated into Pandat™ as a specific module that extends the capability of Pandat™ for kinetic simulations, while taking full advantage of the automatic thermodynamic calculation engine (PanEngine) and the user- friendly Pandat™ Graphical User Interface (PanGUI). For this reason, precipitation simulations for highly complex alloys under arbitrary heat treatment conditions can be accomplished with only a few operations. Figure 5.1 shows the three-layered architecture of this modeling tool.
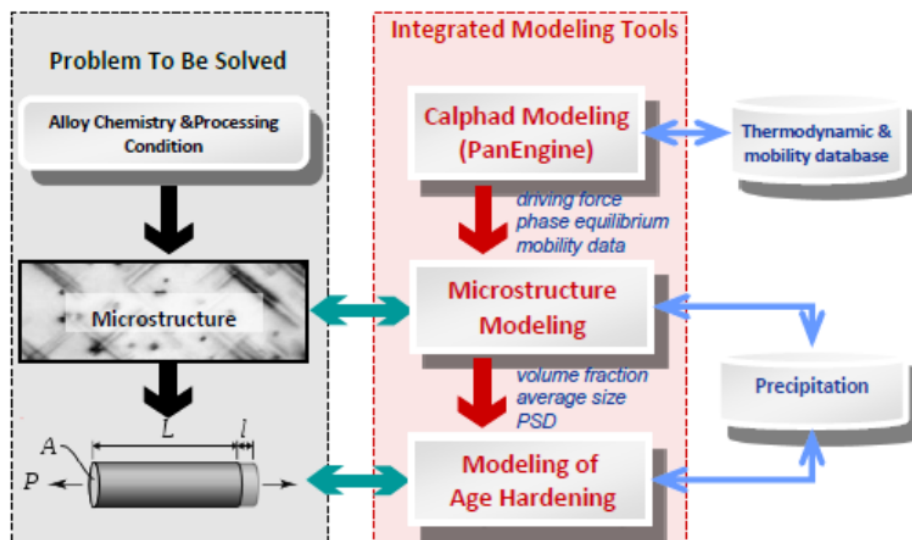
Figure 5.1: The three-layered architecture of the integrated modeling tool

5.1 Features of PanEvolution/PanPrecipitation

5.1.1 Overall Design

- Concurrent nucleation, growth/dissolution, and coarsening of precipitates

- Simulation of concurrent grain growth and precipitation

- Temporal evolution of average particle size and number density

- Temporal evolution of particle size distribution

- Temporal evolution of volume fraction and composition of precipitates

- Simulation of dynamic recrystallization process

- Simulation of dislocation density evolution during thermo-mechanical processes

5.1.2 Data Structure

PanEvolution is a purely object-oriented module written in C++ with generic data structures like PanEngine, balancing performance, maintainability and scalability. The basic data structure for storing precipitation information in the system of interest is schematically shown in Figure 5.2.
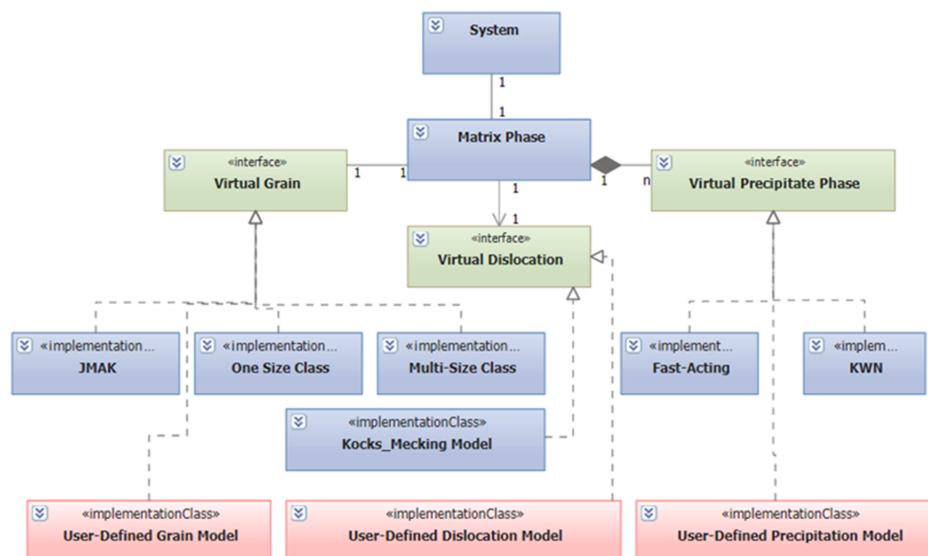
Figure 5.2: Data structure of system information in PanEvolution

In general, a system contains a matrix phase, which is defined by a grain structure, and a number of precipitate phases. Different precipitate phases may behave differently, which should be described by different kinetic models. On the other hand, the calculation speed is directly related to the complexity of the model. It therefore requires multi-level models for different purposes. The current PanEvolution module includes two built-in models for precipitation: the Kampmann/Wagner Numerical (KWN) model and the Fast-Acting model. In order to simulate the grain growth/coarsening behavior, "one-size" and "multi-size" models have been developed. One unique advantage is that this generic data structure allows for concurrent simulation of grain growth and precipitation as well as easy integration of other precipitation, grain growth, dislocation density and recrystallization models with the PanEvolution module as shown in Figure 5.2. This gives users great flexibility in choosing the proper kinetic models, including their own user-defined models, for custom applications.

Based on the above data structure, input parameters for the matrix and its grain/precipitate phases are organized in "Extensible Markup Language" (XML) format, which is a standard markup language and well-known for its extendability. In accordance with the XML syntax, a set of well-designed tags are used to define the models for grain evolution and for every precipitate phase along with the corresponding model parameters, such as grain boundary energy, interfacial energy, molar volume and nucleation site parameter. The defined models and model parameters, which should be calibrated by the available experimental data, are stored in the so-called "Kinetic Parameters Database (KDB)". In PanEvolution, two kinetic models known as the KWN and Fast-Acting were implemented for precipitation simulation and available for user's choice. Both models can be used to simulate the co-precipitation of phases with various morphologies (sphere and lens), with concurrent processes of nucleation, growth and coarsening. With the selection of the KWN model, the particle size distributions (PSD) of various precipitate phases can be obtained in addition to the temporal evolution of the average size and volume fraction as obtained from the Fast-Acting model. Therefore, the KWN model is recommended by default.

It should be noted that in the KDB, models with closed-form equations can be defined to replace the built-in sub-models. This provides a database/script file level plug-in method for user-defined models.

5.2 Kinetic Models

In PanEvolution , KWN model is based on the Kampmann and Wagner's work as implemented in a numerical framework, and extended to handle both homogeneous and heterogeneous nucleation, dealing with various morphologies for the simulation of precipitation kinetics of multi-component alloys under arbitrary heat treatment conditions. The following is a brief introduction to the KWN model along with its sub-models for nucleation, growth and coarsening.

Specifically, in the KWN model the continuous PSD is divided into a large number of size classes. The program takes a simulation step at every sample time hit. To maintain both accuracy and efficiency between two adjacent simulation steps, a fifth-order Runge-Kutta scheme is used to generate an adaptive step size based on the continuity equation.

Please note that all the variables used in the equations in this section are summarized in the tables in Section 11.3 . Please refer to these tables and examples in Section 5.3 for setting up these variables in the KDB files.

5.2.1 Precipitation Nucleation

(a) Homogeneous nucleation

At each simulation step, the number of new particles is first calculated using classical nucleation theory and then these new particles are allocated to an appropriate size class. The transient nucleation rate is given by,

$$J = N_\nu Z \beta^* \exp\left(-\frac{\Delta G^*}{k_B T}\right) \exp\left(-\frac{\tau}{t}\right) \tag{5.1}$$

The pre-exponential terms in Eq. 5.1 are: N_v, the nucleation site density, Z , the Zeldovich factor and β^*, the atomic attachment rate. t is the time, τ the incubation time for nucleation, k_B the Boltzmann constant and T the temperature.

The nucleation barrier is defined as $\Delta G^* = \frac{4\pi}{3}(R^*)^2 \sigma_{\alpha\beta}$ where R^* is the radius of the critical nucleus and $R^* = -\frac{2\sigma_{\alpha\beta}}{\Delta G_{vol}}$. The nucleation barrier can be written as,

$$\Delta G^* = \frac{16\pi}{3} \frac{\sigma_{\alpha\beta}^3}{(\Delta G_V + \Delta G_S)^2} \tag{5.2}$$

where ΔG_V is the chemical driving force per volume for nucleation and calculated directly from thermodynamic database with $\Delta G_V = \frac{\Delta G_m}{V_m}$, ΔG_S is the elastic strain energy per volume of precipitate. $\sigma_{\alpha\beta}$ is the interfacial energy of the matrix/particle interface. For a spherical nucleus,

$$Z = \frac{V_\alpha}{2\pi N_A R^{*2}}\sqrt{\frac{\sigma_{\alpha\beta}}{k_B T}} \tag{5.3}$$

$$\beta^* = \frac{4\pi R^{*2}}{a^4}D_{eff} \tag{5.4}$$

$$\tau = \frac{1}{2Z^2\beta^*} \tag{5.5}$$

Where V_α is the molar volume of the matrix, and a is the lattice constant of the precipitate phase and D_{eff} is the effective diffusivity $D_{eff} = \left[\sum_{i=1}^{n}\frac{(C_{pi}-C_{0i})^2}{C_{0i}D_{0i}}\right]^{-1}$ and is defined by [2004Svo], where C_{pi} is the mean concentration in the precipitate and C_{0i} is the mean concentration in the matrix phase. D_{0i} is the matrix diffusion coefficient of the component i.

The calculation of the elastic strain energy is given by Nabarro [1940Nab] for a homogeneous inclusion in an isotropic matrix,

$$\Delta G_s = \tfrac{2}{3}\mu\Delta^2 f(A_R) \tag{5.6}$$

Where μ is the shear modulus of the matrix. Thus the elastic strain energy is proportional to the square of the volume misfit Δ^2. The function $f(A_R)$ provided in [1940Nab] is a factor that takes into account the shape effects. For a given volume, a sphere ($A_R= 1$) has the highest strain energy while a thin plate ($A_R\rightarrow 0$) has a very low strain energy, and a needle shape ($A_R\rightarrow \infty$) lies between the two. Therefore the equilibrium shape of precipitates will be reached by balancing the opposing effects of interfacial energy and strain energy. When Δ is small, interfacial energy effect should dominate and the precipitates should be roughly spherical.

In homogeneous nucleation, the number of potential nucleation sites can be estimated by,

$$N_v = N_f\frac{N_A}{V_\alpha} \tag{5.7}$$

Where N_A is the Avogadro number. The value of nucleation site parameter N_f is an adjustable parameter and usually chosen close to solute concentration for homogeneous nucleation.

(b) Heterogeneous nucleation

For heterogeneous nucleation on dislocations, grain boundaries (2-grain junctions), grain edges (3-grain junctions) and grain corners (4-grain junctions), the potential nucleation sites N_v and the nucleation barrier ΔG^* must be adjusted accordingly.

There are two ways to account for heterogeneous nucleation in PanEvolution. One way is to treat nucleation site parameter N_f in Eq. 5.7 as phenomenological parameters and replace Eq. 5.2 by a user-defined equation. An example is given at Section 5.3.4 . The other way is to theoretically estimate the potential nucleation sites and calculate the nucleation barrier by assuming an effective interfacial energy for each nucleation site. For nucleation on grain boundaries, edges or corners, the nucleation barrier can be calculated by [1955Cle],

$$\Delta G^* = \frac{4}{27} \frac{(b\sigma_{\alpha\beta} - a\sigma_{\alpha\alpha})^2}{(c\Delta G_{vol})^2} \tag{5.8}$$

Where $\sigma_{\alpha\beta}$ is the interfacial energy and $\sigma_{\alpha\alpha}$ is the grain boundary energy and a, b, c are geometrical parameters, which are evaluated for grain boundaries, edges and corners. By introducing a contact angle, the grain boundary energy can be calculated from $\sigma_{\alpha\alpha} = 2\cos\theta\sigma_{\alpha\beta}$. An effective interfacial energy σ_{eff} is then introduced and the nucleation barrier is given as $\Delta G^* = \frac{16\pi}{3} \frac{(\sigma_{eff})^3}{(\Delta G_{vol})^2}$ with

$$\sigma_{eff} = \frac{b\sigma_{\alpha\beta} - a\sigma_{\alpha\alpha}}{c^{\frac{2}{3}}} \left(\sqrt[3]{\frac{1}{36\pi}} \right) \tag{5.9}$$

The effective interfacial energy σ_{eff} can be applied in the framework of classical nucleation theory. One can verify that the homogenous nucleation equations are recovered with $a=0, b=4\pi, c=4\pi/3$.

The potential nucleation sites on grain boundaries can be estimated from the densities for grain boundary area, grain edge and grain corner, which depend on the shape and size of grain in the matrix phase,

$$N_i = \rho_i \left(\frac{N_A}{V_\alpha}\right)^{i/3} \qquad i= 3, 2, 1, 0 \qquad\qquad (5.10)$$

and

$$N_v = N_f N_i \qquad i= 3, 2, 1, 0 \qquad\qquad (5.11)$$

where ρ_i are nucleation densities, $i=3$ for bulk nucleation, $i=2$ for grain boundary nucleation, $i=1$ for grain edge or dislocation nucleation and $i=0$ for grain corner nucleation. The nucleation densities for grain boundary, edge or corner are dependent on the grain size D and the aspect ratio $A=H/D$ in tetrakaidecahedron shape,

$$\rho_i = f_i\left(A\right) D^{3-i} \qquad i= 2, 1, 0 \qquad\qquad (5.12)$$

where $f_i(A)$ is a function of A and can be estimated for each case.

For nucleation on dislocations, the potential nucleation sites can be calculated from Eq. 5.11 if a dislocation density $\rho_1=\rho_d$ is given.

(c) Estimation of interfacial energy

The interphase boundary energy between the matrix and a precipitate phase or interfacial energy is the most critical kinetic parameter in the precipitation simulation. The generalized broken bond (GBB) method is used to estimate the interfacial energy for different alloys and temperatures,

$$\sigma_{sphere,dff} = 0.329\alpha_{SCF}\beta_{diff} \left(N_A V_\alpha^2\right)^{-1/3}\Delta H_{sol} \qquad\qquad (5.13)$$

The prefixed structure factor of 0.329 reflects the average proportion of broken bonds due to precipitations for fcc and bcc matrix. ΔH_{sol} is the solution enthalpy and calculated from thermodynamic database. α_{SCF} is a correction factor for the size of the precipitate particle. β_{dff} is a correction factor for diffuse interfaces. It is a function of T/T_C with T_C being the highest or critical temperature at which two phases are present in the system. At T_C, the composition of matrix and precipitate phases is the same and the interface energy equals to zero. At T=0 K, an ideal sharp interface is present. T_C can be provided by the user in the kdb file. If T_C is unknown and not provided, β_{dff} =1. Such an example is given in Section 5.3.2 .

5.2.2 Precipitation Growth

Two kinds of precipitation growth models are implemented in current PanEvolution module:

(a) Simplified Growth model

The growth of existing particles for each size class is computed by assuming diffusion-controlled growth, where the Gibbs-Thomson size effect is also taken into account. The growth model for multi-component alloys proposed by Morral and Purdy [1994Mor] is adopted in PanEvolution and modified to handle the growth/dissolution of various precipitate phases with different morphologies. The motion rate of the curved interface, e.g., the interface of a spherical or lens-like precipitate is given by,

$$v = \frac{dR}{dt} = \frac{K}{R}\left(\frac{2\sigma V_m}{R^*} - \frac{2\sigma V_m}{R}\right)$$

(5.14)

where R is the radius of the interface. R^* is the radius of the critical nucleus and $K = \frac{1}{\Delta C^{\alpha\beta}[M]^{-1}\Delta C^{\alpha\beta}\big)} \cdot \Delta C^{\alpha\beta}$ and $\Delta C^{\alpha\beta}$ are the row and column vector of the solute concentration difference between α (matrix phase) and β (precipitate phase), and [M] is the chemical mobility matrix. One can verify that the Eq. 5.14 can be further simplified and written as,

$$v = \frac{dR}{dt} = \frac{K}{R}\Delta G_m^*$$

(5.15)

where K is the kinetic parameter, and ΔG_m^* is the transformation driving force defined as $\Delta G_m^* = \Delta G_m - \Delta G_T$ with ΔG_m being the molar chemical driving force and $\Delta G_T = \frac{2\sigma V_m}{R}$ compensating the energy difference due to the Gibbs-Thompson effect.

(b) SFFK (Svoboda-Fischer-Fratzl-Kozeschnik) Model

The SFFK model [2004Svo] presents a set of linear equations describing the rate of change of radius and chemical composition of each precipitate in the system. Let the system consist of a matrix and a number of precipitates. The composition of each precipitate phase has boundary, which is determined by the complex lattice structure of the precipitate and the thermodynamic model used to describe the phase. The constraints can be written as,

$$\sum_{i=1}^{n} a_{ij} C_i = U_j, (j = 1, \ldots, p) \tag{5.16}$$

where the parameters a_{ij} take on the values 0 or 1.

For the description of the state of a closed system under constant temperature and pressure, the state parameters q_i can be chosen. Then under several assumptions for the geometry of the system and/or coupling of process, the total Gibbs energy G of the system can be expressed by means of the state parameters, and the rate of the total Gibbs energy dissipation Q can be expressed by means of $\dot{q}_i$. In the case of Q being a positive definite quadratic form of the rates $\dot{q}_i$, the evolution of the system is given by the requirement of the maximum of the total Gibbs energy dissipation Q constrained by $\dot{G} + Q = 0$ and by additional constrains which stem from the physical nature of the problem. Such a treatment is based on the thermodynamic extreme principle, which was formulated by Onsager in 1931 [1931Ons1, 1931Ons2].

Let μ_{0i} (i=1,...,n) be the chemical potential of component i in the matrix and μ_i (i=1,...,n) be the chemical potential ofcomponent i in the precipitate. All chemical potentials can be expressed as functions of the concentrations C_i. The total Gibbs energy of the system, G, is given by,

$$G = \sum_{i=1}^{n} N_{0i} \mu_{0i} + \frac{4\pi R^3}{3} \left(\lambda + \sum_{i=1}^{n} C_i \mu_i\right) + 4\pi R^2 \sigma \tag{5.17}$$

where σ is the interfacial energy and λ accounts from the contribution of the elastic energy and plastic work due to volume change of precipitates.

The evolution of the system corresponds to the maximum total dissipation rate Q and constrained $\dot{G} + Q = 0$ where,

$$\dot{G} = \frac{\partial G}{\partial R} \dot{R} + \sum_{i=1}^{n} \frac{\partial G}{\partial C_i} \dot{C}_i \tag{5.18}$$

The problem can then be expressed by a set of linear equations. By solving the linear equation Ay=B, one can obtain the particle growth rate as well as the composition change rate of each precipitate phase. The detail description of the SFFK model can be found in reference [2004Svo].

(c) Morphology evolution

The driving force for the evolution of the aspect ratio of the precipitate stems from the anisotropic misfit strain of the precipitate and from the orientation dependence of the interface energy. The SFFK model was modified to account for shape factor and its evolution during precipitation process [2006Koz, 2008Svo]. In the model, the precipitate shape is approximated by a family of cylinders having a length L and a diameter D as shown in Figure 5.3. The aspect ratio A_R is used to describe the precipitate shape given by A_R = L/D. The quantity R is the equivalent precipitate radius of the spherical shape with the same volume of the cylindrical precipitate. With this definition, small values of A_R represent discs, whereas large values of A_R represent needles as shown in Figure 5.3.
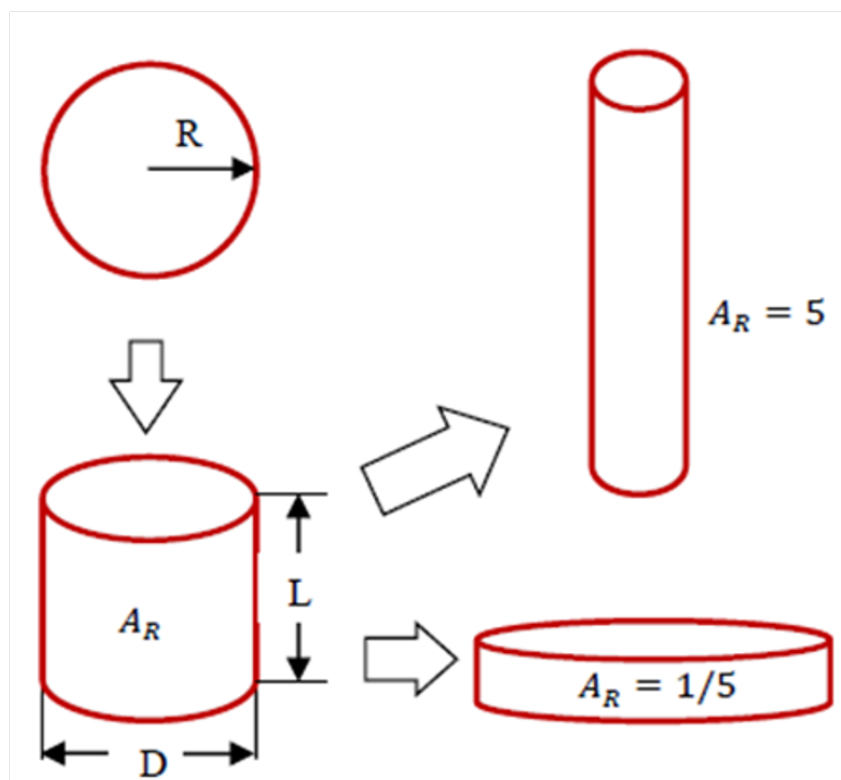
Figure 5.3: Schematic plot illustrating aspect ratio

In PanEvolution, the aspect ratio A_R of a precipitate phase can be treated as either a constant or a variable during evolution. In the former case, a set of shape factors are calculated based on A_R and the growth models are adjusted accordingly [2006Koz]. The shape factor S relating the surface of the cylindrical precipitate to the spherical precipitate is given by,

$$S = \frac{S_{cylinder}}{S_{sphere}} = 0.7631 A_R{}^{1/3} + 0.3816 A_R{}^{-2/3} \tag{5.19}$$

The shape factor K for interface migration of the precipitate is given by $K = 0.2912 A_R{}^{2/3} + 0.5824 A_R{}^{-1/3}$. The shape factor I for diffusion inside the precipitate is given by $I = 0.4239 A_R{}^{4/3} + 0.6453 A_R{}^{-2/3}$. The shape factor O for diffusion outside the precipitate is given by $O = 1.0692 A_R{}^{2/3}$.

In the latter case, the original SFFK model was modified and the evolution equations are described by a set of independent parameters including the effective radius (R), mean chemical composition (C_{ki}) and the aspect ratio A_R of each precipitate phase. The evolution rates of these parameters $\dot{R}$, $\dot{C}_{ki}$ and $\dot{A}_R$ are obtained by solving the linear equation Ay=B [2008Svo]. In the modified SFFK, the shape evolution is determined by the anisotropic misfit strain of the precipitate and by the orientation dependence of the interface energy. The total Gibbs energy of the system, G, is given by

$$G = \sum_{i=1}^{n} N_{0i} \mu_{0i} + \sum_{k=1}^{m} \frac{4\pi R_k^3}{3} \left(\lambda_k \left(A_{Rk} \right) + \sum_{i=1}^{n} c_{ki} \mu_{ki} \right) \\ + \sum_{k=1}^{m} 4\pi R_k^2 \frac{\beta_k^2}{8} \left(\sigma_k^L + 2 A_{Rk} \sigma_k^D \right) \tag{5.20}$$

The first term is the chemical part of the Gibbs energy of the matrix, the second term corresponds to the stored elastic energy and the chemical part of the Gibbs energy of the precipitates and the third term represents the total precipitate/matrix interface energy. The subscripts "0" denote quantities related to the matrix, e.g., N_{0i} is the number of moles of component i in the matrix and μ_{0i} is chemical potential in the matrix. The quantity $\lambda_k A_R$ accounts for the contribution of elastic strain energy due to the volume misfit between the precipitate and the matrix and is calculated from Eq. 5.6. μ_{ki} are the values of chemical potentials in the precipitates corresponding to c_{ki}. In the model, there are two interface

energies that must be assigned: σ_k^D at the mantle of the cylinder and σ_k^L at the bottom and top of the cylinder. The shape factor $\beta_k = \sqrt[3]{\frac{16}{3A_{Rk}}}$ is obtained by calculating equivalent radius of a sphere $R_k = D_k/\beta_k$ with the same volume of cylinder.

5.2.3 Precipitation Strengthening Model

The precipitation hardening arises from the interactions between dislocations and precipitates. Specifically, the dispersed precipitate particles act as pinning points or obstacles and impede the movement of dislocations through the lattice, and therefore strengthen the material. In general, either a dislocation would pass the obstacles by cutting through the small and weak particles (the shearing mechanism) or it would have to by-pass the strong impenetrable precipitates (the by- passing mechanism). The shearing mechanism is believed to predominate in lightly aged alloys with fine coherent precipitates or zones, while the by-passing mechanism is more characteristic of over-aged alloys with coarser precipitates.

Following an usual approach to determine the critical resolved shear stress at which a dislocation overcomes the obstacles in the slip plane, the response equation can be derived and written as [1979Ger,1998Des],

$$\sigma_P = \frac{M\bar{F}}{bL} \tag{5.21}$$

where M is the Taylor factor, $\bar{F}$ the mean obstacle strength, b the Burgers vector, and L the average particle spacing on the dislocation line. It has been shown that the Friedel statistics gives fairly good results in the calculation of the average particle spacing [1998Des]. Accordingly, this statistics is adopted and Eq. 5.21 becomes,

$$\sigma_P = \frac{M}{b\sqrt{2\beta Gb^2}}\sqrt{\frac{3V_f}{2\pi}}\frac{\bar{F}^{3/2}}{\bar{R}} \tag{5.22}$$

where G is the shear modulus, β a constant close to 0.5, V_f the particle volume fraction and $\bar{R}$ the mean particle size.

The calculation of the mean obstacle strength $\bar{F}$ in Eq. 5.22 is determined by the obstacle size distribution and the obstacle strength:

$$\bar{F} = \frac{\sum_i N_i F_i}{\sum i N_i} \tag{5.23}$$

where N_i is the obstacle number density of the size class R_i and F_i is the corresponding obstacle strength, which is related to the size of the obstacles and how the obstacles are overcome (particle shearing or particle by-passing).

In the case of particle shearing for weak obstacles, a rigorous expressions of the obstacle strength is quite complex and depends on different strengthening mechanisms (e.g., chemical strengthening, modulus hardening, coherency strengthening, ordering strengthening and so on). Therefore, we will not consider the detailed mechanisms involved. Instead, a more general model proposed by Gerold [1979Ger] is used in the present study and the obstacle strength of a precipitate of radius R is given by,

$$F = kGbR \tag{5.24}$$

where k is a constant and G is the shear modulus.

On the other hand, the obstacle strength is constant and independent of particle radius in the case of particle by-passing [1979Ger],

$$F = 2\beta Gb^2 \tag{5.25}$$

where β is a constant close to 0.5. From Eq. 5.24 and Eq. 5.25, one can find that the critical radius for the transition of the shearing and the by-passing mechanism is $R_C = \frac{2\beta b}{k}$. By treating R_C an adjustable parameter as suggested by Myhr et al. [2001Myh], the equations Eq. 5.24 and Eq. 5.25 can be written as,

$$F_i = \begin{cases} 2\beta Gb^2 \left(\frac{R_i}{R_C} \right) & if\ R_i < R_C\ \text{(weak particles)} \\ 2\beta Gb^2 & if\ R_i \geq R_C\ \text{(strong particles)} \end{cases} \tag{5.26}$$

Combining equations Eq. 5.22, Eq. 5.23 and Eq. 5.26 gives the yield strength contributed by a particle of radius R_i,

$$\sigma_{P,i} = \begin{cases} k_P \dfrac{\sqrt{V_f}}{\bar{R}} \left(\dfrac{\sum_i N_i \frac{R_i}{R_C}}{\sum_i N_i} \right)^{3/2} & if\ R_i < R_C\ \text{(weak particles)} \\ k_P \dfrac{\sqrt{V_f}}{\bar{R}} & if\ R_i \geq R_C\ \text{(strong particles)} \end{cases} \tag{5.27}$$

where $k_P = 2\beta GbM\sqrt{\frac{3}{2\pi}}$. This leads to the expression for the yield strength arising from precipitation hardening in the general case where both shearable (weak) and non-shearable (strong) particles are present,

$$\sigma_P = \sum_i \sigma_{P,i} \tag{5.28}$$

Eq. 5.27 and Eq. 5.21 yield the relationships $\sigma_P \sim \sqrt{V_f \overline{R}}$ and $\sigma_P \sim \frac{\sqrt{V_f}}{\overline{R}}$, respectively, in the two extreme case of pure shearing (i.e., all particles are small and shearable) and pure by-passing (i.e., all particles are bigger than the critical radius and non-shearable). The two relationships are consistent with the usual expressions derived from the classical models.

Besides the precipitate hardening σ_P , the other two major contributions should be considered in the calculation of the overall yield: 1) σ_0, the baseline contribution including lattice resistance σ_i, work-hardening σ_{WH} and grain boundaries hardening σ_{GB}; 2) σ_{SS}, the solid solution strengthening. If all three contributions can be estimated individually, the overall yield strength of the material can thus be obtained according to the rule of additions. Several types of these rules have been proposed to account for contributions from different sources [1975Koc, 1985Ard]. A general form is written as [1985Ard],

$$\sigma^q = \sum \sigma_i^q \tag{5.29}$$

when q = 1, it is a linear addition rule and it becomes the Pythagorean superposition rule when q = 2. The value of q can also be adjusted between 1 and 2 in terms of experimental data. When applying to aluminum alloys, the linear addition rule has been shown to be the appropriate one [1985Ard, 1998Des, 2001Myh, 2003Esm] and therefore the overall yield strength is given by the following equation,

$$\sigma = \sigma_0 + \sigma_{SS} + \sigma_P \tag{5.30}$$

where $\sigma_0 = \sigma_i + \sigma_{WH} + \sigma_{GB}$; it does not change during precipitation process. σ_{SS} is the solid solution strengthening term, which depends on the mean solute concentration of each alloying element [1964Fri, 2001Myh],

$$\sigma_{SS} = \sum_j a_j W_j^{2/3} \tag{5.31}$$

where W_j is the weight percentage of the jth alloying element in the solid solution matrix phase and a_j is the corresponding scaling factor. σ_P is the precipitation hardening term defined by Eq. 5.21. By applying a regression formula [1997Gro, 2001Myh], the yield strength σ in MPa can be converted to hardness HV in VPN as,

$$HV = A\sigma + B \tag{5.32}$$

where A and B are treated as fitting parameters in terms of experimental data.

5.2.4 Precipitation Hardening Model with Multiple Particle Groups

The calculation of precipitation strengthening σ_P in Eq. 5.30 can be extended to multiple particle groups with different strengthening mechanism.

$$\sigma_p = \sum_g f_g \sum_j W_j \sigma_p^j \tag{5.33}$$

Where g is the size group defined by a particle size range and f_g is the volume fraction of the group. j is the strengthening mechanism in each size group. $W_j = \frac{\sum_{i,R_i>R_C} N_i}{\sum_i N_i}$ is the weight fraction of the mechanism with R_C the critical radius for transition from one mechanism to another.
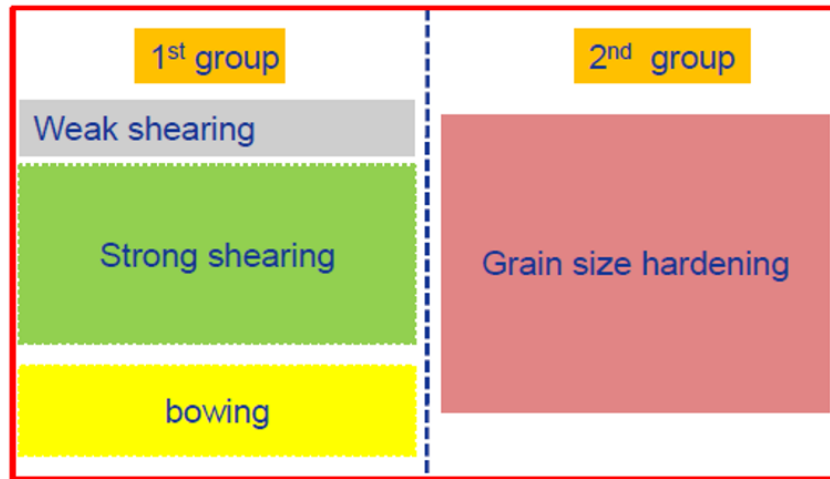
Figure 5.4: Schematic plot for strengthening groups

As an example shown in Figure 5.4, the first group has three mechanisms "weak shearing", "strong shearing" and "bowing" while the second group has only one mechanism. This is very typical in the γ' strengthened Ni-based superalloys. The big primary γ' acts to

limit grain growth during solution treatment and enhances grain boundary strengthening while the tertiary/secondary γ' particles strengthen materials through the weak/strong shearing and bowing mechanisms. In [2015Gal], Galindo-Nava proposed a unified model for weak/strong shearing and bowing mechanisms. The transitions from weak to strong shearing and to bowing are automatically taken care. The unified critical resolved shear stress $\tau_{shearing}$ is calculated as,

$$\tau_{shearing} = \frac{\gamma_{APB}l_1}{2b(\Lambda_1 - 2R)} \tag{5.34}$$

where l_1 is the length of the leading dislocation cutting the precipitates and defined as,

$$l_1 = \begin{cases} 2R & \text{if } R < R_c \text{ (weak pair-coupling)} \\ 2\sqrt{R^2 - (R - R_c)^2} & \text{if } R \geq R_c \text{ (strong pair-coupling)} \end{cases} \tag{5.35}$$

with the critical radius defined as,

$$R_C = \frac{\mu b^2}{2\gamma_{APB}} \tag{5.36}$$

where μ is the shear modulus and b is the magnitude of the Burgers vector. In Eq. 5.34, the mean distance Λ_1 is given by $\Lambda_1 = max\left(L\sqrt{\frac{T_{ten}}{\gamma_{APB}R}}, L - l_1\right)$ with $T_{ten} = \frac{\mu b^2}{2}$ and $L = R\sqrt{\frac{2\pi}{3V_f}}$.

The Orowan yield stress is given by,

$$\sigma_{bowing} = M\frac{3\mu b}{2L} \tag{5.37}$$

where M is the Taylor orientation factor. The total yield strengthen due to precipitation hardening can be calculated by Eq. 5.33.

5.2.5 Grain Growth/Coarsening

(a) One-size model

As a first approximation, a one-size grain evolution model is implemented. In this model, by giving an initial grain structure including average grain size and volume fraction, the new grain size at each time step is updated based on the coarsening rate defined in Eq. 5.38:

$$\dot{R} = M_{eff}P$$

(5.38)

In this model, the growth rate of recrystallized grains, $\dot{R}$ is expressed by the product of an effective mobility, M_{eff}, and the driving force **P**. The grain number density is then adjusted according to the conservation equation by maintaining the constant volume fraction. The grain boundary mobility is given by [2009Pay]:

$$M_{eff} = \frac{A_2}{\delta}\bar{M}$$

(5.39)

where $\bar{M}$ is the effective atomic mobility for a multi-component system, which is computed directly from mobility database $\bar{M} = \left[\sum_{i=1}^{n} \frac{C_i}{M_i}\right]^{-1}$ (C_i is the instant composition of the matrix phase and M_i is the atomic mobility); δ is the grain boundary width. A_2 is a factor related to the fraction of atomic jumps which result in a 'permanent' motion of the grain boundary in the direction of the driving force. The grain coarsening driving force **P**=**P**$_C$+**P**$_Z$ related to the local grain boundary curvature **P**$_C$ and the Zener drag force **P**$_Z$ due to the pinning effect of the precipitates [1985Nes]. **P**$_C$ is given by:

$$P_C = \frac{2\gamma V_m}{R}$$

(5.40)

where R is the grain radius and γ is the grain boundary interfacial energy.

$$P_Z = \beta\gamma\sum_p \frac{f_p}{r_p}$$

(5.41)

where f_p and r_p are the volume fraction and average radius of each precipitate phase, which can be obtained from the precipitation modeling. β is a Zener drag force factor and the most used value is β=3/2.

(b) Multi-size model

In order to track the evolution of grain size distribution (GSD), the Multi-size model is implemented based on the KWN model developed for the simulation of recrystallization (not implemented in this version) and grain coarsening processes. In the Multi-size model, the continuous grain size distribution (GSD) is divided into a large number of size classes. The

program takes a simulation step at every sample time hit. To maintain both accuracy and efficiency between two adjacent simulation steps, a fifth-order Runge-Kutta scheme is used to generate an adaptive step size based on the continuity equation and the growth/coarsening rate.

The coarsening of existing grains for each size class is given by [2009Pay]:

$$\dot{R}_i = \alpha M_{eff} \gamma V_m \left(\frac{1}{R_c} - \frac{1}{R_i} + Z \right)$$

(5.42)

where R_c is the critical radius and can be calculated from by $R_c = \frac{\bar{R^2}}{\bar{R}}$, α is a constant and α = 2 for spherical grains. The effective grain boundary mobility M_{eff} is defined in Eq. 5.39. The Zener drag effect **Z** can be obtained from Eq. 5.41.

After the coarsening rate of each size class is calculated, the number of grains in each size class is re-distributed and normalized to maintain the constant volume fraction of the system. To improve the efficiency, the number of size classes is fixed in the Multi-size model and the width of each size class is automatically adjusted according to the average grain size.

5.2.6 Evolution of Dislocation Density

In PanEvolution, the evolution of dislocation density for the deformed (DEF) matrix grains and new recrystallized (DRX) grains are both evaluated based on the Kocks-Mecking (KM) model [2003Koc, 2016Lin]. For the DEF grains, the evolution of dislocation density is calculated by,

$$\dot{\rho}_{DEF} = \dot{\rho}^+ - \dot{\rho}_{drv}^-$$

(5.43)

where $\dot{\rho}^+$ is due to work hardening, $\dot{\rho}_{drv}^-$ is due to dynamic recovery. And the increase of the dislocation density during hot deformation is termed as,

$$\dot{\rho}^+ = (f_w \sqrt{\rho_i} + 1/(bD))\dot{\varepsilon}$$

(5.44)

Where f_w is the work hardening coefficient and relevant to the dislocation substructures, grain boundaries, precipitates, etc. ρ_i is the mean dislocation density for the DEF grain or the DRX grain. D is the average size of deformed grains and b is Burgers vector.

The dynamic recovery term $\dot{\rho}_{drv}^-$ is proportional to dislocation density,

$$\dot{\rho}_{drv}^- = f_v \rho_i \dot{\varepsilon}$$

(5.45)

where f_v is the dynamic recovery coefficient and depends on initial deformation conditions.

For the DRX grains, the evolution of dislocation density is also calculated by Eq. 5.43 with an additional softening factor f_x is introduced as [2020Cai],

$$\dot{\rho}_{DRX} = f_x \dot{\rho}^+ - \dot{\rho}_{drv}^-$$

(5.46)

Then the evolution of mean dislocation density can be obtained by,

$$\dot{\rho}_{mean} = v_{DEF}\dot{\rho}_{DEF} + v_{DRX}\dot{\rho}_{DRX}$$

(5.47)

where v_{DEF} and v_{DRX} are the volume fractions of DEF and DRX grains, respectively. Then, the relationship between the stress arising from the dislocation interactions, σ_M, and the mean dislocation density of the material, ρ_{mean} , can be described as,

$$\sigma_M = M\alpha\mu b\sqrt{\rho_{mean}}$$

(5.48)

where M is the Taylor factor. α is a constant depending on the dislocation/dislocation interaction and can be taken as 0.3. μ is shear modulus, and b is Burgers vector. Then the overall yield strength in Eq. 5.30 becomes $\sigma = \sigma_i + \sigma_{ss} + \sigma_P + \sigma_M$.

5.2.7 The JMAK Model for Recrystallization

As a first approximation, the phenomenological JMAK model (Johnson-Mehl-Avrami-Kolmogorov) [1937Kol, 1939Joh, 1939Avr, 1940Avr] has been implemented to describe the recrystallization process:

$$X(t) = 1 - exp[-\ln 2(\frac{t}{t_{0.5}})^n]$$

(5.49)

where X is fraction recrystallized at time t and n is a constant. The recrystallization kinetics expressed as the time to reach a certain level of recrystallization depends on chemical composition, applied strain ε, strain rate $\dot{\varepsilon}$, initial grain size D_0, and temperature T. According to Sellars [1979Sel], the time for 50% RX can be estimated by,

$$t_{0.5SRX} = f(\zeta_i)Z^p$$

(5.50)

where p is an exponent, $f(\zeta_i)$ a function of a parameter set ζ_i which often include strain rate, initial grain size and temperature. Z is the Zener-Hollomon parameter and is defined as

$$Z = \dot{\varepsilon}\,\exp(\tfrac{Q_d}{RT})$$

(5.51)

with $\dot{\varepsilon}$ is the strain rate, R is the gas constant and Q_d a deformation activation energy that is a material parameter. Then the recrystallized grain size D_{RX} can be estimated from a relation in the form,

$$D_{RX} = f(\zeta_j)Z^q$$

(5.52)

here q is an exponent and $f(\zeta_j)$ a function of a parameter set ζ_j which often include the initial average grain size and the accumulated plastic strain. These parameters are defined in KDB file and can be adjusted based on the experimental data.

Thus, the average size of all grain D can be written as,

$$D = XD_{RX} + (1 - X)D_0$$

(5.53)

where D_0 and D_{RX} are initial and recrystallize grain sizes, respectively. X is recrystallized volume fraction.

5.2.8 The Fast-Acting Physical Model for Recrystallization

The Fast-Acting (F-A) model for RX is based on the mean field theory by considering both nucleation and growth of the new recrystallized grains. The nucleation of recrystallization is assumed to occur at deformed grain boundaries once the accumulation of dislocations reaches the critical dislocation density.

During deformation, the nucleation rate is evaluated as a function of both temperature T and $\dot{\varepsilon}$ strain rate. The growth of the new grains (mean grain size) is determined by both interface mobility and nucleation rate of small grains at the critical size. The F-A model uses the dislocation density model as input to estimate the driving force for the growth of the new recrystallized grains.

5.2.8.1 Nucleation and growth of recrystallization

The recrystallization model is based on dislocation density theory where the generated dislocations during deformation provide the driving force for recrystallization. The nucleation of the DRX occurs at grain boundaries once the accumulation of dislocations reaches the critical dislocation density defined by,

$$\rho_c = \left(\frac{20\gamma_{HB}\dot{\varepsilon}}{3bLM_{HB}\tau^2}\right)^{\frac{1}{2}}$$

(5.54)

Where γ_{HB} is high-angle grain boundary energy; $L = \frac{K_{\mu b}}{\sigma} = \frac{K}{\sqrt{\rho_{mean}}}$ is the dislocation mean free path, which is ~ $\frac{10}{\sqrt{\rho_{mean}}}$ for the most metals; $\tau = \beta\mu b^2$ is the dislocation line energy, where β is a constant of the order of 0.5; M_{HB} is the high-angle grain boundary mobility.

It is assumed that a nucleation event can occur during RX when a nucleus located on a grain boundary reaches a critical radius, R_c. The critical nucleus size corresponds to the condition when the stored energy difference between the DEF grain and the new RX grain or the nucleation driving force, $\Delta G = \beta\mu b^2(\rho_{DEF} - \rho_{RX})$, is large enough to overcome the capillary force of the nucleus. Then, the critical nucleus size is defined as,

$$R_c = \frac{2\gamma_{HB}}{\Delta G}$$

(5.55)

The nucleation rate for DRX as a function of both temperature T and strain rate $\dot{\varepsilon}$ is calculated by [1993Pec]:

$$\dot{N}_{DRX} = C\dot{\varepsilon}^\eta e^{-\frac{Q_N}{RT}}\left(1 - X_{RX}\right)$$

(5.56)

where C is the nucleation parameter,

$$C = N_f N_i = N_f \rho_i \left(\frac{N_A}{V_\alpha}\right)^{2/3}$$

(5.57)

with $\rho_i = \frac{1}{D}$ being the nucleation densities, D the grain size and N_f the nucleation site parameter, which can be estimated either by experiment or the inverse analysis method; Q_N is the activation energy of nucleation and exponent η is usually set to be 1. X_{RX} is the recrystallized volume fraction.

The nucleation rate for SRX (Static Recrystallization) is given by [2006Iva],

$$\dot{N}_{SRX} = C(E_{def} - E_c)e^{-\frac{Q_N}{RT}}(1 - X_{RX})$$

(5.58)

where E_{def} is the deformation-stored energy and can be calculated from the dislocation density, ρ, as $E_{def} = \beta \mu b^2 \rho$. E_c is the critical stored energy for initiating SRX, which can be determined from the critical deformation strain as $E_c = 10^7 \gamma_{HB} \frac{\varepsilon_c}{2.2\varepsilon_c + 1.1}$, where the critical strain ε_c generally ranges from 0.05 to 0.1 for different materials.

Once nucleated, the growth rate of recrystallized grains, $\dot{R}$ [m/s] is expressed by the product of an effective mobility for high angle grain boundary, M_{eff}, and the driving pressure as:

$$\dot{R} = M_{eff}P$$

(5.59)

In Eq. 5.59, the effective grain boundary mobility linking grain boundary migration velocity to driving pressure is given by [2009Pay]:

$$M_{eff} = \frac{A_2 V_m D_{eff}}{\delta RT}$$

(5.60)

Where D_{eff} is the effective self-diffusivity of the bulk atoms for a multi-component system, which is computed directly from CompuTherm mobility database; V_m is the molar volume. δ is the grain boundary width. A_2 is a factor related to the fraction of atomic jumps which result in a 'permanent' motion of the grain boundary in the direction of the driving force and a value of "1.0" is assumed. The driving pressure P contains one component $P_C = -(2\gamma_{HB})/R$ related to the local grain boundary curvature. This pressure component is derived from the grain boundary energy and is also a function of the local grain boundary curvature. Additionally,

the driving pressure P contains a term $P_D = \tau[\rho]$ that is derived from the stored energy and hence related to the jump in dislocation density across the grain boundary. Finally, the driving pressure P will also be influenced by a term P_Z due to the presence of second phase precipitation. The influence of precipitates on recrystallization kinetics is incorporated via the Zener pressure, which is given for a spatial distribution of spherical precipitates by Neset al. [1985Nes] as $P_Z = -\frac{3f\gamma_{HB}}{2r}$ with f the volume fraction of precipitates in mean radius r. Then, the total driving pressure can be calculated by,

$$P = P_D + P_C + P_Z = \tau[\rho] - \frac{2\gamma_{HB}}{R} - \frac{3f\gamma_{HB}}{2r} \tag{5.61}$$

In the Fast-Acting model, only the mean radius R and mean dislocation density ρ_{DRX} is tracked for the recrystallized grains. At each time step, the mean values must account concurrently for the increases of existing grains and the contributions due to the nucleation of the new grains.

5.2.9 Analytical models for Stress-Strain (σ-ε) curve at low/ambient temperatures

In PanEvolution, the modified tri-stage constitutive model [2023Yan] was adopted to describe the σ-ε behaviors at low to ambient temperatures as shown in Figure 5.5
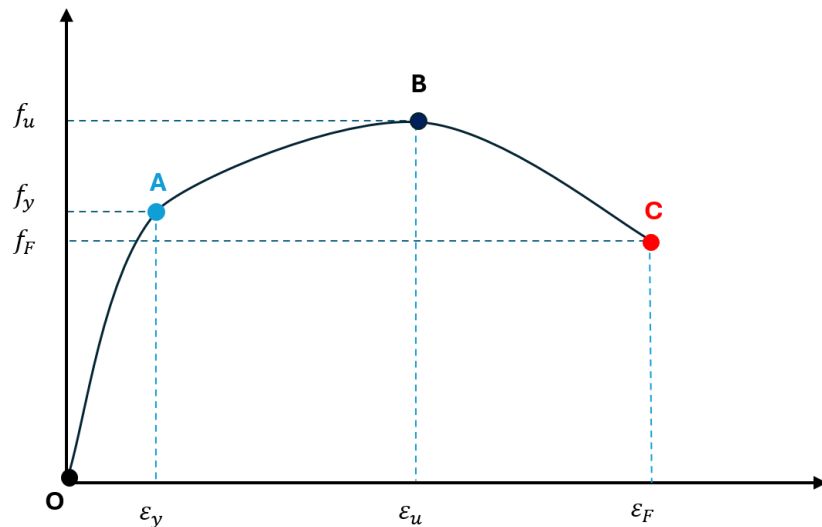
Figure 5.5: The tri-stage model at low/ambient temperatures

In this model, before or after the yielding point A, the relationship between stress (σ) and strain (ε) can be described by the modified RO model by Hill [1944Hill]. Beyond the ultimate strength point B, the recession branch of σ-ε models is adopted [2019Tao]. Eq. 5.62 - Eq. 5.64 summarizes the models at different stages:

$$\sigma_{O-A} = f_y * \left(\frac{\varepsilon}{0.2\%}\right)^{n_y} \tag{5.62}$$

$$\sigma_{A-B} = f_u * \left(\frac{\varepsilon - \varepsilon_y}{\varepsilon_u - \varepsilon_y}\right)^{n_u} \tag{5.63}$$

$$\sigma_{B-C} = f_u * \left(1 + K * \left(\frac{\varepsilon - \varepsilon_u}{\varepsilon_F - \varepsilon_u}\right)^{n_u}\right) \text{ where } K = \frac{f_F - f_u}{f_u} \tag{5.64}$$

Where f_y (or ε_y), f_u (or ε_u) and f_F (or ε_F) are yield strength (or strain), ultimate strength (or strain) and fracture strength (or strain), respectively. n_y and n_u are the strain-hardening indexes before and after yielding point while n_F is the recessing index. These parameters influence the material's behavior and are often different from alloy systems. Constant or analytical expression of these parameters can be defined in the KDB and then optimized for various applications.

5.3 The Kinetic Parameters Database (KDB file) Syntax and Examples

The kinetic parameters database (.KDB file) uses the XML format, which defines the kinetic model for each precipitate phase and its corresponding model parameters such as interfacial energy, molar volume, nucleation type, morphology type, and so on. In the KDB, a series of alloys can be defined. Each alloy has a matrix phase with one or multiple precipitate phases. A sample kdb structure is shown as follows,

```
<Alloy name="Fe-Mn-C">
<MatrixPhase name="Bcc">
  <ParameterTable type="kinetic" name="Parameters for Bcc">
  </ParameterTable >
  <PrecipitatePhase name="Cementite_GB"
        phase_name="Cementite" model="kwn"
   morphology="Sphere" nucleation="Grain_Boundary" growth="simp">
    <ParameterTable type="kinetic" name="Parameters for Cementite">
```

```
        <Parameter   type="Molar_ Volume"   value="6E- 6"   description="Molar
Volume" />
      <Parameter type="Interfacial_Energy" value="0.2"
          description="Interfacial Energy" />
    </ParameterTable >
  </PrecipitatePhase >
  <PrecipitatePhase name="M7C3_GB" phase_name="M7C3" model="kwn"
   morphology="Sphere" nucleation="Grain_Boundary"  growth="simp">
   <ParameterTable type="kinetic" name="Parameters for M7C3">
    <Parameter type="Molar_Volume" value="6E-6"
        description="Molar Volume" />
    <Parameter type="Interfacial_Energy" value="0.1" description=
       "Interfacial Energy" />
   </ParameterTable >
  </PrecipitatePhase >
  </MatrixPhase >
 </Alloy >
```

In this sample KDB, an alloy "`Fe-Mn-C`" is defined with the matrix phase "`Bcc`", which has two precipitate phases "`Cementite_GB`" and "`M7C3_GB`". The precipitate name, kinetic model, morphology, as well as nucleation and growth model can be defined for each precipitate phase. Available options for the kinetic models and morphology are given in Table 11.6 in Section 11.3 . A set of parameters for each phase, such as molar volume, interfacial energy, and so on, can be defined in "`ParameterTable`". The kinetic and mechanical model parameters that can be defined under "`ParameterTable`" are listed in Table 11.8 to Table 11.11 in Section 11.3 .

A few examples are given below to explain the content of the precipitation database in detail. In the reference folder, test batch files (.pbfx) are prepared with .kdb files for running example simulations.

5.3.1 An Example KDB File for Ni-14Al (at%) Alloy

Reference Folder: `$Pandat_Installation_Folder\Pandat 2024 Examples\PanEvolution\Ni-14%Al\Ni-14Al_Precipitation.kdb`

This KDB file defines the kinetic parameters for the L12_Fcc (γ') phase in the Ni-Al binary system at the Ni-rich side. The matrix phase is Fcc_A1 (γ) phase. The detail explanations of kinetic parameters are listed in Table 11.8 in Section 11.3 . This KDB file can be used for multi-component Ni-based superalloys as well, but some of the key parameters, such as the interfacial energy, and/or nucleation site parameter, may need to be calibrated accordingly. The molar volume and atomic spacing for various types of crystal structures can be found or estimated following reference of ASM handbook. The interfacial energy between γ and γ' for nickel alloys is usually in the range of 0.02-0.035 J/m^2, it is given as a constant in this example. Interfacial energy can be estimated as discussed in Section 5.2.1 , and an example is given in the Section 5.3.2 .

```xml
<Alloy name="NI-14Al_KWN">
<MatrixPhase name="Fcc">
  <ParameterTable type="kinetic" name="Parameters for gamma">
   <Parameter type="Molar_Volume" value="7.1E-6"
        description="Molar Volume" />
  </ParameterTable >
<PrecipitatePhase name="L12_FCC" model="KWN" morphology="Sphere"
        nucleation="Modified_Homo" growth="SFFK">
  <ParameterTable type="kinetic" name="Parameters for Gamma_prime">
   <Parameter type="Molar_Volume" value="7.1E-6"
        description="Molar Volume" />
   <Parameter type="Interfacial_Energy" value="0.025"
        description="Interfacial Energy" />
   <Parameter type="Atomic_Spacing" value="3.621E-10"
        description="Atomic Spacing" />
   <Parameter type="Nucleation_Site_Parameter" value="0.001"
        description= "Nucleation Site Parameter" />
  </ParameterTable >
</PrecipitatePhase >
</MatrixPhase >
</Alloy >
```

5.3.2 An Example KDB File for Ni-14Al (at%) Alloy Using the Calculated Interfacial Energy

Reference Folder: `$Pandat_ Installation_ Folder\Pandat 2024 Examples\PanEvolution\Ni-14%Al_IFE_CALC\Ni-14Al_Precipitation.kdb`

In KDB file, use keyword "`IFE_CALC(*)`" to get the calculated interfacial energy. The value can also be an expression, such as: `value="1.2 * IFE_CALC(*)"`. The optional parameter "`Interfacial_Energy_Tc`" is used to define the temperature at which γ' forms from γ congruently, and T_C represents the congruent point. In many cases this point does not exist. This parameter is optional and users should not give an arbitrary value to this parameter since it will lead to unrealistic results. In this example, this parameter is commented out in the Ni-14Al_Precipitation.kdb kinetic database. This means the correction factor β_{eff} in Eq. 5.13 is set to be 1.

```
   <Parameter type="Interfacial_Energy" value = "IFE_CALC(*)" description
= "Interfacial Energy" />
```

5.3.3 An Example KDB File for AA6005 Al Alloy

Reference Folder: `$Pandat_ Installation_ Folder\Pandat 2024 Examples\PanEvolution\AA6005_yield_strength\AA6xxx.kdb`

This KDB file defines the kinetic parameters for the Mg_5Si_6 (β') phase within the Al-Mg-Si alloys at the Al-rich side. The matrix phase is Fcc_A1-(Al) phase. This example `AA6xxx.kdb` can be used for most of the AA6xxx and AA3xx series of alloys. Again, some of the key kinetic parameters may need to be slightly revised according to the chemical compositions. Note that, the strengthening model is also included, which enables us to simulate the yield strength (σ) as well as its contributions (σ_i, σ_{ss}, σ_p). Using the equation Eq. 5.30 , the hardness can then be obtained as well. Table 11.11 in Section 11.3 lists the strengthening model parameters defined in the precipitation database. These model parameters are adjustable and obtained via optimization to describe available experimental data.

```
   <Alloy name="AA6xxx">
   <MatrixPhase name="Fcc">
       <ParameterTable type="Strength" name="Parameters for Fcc">
```

```
        <Parameter type="Solution_Strengthening_Factor" name="Si"
          value="66.3"
          description="solid solution strengthening scaling factor" />
        <Parameter type="Solution_Strengthening_Factor" name="Mg"
          value="29"
          description="solid solution strengthening scaling factor" />
        <Parameter type="Solution_Strengthening_Factor" name="Cu"
          value="46.4"
          description="solid solution strengthening scaling factor" />
        <Parameter type="Solution_Strengthening_Factor" name="Li"
          value="20"
          description="solid solution strengthening scaling factor" />
        <Parameter type="Intrinsic_Strength" value="10" description=
          "intrinsic strength" />
        <Parameter type="Hardness_Factor" value="0.33" description=
          "intrinsic strength" />
        <Parameter type="Hardness_Constant" value="16" description=
          "intrinsic strength" />
      </ParameterTable >
      <ParameterTable type="kinetic" name="Parameters for (Al)">
        <Parameter type="Molar_Volume" value="1.0425E-5" description=
          "Molar Volume" />
      </ParameterTable >
    <PrecipitatePhase name="Mg5Si6" model="KWN" morphology="sphere"
      nucleation="Modified_Homo" growth="SFFK">
      <VariableTable name="Variables replacing built-in variables">
        <Parameter type="Nucleation_Barrier_Energy" value="7.475e-12/dgm
(*)^2"
          description="Nucleation_Barrier_Energy" />
      </VariableTable >
        <ParameterTable  type="kinetic"  name="Parameters  for  kinetic
model">
        <Parameter type="Molar_Volume" value="3.9e-5"
          description="Molar Volume" />
        <Parameter type="Interfacial_Energy" value="0.4"
          description="Interfacial Energy" />
        <Parameter type="Atomic_Spacing" value="4.05E-10"
          description="Atomic Spacing" />
       <Parameter type="Nucleation_Site_Parameter" value="0.3e-5"
```

```
            description="Nucleation Site Parameter" />
        <Parameter type="Steady_State_Nucleation_Rate" value="1"
         description="Indicate whether or not steady state nucleation
rate" />
        <Parameter type="Driving_Force_Factor" value="1.0"
          description="Driving Force Factor" />
        <Parameter type="Kinetic_Parameter_Factor" value="3.0"
          description="a factor for kinetic parameter" />
        <Parameter type="Effective_Diffusivity_Factor" value="3"
          description="a factor for effective diffusivity" />
      </ParameterTable >
      <ParameterTable type="strength" name="Parameters for
        strengthening models">
          <Parameter type="Strength_Parameter" value="1.1e-5"
          description="th k_ppt in the equation" />
        <Parameter type="Shearing_Critical_Radius" value="5.0e-9"
           description="Critical  size  of  shifting  from  shearing  to
looping" />
        </ParameterTable >
      </PrecipitatePhase >
    </MatrixPhase >
    </Alloy >
```

5.3.4 An Example KDB file for Heterogeneous Nucleation

Reference   Folder: $Pandat_ Installation_ Folder\Pandat   2024
Examples\PanEvolution\AA6005_yield_strength\AA6xxx.kdb

There are two parameters need to be adjusted in order to consider heterogeneous nucleation. In this example, the heterogeneous nucleation sites and barrier energy are adjusted manually by user:

a. Nucleation_Site_Parameter; in the above example of AA6005 alloy, the value of 0.3e-5 was chosen based on experimental data.

b. Re-define the nucleation barrier energy. In this example, the nucleation barrier energy for the heterogeneous nucleation is manually adjusted. For homogeneous nucleation, the nucleation barrier energy is defined by Eq. 5.2 with
$$\Delta G^* = \frac{16\pi}{3} \frac{(\sigma_{\alpha\beta})^3}{(\frac{\Delta G_m}{V_m})^2} = \frac{16\pi}{3} \frac{(V_m)^2(\sigma_{\alpha\beta})^3}{(\Delta G_m)^2} = \frac{16\pi}{3} \frac{(V_m)^2(\sigma_{\alpha\beta})^3}{dgm(*)^2}$$ Taking Vm = 3.9e-5 and σ = 0.4 in

the kdb file as shown in Section 5.3.3 , we get the nucleation barrier energy for the homogeneous nucleation as `1.63e-9/ dgm(*)^2`. For heterogeneous, the nucleation barrier energy is ~2-3 order smaller than that of homogeneous nucleation. In this example, a value "`7.475e-12/dgm(*)^2`" is used. This value is optimized by using experimental data.

```
<VariableTable name="Variables replacing built-in variables">
<Parameter type="Nucleation_Barrier_Energy" value="7.475e-12/dgm(*)^2"
description="Nucleation_Barrier_Energy"/>
</VariableTable >
```

where dgm(*) is the calculated thermodynamic driving force, which will be update at every simulation steps.

5.3.5 Another Example for Heterogeneous Nucleation

Reference Folder: `$Pandat_ Installation_ Folder\Pandat 2024 Examples\PanEvolution\Fe-Mn-C_Heterogeneous_Nucleation\PanFe.kdb`

In the example of Section 5.3.4 , the heterogeneous nucleation sites and nucleation barrier energy were treated as adjustable parameters and were given manually by the user based on available experimental data. In this example, the potential heterogeneous nucleation sites and nucleation barrier energy are estimated by theoretical models as discussed in Section 5.2.1 (b) depending on the selected nucleation type.

```
<PrecipitatePhase    name="Cementite_ GB"    phase_ name="Cementite"
model="kwn"   morphology="Sphere"   nucleation _ = _ "Grain_ Boundary"
growth="simp">
<PrecipitatePhase name="Cementite_Dislocation" phase_name="Cementite"
model="kwn" morphology="Sphere" nucleation="Dislocation" growth="simp">
```

The nucleation type can be specified when defining a precipitate phase. In this example, two different type of heterogeneous nucleation are defined for "`Cementite`" phase: one at "`Grain_Boundary`" and the other at "`Dislocation`".

5.3.6 An Example KDB File for a Mg-based AZ91 Alloy Considering Shape Factor and Shape Evolution

Reference   Folder: `$Pandat_  Installation_  Folder\Pandat  2024 Examples\PanEvolution\AZ91_Morphology\AZ91.kdb`

- Consider shape factor but shape factor not evolve (AZ91_200C.pbfx)

```
<PrecipitatePhase name="ALMG_GAMMA" model="kwn"
morphology="Cylinder" nucleation="Modified_Homo" growth="sffk">
```

In this case, the morphology must be set to "`morphology="Cylinder"`" and the keyword "Aspect_Ratio" or "A_R" is used to set the aspect ratio of the particles: `<Parameter type="A_R" value="0.15" description="aspect ratio" />`

In this example, the Aspect Ratio is set as a constant during the particle evolution.

- Shape evolution (AZ91_200C_shape_evolution.pbfx)

```
<PrecipitatePhase name="ALMG_GAMMA" model="KWN" morphology="Cylinder"
nucleation="Modified_Homo" growth="SFFK_Shape_Evolution">
```

In order to consider shape evolution, the morphology must be set to "`morphology="Cylinder"`" and the growth model should be set to "`growth="SFFK_Shape_Evolution"`".

In this case, the aspect ratio value set in "`<Parameter type="A_R" value="1" description="Initial aspect ratio" />`" will be used as a starting value and evolves during precipitation process.

There are two sets of parameters that can control the shape evolution. The first set includes the interfacial energies of the two different directions:

```
<Parameter type="Interfacial_ Energy" value="0.25" description=
"Interfacial Energy" />
<Parameter type="Interfacial_ Energy_ L" value="0.05" description=
"Interfacial Energy in L direction"/>
```

The second set includes the anisotropic misfit strain of the precipitate and its model parameters can be defined as follows:

```
<Parameter type="Shear_Modulus" value="30e9" description="the shear
modulus, in Pa"/>
```

```
   <Parameter type="Volume_Misfit" value="0.02" description="the volume
misfit"/>
```

It should note that user can set parameters for either interfacial energy or strain energy or both to control the shape evolution.

5.3.7 An Example to Show How to Set an Initial Microstructure for Simulation

Reference Folder: `$Pandat_ Installation_ Folder\Pandat 2024 Examples\PanEvolution\Ni-16%Al_Dissolution\Al-Ni.ini`

This example is to demonstrate how to set up the initial microstructure for dissolution simulation. In this example, the alloy composition is Ni-16Al at.%, which is isothermally annealed at 1444K for 100 seconds. The alloy composition and heat treatment condition is given in Ni-16Al_dissolution.pbfx. Initially, there is 10% (volume fraction) L12_Fcc (γ') phase with average particle size 1.2μm. The initial microstructure is set in Al-Ni.ini as shown below:

```
  <Condition name="C1">
    <MatrixPhase name="Fcc">
      <PrecipitatePhase name="L12_FCC">
        <Parameter name="size" value="1.2e-6" description="average size"
/>
         <Parameter  name="volume_ fraction"  value="0.1"  description=
"volume fraction" />
        <Parameter  name="particle_size_distribution"  value="2"  number_
cells =
          "200" sigma="0.25" description="initial psd shape: 0-uniform;
1-normal; 2-lognormal;10: user-defined psd" />
        <psd>
           <cell size="1.2E-08" number_density="170191212760.267" />
           <cell size="2.4E-08" number_density="199242482086.767" />
           <cell size="3.6E-08" number_density="232879833831.677" />
           <cell size="4.8E-08" number_density="271760886495.754" />
           <cell size="6E-08" number_density="316626417716.63" />
           ......
        </psd>
      </PrecipitatePhase >
    </MatrixPhase >
```

```
</Condition>
```

The "`<psd>` " section can be empty if uniform, normal or lognormal distribution is selected and the initial PSD is automatically generated based on the defined number_cells and standard deviation sigma. If the "`<psd>` " section is given, the provided psd data will be used and the defined `initial psd shape` (uniform, normal or lognormal distribution) is neglected.

User can also define the initial structure from GUI. User first chooses the target phase to set the initial structure and then set the average quantities including the volume fraction and average size. The PSD type can be "`uniform`", "`normal`", "`log_normal`" or "`user_defined`". When "`normal`" and "`log_normal`" are selected, the PSD data will be generated and the corresponding plot is shown on the right panel. In order to load a user defined PSD, the user can set PSD type to be "`user_defined`" and then load the PSD data file by clicking the import button. An example PSD data file "`psd_test.dat`" can be found in the folder: `$Pandat_ Installation_ Folder\Pandat 2024 Examples\PanEvolution\Ni-16%Al_Dissolution\`.
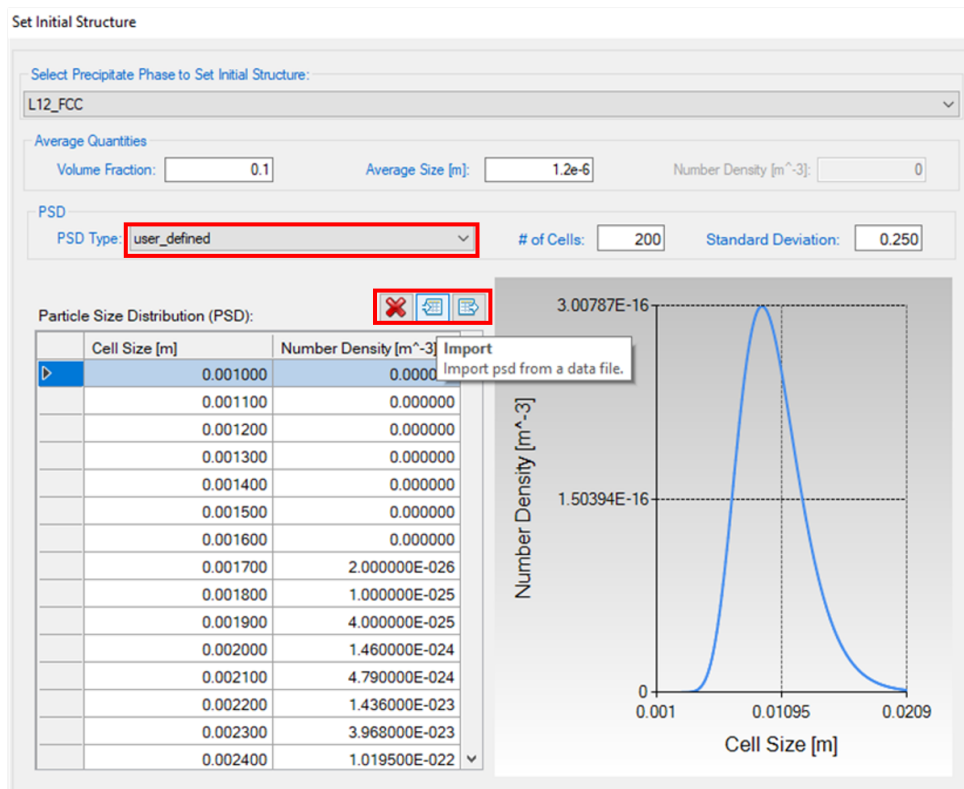
Figure 5.6: Dialog to define the initial structure from GUI

5.3.8 User-defined Growth Rate Model for Grain Coarsening

Reference   Folder: `$Pandat_  Installation_  Folder\Pandat    2024 Examples\PanEvolution\Grain_Coarsening\User_Defined_Model\Ni-14Al_ Grain_User_Defined_GrowthRate.kdb`

This example is to demonstrate how to set up a user-defined growth rate model for grain growth simulation. In this example, the built-in growth model can be replaced by the user-defined variable "`Growth_Rate`". In the built-in growth model, the growth rate is given by Eq. 5.38~Eq. 5.41, and it can be replaced by the expression `value=" KP (@Grain)/2.86e-10*2*0.75*7.1e-6/s(@Grain)"` where "`KP(@Grain)`" is to get the effective atomic mobility calculated from mobility database and "`s(@Grain)`" is to get the average grain size. The grain boundary width is "`2.86e-10`", grain boundary energy is "`0.75`" and molar volume is "`7.1e-6`". Below is the kdb defining the grain growth model

```
   <Grain     name="Fcc"     model="Grain_  OneSize"     morphology="Sphere"
nucleation=""growth="Grain_Simplified">
   <VariableTable name="Variables replacing built-in variables">
      <Parameter type="Growth_Rate"
   value="KP (@Grain)/2.86e- 10*2*0.75*7.1e- 6/s (@Grain)"   description="
Grain growth rate"/>
      </VariableTable>
      <ParameterTable type="kinetic" name="Parameters for grain">
   <!-- Using  Standard  Unit  --><Parameter  type="Interfacial_ Energy"
value="0.75" description="Interfacial Energy" />
      </ParameterTable >
   </Grain >
```

5.3.9 Concurrent Precipitation and Grain Growth Simulation

Reference   Folder: `$Pandat_  Installation_  Folder\Pandat    2024 Examples\PanEvolution\Grain_ Precipitation\Ni- 14Al_ Microstructure_ Zener.kdb`

This example is to demonstrate how to perform a simulation of concurrent precipitation and grain growth. In this example, the alloy composition is Ni- 14Al at.%, which is first isothermally annealed at 800 °C for 10000 seconds and then continuously cooled to 550 °C

from 10000 to 30000 seconds. The alloy composition and heat treatment condition are given in "`Ni-14Al_Microstructure.pbfx`". In the first simulation, the Zener pinning effect is not considered by default. Initially, there is no L12_Fcc (γ') phase and average grain size is 1.0μm in lognormal distribution. The initial microstructure is set in `Al-Ni.ini` as shown below

```
<Grain name="Fcc">
    <Parameter name="size" value="1e-6" description="average size" />
    <Parameter  name="volume_ fraction"  value="1"  description="volume
fraction" />
    <Parameter  name="size_distribution"  value="2"  number_cells="1000"
sigma="0.3"  description="initial  psd  shape:  0- uniform;  1- normal;  2-
lognormal;10: user-defined psd" />
</Grain>
```
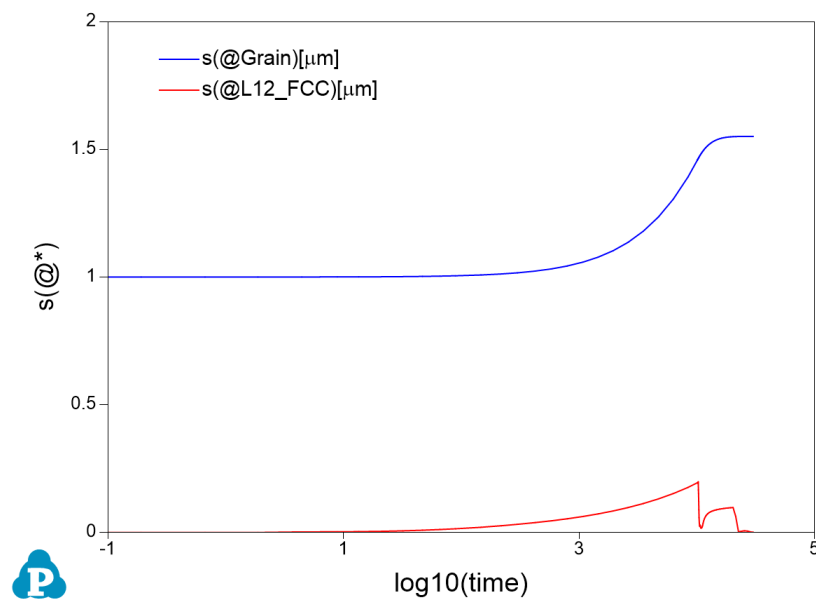

Figure 5.7: Evolution of the average size of the grains and the precipitates
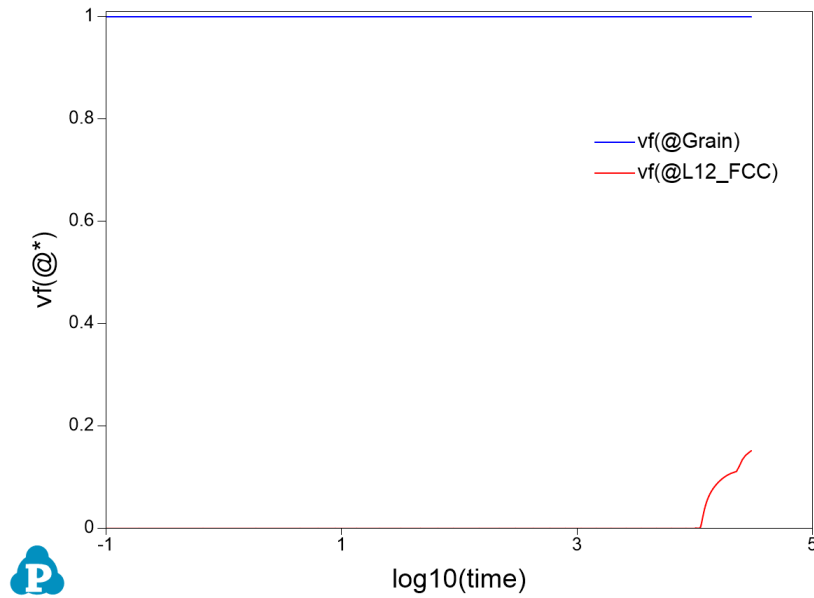
Figure 5.8: Evolution of the volume fractions of the grain and the precipitate phase

Figure 5.7 and Figure 5.8 show the evolution of the average size and volume fraction of the grain and precipitate phase. As can be seen, during the isothermal annealing at 800 °C, the precipitate phase nucleates. Both grain and precipitate phase start to grow. At the end of the isothermal annealing, there is around 10% γ' phase and the average grain size grows from 1 μm to around 1.5 μm. In the following continuous cooling stage, secondary and tertiary γ' form as indicated in Figure 5.9. Accordingly, the average particle size decreases due to the precipitation of these small particles (Figure 5.7 in red color). The grain growth rate slows down at low temperatures (Figure 5.7 in blue color).
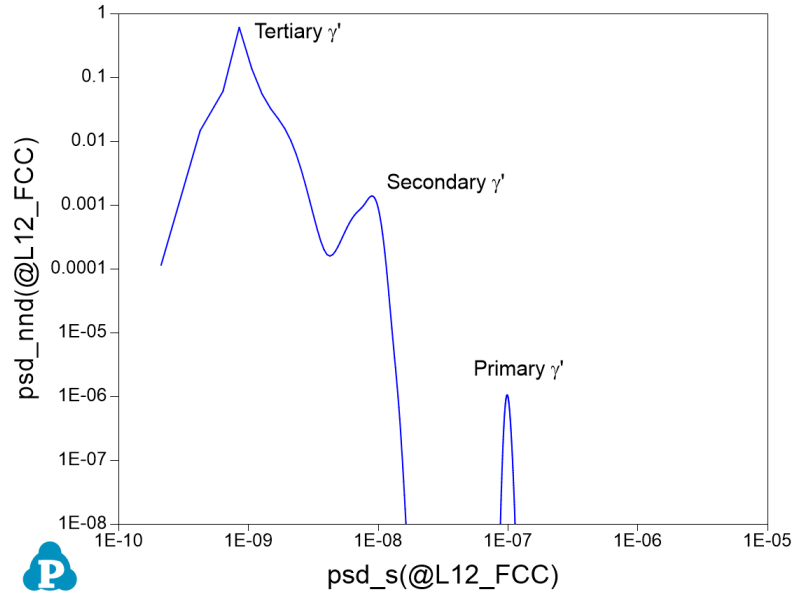
Figure 5.9: Particle Size Distribution (PSD) of γ' phase at the end of the simulation

5.3.10 Dynamic Recrystallization (DRX) Simulations with Built-in and User-defined Models

Reference   Folder: `$Pandat_  Installation_  Folder\Pandat    2024 Examples\PanEvolution\Grain_Recrystallization\Physical\`

In order to run a recrystallization simulation, four files are needed as shown in Figure 5.10: 1) "`Al-Ni.tdb`" – a Ni-Al binary thermodynamic and mobility database; 2) "`Ni-14Al_RX_ FA_DRX.kdb`" – a kdb file for kinetic models and parameters; 3) "`Al-Ni.ini`" – a file to define the initial grain structure. The initial grain size (radius) is set to 25.0µm in this case. 4) "`Ni-14Al_RX_950_FA_DRX.pbfx`" – a Pandat batch file to set up the simulation conditions including alloy chemistry, heat treatment schedule and output tables/graphs.

Figure 5.10: Files need for a recrystallization simulation
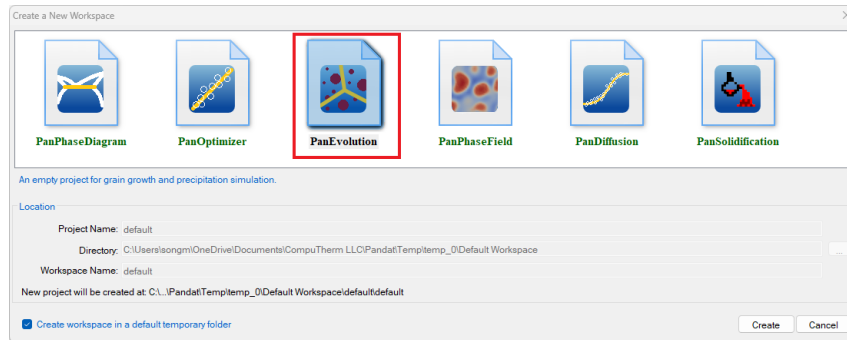
Figure 5.11: PanEvolution project in Pandat workspace

For model validations, all the simulations were performed in Pandat™ under a PanEvolution Project as shown in Figure 5.11. The first one is to simulate the dynamic recrystallization process by using the "`Fast-Acting`" model with the built-in nucleation model "`DRX`" and growth model "`RX_Simplified`". The processing history is defined in .pbfx file as:

```
<processing_history useRate="true">
    <node time="0" T="950" dT="0" EPS="0" dEPS="0.01"/>
    <node time="260" T="0" dT="0" EPS="0" dEPS="0"/>
</processing_history>
```

PanEvolution was designed to handle any arbitrary processing history. For demonstration, the processing condition is set to be 950 °C with strain rate at $0.01s^{-1}$. The simulation results are shown in Figure 5.12.
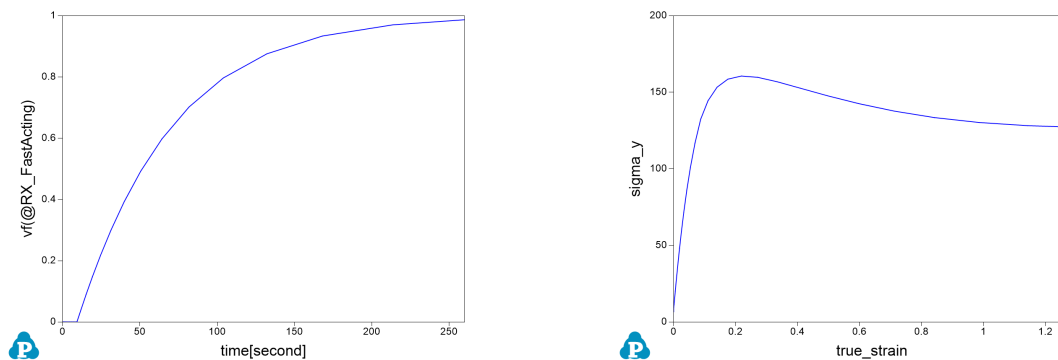
Figure 5.12: Simulations of DRX with the Built-in and User-Defined models

In order to run a user-defined growth model in KDB, another simulation was performed with the user- defined nucleation rate to overwrite the built- in model. In this case, a "`<VariableTable>`" sections is included in the KDB and the nucleation rate is defined by

```
<Parameter  type="Nucleation_ Rate"  value="0.1* (6.02e23/7.1E- 6)^
(2.0/3.0)/(max(1e- 6, Grain_Size_0*2))*fv(@Grain)*strain_rate*exp(-
224000/Rg/T)" description="user defined nucleation rate"/>.
```

In this case, the potential nucleation sites are less, and the nucleation of DRX is significantly lower and softening behavior due to DRX is also reduced as shown in Figure 5.13 (red color).
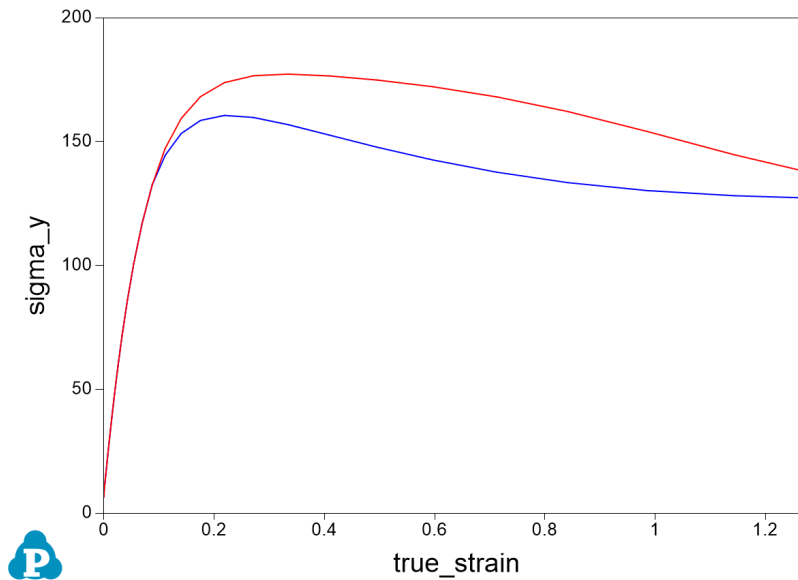
Figure 5.13: Simulations of DRX with the User-Defined models

5.3.11 Create or Edit kdb Parameters Through Pandat GUI

Since Pandat™ 2022, a Pandat™ GUI interface is introduced for users to create or edit kdb parameters, and then easily save and load the parameters. Click "`Create or Edit KDB`" from the PanEvolution menu. The "`Create or Edit kdb Parameters`" dialog will pop up as shown in Figure 5.14. Through this interface, users can set up the kinetic parameters for their precipitation and grain growth simulations from the default values
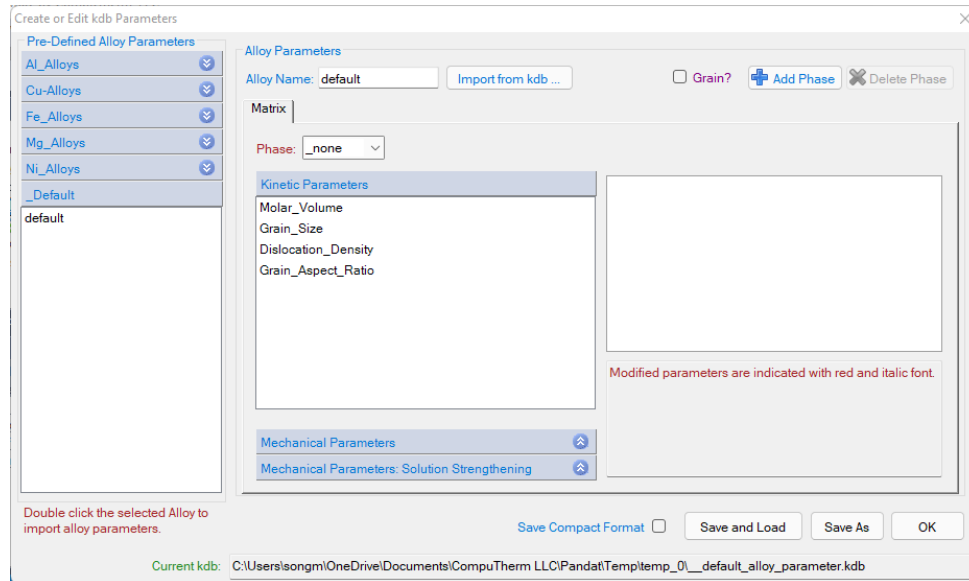
Figure 5.14: Dialog to create or edit kdb parameters

In this dialog, some pre-defined alloy parameters for Al alloys, Cu alloys, steel (Fe alloys), Mg alloys and Ni alloys are aslo provided for users to get started for developing their own kdb files. In order to import the parameters for a pre-defined alloy, one can just simply select and double click the alloy from the list. For example, after loading the thermodynamic and mobility database for Ni alloys, double click the IN100 alloy, the matrix Fcc (γ) phase related kinetic parameters are imported as shown in Figure 5.15, and the precipitate L12_ Fcc (γ') phase related parameters in Figure 5.16. The parameters highlighted in red italic are imported values different from the default values. User can further adjust the parameters for their own alloys systems. After all the parameters are properly set, one can click "`Save and Load`" , and then run the simulation. Or just click "`Save As`" to save a new kdb file for future simulations.
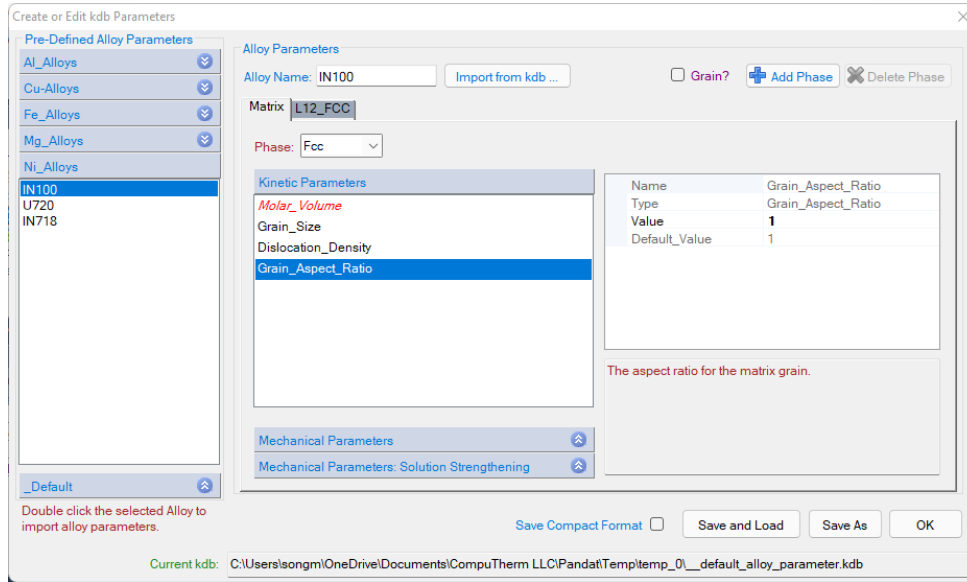
Figure 5.15: Parameters for the matrix phase in the Create or Edit kdb Parameter dialog
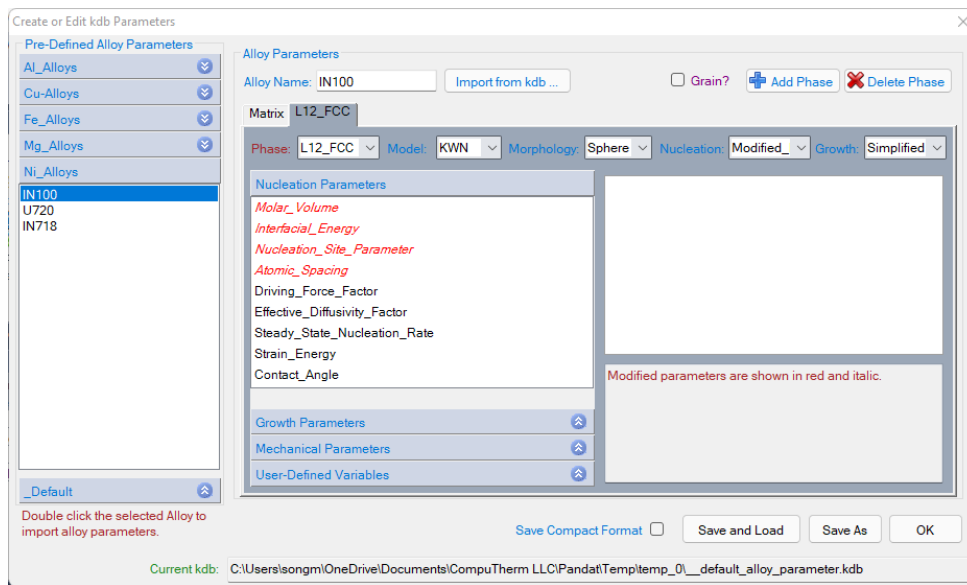
Figure 5.16: Parameters for the precipitate phase in the Create or Edit kdb Parameter dialog

When select the "Grain?" option, an item of "Matrix->Grain" will appear as shown in Figure 5.17. From here user can define the parameters required for grain growth simulation as described in Section 5.3.8   and Section 5.3.9 .
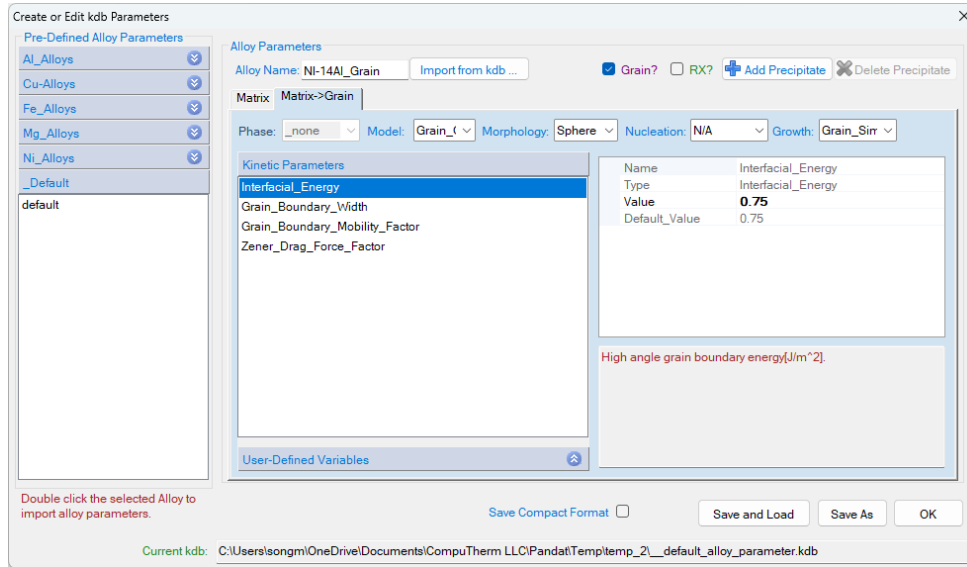
Figure 5.17: Parameters for the Matrix grain in the Create or Edit kdb Parameter dialog

After selected the "Grain?", another selection box of "RX?" will also appear for the recrystallization (RX) simulation. Select "RX?", the items of "`Grain->Dislocation`" and "`Grain->RX`" will appear for user to define the required dislocation density and recrystallization parameters.
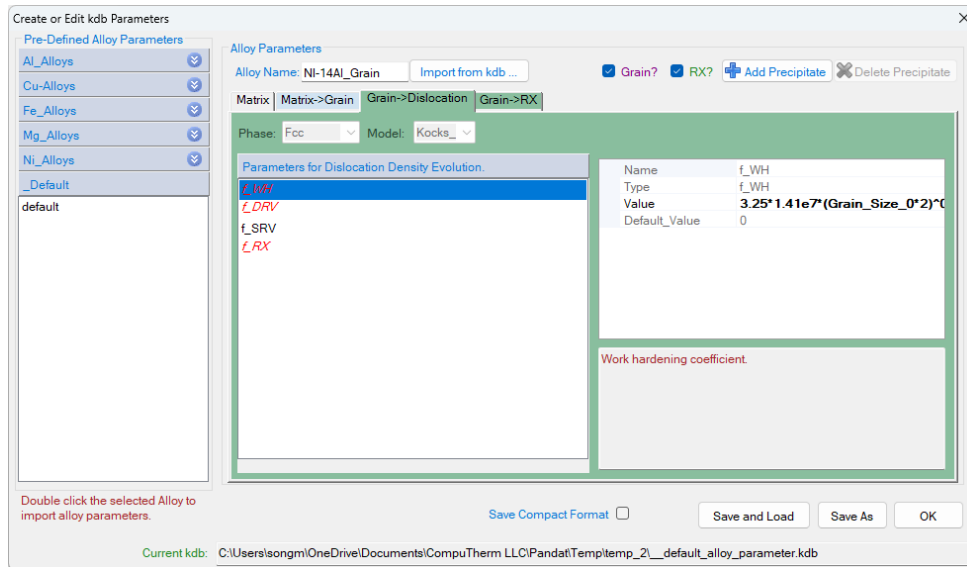
Figure 5.18: Parameters for the dislocation density and recrystallization grain(RX) in the Create or Edit kdb Parameter dialog

In the pre-defined alloy parameters, the matrix and the precipitate phase names are based on the demo databases used in "Pandat 2024 examples" folder and "Pandat 2024 example book" folder. One needs to make sure the phases in the pre-defined alloys are included and have the same phase name in the loaded thermodynamic and mobility database. Otherwise, users need to add the matrix and the precipitate phases parameters from the default setting. The kdb files for these pre-defined alloys are included in the folder :

"..\Program Files\CompuTherm LLC\Pandat 2024\bin\Database\kdb"

Users can check the phase names and parameters directly from these kdb files. User can also add their own kdb files in this folder, the added alloys will be automatically shown on this "**Create or Edit kdb Parameter dialog**".

5.4 PanEvolution Functions

The PanEvolution/PanPrecipitation menu includes three groups of major functions: The first group are precipitation kinetic database related functions: **Create or Edit KDB**; **Load KDB or EKDB** and **Select Alloy Parameter**. The second group are functions related to the traditional precipitation simulation : **Interfacial Energy (IFE) Calculation**; **Precipitation Simulation and TTT simulation**. The third group are the **Grain growth** Grain **Growth/Recrystallization** and**Concurrent Grain Growth and Precipitation**.
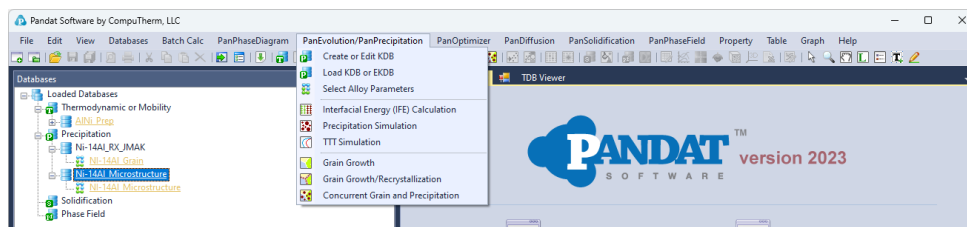
Figure 5.19: Menu functions of PanEvolution

The shortcuts for those functions are also displayed in the Toolbar as shown in Figure 5.20. When a thermodynamic together with mobility database (TDB or PDB file) and the kinetic parameter database (KDB) file are properly loaded. These functions will be activated and these icons will be highlighted.
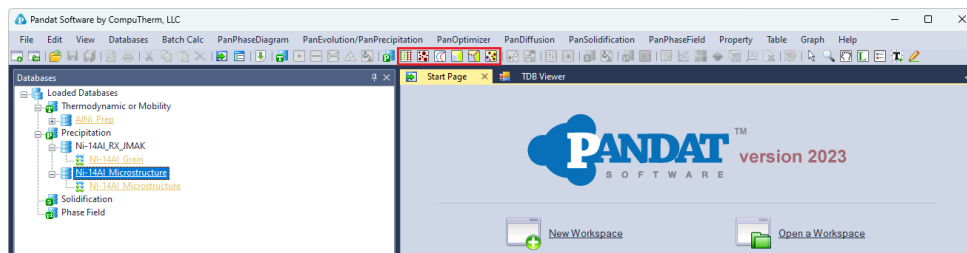
Figure 5.20: Toolbar icons of PanEvolution

5.5 Tutorial

In general, user should follow five steps to carry out a precipitation or grain growth simulation:

- **Create workspace and project:** Create a PanEvolution module project in an existing or a new workspace.

- **Load thermodynamic and mobility database:** Load thermodynamic and mobility database (.tdb or .pdb file). The database must contain both thermodynamic and mobility data.

- **Create or Load kinetic parameter database (.kdb file):** Create a kdb file from the template and set proper parameters for the kdb file (New function in Pandat 2022), or load a predefined kdb file with proper parameters. Then select the matrix phase and precipitate phases available in precipitation database for simulation

- **Simulation conditions setting:** Set the conditions for Precipitation or/and Grain growth Simulation, including alloy compositions, initial conditions and thermal history, then perform the simulation.

- **Post Calculation Operation**:Edit tables and graphs from the calculated results.

5.5.1 Ni-14at.%Al Precipitation Simulation

In this tutorial, the Ni‑14 at% Al alloy is taken as an example to demonstrate the functionalities provided by PanEvolution. The four files mentioned in this section (“`AlNi_Prep.tdb`”, “`Ni-14Al_Precipitation.kdb`”, “`Ni-14Al_Exp.dat`” and “`Ni-14Al_Precipitation.pbfx`”) can be found in the installation folder of Pandat™.

5.5.1.1 Step 1: Create a Workspace

Let's start this tutorial by "`creating a workspace`". By clicking the button on the toolbar, the "`Create a New Workspace`" popup window will open. User can choose the "PanEvolution" icon and define the location (project name, directory and/or workspace name), then click the "Create" button to create a precipitation workspace. User can also create a default precipitation workspace by double-click the "PanEvolution " icon as shown in Figure 5.21.
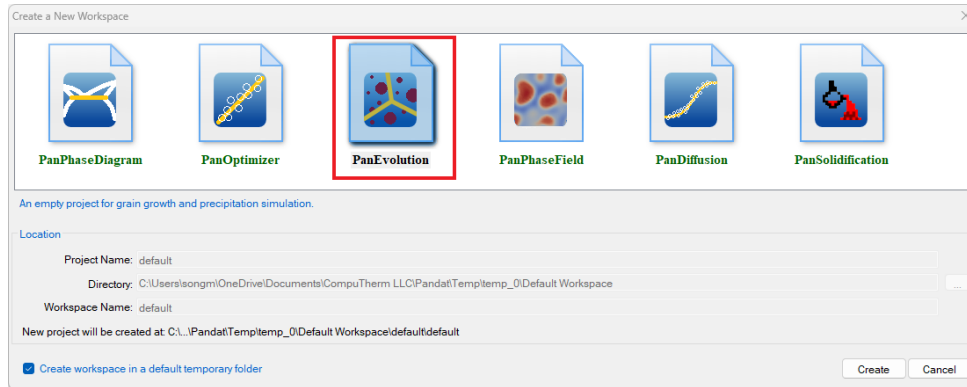
Figure 5.21: Dialog window for creating a workspace

5.5.1.2 Step 2: Load Thermodynamic and Mobility Database

The next step is to load the database, which is `AlNi_Prep.tdb` in this example. Different from the normal thermodynamic database, this database also contains mobility data for the matrix phase (Fcc) in addition to the thermodynamic model parameters. Both are needed for carrying out precipitation simulation. By clicking the icon on the toolbar, a popup window will open, allowing user to select the database file. User may select the TDB file and click on `Open` button or just double click on the TDB file to load it into the PanEvolution module as shown in Figure 5.22.
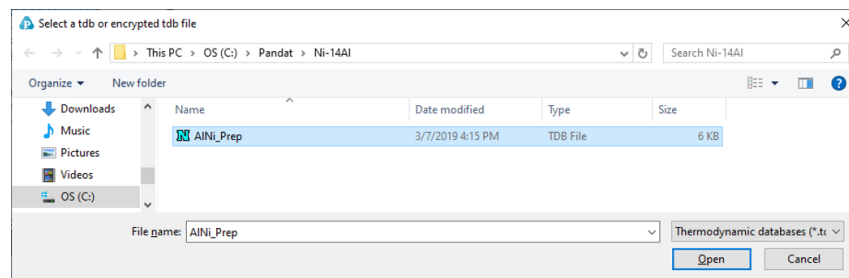
Figure 5.22: Dialog window for loading thermodynamic and mobility database

5.5.1.3 Step 3: Load Kinetic Parameters Database (KDB file)

A Kinetic Parameters Database (KDB file) is required for precipitation simulation. This file contains kinetic parameters which are alloy dependent. Each alloy is composed of one matrix phase and a number of precipitate phases. To organize these parameters in a more intuitive way, the standard XML format is adopted and a set of well-formed tags are deliberately designed to define the kinetic model (KWN or Fast_Acting) for each precipitate phase and its corresponding model parameters such as interfacial energy, molar volume, nucleation type, and morphology type.

In this example, the `Ni-14Al_Precipitation.kdb` is prepared. This file can be opened by click 📂 icon in the toolbar and be viewed in the Pandat™ workspace or through third-party external editors, such as NotePad, WordPad etc. The advantage to open this file in Pandat™ workspace is that all the key words will be highlighted which makes it easy to read. In this example, the matrix phase for this alloy is "`Fcc`", which has one precipitate phase "`L12_FCC`". The "`KWN`" model with "`Sphere`" morphology and "`Modified Homogeneous`" nucleation type are selected for precipitation simulation. There are five model parameters: `Molar_Volume`, `Interfacial_Energy`, `Atomc_Spacing`, `Nucleation_Site_Parameter`, and `Driving_Force_Factor`. These parameters have been developed for the Ni-14Al alloy in this particular example; these parameters need to be optimized for different alloys in terms of experimental data.

To load a KDB file, user should navigate the command through menu "`PanEvolution/PanPrecipitation → Load KDB or EKDB`", or click 📘 icon from the toolbar. After `Ni-14Al_Precipitation.kdb` is chosen, a dialog box pops out automatically for user to select the model, as shown in Figure 5.23, the "`Ni-14Al_KWN`" is chosen in this example. The Matrix Phase window allows user to select the matrix phase (Fcc in this case) and the Precipitates window allows user to choose one or more precipitate phases of interest for simulation (L12_Fcc in this case). To select several phases at one time, press and hold the "`Ctrl`" key.

Click "`View Parameters`", The "`Create or Edit kdb Parameters`" dialog as shown in Figure 5.14 will pop out for user to view and edit KDB parameters.
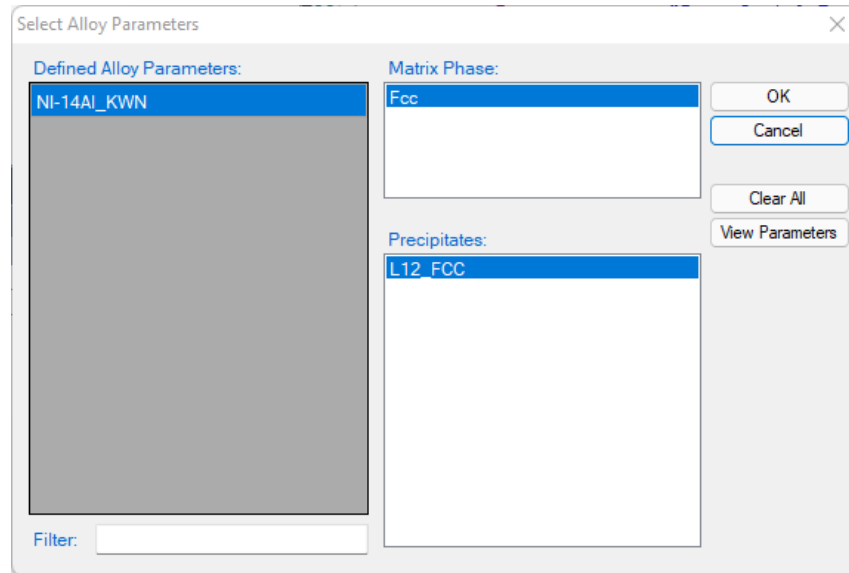
Figure 5.23: Dialog box for selecting alloy parameter

5.5.1.4 Step 4: Precipitation Simulation

After successfully loading the thermodynamic and mobility database (TDB or PDB file) and Kinetic Parameters Database (KDB file), the function for precipitation simulation is then being activated. To perform a precipitation simulation, choose from Pandat™ GUI menu "`PanEvolution/PanPrecipitation` → `Precipitation Simulation`", or click 🎲 icon from the toolbar. A dialog box entitled "`Precipitation Simulation`", as shown in Figure 5.24, pops out for user's inputs to set up the simulation conditions: alloy composition and thermal history.
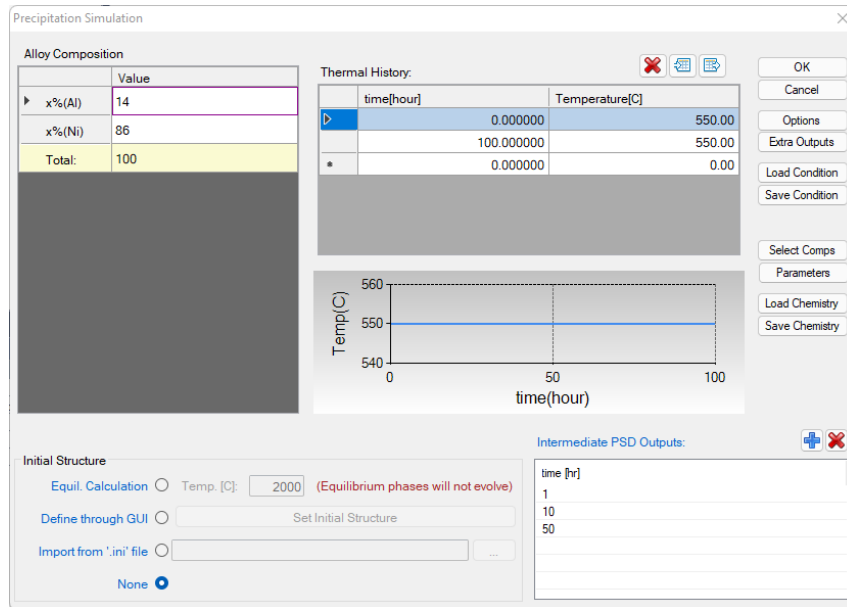
Figure 5.24: Dialog box for setting precipitation simulation conditions

- **Alloy Composition:** User can set alloy composition by typing in or use the `Load Chemistry` function. User can also save the alloy composition through `Save Chemistry`. This is especially useful when working on a multi-component system, so that user does not need to type in the chemistry every time.

- **Thermal History:** Arbitrary heat treatment schedule can be inputted, with a linear Time-Temperature relationship (i.e., constant cooling or heating rate) at each two consecutive rows. Time at the first row must be zero representing the initial time. The thermal history set up in the Figure 5.24 represents an isothermal heat treatment for 100 hours at 550 °C. If the temperatures in the two rows are different, it represents constant cooling or heating. Multi-stages of heat treatment can be set by adding more rows in the Thermal History column.

- **Set intermediate PSD outputs:** Based on the specified intermediate PSD outputs, psd tables are automatically generated and the psd plot for the final time step will be created. As shown in Figure 5.24, the intermediate PSD outputs are set up at 1, 10, 50, and 100 hours, respectively.

User can access the Options window by clicking the **Options** button. The **Options** window allows user to change the units for pressure, temperature, time and composition required in the simulation (as shown in Figure 5.25). User can also define the output **Table** and **Graph** within the **Output Options** window (as shown in Figure 5.26).
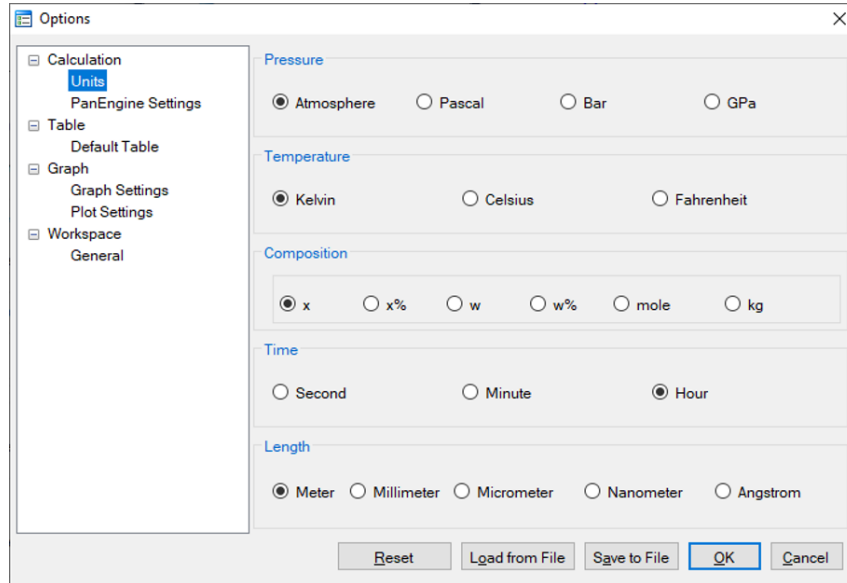
Figure 5.25: Dialog box for setting simulation units
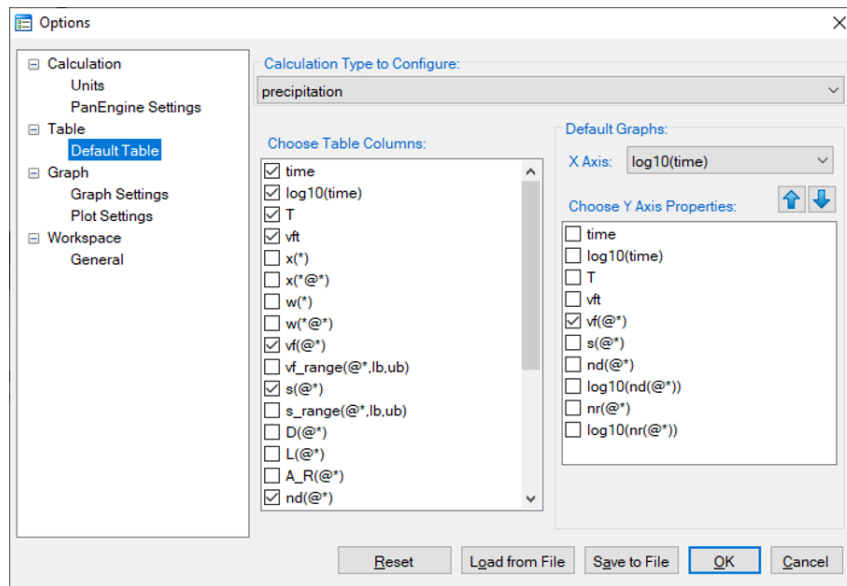
Figure 5.26: Dialog box for setting simulation outputs
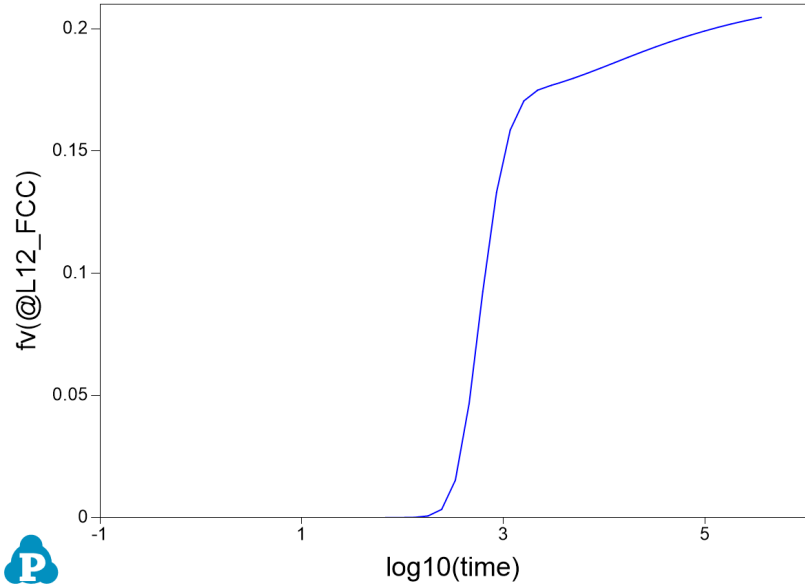
Figure 5.27: Default graph plotting volume fraction changes
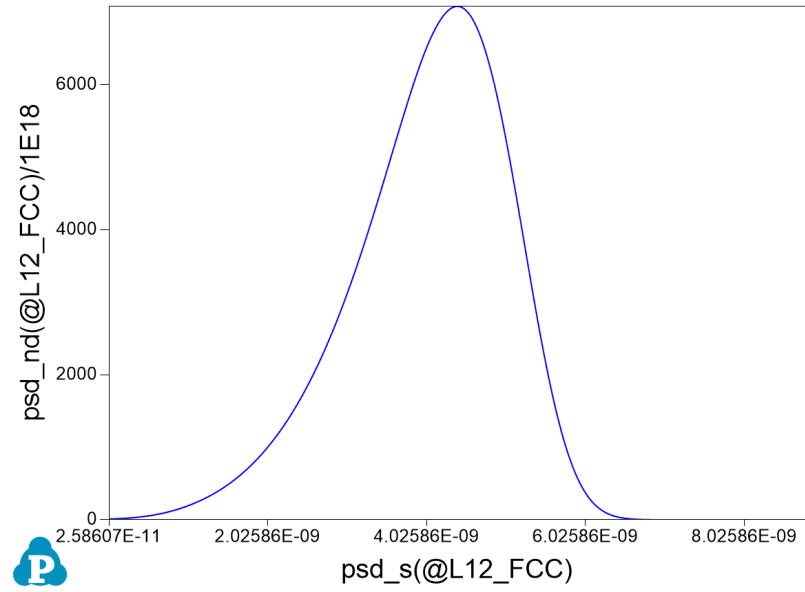
Figure 5.28: Calculated PSD at 100 hour
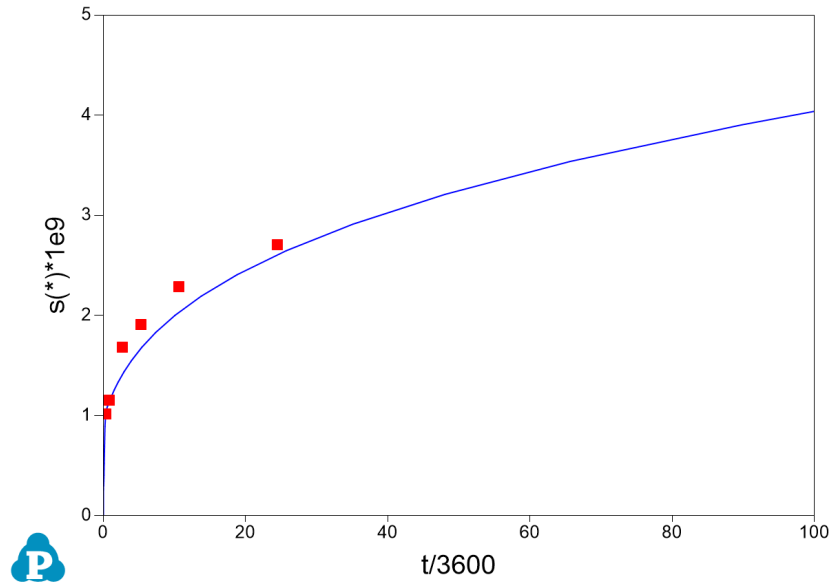
Figure 5.29: Compare the simulated particle size evolution with experimental data

In Precipitation Simulation window (as shown in Figure 5.24), click ΟΚ to run the simulation. After the simulation is completed, its results are displayed in two types of formats in the Pandat™ Explorer window: Graph and Table. By default, the graphs plotting volume fraction evolution and PSD at 100 hours are displayed in the Pandat™ main window as shown in Figure 5.27 and Figure 5.28, respectively. To view detailed simulation results, user can switch from Graph view to Table View by clicking tabs in the Pandat™ Explorer window.

User can customize the output results with Pandat™ batch file. Experimental data may be imported as a separate table using the syntax below:

```
<table source="Ni-14Al_Exp.dat" name="exp"/>
```

The experimental data file should be in the same folder where the batch file locates, otherwise a full path is needed for the experimental file as shown below:

```
<table source="D:\Pandat\Precipitation\Ni-14Al_Exp.dat" name="exp"/>
```

And the experimental data can be plotted together with the calculated results using the syntax below to obtain a diagram as shown in Figure 5.29.

```
<graph name="size">
    <plot table_name="default" xaxis="t/3600" yaxis="s(*)*1e9"/>
    <plot type="point" table_name="exp" xaxis="t(hr)" yaxis="radius
(nm)"/>
</graph>
```

5.5.1.5 Step 5: Customize Simulation Results

As all other calculations available in Pandat™, upon the completion of the precipitation simulation, a default table with related kinetic properties (such as time, total transformed volume fraction, average size of each precipitate phase, number density of each precipitate phase and nucleation rate of each precipitate phase) is automatically generated and a default graph for volume fraction changes is displayed Figure 5.27.

However, it should be emphasized that, in addition to the default tables, variety of properties, such as temperature, volume fraction of each individual precipitate phase, instant composition of matrix phase and also size distribution related information if the KWN model is used for simulation, can be retrieved through "`Add a new table`". Symbol and syntax for retrieving quantities of matrix phase, precipitate phases or grain are listed in Table 11.15.

Add a New Table

"`Add a new table`" function can be activated by the following ways: (1) selecting the "`Table`" node in the Explorer window, then right click the mouse and select "`Add a New Table`" option (as shown in Figure 5.30); or (2) choose "`Add or Edit a Table`" from the "`Table`" menu. This function allows user to create a new table at their own choices.

Figure 5.30: Dialog of add a new table

Figure 5.31: Dialog window of table editor

The basic layout of the window of "`Add a new table`" is shown in Figure 5.31. There are two major parts in this window. The left part is the available variables and contents. User first chooses the type of the tables from the "`Table Type`" drop-down list as highlighted in Figure 5.31, to show the list of available variables. The right part is the generated table field. User can drag the available variables from the left column to the right side by clicking and holding the left mouse button.

Table Format Syntax

Moreover, mathematical calculation over these properties is allowed in PanEvolution. Accordingly, a variety of property diagrams can be generated based on the customized tables, which offers users an excellent flexibility for different applications. A detailed description of table format is given below. Figure 5.32 shows the dialog window of table

editor for customized new table. The symbol "*" can be used to get quantities of all the phases or all components. As an example, s(*) retrieves the average size of phase L12_ FCC, which is equivalent to s(L12_FCC).

Figure 5.33 shows the dialog window of table editor for user to get a psd table by specifying time. For example, the psd table at 100 hours can be obtained by setting "`time = 360000`". Note that the default unit for time is second.

Figure 5.32: Dialog window of table editor for customized new table

Figure 5.33: Dialog to get a psd table at a specified time

5.5.2 Concurrent Grain Growth and Precipitation Simulation of Ni-14at.%Al Alloy

In this tutorial, the Ni‑14 at% Al alloy is also used for concurrent grain growth and precipitation simulation in PanEvolution. The related batch file (`Ni‑ 14Al_ Microstructure.pbfx`), thermodynamic and mobility database (`AlNi_Prep.tdb`) and kinetic parameters database (`Ni‑14Al_ Microstructure.kdb`) can be found in the Pandat™ installation folder "`\Pandat 2024 Examples\PanEvolution\Grain_ Precipitation\`":

5.5.2.1 Step 1: Create a Workspace

The same procedure as described in the previous tutorial Section 5.5.1.1

5.5.2.2 Step 2: Load Thermodynamic and Mobility Database

The same procedure as described in the previous tutorial Section 5.5.1.2

5.5.2.3 Step 3: Load Kinetic Parameters Database (kdb file)

The procedure is similar to the description in the previous tutorial Section 5.5.1.3 . The "Defined Alloy parameters" in this kdb file is "`Ni-14Al_Microstructure`", as shown in Figure 5.34. It was "`Ni-14Al_KWN`" in the previous tutorial as shown in Figure 5.23.

Figure 5.34: Dialog box for selecting alloy parameter

Click the "`View Parameters`", the "`Create or Edit kdb Parameters`" dialog pops out as shown in Figure 5.35. The parameters in this kdb file contains not only the "`matrix`" phase and the "`L12_Fcc`" precipitate phase, but also the "`Matrix:: Grain`" parameters. Those "`kinetic parameters`" and some "`User-defined Variables`" are required for grain growth simulation.

Figure 5.35: Parameters for matrix grain in the "Create or Edit kdb Parameter" dialog

5.5.2.4 Step 4: Concurrent Grain Growth and Precipitation simulation

After successfully loading the thermodynamic and mobility database (TDB and PDB file) and the kinetic parameters database (KDB file), which contains kinetic parameters for both the matrix phase and precipitate phases, the function for grain growth and precipitation simulation is then being activated. To perform a concurrent grain growth and precipitation simulation, choose from Pandat™ menu "`PanEvolution → Concurrent Grain Growth and Precipitation`", or click icon 🔳 from the toolbar. A dialog box entitled "`Grain Growth and Precipitation Simulation`", as shown in Figure 5.36, pops out for user's inputs to set up the simulation conditions: alloy composition, thermal history and initial structure.

Figure 5.36: Dialog box for setting precipitation simulation conditions

- **Alloy Composition:** User can set alloy composition by typing in or use the `Load Chemistry` function.

- **Thermal History:** Arbitrary heat treatment schedule can be inputted, with a linear Time-Temperature relationship (i.e., constant cooling or heating rate) at each two consecutive rows. Time at the first row must be zero representing the initial time. The

thermal history set up in the Figure 5.36 represents an isothermal heat treatment for 10000 s at 800 °C, then constantly cooling to 550 °C within from 10000s to 30000 s. Multi-stages of heat treatment can be set by adding more rows in the Thermal History column.

- **Set initial Structure:** Users can set initial structure through the function "`import from '.ini' file`" or "`Define through GUI`". Click "`Set Initial Structure`", the dialog as shown in Figure 5.37 pops out for users to set initial structure. From this dialog, users can set the initial structure for both the grain and precipitate phase. In this tutorial, the initial average grain size is set as 1e-6 m (1µm), with "log_normal" distribution. After setting the initial structure, click "`Export Ini. File`" to save the initial condition file.

Figure 5.37: Dialog box for setting initial structure

5.5.2.5 Step 5: Customize Simulation Results

The same as all other calculations available in Pandat™, upon the completion of the precipitation simulation, a default table with related kinetic properties (such as time, total

transformed volume fraction, average size of each precipitate phase, number density of each precipitate phase and nucleation rate of each precipitate phase) is automatically generated and a default graph for average size changes is displayed in Figure 5.38.

Figure 5.38: Default graph plotting average grain and precipitate particle size changes

5.5.3 Dynamic Recrystallization Simulation of Ni-14at.%Al Alloy

In this tutorial, the Ni -14 at% Al alloy is also used for dynamic recrystallization simulation in PanEvolution. The related batch file (`Ni-14Al_RX_JMAK.pbfx`), thermodynamic and mobility database (`AlNi_Prep.tdb`), kinetic parameters database (`Ni-14Al_RX_JMAK.kdb`) and the initial condition file (`Al-Ni.ini`) can be found in the Pandat™ installation folder "`\Pandat 2024 Examples\PanEvolution\Grain_ Recrystallization\JMAK`":

5.5.3.1 Step 1: Create a Workspace

The same procedure as described in the previous tutorial Section 5.5.1.1

5.5.3.2 Step 2: Load Thermodynamic and Mobility Database

The same procedure as described in the previous tutorial Section 5.5.1.2

5.5.3.3 Step 3: Load Kinetic Parameters Database (kdb file)

The procedure is similar to the description in the previous tutorial Section 5.5.1.3 . The "Defined Alloy parameters" in this kdb file is "`Ni-14Al_Grain`", as shown in Figure 5.39.

Figure 5.39: Dialog box for selecting alloy parameter

Click the "`View Parameters`", the "`Create or Edit kdb Parameters`" dialog pops out as shown in Figure 5.40. The parameters in this kdb file contains not only the "`matrix`", but also some other items "`Matrix->Grain`" to define the properties of the original grains, "`Grain->Dislocation`" to define the dislocation density related properties, "`Grain->RX`" to define the parameters related to the recrystallization grains. The meaning of the parameters are detailed shown in Section 11.3 .

Figure 5.40: Parameters for matrix grain in the "Create or Edit kdb Parameter" dialog

5.5.3.4 Step 4: Grain Growth and Recrystallization simulation

After successfully loading the thermodynamic and mobility database (TDB and PDB file) and the kinetic parameters database (KDB file), which contains kinetic parameters for the matrix grains, dislocation density and recrystallization grains, the function for grain growth and Recrystallization simulation is then being activated.

To perform a grain growth and recrystallization simulation, choose from Pandat™ menu "`PanEvolution/PanPrecipitation → Grain Growth/Recrystallization`", or click icon ![icon] from the toolbar. A dialog box entitled "`Grain Growth and Recrystallization Simulation`", as shown in 5.5.3, pops out for user's inputs to set up the simulation conditions: alloy composition, processing history and initial structure.

Figure 5.41: Dialog box for setting grain growth and recrystallization simulation conditions

- **Alloy Composition:** User can set alloy composition by typing in or use the `Load Chemistry` function.

- **Processing History:** Select the "Use Rate?" to define the conditions as Rate. This condition shown in Figure 5.41 is at 1193 K with strain rate of 0.01 s^{-1} for 260 seconds.

- **Set initial Structure:** Users can set initial structure through the function "`import from '.ini' file`" or "`Define through GUI`". In this example, the "`Al-Ni.ini`" file is imported as the initial structure.

5.5.3.5 Step 5: Customize Simulation Results

The same as all other calculations available in Pandat™, upon the completion of the recrystallization simulation, a default table with related kinetic properties, such as time, strain and true strain, volume fraction grain and recrystallization grains, grain size, dislocation density and yield strength (sigma_y), is automatically generated. Two default graphs for the volume fraction of the grain and recrystallization grains, and the yield strength vs true strain are plotted as show in Figure 5.42.

(a)   (b)

Figure 5.42: Default plots in the recrystallization simulation

Based on the data in the default table, users can also plot other properties, for example, the recrystallizated grain size and the average grain size as shown in Figure 5.43.

Figure 5.43: The simulated average grain size during the dynamic recrystallization process

6 PanDiffusion

PanDiffusion is a module of Pandat™ software designed to simulate kinetic processes dominated by elemental diffusion. PanDiffusion provides a rich variety of applications including particle dissolution, carburization, decarburization, homogenization, phase transformation, diffusion couple, and so on.

PanDiffusion is seamlessly integrated with the user-friendly Pandat™ Graphical User Interface (PanGUI) as well as thermodynamic calculation engine, PanEngine. The interface between PanDiffusion and PanEngine is managed through PanDataNet. The implementation of PanEngine guarantees reliable input data, such as chemical potential, phase equilibrium and mobility. Figure 6.1 shows an overall architecture of the PanDiffusion module.

Figure 6.1: An overall architecture of the PanDiffusion module

6.1 Features of PanDiffusion

6.1.1 Overall Design

- Time evolution of composition profile, phase volume fraction, and phase composition.

- Multiple selections of thermal history, boundary condition and geometry.

- Applications including particle dissolution, carburization, decarburization, homogenization, phase transformation, and diffusion couple.

6.1.2 Kinetic Model

Figure 6.2 shows a schematic plot of a composition profile in a diffusion couple. In the simulation, the sample is divided by grids with equal width. The numbers, 1, 2, 3, …, n, indicate grid id. The solid black and dashed red lines means inter-grid interface. What is more, the red dashed line between (i-1)-th and i-th grids indicates the position of a sharp interface of this diffusion couple. Inter-grid flux is calculated following Fick's first law. The interface between different phases have different special treatments such as "moving-boundary" model and " plain" model. Moving-boundary model is especially required for simulating Solid-Liquid interfaces. An example for simulating the diffusion between solid Al and liquid Zn is provide in the example folder (See `AlZn_Solid_Liquid_diffusion_couple.pbfx`). Composition of each grid, which is indicated by solid blue line in Figure 6.2, is calculated following Fick's second law. The discrete composition profile represents a continuous composition profile of gray dashed line. Chemical potential, mobility, phase equilibrium and related properties are updated for each grid after calculating the composition.

Figure 6.2: A schematic plot of a composition profile in a diffusion couple

6.1.2.1 Flux Model

The evolution of each grid's composition is controlled by the inter-grid flux which is calculated based on absolute reaction rate theory [1941Gla]. At the lattice-fixed frame of reference, the flux is:

$$J_k = -\frac{M_k RT}{V_m \Delta z}\sqrt{x_k^i x_k^{i+1}} \times 2\sinh\left(\frac{\Delta\mu_k}{2RT}\right)$$

(6.1)

Where J_k is the flux of k-th element between the grid i and the grid i+1. M_k is effective mobility, R is the gas constant, T is temperature in Kelvin, V_m is molar volume, Δz is the thickness of grid interface and in most cases calculated as average grid size, x_k^i and x_k^{i+1} are molar fraction of k-th element at the grid i and the grid i+1, $\Delta\mu_k$ is the chemical potential difference of k-th element between the grid i and the grid i+1. The $\texttt{sinh}$ is the hyperbolic sine function. For convenience of calculation, a composition profile is observed at a volume-fixed frame of reference, and the flux of substitute element is transformed to $J_k^V = J_k - x_k\sum J_k$.

Then the amount of k-th element (n_k) of each grid can be updated according Fick's 2nd law:

$$\frac{dn_k}{dt} = -\iint_S \vec{J}_k^V \cdot d\vec{S}$$

(6.2)

Where unit normal vectors point outward from the grid.

6.2 Get Started

Diffusion simulation usually follows the following procedures.

6.2.1 Create a PanDiffusion Project

User can create a PanDiffusion project through menu "`File → Create a New Workspace`" or "`File → Add a New Project`" in an existing workspace. The "`Module Window`" pops out for user to choose a module for the new project as shown in Figure 6.3. Choose "PanDiffusion" module for diffusion simulation, and the PanDiffusion project will be created after user click on `Create` button or double click on the `PanDiffusion` icon.

Figure 6.3: Creating a PanDiffusion workspace

6.2.2 Load Thermodynamic and Mobility Database

The next step is to load the database, which is `FeCrNi.tdb` in this example. Different from the normal thermodynamic database, this database also contains mobility data for the phases of interest in addition to the thermodynamic model parameters. Both are needed for carrying out diffusion simulation. By clicking the 🔲 icon on the toolbar, a popup window will open, allowing user to select the database file.

6.2.3 Start PanDiffusion Module

Perform diffusion simulation through the menu bar (Figure 6.4) or the tool bar icons (Figure 6.5). PanDiffusion deals with diffusion process in two strategies, Settings in General Diffusion Simulation and Settings in Particle Dissolution Simulation.

Figure 6.4: Menu functions of PanDiffusion

Figure 6.5: Toolbar icons of PanDiffusion

6.3 Settings in General Diffusion Simulation

User can perform variety of diffusion simulations by PanDiffusion module. All types of simulations, except for particle dissolution, share the same general graphic user interface (GUI) to set up simulation conditions. This GUI can be accessed through the menu bar "`PanDiffusion→ Diffusion Simulation`", or by clicking 🔲 on the Toolbar. Figure 6.6 shows the GUI interface of general diffusion simulation.

Figure 6.6: The general diffusion simulation condition setting GUI

6.3.1 Set Units

In the GUI of PanDiffusion, unit settings can be accessed through "`Options → Calculation → Units`". Please refer to Section 3.2.3 for details.

6.3.2 Select Phases

By clicking "`Select Phases`" as highlighted by the **red box** in Figure 6.6, user can select phases involved in the diffusion simulation on the system level. By default, all the phases in the alloy system are selected, while user can deselect some of them as wishes. In addition,

user can also select phases for each region by clicking "`Select Phases`" as highlighted by the `green box` in Figure 6.6. By default, phase selection for each region follows the global setting in the system level.

6.3.3 Set Initial Composition Profile

In the GUI of PanDiffusion, initial composition profile is assigned region-by-region. In each region, composition profile can be:

- **Uniform:** the composition of each element in the selected Region is a constant. When "`uniform`" is selected, a homogenous "`Region Composition`" can be set.

Figure 6.7: "Uniform" composition of a region

- **Linear**: a linear composition interpolation is set from the left edge to the right edge of the selected Region. When "`linear`" is selected, both "`Left End`" and "`Right End`" compositions need to be set.

Figure 6.8: "Linear" composition of a region

- **Input_file**: Load composition profile from a tab-delimited `.dat` (or `.txt`) file with the following two kinds of formats: Format (I): the distance is specified for each element, which is usually the case when using experimental data. The 1st row contains the name of each column, the 2nd row contains the unit definition, and value starts from the 3rd row.

distance(Cr)	x(Cr)	distance(Ni)	x(Ni)
meter	x	meter	x
3.92E-07	0.307	3.90E-07	0.693

Format (II): the distance is specified in the 1st column, the composition of every element correspond to the same distance. This format is usually obtained by digitizing data from literature. The 1st row contains the name of each column, the 2nd row contains the unit definition, and value starts from the 3rd row.

distance(*)	x(Cr)	x(Ni)
meter	x	x
3.92E-07	0.307	0.693

The unit of distance can be: meter (or *m*), millimeter (or *mm*), micrometer (or *um*), nanometer (or *nm*) and angstrom (Å). The unit of composition can be: `x` for mole fraction, `x%` for mole percentage, `w` for weight fraction, and `w%` for weight percentage. Note that, the units in the input file will override setting in "`Options → Calculation → Units`" and the input file is case insensitive.

6.3.4 Delete or Add a Region

In the GUI of PanDiffusion, there are two Regions by default. In the following cases, single region is recommended:

- Only one input composition profile

- Carburization, decarburization, or other surface flux process, with a homogeneous

initial composition profile

- Homogenization of a linear composition profile

The redundant region can be removed by first selecting the Region, then clicking ✖ button as shown in 6.3. If diffusion is among three or more Regions, Regions can also be added one by one by clicking ➕.

Figure 6.9: Add or delete a region calculation

6.3.5 Set Regional Length

In the GUI of PanDiffusion, "`Diff. Length`" is the length of a region.

Figure 6.10: Length in a region

6.3.6 Set # of Grids

In the GUI of PanDiffusion, number of grids is set for the entire simulation box and is distributed automatically to each region.

Figure 6.11: Number of grids

6.3.7 Set Thermal History

In the GUI of PanDiffusion, "`Thermal History`" controls temperature and the duration at each temperature. The setting in Figure 6.12 indicates that the temperature increases linearly from 500°C to 1000°C by 100 hours (5°C/hour).

Figure 6.12: A linear increment of temperature

The setting shown in Figure 6.13 means that the sample is held at 500°C for 100 hours, and then the furnace temperature is suddenly increased to 1000°C and held there for another 100 hours.

Figure 6.13: A step-like thermal history

6.3.8 Add Moments for Profile Outputs

For a calculation set up through GUI of PanDiffusion, two moments of profiles, initial and final, will be provided by default. To add more intermediate moments of profiles, click ✚ besides the "`Moments for Profile Outputs`", and edit time of the expected moment.

Figure 6.14: Add a moment for profile output at 1.3 hour

6.3.9 Set Geometry

In the GUI of PanDiffusion, "`Geometry`" under "`Simulation Conditions`" decides the shape of inter-grid interface. By default, "`planar`" is used for most diffusion couple simulations. When particle homogenization or transformation is performed, "`spherical`" or "`cylindrical`" could be selected depending on the problem of interest.

Figure 6.15: Select the geometry

6.3.10 Set Inner Radius

When geometry other than "`planar`" is selected, user can set inner radius of cell, to simulate for a tube or shell geometry. When "`cylindrical`" geometry is selected, and the inner radius is non-zero, tube geometry is set up. When "`spherical`" geometry is selected, and the inner radius is non-zero, shell geometry is set up.

Figure 6.16: Set Inner Radius

6.3.11 Set Interface Flux Model

In the GUI of PanDiffusion, "`Interface Flux Model`" under "`Simulation Conditions`" decides how the grid composition evolves at the interface between different phases. The option "`automatic`" is selected by default, which chooses the interface flux model automatically based on the number of phases selected in each region. When "`plain`" is selected, the mobility of the solute elements is determined by the weighted average of the mobilities of the individual phases occurring in a grid point (See the Section 6.3.13). When "`moving_boundary`" is selected, a sharp interface model is used to calculated the flux across the phase boundary which does not require the effective mobility.

Figure 6.17: Select interface flux model

6.3.12 Set Boundary Conditions

There are "`Upper Boundary Condition`" and "`Lower Boundary Condition`" in PanDiffusion. Their definitions are illustrated in the following figure:

Figure 6.18: Definition of boundary conditions

- **Fixed activity** at boundaries can be set following the format:

Figure 6.19: Set fixed activity as a boundary condition

- **Mass flux expression** at boundaries can be set following the format:

Figure 6.20: Set mass flux expression as a boundary condition

- **Fixed composition** of boundaries can be set following the format:

Figure 6.21: Set fixed composition as a boundary condition

6.3.13 Set Effective Mobility (Advanced option)

If the " `Interface Flux Model` " chosen is "`plain`", then the choice of the effective mobility determines the flux across the interface. More details about the various effective mobilities definition can be found in [2009Lar] The default choice for the effective mobility is "`HS_lower`". It would be mentioned in the input batch file as shown below.

```
<diffusion>
  <integral_settings value="automatic" />
  <Runge_Kutta_order value="1" />
  <relative_tolerance value="1e-05" />
  <profiles>
    <profile time="0" />
    <profile time="1" />
    <profile time="10" />
  </profiles>
  <interface_flux_model value="plain" />
  <effective_mobility_model value="HS_lower" />
</diffusion>
```

Figure 6.22: Define effective mobility in the batch file

Based on the expected distribution of phases within the grid point, the effective mobility should be chosen as shown in the table below. Sometimes, the user may not know the choice between "`HS_upper`" and "`HS_lower`" as they don't know the mobility of individual phases. In such cases, they can use the option "`HS_matrix:YYY`" where YYY is the expected matrix phase (i.e. the connected phase) in the microstructure. Additionally, "`H_major`" option is also available where the majority phases will be considered as the connected phase.

Table 6.1: Options for effective mobility in PanDiffusion

Illustration of Phase Distribution	Effective Mobility
	HS_ upper (Hashin- Shtrikman Upper Bound)
	HS_ lower (Hashin- Shtrikman Lower Bound)
	WA_upper (Weiner Upper Bound)
	WA_lower (Weiner Lower Bound)

α , β : Phase distribution α and β within the grid point,
Individual phase mobilities $M^\alpha > M^\beta$

6.4 Settings in Particle Dissolution Simulation

The features of general GUI for diffusion simulation have been demonstrated in Section 6.3 In this section, we will present the features of the special GUI for dissolution simulation, which can be accessed through "`PanDiffusion → Dissolution Simulation`", or by clicking ▣ on the Toolbar.

Settings for units, number of grids, geometry, interface flux model, thermal history and output profiles follow the same way as those in general GUI. The phases are selected globally in dissolution GUI. There is no regional setting in dissolution simulation. The global phase setting follows the same way as the general GUI.

6.4.1 Set Alloy Composition

In the dissolution GUI, overall composition of an alloy is set in the following section:

Alloy Composition	Value
▶ x%(Al)	98.45
x%(Cu)	1.55
Total:	100

Figure 6.23: Alloy composition in dissolution simulation GUI

6.4.2 Set Matrix and Particle Information

Matrix phase and particle phases are explicitly selected in the "`Phase Information`" section. There can be more than one particle phases. For each particle phase, particle radius and volume fraction need to be set up.

Phase Information

Matrix Phase: Fcc

Particles: ➕ ✖

Phase Name		Particle Size[um]	Vol. Fraction
AlCu_Theta	∨	3	0.008
AlCu_Theta	∨	1	0.001

Figure 6.24: Matrix and particle information in dissolution simulation

6.5 Tutorial

Several examples are given below to demonstrate how to use PanDiffusion module.

6.5.1 Diffusion Couple with Uniform Composition Input

In this tutorial, the diffusion simulation on a diffusion couple with uniform composition at each side (as shown in Figure 6.25) is carried out. The annealing temperature is 1000 °C and the duration is 2 hours. "`Region_1`" represents the left-hand side of this diffusion couple, and "`Region_2`" represents the right-hand side of this diffusion couple. Each region is 100 μm and the simulation box is discretized to 100 grids.

Figure 6.25: Initial conditions of the diffusion couple simulation

6.5.1.1 Set up Initial Condition for Diffusion Simulation

In order to set the above conditions in PanDiffusion, click through the menu bar "`PanDiffusion→ Diffusion Simulation`", or by clicking ▦ on the Toolbar., a popup window will show and allow user to set up calculation conditions (as shown in Figure 6.26 and Figure 6.27).

Figure 6.26: Dialog window for user to set the composition of Region_1

Figure 6.27: Dialog window for user to set the composition of Region_2 and diffusion simulation conditions

Click **Region_1** to set up the composition of the left-hand side as shown in Figure 6.26, and click **Region_2** to set up composition of the right-hand side as shown in Figure 6.27. Details regarding the interface design will be described in Section 6.3 . Constant annealing temperature is set as 1000 °C, and annealing time is 2 hours.

As is highlighted by the green box in Figure 6.27, the simulation condition can be saved to a ".pbfx" file by selecting "**Save Condition**". The saved condition can be loaded by clicking "**Load Condition**" for the future usage. When a condition is loaded from ".pbfx" file, the settings in the GUI are updated accordingly.

6.5.1.2 Simulation Results

Click OK button in Figure 6.27 to start the simulation after the initial condition is set properly. The simulation result is displayed in Figure 6.28.

In this case, only the initial and final composition profiles are presented: the dotted lines with sharp interface are for the initial stage and the solid lines with smooth interface are for the final stage.

Figure 6.28: Output graph showing the initial and final composition profiles

6.5.1.3 Customize Simulation Results

As all other calculations available in Pandat™, upon the completion of the diffusion simulation, a default table of properties (such as time, temperature, grid size, distance,

composition distribution, volume fraction and chemical potential) is automatically generated and a default graph of composition profiles is displayed. User can refer to Section 2.4 and Section 2.5 to learn how to customize simulated graph and table.

6.5.2 Diffusion Simulation with Composition Profile Input

In this tutorial, the `Fe-Ni-Cr` alloy is taken as an example to demonstrate a diffusion simulation with a composition profile input. The format of input file is discussed in Section 6.3.3 . The database file mentioned in this section, "`FeCrNi.tdb`", can be found in the installation folder of Pandat™. In general, user should follow the following steps to carry out a diffusion simulation:

6.5.2.1 Set up Initial Condition for Diffusion Simulation

Figure 6.29 demonstrates the conditions of this simulation. The initial composition profile from the input file is displayed in this figure. The length of the profile is 200 μm and is discretized to 100 grids.

To set the above conditions in PanDiffusion, click through on the tool bar, or from the menu "**PanDiffusion→ Diffusion Simulation**" a popup window as shown in Figure 6.26 will allow user to set up calculation conditions. In Figure 6.30, "**Region_2**" is deleted firstly. Please refer to Section 6.3.4 on how to delete a region. In the remaining "**Region_ 1**", select "**input_file**" from "**Region Composition Distribution**", and click "**Browse**" to load a `.dat` (or `.txt`) file which contains composition profile. Annealing time (100 hours in total) and annealing temperature (1200 °C) are set. Add an intermediate output at 10 h in the "**Moments of Profile Outputs**" to observe simulation process. Please refer to Section 6.3.8 on how to add a moment of profile output.

Figure 6.29: Initial conditions of the diffusion simulation with a composition profile input

Figure 6.30: Dialog window for user to input diffusion simulation conditions and composition of Region_1

6.5.2.2 Simulation Results

Click OK button in Figure 6.30 to start the simulation after the initial condition is set properly. The simulation result is displayed in Figure 6.31. Please refer to Section 2.4   of this user manual to add text, legend, and change the appearance of the figure.

As is seen in this figure, the initial serrated composition profiles were smoothed out after 10 hours of annealing at 1200 °C, and the material is homogenized after 100 hours of annealing.

Figure 6.31: Output graph showing the initial and final composition profiles

6.5.3 Dissolution Simulation

In this tutorial, the `Al-Cu` alloy is taken as an example to demonstrate a dissolution simulation. The GUI of dissolution is discussed in Section 6.4 . In general, user should follow the following steps to carry out a dissolution simulation:

6.5.3.1 Set up Initial Condition for Dissolution Simulation

The particle is assumed to be spherical and have a radius of 3 μm, and its volume fraction is 0.008. The overall composition of the system is `98.45Al-1.55Cu (at%)`. The simulation box is digitized to 100 grids.

To set the above conditions in PanDiffusion, click 🔴 on the Toolbar or from the menu "`PanDiffusion → Dissolution Simulation`", a popup window as shown in Figure 6.32 will allow user to set up calculation conditions. In "`Alloy Composition`" section, the overall composition of `Al-Cu` alloy is set to `98.45Al-1.55Cu (at%)`. In "`Phase Information`" section, set "`Matrix Phase`" as `Fcc`. In "`Particles`" subsection, click ➕ bottom to add a particle, set "`Phase Name`" as `AlCu_Theta`, set "`Particle Size`" as 3 μm, and set "`Vol. Fraction`" as `0.008`. In "`Simulation Conditions`" Section, set

"**Geometry**" as `spherical`, and set "**# of Grids**" as `100`. In "**Thermal History**" section, **temperature** is set as `550°C`, and **heat treatment duration** is set as `2000 seconds`.

Add additional intermediate outputs at 100s, 500s and 1000s in the section "**Moments for Profile Outputs**", then click "**OK**" to start the calculation.

Figure 6.32: Dialog window for user to input dissolution simulation conditions

6.5.3.2 Simulation Results

Click OK button in Figure 6.32 to start the simulation after the initial condition is set properly. The simulation results are displayed in Figure 6.33 and Figure 6.34. Please refer to Section 2.4 of this user manual to add text, legend, and change the appearance of the figure. Figure 6.33 shows the particle size change with time, from 3 *μm* (radius) at beginning to zero after annealing at 550 °C for ~1000 seconds. Figure 6.34 shows diffusion between particle and matrix and the composition profiles at 0s, 100s, 500s, 1000s, and 2000s. It is seen from Figure 6.15 that the particles are completely dissolved after annealing for 2000s.

Figure 6.33: Output graph showing the time evolution of particle radius

Figure 6.34: Output graph showing the time evolution of composition profile

7   PanSolidification

PanSolidification is a module of Pandat™ software designed to simulate solidification behavior under a variety of conditions with different cooling rates. It is an extension of the Scheil model taking into consideration of back diffusion in the solid, secondary dendrite arm coarsening, and the formation of eutectic structure.

Figure 7.1: An overall architecture of the PanSolidification module

It is seamlessly integrated with the user-friendly Pandat™ Graphical User Interface (PanGUI) as well as thermodynamic calculation engine, PanEngine. The implementation of PanEngine guarantees reliable input data, such as chemical potential, phase equilibrium and mobility. Figure 7.1 shows an overall architecture of the PanSolidification module.

7.1 Features of PanSolidification

7.1.1 Overall Design

- The system composition profile, phase fraction, and phase concentration evolution during solidification

- Secondary dendrite arm spacing (`SDAS`) evolution during solidification.

- Back diffusion in the primary matrix phase during the solidification process.

7.1.2 Data Structure

Thermodynamic and mobility parameters are stored in `TDB` file, and the kinetic parameters for undercooling and coarsening effects are stored in an `SDB` file in "`Extensible Markup Language`" (`XML`) format, which is a standard markup language and well-known for its extendability. In accordance with the `XML` syntax, a set of well-formed tags are specially designed to define the back diffusion model for the morphology of primary phase and its corresponding model parameters such as interfacial energy, latent heat, coarsening geometric factor, dendrite tip factor, solute trapping parameter, solid diffusivity factor and boundary layer factor.

7.2 Numerical Model

The PanSolidification module, which is developed by coupling a solidification micro-model with PanEngine, is basically a modified Scheil model incorporating back-diffusion, undercooling, and dendrite arm coarsening. Figure 7.2 shows a sketch of dendrite, with a big solid trunk as the primary dendrite arm and fine secondary dendrite arms symmetrically distributed at the sides; the `SDAS` is indicated as λ_2. A one-dimensional morphology within the interdendritic region of secondary arms is usually used to describe the solidification processing (as enlarged and shown at the bottom part of Figure 7.2. Because of the symmetry of the dendrite arms, there is no mass flow through the arm center. Therefore, only half of the arm spacing is considered.

Figure 7.2: A schematic diagram of dendrites in the solid and liquid region

7.2.1 Back Diffusion in the Solid

The evolution of the concentration profile for component *i* in the considered dendrite arm is shown schematically in Figure 7.3.

Figure 7.3: A schematic plot showing the composition distribution of component *i* in a dendrite arm at time *t* and *t+Δt*

C_L^i and C_S^i are compositions of component *i* within the liquid and solid phases (given the unit of wt.% in this work), respectively. *V* is the velocity of *S/L* interface. During the time interval Δ*t*, the S/L interface advances Δx_S (due to solidification) and the length of the solidification region increases by Δ*L* (due to the SDAS coarsening). For the current solidification simulation at each time step, three major tasks are carried out: (1) calculate the composition of each component at the S/L interface including the undercooling effects

and local-equilibrium conditions; (2) solve the diffusion equations within the solid phase; (3) update the length scale to conserve mass balance for every component. More detailed description on the back diffusion can be found in some textbooks [1974Fle, 1985Kur].

7.2.2 Micro-Model for Dendrite Arm Coarsening

The initial SDAS is about twice of the dendrite tip radius: $\lambda_0 \cong 2r_T$ and r_T is described as a function of initial alloy composition, growth rate, and independent of temperature gradient:

$$r_T^2 = \frac{2\gamma D_L T_0 \delta}{V \Delta T_0 k_e \Delta H_f} \tag{7.1}$$

where V, ΔT_0, ke are the interface solidification velocity, freezing temperature range, and equilibrium partition coefficient, respectively. δ is a constant being dependent on the harmonic of the perturbation.

The dendrite arm spacing needs to be known since it sets the diffusion distances in the liquid and solid phases. Owing to the re-melting and re-solidification mechanism, dendrite arm coarsening contributes significantly to homogenization during solidification. The calculation of coarsening is described as below [1986Roo]:

$$\lambda^3 - \lambda_0^3 = \int_0^t gM dt \tag{7.2}$$

λ_0 is the initial SDAS obtained from the calculated dendrite tip radius as described in above Eq. 7.1, and λ is the model predicted SDAS at a certain time. M is coarsening parameter which is proportional to $\lambda^{1/3}$, t is time and g is the geometry factor representing the influence of the dendrite geometry.

For a binary system, the coarsening parameter M is defined as [1990Roo]:

$$M = \frac{\gamma D_L T}{\Delta H_f m_L^v (1 - k_v) C_L^v} \tag{7.3}$$

For a multi- component system, the coarsening parameter must be calculated separately for each alloying element. Then, the following model is used to take into consideration all the solute elements:

$$M = \frac{1}{\sum_{j=1}^n 1/M_j} \tag{7.4}$$

All phase equilibrium related quantities needed in the above equations (such as m_L and k_e) are directly calculated via PanEngine [2009Cao] at each time step by assuming the local equilibrium at the liquid/solid interface.

7.3 The Solidification Kinetic Database Syntax and Examples

The Solidification kinetic database (`.SDB`) uses the XML format, which defines the back diffusion model for the morphology of primary phase and its corresponding model parameters such as interfacial energy, latent heat, coarsening geometric factor, dendrite tip factor, solute trapping parameter, solid diffusivity factor, boundary layer factor, and so on.

In the `SDB`, a series of alloys can be defined. A sample `SDB` file is given below,

```xml
<Alloy name="Mg alloys">
    <solvent name="Mg"/>
    <primary_phase name="Hcp"/>
    <ParameterTable name="">
        <Parameter name="coordinate" value="0" description = "geometry
        of dendrite. 0 for plate; 1 for cylinder; 2 for sphere" />
        <Parameter name="interfacial_energy" value="0.065" description
        = "interfacial energy, unit = J/m^2"/>
        <Parameter name="latent_heat" value="5.5e8" description =
        "latent heat, unit=J/m^3"/>
        <Parameter name="solute_trapping_parameter" value="1e- 9"
        description = "solute trapping parameter, unit=m"/>
        <Parameter name="sound_velocity" value="1000" description =
        "sound velocity, unit=m/s"/>
        <Parameter name="coarsening_geometric_factor" value="40"
        description = "No unit"/>
        <Parameter name="dendrite_tip_factor" value="1" description =
        "No unit"/>
        <Parameter name="solid_diffusivity_factor" value="0.2"
        description = "No unit"/>
        <Parameter name="boundary_layer_factor" value="1" description
        = "No unit"/>
    </ParameterTable>
</Alloy>
</sdb>
```

In this sample `SDB`, "`Mg alloys`" is defined as the name of the alloy, the primary phase is thus set as "`Hcp`" phase. A set of parameters for each phase, such as interfacial energy, latent heat and so on, can be defined in "`ParameterTable`". The kinetic model parameters that can be defined under "`ParameterTable`" are listed in Table 7.1.

Table 7.1: Kinetic model parameters in `sdb`

Name	Unit	Description
Coordinate	N/A	Describe the geometry of dendrite for back diffusion model `<Parameter name="coordinate" value="0" description = "geometry of dendrite. 0 for plate; 1 for cylinder; 2 for sphere" />`
Interfacial_ Energy	J/m^2	Interfacial energy `<Parameter name="interfacial_ energy" value="0.065" description = "interfacial energy, unit = J/m^2"/>`
Latent_heat	J/m^2	Latent Heat of the alloy `<Parametername="latent_ heat" value="5.5e8" description = "latent heat, unit=J/m^3"/>`
Solute_ Trapping_ Parameter	m	Solute trapping parameter `<Parametername="solute_ trapping_ parameter" value="1e- 9" description = "solute trapping parameter, unit=m"/>`
Sound_velocity	m/s	Sound velocity `<Parametername="sound_velocity" value="1000" description = "sound velocity, unit=m/s"/>`
Coarsening_ Geometric_ Factor	N/A	A factor adjusting adjust the coarsen speed of the dendrite `<Parametername="coarsening_geometric_factor" value="40" description = "No unit/>`

7.4 Tutorial

7.4.1 Create a PanSolidification Project

user can create a PanSolidification project through menu "`File → Create a New Workspace`" or "`File → Add a New Project`" in an existing workspace. The "Module Window" pops out for user to choose a module for the new project as shown in Figure 7.4. Choose "`PanSolidification`" module and then click on `Create` button or double click on the `PanSolidification` icon, then a `PanSolidification` project will be created.

Figure 7.4: Creating a PanSolidification workspace

7.4.2 Load Thermodynamic and Mobility Database

The next step is to load the database, which is `MgAl.tdb` in this example. Different from the normal thermodynamic database, this database also contains mobility model parameters for the phases of interest in addition to the thermodynamic model parameters. Both are needed for carrying out solidification simulation. By clicking the 🔳 button on the toolbar, a pop-up window as shown in Figure 7.5 will open, allowing user to select the database file. And Click "`Open`" to select the database, then a window as Figure 7.6 will pop up for user to select components for PanSolidification simulation.

Figure 7.5: Dialog window for loading thermodynamic and mobility database

Figure 7.6: Dialog window for components selection

7.4.3 Load Solidification Kinetic Database

A solidification kinetic database is required for solidification simulation. Such a database contains kinetic parameters which are alloy dependent. To organize these parameters in a more intuitive way, the standard XML format is adopted and a set of well-formed tags are deliberately designed to define back diffusion model for the morphology of primary phase (which could be plate, cylinder and sphere) and its corresponding model parameters such as interfacial energy, latent heat, coarsening geometric factor, dendrite tip factor, solute trapping parameter, solid diffusivity factor and boundary layer factor.

In this example, the `MgAlloys.sdb` is prepared. To load a solidification database, user should navigate the command through menu "**PanSolidification → Load SDB**", or click icon 📗 from the toolbar. After `MgAlloys.sdb` is chosen, a dialog box pops out automatically for user to select the alloy for the simulation. As shown in Figure 7.7, Mg alloys is contained in this `SDB` file.

Figure 7.7: Dialog box for selecting alloy parameter

7.4.4 Solidification Simulation

Perform solidification simulation through menu bar "**PanSolidification → Solidification Simulation**" with Back Diffusion or click icon 🔧 from the tool bar. As shown in Figure 7.8, a dialog box entitled "**Solidification Simulation with Back Diffusion**" pops out for user's inputs to set up the simulation conditions: alloy composition and solidification conditions. When setting the solidification conditions, user need to be careful about the units used for the conditions. user can change the units by clicking on the "**options**" button. Please refer to Section 3.2.3 about the **options** in detail.

Figure 7.8: Dialog box for setting solidification simulation conditions

- **Alloy Composition:** User can set alloy composition by typing in or use the `Load Chemistry` function. User can also save the alloy composition through `Save Chemistry`. This is especially useful when working on a multi-component system, so that user does not need to type in the chemistry every time.

- **Solidification Conditions:** The `Cooling Rate (CR)` of solidification can be defined through `Thermal History` window. The cooling rate determined by cooling curves can also be imported by click the icon 🔄 as shown in Figure 7.8. The `Solidification Rate (V)` and `Temperature Gradient (G)` can be defined from this interface. As these parameters have a relationship of `CR = G*V`, user may choose to provide either `Solidification Rate` or `Temperature Gradient`.

7.4.5 Customize Simulation Results

As all other calculations available in Pandat™, upon the completion of the solidification simulation, a default table with related solidification related properties (time, temperature, secondary dendrite arm spacing, solid and liquid phase fractions, etc.) is automatically

generated and a default graph for temperature (T) vs solid fraction (fs) is displayed as shown in Figure 7.9. User can refer to Section 2.4 and Section 2.5 to learn how to customize simulated graph and table.

Figure 7.9: Default graph plotting Temperature vs fs during solidification

8 PanPhaseField

PanPhaseField is a module of Pandat™ software designed to simulate microstructure evolution. It is also a framework allowing user to implement their own phase-field models.PanPhaseField provides two built-in applications of precipitation and spinodal decomposition.

PanPhaseField is seamlessly integrated with the user-friendly Pandat™ Graphical User Interface (PanGUI) as well as the thermodynamic calculation engine,PanEngine, through an innovative data management system, PanDataNet. PanEngine provides PanPhaseField with reliable thermodynamic data, such as chemical potential, phase equilibrium and mobility of multi-component system. PanDataNet provides data management support which makes phase field simulation of multi-component commercial alloys feasible. Figure 8.1 shows an overall architecture of the PanPhaseField module. With the PanPhaseField SDK, researchers can develop their own phase-field application plugins. Plugins can directly get the needed thermodynamic input from CALPHAD database and use Pandat™ GUI to visualize their simulation results.

Figure 8.1: An overall architecture of the PanPhaseField module

Pandat™ can visualize simulation results in 1-D, 2-D and 3-D. Snapshots of a visualization of 3D spinodal decomposition in Nb-Mo-Ta-Ti-V-Zr is displayed in Figure 8.2

Figure 8.2: Snapshots of visualization through PanPhaseField. 3D Composition fields of Zr, Nb, Ta, Ti, V and Mo from a spinodal decomposition are displayed

Besides using PanGUI to visualize the simulation result, user can also use third-party visualization software compatible with VTK-format, such as ParaView. This is because PanPhaseField saves simulations result in standard .vtk files. Please refer to Section 8.4.8 regarding VTK files from PanPhaseField simulations.

8.1 Phase-Field Method

Phase-field model [2010Wan, 2002Chen] is widely used to simulate microstructural evolution of compositional/structural domains and kinetics of interfaces. By applying Khachaturyan's microscopic elasticity theory [2013Kha], phase-field model can precisely predict precipitate morphologies and effects of point/line/planar/bulk defects. Phase-field model can also predict microstructure evolution in existence of external fields such as elastic field, electric field, and magnetic field. A general expression of phase field model is

$$
\begin{aligned}
F\left[\{\phi_\alpha\left(\vec{r}\right)\}, \{c^i\left(\vec{r}\right)\}\right] =\ & \\
\int dV\ & \left[f\left(\{\phi_\alpha\left(\vec{r}\right)\}, \{\mathbf{c}_\alpha\left(\vec{\mathbf{r}}\right)\}\right) + f^{intf}\left(\{\phi_\alpha\left(\vec{r}\right)\}\right) + f^{grad}\left(\{\mathbf{c}_\alpha\left(\vec{\mathbf{r}}\right)\}\right)\right] \\
+ & \iint G\left(r - r'\right) d^3r d^3r'
\end{aligned}
\tag{8.1}
$$

In the first part of the integration, energy terms such as chemical energy and self-elastic energy are considered in $f(\{\phi_\alpha(\vec{r})\},\{\mathbf{c}_\alpha(\vec{r})\})$, where ϕ_α is some field such as order parameter, and $\mathbf{c}_\alpha$ is usually composition field. Interfacial energy and chemical gradient energy are calculated in f^{intf} and f^{grad}. When applying long-range external field such as elastic field and magnetic field, $G(r-r')$ counts the long-range interactions. Time evolution of microstructure is controlled by the total energy F.

8.2 Built-in Phase-Field Applications

8.2.1 Phase-Field Modeling of Precipitation

In PanPhaseField, a multi-component multi-phase phase field method (MPFM) [1996Ste, 2011Guo, 2018Shi] is adopted to simulate precipitation kinetics. Elastic constants and stress-free transformation strain (SFTS) are used to calculate elastic energy of precipitate and particle morphology can be precisely predicted.

For a system with α =1,2,…,N phases and i=1,2,…,M components, its total free energy functional (F) is given as a sum of the interfacial energy density (f^{intf}), the bulk chemical free energy density (f^{chem}) and the elastic energy (E^{el}).

$$F\left[\{\phi_\alpha(\vec{r})\},\{c^i(\vec{r})\}\right] =$$

$$\frac{1}{V_m}\int dV\ \left[f^{chem}(\{\phi_\alpha(\vec{r})\},\{c_\alpha(\vec{r})\}) + f^{intf}(\{\phi_\alpha(\vec{r})\})\right] \qquad (8.2)$$
$$+ E^{el}\left[\{\phi_\alpha(\vec{r})\},\{c_\alpha(\vec{r})\}\right]$$

The order parameter field, $\phi_\alpha(\vec{r})$, represents the local volume fraction of the phase with its concentration field described by the vector:

$$\mathbf{c}_\alpha(\vec{r}) = (c_\alpha^{j=1}(\vec{r}),\ldots,c_\alpha^{j=M}(\vec{r})) \qquad (8.3)$$

Concentration field of the i-th component is represented as $c^i(\vec{r})$. In MPFM, the order parameter fields follow the constraint:

$$\sum_\alpha \phi_\alpha(\vec{r}) = 1 \qquad (8.4)$$

From the order parameter $\phi_\alpha(\vec{r})$ and the phase composition $\mathbf{c}_\alpha(\vec{r})$, the local concentration $c^j(\vec{r})$ can be determined using the mass conservation relation:

$$c^j(\vec{r}) = \sum_\alpha \phi_\alpha(\vec{r}) c^j_\alpha(\vec{r}) \tag{8.5}$$

PanEngine/PanDataNet is coupled with MPFM through the chemical energy density f^{chem},

$$f^{chem}\left(\{\phi_\alpha(\vec{r})\},\{c(\vec{r})\}\right) = \sum_\alpha \phi_\alpha(\vec{r}) f^\alpha(c_\alpha(\vec{r})) \tag{8.6}$$

Free energy of each individual phase, $f^\alpha(\{c_\alpha(\vec{r})\})$, is calculated from PanEngine. For the given local composition $c(\vec{r})$ and phase field $\phi_\alpha(\vec{r})$, individual phase composition $c_\alpha(\vec{r})$ vectors are evaluated through KKS model [1999Kim].

To solve for the individual phase concentration, it is assumed that local diffusion potentials are same in all the phases, which is expressed mathematically [1999Kim] as

$$\frac{\partial f^{chem}_\alpha}{\partial c^i_\alpha} = \frac{\partial f^{chem}_\beta}{\partial c^i_\beta} \tag{8.7}$$

where α and β are the phase indices and *i* is the component index. For a given set of order parameter fields $\phi_\alpha(\vec{r})$ and composition fields $c_\alpha(\vec{r})$, phase compositions are uniquely determined by the Eq. 8.5 and Eq. 8.7.

The interfacial free energy density is given by

$$f^{intf}\left(\{\phi_\alpha(\vec{r})\}\right) = \sum_\alpha^N \sum_{\beta>\alpha}^N -\frac{\kappa_{\alpha\beta}}{2}\nabla\phi_\alpha(\vec{r}).\nabla\phi_\beta(\vec{r}) + \omega_{\alpha\beta}\left|\phi_\alpha(\vec{r})\phi_\beta(\vec{r})\right| \tag{8.8}$$

where $\kappa_{\alpha\beta}$ and $\omega_{\alpha\beta}$ are the gradient energy coefficients and potential energy humps across α/β interface and they together determine the interfacial energy $\sigma_{\alpha\beta}$ and boundary width $\Lambda_{\alpha\beta}$.

The interfacial energy between two phases α and β is evaluated to be:

$$\sigma_{\alpha\beta} = \frac{\sqrt{\kappa_{\alpha\beta}\omega_{\alpha\beta}}}{3\sqrt{2}} \tag{8.9}$$

The elastic strain energy dominates particle morphology and variant selection at coarsening stage. Elastic strain energy functional, $E^{el}\left(\{\phi_\alpha(\vec{r})\}\right)$, is formulated according to

Khachaturyan's microscopic elasticity theory [2013Kha] and modified in the framework of MPFM [2006Ste]. The key inputs in the formulation of $E^{el}\left(\{\phi_\alpha(\vec{r})\}\right)$ are the SFTS (stress-free transformation strain) of each phase and the elastic constants C_{ijkl} of each phase.

The SFTS is defined in form of a matrix:

$$
\varepsilon = \begin{bmatrix} \varepsilon_{11} & \varepsilon_{12} & \varepsilon_{13} \\ \varepsilon_{21} & \varepsilon_{22} & \varepsilon_{23} \\ \varepsilon_{31} & \varepsilon_{32} & \varepsilon_{33} \end{bmatrix}
$$

Generally, matrix phase, for example Fcc, plays as reference phase of this SFTS matrix. When a cubic-to-cubic transformation from matrix to precipitate phase happens, ε_{11} = ε_{22} = ε_{33} and ε_{ij} = 0 for $i \neq j$. When a cubic -> tetragonal transformation occurs, the SFTS could be one of ε_{22} = ε_{33} or ε_{11} = ε_{33} or ε_{11} = ε_{22} depending on the orientation variants of precipitation. Other symmetry breaking will introduce a much more complex form of SFTS, which is beyond discussion in this manual.

The coherency elastic strain energy is formulated following Khachaturyan's microscopic elasticity theory [2013Kha] under the homogeneous modulus assumption when external applied stress is zero:

$$
E^{el} = \frac{1}{2} \int \sum_{p,q=1,2} \frac{d^3k}{2\pi} B_{pq}\left(\boldsymbol{n}\right) \left|\widetilde{\phi}_p\right| \left|\widetilde{\phi}_q\right|^* \tag{8.10}
$$

Where p and q are the index of two phases. $\boldsymbol{n}$ is a unit vector in the reciprocal space along the direction of k. $\left|\widetilde{\phi}_p\right|$ is the Fourier transformation of the absolute value of the structural order parameter ϕ_p. The superscript asterisk in $\left|\widetilde{\phi}_q\right|^*$ denotes the complex conjugate. $B_{pq}\left(\mathbf{n}\right)$ is calculated as:

$$
B_{pq}\left(\boldsymbol{n}\right) = \begin{cases} \boldsymbol{n} = 0: \ B_{pq}\left(\boldsymbol{n}\right) = 0 \\ \boldsymbol{n} \neq 0: \ B_{pq}\left(\boldsymbol{n}\right) = C_{ijkl}\varepsilon_{ij}^{*p}\varepsilon_{kl}^{*q} \end{cases} \tag{8.11}
$$

The phase fields $\phi_\alpha(\vec{r})$ and concentration fields $c^j(\vec{r})$ are evolved in time through MPFM and Cahn-Hilliard dynamics respectively,

$$
\frac{\partial \phi_\alpha}{\partial t} = - \sum_{\beta=1}^{\widetilde{N}} \frac{L_{\alpha\beta}}{\widetilde{N}} \left(\frac{\delta F}{\delta \phi_\alpha} - \frac{\delta F}{\delta \phi_\beta} \right) \tag{8.12}
$$

$$\frac{1}{V_m^2} \frac{\partial c^j}{\partial t} = \sum_{k=1}^{n-1} \nabla \left(M^{kj} \nabla \left(\frac{\partial f^{chem}}{\partial c^k} - \kappa_k \nabla^2 c^k \right) \right) \qquad \textbf{(8.13)}$$

where M^{kj} is the chemical mobility [2004Che], $L_{\alpha\beta}$ is the interface mobility and $\widetilde{N}$ is the number of phases that coexist locally. $\frac{\partial f}{\partial c^k} = \mu^k - \mu^n$, where μ^k and μ^n are the chemical potentials of components k and n, respectively. Chemical mobility M^{kj} of a phase [2004Che] is given as

$$M^{kj}(\{c^j\}, T) = \frac{1}{V_m} \sum_{l}^{n} \left(\delta_{kl} - c^k \right) \left(\delta_{jl} - c^j \right) c^l M^l \qquad \textbf{(8.14)}$$

where M^l is the atomic mobility and c^l is the composition of l-th element. The Eq. 8.12 and Eq. 8.13 are solved concurrently.

In the precipitation model, there are several treatments related with kinetics of order parameters:

- **Driving-force scaling method**: Sometimes, when we model the precipitate growth, the driving force for transformation may become extremely large which may cause numerical instability. Hence, we use a scaling approach to reduce the driving force for phase transformation.

- **Mobility of order parameter**: For diffusion-controlled precipitation processes, the mobility of the order parameter (M_η) should be large enough to ensure diffusion-controlled interface movement. But if M_η is too large, we will observe the congruent transformation. Hence, we choose an appropriate M_η inside PanPhaseField automatically to maintain this diffusion-controlled assumption.

- **Interface width**: The interface width in the KKS model is simply a numerical parameter and does not correspond to the real interface width. For numeric stability, we recommend an interface width of 5 grid points.

8.2.1.1 Explicit nucleation method

In the nucleation simulation of PanPhaseField, the transient nucleation rate is calculated by:

$$J = N_v Z \beta^* \exp\left(-\frac{\Delta G^*}{k_B T}\right) \exp\left(-\frac{\tau}{t}\right)$$

(8.15)

The pre-exponential term in Eq. 8.15 are: N_v, the nucleation site density, Z, the Zeldovich factor and β^*, the atomic attachment rate. t is the time, τ is the incubation time for nucleation, k_B the Boltzmann constant and T the temperature. The ΔG^* is the nucleation barrier which is calculated according to interfacial energy, chemical driving force and elastic strain. For details of the nucleation rate calculation, please refer to Section 5.2.1 .

The nucleation implanting algorithm in PanPhaseField follows a probabilistic Poisson seeding process at a probability of forming a critical nucleus [2019Shi]:

$$P(r, t) = 1 - \exp(-J(r, t) \Delta V \Delta t)$$

(8.16)

Where $P(r,t)$ is the probability of forming a nucleus with critical radius in a volume ΔV and a time interval Δt and position r starting from time t.

8.2.2 Phase-Field Modeling of Spinodal Decomposition

As a special case of microstructure evolution model, spinodal decomposition does not consider interactions between phases, and the elastic energy is dominated by chemical misfit instead of phase misfit. Gradient term in spinodal decomposition is calculated by chemical gradient instead of order parameter gradient. As a result, order parameter evolution (the Eq. 8.12) is not relevant and only Cahn-Hilliard (the Eq. 8.13) dynamics needs to be solved. The formula of elastic energy in spinodal decomposition is constructed as:

$$E^{el} = \frac{1}{2} \sum_{p,q} B_{pq} \, \tilde{c}_p^* \tilde{c}_q$$

(8.17)

When spinodal decomposition is simulated, the gradient term will be replaced by using concentration field instead of phase field:

$$f^{intf}\left(\{c(\vec{r})\}\right) = \sum_{j=1}^{n-1} \frac{\kappa_j}{2} \left(\nabla c^j(\vec{r})\right)^2$$

(8.18)

where κ_j is the gradient energy coefficient.

In spinodal decomposition, a homogeneous solid solution decomposes into two or more isostructural phases which are different in composition. Certain wavelength of decomposition grows faster than the other wavelengths owing to thermodynamic driving force and kinetic factors. By linearizing the Cahn-Hillard equation for a binary alloy, the maximally growing wavelength in the initial stages of the decomposition can be calculated by

$$\lambda_{max} = 2\pi\sqrt{2\kappa/(-f'')}$$

(8.19)

where κ is the gradient energy coefficient and f'' is the second derivative of Gibbs free energy.

Table 8.1 lists the symbols used in the phase-field models.

Table 8.1: Symbols used in the phase-field model

Symbol	Description
α, β	Index used for various phases
i, j, k, l	Index used for various components
$\phi_\alpha(\vec{r})$	Structural order parameter of the α phase
$c_\alpha(\vec{r})$	Composition vector of the α phase
$c^i(\vec{r})$	Compositional order parameter of the i-th component
$c_\alpha^i(\vec{r})$	Phase concentration of i-th component in α phase
f^{chem}	Chemical free energy density
C_{ijkl}	Stiffness tensor
ε_{ij}	Stress free transformation strain
$\kappa_{\alpha\beta}$	Gradient energy coefficient between α and β phases
$\sigma_{\alpha\beta}$	Interfacial energy (J/m^2)
κ_i	Gradient energy coefficient for component i
$L_{\alpha\beta}$	Interface mobility
M^{kj}	Chemical mobility
M^l	Atomic mobility for l-th component

8.3 Syntax and Examples of Phase-Field Database File (.PFDB)

The `phase-field database file` (`.pfdb`) uses the XML format, which defines the phase-field model related parameters as well as physical properties for each phase. Phase-field model related parameters include order parameter mobility, chemical gradient term, smoothness factor, driving force scaling factor, interface width, chemical random amplitude, chemical gradient factor and so on. Physical parameters of each phase include molar volume, stiffness tensor, chemical misfit strain, stress-free transformation strain, interfacial energy and so on. Nucleation related parameters can also been set through `.pfdb` file. The database file is designed for easy extension when user uses a customized phase-field model plug-in.

In a `.pfdb` file, a series alloys can be defined. Each alloy has a matrix phase with one or multiple precipitate phase. User can define new types of phases when models other than precipitation are used as plugin. A sample `.pfdb` file structure is shown as follows,

```xml
<Alloy name="AlNi_Precipitation">
    <Phase type="Matrix" name="Fcc">
        <ParameterTable type="phase_field_general" name="">
            <Parameter  type="Order_ Parameter_ Mobility"  value="0.1"
description=""/>
            <Parameter  type="Driving_ Force_ Scaling_ Factor"  value="8.0"
option="enable"  description="A  factor  to  scale  order  parameter  driving
force"/>
            <Parameter  type="Interface_ Width"  value="5"  description="grid
number of interface"/>
        </ParameterTable>
        <ParameterTable type="molar_volume" name="">
            <Parameter  name="*"  value="1E- 5"  description="Default  value
1E-5"/>
        </ParameterTable>
        <ParameterTable type="stiffness_tensor" name=" description=" ">
            <Parameter name="C11" value="2.03E12" description=" "/>
            <Parameter name="C12" value="1.50e12" description=" "/>
            <Parameter name="C44" value="1.35e12" description=" "/>
        </ParameterTable>
            <Phase type="Precipitate" name="L12_Fcc">
        <ParameterTable type="inter_phase" name="" reference="Fcc">
```

```
        <Parameter      type="Interfacial_  Energy"      value="0.025"
description="J/m^2 "/>
        <Parameter  type="e11"  value="- 0.0005"  description="Misfit
strain"/>
      </ParameterTable>
    </Phase>
  </Phase>
  </Alloy >
```

In this sample `.pfdb` file, an alloy "`AlNi_Precipitation`" is defined with the matrix phase "`Fcc`", which has a precipitate phase "`L12_Fcc`". More than one precipitate can be included. The precipitate name and physical properties can be defined for each precipitate phase. A set of physical parameters for each phase, such as molar volume, elastic modulus, stress-free transformation strain, interfacial energy, and so on, can be defined in "`ParameterTable`". Phase-field related parameter defined in the "`ParameterTable`" with type of "`phase_field_general`" are listed in Table 8.2. The physical parameters available for the precipitation model and spinodal decomposition model are listed in Table 8.3 and Table 8.4. The nucleation model related parameters are listed in Table 8.5, which are consistent with the definition in Table 11.8 in Section 11.3 .

Table 8.2: Phase-field model parameters in pfdb

Name	Description
Order_Parameter_ Mobility	Mobility of order parameter. This parameter decides the rate of time-evolution controlled by the Eq. 8.12
Driving_Force_ Scaling_Factor	A scale-factor for the driving force for phase transformation. To disable it, set option to "disable"
Interface_Width	Number of grids which sample an interface between two phases. Default value is 5 grids.
Chemical_ Gradient_Factor	For spinodal decomposition model only. It is used to calculate gradient energy term introduced by chemical inhomogeneous.
Chemical_ Random_ Amplitude	For spinodal decomposition model only. It is used to generalize initial composition profile with a sufficient degree of randomness which can initiate decomposition.
Bpq_Substract	For precipitation simulation, it decides if average elastic energy is subtracted from local elastic energy. When average elastic energy is subtracted, the overall volume fraction of precipitate phase should be independent of elastic energy. When it is set to 1, average elastic energy is subtracted from local elastic energy. Default value is 1.
Enable_ Precipitate_ Diffusion	For precipitation simulation, chemical mobility can be set to zero inside precipitate If matrix-diffusion is assumed to dominating the kinetics. When it is set to 1, diffusion is considered inside precipitate phases. Default value is 1.
Mean_Field_ Driving_Force_ Nucleation	For nucleation model, nucleation driving force can be calculated by either chemical composition or average chemical composition (mean-field) in matrix. The mean-field assumption is computationally faster. When it is set to 0, local chemical composition is used to calculate driving force of nucleation. Default is 0.

Table 8.3: Single phase physical properties in pfdb

Name	Unit	Description

molar_volume	m³/mole	Molar volume of matrix or precipitate phase
		
		`<ParameterTable type="molar_volume" name="">`
		`<Parameter name="*" value="1E-5" description=""/>`
		`</ParameterTable>`

stiffness_
tensor
	Pa	Stiffness tensor following Voigt notation
		
		`<ParameterTable type="stiffness_tensor" name=""`
		`description="Voigt   notation;   Orthotropic   elasticity   is`
		`applied  when  only  C11,  C12,  C44  are  set;  Otherwise,  set`
		`stiffness tensor explicitly">`
		`<Parameter name="C11" value="2.03E12" description=""/>`
		`<Parameter name="C12" value="1.50e12" description=""/>`
		`<Parameter name="C44" value="1.35e12" description=""/>`
		`<!--Default values of C22 and C33 are C11-->`
		`<!--Default values of C21, C23, C32, C31 and C13 are C12-`
		`-->`
		`<!--Default values of C55, C66 are C44-->`
		`<!--Default values of C45, C56 and C46 are 0.0-->`
		`<!--Default values of C14, C15, C16, C24, C25, C26, C34,`
		`C35 and C36 are 0.0-->`
		`</ParameterTable>`

chemical_
misfit_strain
	N/A	For spinodal decomposition model only. Chemical misfit strain
		
		`<ParameterTable type="chemical_misfit_strain"`
		`name="Chemical_Misfit_Strain" reference="Fe">`
		`<Parameter name="Ni" value="-0.01"`
		`description="Default value is 0.0"/>`
		`</ParameterTable>`

mobility_
scaling_factor
	N/A	Scaling factor for the mobility from CALPHAD database
		
		`<ParameterTable type="mobility_scaling_factor"`
		`name="Mobility_Scaling_Factor">`
		`<Parameter name="Mo" value="6.3e-19"`
		`description="Default value is 1.0"/>`
		`</ParameterTable>`

Table 8.4: Inter-phase physical properties in pfdb

Name	Unit	Description
Interfacial_Energy	J/m^2	Interfacial energy between precipitate phase and matrix phase `<Parameter type="Interfacial_Energy" value="0.05" description=" "/>`

e11, e12, e13, e21, e22, e23, e31, e32, e33	N/A	Stress-free transformation strain between precipitate phase and matrix phase
		`<Parameter type="e11" value="-0.003" description=" "/>`
		`<!--Default value of e22 and e33 is e11-->`
		`<!--Default value of e12, e13, e23 is 0.0-->`
		`<!--Default value of e21 is e12-->`
		`<!--Default value of e31 is e13-->`
		`<!--Default value of e32 is e23-->`

Table 8.5: Nucleation model related parameters in pfdb

Name	Unit	Description
Atomic_Spacing	*m*	Usually use lattice constant `<Parameter type="Atomic_Spacing" value="7.6E10" description="Atomic Spacing" />`
Nucleation_ Site_ Parameter	N/A	Homogenous model only for the pre-installed precipitation model
Steady_State_ Nucleation_Rate	N/A	Set to 1 if steady state nucleation rete is applied. `<Parameter type="Steady_ State_ Nucleation_ Rate" value="1" description="Indicate if steady state nucleation rate is applied" />`
Driving_Force_Factor	N/A	A scaling factor of nucleation driving force
Effective_ Diffusivity_ Factor	N/A	A scaling factor of effective diffusivity for nucleation rate model
Shield_Time	second	A time period in which stability of nuclei is numerically sustained.

8.4 Tutorial

8.4.1   Step 1: Create a PanPhaseField Project

One can create a PanPhaseField project through menu "`File → Create a New Workspace`" or "`File → Add a New Project`" in an existing workspace. The "`Module Window`" pops out for user to choose a module for the new project as shown in Figure 8.3. Choose "`PanPhaseField`" module for phase-field simulation, and the PanPhaseField project will be created after user click on `Create` button or double click on the `PanPhaseField` icon.

Figure 8.3: Creating a PanPhaseField workspace

8.4.2 Step 2: Load Thermodynamic and Mobility Database

The next step is to load the database, which is "`AlNi_Prep.tdb`" in this example. Different from the normal thermodynamic database, this database also contains mobility data for the phases of interest in addition to the thermodynamic model parameters. Both are needed for carrying out phase-field simulation. By clicking the button on the toolbar, a popup window will open, allowing user to select the database file. User may select the `TDB` file and click on `Open` button or just double click on the `TDB` file to load it into the Workspace as shown in Figure 8.4.

Figure 8.4: Dialog window for loading a combined thermodynamic and mobility database

8.4.3 Step 3: Load Phase-Field Database File

In addition to a combined thermodynamic and mobility database, a phase-field database (`.pfdb`) file which contains kinetic parameters for the alloy to be studied is also required for phase-field simulation.

A `pfdb` file is organized by the standard `XML` format, and a set of well-formed tags are deliberately designed to define phase-field model parameters for general and physical parameters of each phase.

In this example, the "`AlNi_diffusion_only.pfdb`" is prepared. This file can be opened by click 📂 button in the toolbar and be viewed in the Pandat™ workspace or through third-party external editors. The advantage to open this file in Pandat™ workspace is that all the key words will be highlighted which makes it easy to read. In this example, the matrix phase for this alloy is "`Fcc`", which has one precipitate phase "`L12_Fcc`". Phase field parameters, such as order parameter mobility and interface width, are defined in matrix.

To load a phase-field database, user should navigate the command through menu "`PanPhaseField → Load PFDB`", or click icon 🔣 from the toolbar. After `AlNi.pfdb` is chosen, a dialog box pops out automatically for user to select alloy and phases as shown in Figure 8.5. To select several phases at one time, for example in precipitation simulation where at least two phases should be selected, press and hold the `<Ctrl>` key.

Figure 8.5: Dialog window for phase-field database (`PFDB`)

8.4.4 Step 4: (Optional) Prepare VTK Files as Input Microstructure

PanPhaseField can use an input microstructure as initial condition. This is especially common when a digitized SEM picture or result from other simulation is used as initial condition. PanPhaseField follows standard VTK file format to load input microstructure. For example, in precipitation simulation, user can prepare a comprehensive VTK file containing order parameter of each phase and composition distribution. An example of this comprehensive VTK file is available in "`input_2D_AlNi`" of the example folder.

Please note that, the dimension in VTK file should be self-consistent, or an error message will popup when PanPhaseField tries to start a simulation. Please follow the examples in "`input_2D_AlNi`" to prepare VTK files as input.

8.4.5 Step 5: PanPhaseField Simulation

After successfully loading both thermodynamic/mobility and phase-field database file, and optionally the input VTK files, the phase-field simulation module is then being activated. To choose from Pandat™ GUI menu "`PanPhaseField → Phase Field Simulation`", or click icon ▶ from the toolbar. A dialog box entitled "`Phase Field Simulation`", as shown in Figure 8.6, pops out for user's inputs to set up the simulation condition: alloy composition, thermal history, and initial microstructure.

Figure 8.6: Dialog box for setting phase-field simulation conditions

- **Alloy Composition:** User can set alloy composition by typing in or use the "`Import from folder`" function under "`Initial Condition`". When "`Import from folder`" function is used, order parameters will also be loaded according to input files (VTK format).

- **Thermal History:** Arbitrary heat treatment schedule can be set, with linear Time-Temperature relationship (i.e., constant cooling or heating rate) at each two consecutive rows. Time at the first row must be zero representing the initial time. The thermal history set up in Figure 8.6 represents an isothermal heat treatment for 5 hours at 800 °C. If the temperature in the two rows is different, it represents constant cooling or heating. Multi-stages of heat treatment can be set by adding more rows in the Thermal History column.

- **Phase Field Model:** Select a built-in application or a user defined plugin. In Figure 8.7, "`spinodal`" and "`precipitation`" are built-in applications, at the same time user defined customized applications (plugins) are named "`USER_PFM_XXX`".

Figure 8.7: Select phase field application from built-in ones or use-defined ones

- **Initial condition:** phase field simulation must be initialized with a pre-existing microstructure without explicitly defining nucleation mechanism by a model. There are three `Pre-Defined` initial conditions: `random`, `sphere`, and `empty`. `Random` is used for spinodal decomposition simulation to apply randomized initial composition profile. `Sphere` is used for precipitation simulation applying a sphere particle with radius of 5 grids at the center of simulation box. `Empty` is used for simulation without explicitly initialize microstructure, for example, a nucleation simulation.

  When "`Import from folder`" function is used, initial composition profile and/or initial order parameters following the data defined in `VTK` file(s) should be prepared before loading it.

- **Run phase-field simulation:** In `Phase Field Simulation` window (Figure 8.6), click `OK` to run the simulation. After the simulation is completed, its results are displayed in two types of formats in Pandat™ Explorer window: `Graph` and `Table`. By default, the graphs plotting microstructure and volume fraction of precipitate are shown in Figure 8.8 and Figure 8.9.

Figure 8.8: Simulated Microstructure of the Ni-15Al alloy after heat treated at 800°C for 5 hours

Figure 8.9: Volume fraction vs. time

8.4.6 Step 6: Customize Simulation Results

Upon the completion of the phase-field simulation, a default table with related properties (such volume fraction) is automatically generated and a default graph is displayed in Figure 8.9.

However, it should be emphasized that, in addition to the default tables, variety of properties, such as temperature, nucleation driving force, nucleation rate, average composition of matrix phase can be retrieved through "**Add a new table**".

8.4.7 Step 7: Add a New Table

To display other properties in addition to those listed in the Default table, add a new table function should be used through GUI.

Add a new table function can be activated by the following ways: (1) selecting the "**Table**" node in the Explorer window, then right click the mouse and select "**Add a New Table**" option (as shown in Figure 8.10); or (2) choose "**Add or Edit a Table**" from the "**Table**" menu. This function allows user to create a new table at their own choices.

Figure 8.10: Dialog of add a new table

Figure 8.11: Dialog window of table editor

The basic layout of the window of "`Add a new table`" is shown in Figure 8.11. User can enter the available variables to the "`Columns`" field. The description of the available variables are listed in Table 8.6.

Table 8.6: Available variables for precipitation simulation with nucleation model

Variable Name	Description
time	Time of elapsed thermal condition
log10(time)	Log-scale time of elapsed thermal condition
T	Temperature
f(@*)	Volume fraction of each phase
x(*)	Overall chemical composition in mole fraction
w(*)	Overall chemical composition in weight fraction
x(*@*)	Average chemical composition of each phase in mole fraction
w(*@*)	Average chemical composition of each phase in weight fraction
nr(@*)	Nucleation rate of each phase. Unite is m^{-3}s^{-1}\
dgm(@*)	Driving force of each phase. Unit is J/mol

8.4.8 Step 8: Get VTK files

For the convenient of visualizing using third-party software such as ParaView, original VTK files of the phase-field simulation can be obtained by navigating to the Table folder follow the steps:

(1) Open any microstructure file in GUI window, as shown in Figure 8.12.

(2) "`Open containing folder`" which contains this microstructure file, as shown in Figure 8.13.

(3) Select all `.VTK` files for usage, as shown in Figure 8.14. (`.VTKB` all binary format of VTK files, which cannot be directly viewed by text editor, but can be opened by ParaVeiw)

Figure 8.12: Open VTK files: Step 1

Figure 8.13: Open VTK files: Step 2

Figure 8.14: Open VTK files: Step 3

8.4.9 Step 9: Load VTK files

Simulation results ($\mathtt{VTK}$ files) from batch mode (without using GUI) can be loaded to Pandat™ workspace for visualization. Users need to prepare two items to visualize $\mathtt{VTK}$ files in Pandat™ workspace:

(1) The batch file which describes the simulation performed in batch mode.

(2) The folder containing `VTK` files.

Batch file and `VTK` files must be consistent, for example they come from a same simulation. Follow the steps to load the `VTK` files:

(1) Create a new result node in Pandat™ workspace.

Figure 8.15: Create a new result node

(2) Select a batch file for the batch mode calculation.

Figure 8.16: Select batch file

(3) Select corresponding `VTK` folder form the same batch mode calculation

Figure 8.17: Select VTK folder

(4) After clicking OK in Figure 8.17, VTK files will be visualized.

Figure 8.18: Visulize VTK files

8.4.10 Use a Plugin Defined by User

User-defined plugins can be loaded through Pandat GUI to perform a simulation using customer's self-developed model.

Plugin should be organized inside "`Pandat 2024/bin/PanPhaseField_Plugins`" folder. Plugins inside this folder will be visible by PanPhaseField's plugin viewer. Choose "`PanPhaseField → Plugins`", the Plugin viewer will show as in Figure 8.19.

Figure 8.19: PanPhaseField's plugin viewer

To perform a spinodal decomposition, follow Section 8.4.2 and Section 8.4.3 to load the database files "`FeCrNi.tdb`" and "`FeCrNi.pfdb`" from "`Pandat 2024 Examples/PanPhaseField/plugin_spinodal`" folder. Set the condition and select `UER_PFM_SPINODAL` plugin from **Phase Field Models**, as is displayed in Figure 8.20. And the simulation result is displayed in Figure 8.21.

Figure 8.20: Simulation condition for `USER_PFM_SPINODAL` plugin

Figure 8.21: Simulation result for `USER_PFM_SPINODAL` plugin

9 Property

Property calculation has been added into Pandat™ and is characterized in three categories, thermodynamic property, physical property and kinetic property. Users can not develop other property calculations from the "User-Defined Property". All the property calculations must be carried out at **PanPhaseDiagram** module. The database format and the equations for these properties are described below.

9.1 Thermodynamic Property

The `Thermodynamic Property` function is used to calculate thermodynamic properties, such as Gibbs energy, enthalpy, entropy, chemical potential, activity, etc. It is under the PanPhaseDiagram module. User can access this function through the menu ("`Property →` `Thermodynamic Property`") after the thermodynamic database is loaded. A popup window will allow user to input calculation conditions as shown in Figure 9.1.

Figure 9.1: Thermodynamic Property calculation dialog

The calculation is set as a line calculation with the desired thermodynamic properties output. User may select the thermodynamic properties as needed by checking the boxes in front of each property as shown in Figure 9.1 and specify the reference state for each element. The selected property will be calculated and graph plotted as shown in Figure 9.2. If two or more thermodynamic properties are selected, they will be plotted separately. Experienced user can also get these results directly from the `Line Calculation` in the `PanPhaseDiagram` module with proper syntax using self-defined table.

Figure 9.2: Thermodynamic Property calculation results

9.2 Physical Property

Physical property calculation implemented in Pandat™ allows user to calculate the molar weight, molar volume, density, surface tension and viscosity. The interface is similar to the Thermodynamic property calculation as shown in Figure 9.3. User can set up the calculation condition as a line calculation and select the properties to be calculated. The default graph is the calculated property as shown in Figure 9.4 as an example. Again, if two or more properties are selected, all of them will be plotted one by one.

The detailed calculation method for each property and the format of the database file are described in the following sections.

Figure 9.3: Physical property calculation dialog

Figure 9.4: Physical property calculation results

9.2.1 Weight

The molar weight for a phase is calculated from the atomic weights of the elements involved in the phase. The unit is `kg/mol·atoms`. Since atomic weights of elements have been included in Pandat™, no input is required in the database `TDB` file.

For example, an `Fcc` phase in `Al-Cu` binary system with x_{Al} = 0:9 and x_{Cu} = 0:1, its molecular weight is

$$
\begin{aligned}
MW^{Fcc} &= x_{Al}M_{Al} + x_{Cu}M_{Cu} \\
&= 0.9 \times 26.982 \times 10^{-3} + 0.1 \times 63.546 \times 10^{-3} \\
&= 30.6384 \times 10^{-3} \quad (kg/mol \cdot atoms)
\end{aligned}
\tag{9.1}
$$

9.2.2 Volume

The molar volume can be calculated by Pandat™ if the model parameters are properly defined in the database file. The molar volume of a pure component is described as a function of temperature and pressure, and the excess molar volume of a phase is described in the format similar to that of excess Gibbs energy. The unit of molar volume is m^3/mol·atoms. In the database (`.TDB`) file, the format of molar volume is:

```
PARAMETER Vm(Fcc,Al;0) 298.15 +9.7743e-006
*exp(6.91213e-005*T+1.62267e-011*T**3+0.413484*T**(-1)); 3000 N !
PARAMETER Vm(Fcc,Fe;0) 298.15 +6.72092e-006*exp(6.97895e-005*T); 3000
N !
PARAMETER Vm(Fcc,Al,Fe:Va;0) 298.15 -3e-6; 3000 N !
```

The calculation of molar volume for a phase is

$$V_m = \sum x_i V_{m_i}^o + \sum x_i x_j V_{m_{ij}} (x_i - x_j)^k \tag{9.2}$$

where $V_{m_{ij}}$ is the parameter for the excess molar volume of this phase.

Molar volume of a system with phase mixture is calculated by

$$V_m = \sum f^\phi V_m^\phi \tag{9.3}$$

where f^ϕ and V_m^ϕ are the molar fraction and molar volume for phase ϕ.

Note that, if a phase is described by a compound-energy-formalism (CEF) with multi-sublattices, the molar volume is automatically described as a function of T, P, and site fraction (y) following the format of the Gibbs energy. This will introduce many end members in the molar volume and may lead to instability in the calculated molar volume. In reality, molar volume should be a function of mole fraction (x) instead of site fraction (y), it is thus recommended to define the molar volume of a phase separately from its Gibbs energy if the phase is described by the CEF model. This can be realized through the user-defined molar volume property (Section 3.3.10).

Using the Al-Ni binary system as an example, type "VARIABLE_X" is defined as property "V_m":

    Type_Definition v GES AMEND_PHASE_DESCRIPTION * VARIABLE_X Vm !

which means any phase with the Type_Definetion "v" will use mole fractions as the variables of V_m. Here are examples to define the molar volume parameters for the Liquid and $L1_2$_FCC phases:

    Phase Liquid %v 1 1 !
    Parameter Vm(Liquid,Al;0) 298.15 +V_Al_Liquid; 3000 N !
    Parameter Vm(Liquid,Ni;0) 298.15 +V_Ni_liquid; 3000 N !
    Phase L12_FCC %v 2 0.75 0.25 !
    Parameter Vm(L12_FCC,Al;0) 298.15 +0.935*V_Al_fcc; 3000 N !
    Parameter Vm(L12_FCC,Ni;0) 298.15 +0.935*V_Ni_fcc; 3000 N !

In this case, the molar volume of L12_FCC is calculated as:

$$V_m = x_A V_{Al}^{L1_2} + x_{Ni} V_{Ni}^{L1_2} \tag{9.4}$$

Although the `L12_FCC` phase is described by `CEF` model with two sublattices. Please refer to the Example 1.21 in the example book for detail information.

9.2.3 Density

Density is calculated from molar volume and the molar weight. It requires the molar volume parameters in database. The unit for density is kg/m^3. Density of a phase is defined as

$$d^\phi = \frac{w_m^\phi}{v_m^\phi} \tag{9.5}$$

The density of a system with phase mixture is calculated from the molar weight and the molar volume of the mixture:

$$d = \frac{w_m}{v_m} \tag{9.6}$$

9.2.4 Viscosity

A function to calculate viscosity of liquid phase is added into current Pandat™. The model used for describing the viscosity of liquid phase is semi-empirical relation presented in the paper by Seetharaman and Du [1994See],

$$\eta = A \exp\left(\frac{\Delta G^*}{RT}\right) \tag{9.7}$$

with

$$A = \frac{hN}{V} \tag{9.8}$$

where V is the molar volume of the melt, h the Plank's constant, N the Avogadro's number. ΔG^* the Gibbs energy of activation, and can be calculated by

$$\Delta G^* = \sum x_i \Delta G_i^\circ + \Delta^m G_{mix} + 3RT \sum \sum x_i x_j \tag{9.9}$$

where x_i and x_j are the molar fractions of component i and j, ΔG_i°, Gibbs energy of activation of component i, and $\Delta^m G_{mix}$, Gibbs energy of mixing. The calculation of molar volume is given in above sections. The parameter for the activation energy of a pure component in database `TDB` file has the format:

```
    Parameter  ActivationEnergy(Liquid,Al;0)  298.15  15051+13.519*T;  2000  N
!
```

9.2.5 Surface Tension

Current Pandat™ has another new function to calculate surface tension of liquid phase. The model for calculating surface tension of liquid is the semi-empirical relationship proposed by Yeum et al. [1989Yeu] to estimate the surface tensions of binary alloys based on the model of Butler [1932But].

$$\sigma = \sigma_i + \frac{RT}{S_i} ln \frac{a_i'}{a_i}$$

(9.10)

where σ_i, a_i' and a_i are the surface tension, activity at the surface and activity in the bulk of component i. And S_i is the surface monolayer area,

$$S_i = bN^{1/3}V_i^{2/3}$$

(9.11)

where b is geometric factor, N the Avogadro's number, and V_i molar volume of component i. This approach was extended to calculate surface tensions of multicomponent liquid alloys [1997Zha]. The calculation of molar volume is the same as given in Section 9.2.2 . The ratio of the coordination number for the surface atoms to that for the atoms in the bulk phase, b, is represented by a parameter `beta`, described in database (`.tdb`) file in the format as

```
    Parameter  Beta(Liquid,Al;0)  298.15 0.83;   2000  N !
```

9.3 Kinetic Property

Diffusivity related properties can be calculated from Pandat™ as long as the corresponding species mobility parameters are available in the database. User can input the calculation condition through the interface shown in Figure 9.5 and select the desired properties for output. The default graph is the selected property as shown in Figure 9.6. If two or more properties are selected, they will be plotted separately.

Details on the kinetic models for the multicomponent diffusion are referred to the literature [1982Ågr, 1992And]. Only the key equations related to the properties of mobility, tracer diffusivity and chemical diffusivity are given in the below sections.

Figure 9.5: Kinetic Property calculation dialog

Figure 9.6: Calculated diffusivity in the Al-Ni system

9.3.1 Atomic Mobility

In order to simulate diffusivity related properties such as chemical diffusivities of components, atomic mobility data of species in phases are required and stored in the database. Mobility of species k is related to its activation energy (Q_k) by

$$M_k = M_k^\circ e^{-Q_k/RT}$$

$$(9.12)$$

where M_k° is a frequency factor, R is the gas constant and T the temperature in Kelvin. Define MQ_k as

$$MQ_k = RT \ln M_k = RT \ln M_k^\circ - Q_k$$

$$(9.13)$$

Then, atomic mobility can be calculated from MQ_k. MQ_k is a function of composition, temperature and pressure and can be expressed as Redlich-Kister type of polynomial expansion as for the excess Gibbs energy [1982Ågr]. Each polynomial coefficient is stored in database. For example, the coefficient for term contributed to Al from `Ni` in `Fcc` phase ($MQ_{Al}^{Ni,\circ}$) is described in `TDB` file as follows,

```
Parameter MQ(Fcc&Al,Ni;0) 298.15 -285517+R*T*Ln(0.0007933); 6000 N !
```

In Pandat™, atomic mobility of species can be obtained through table operation with field of "M(*@*)". For example, `M(Al@Fcc)` represents the atomic mobility of `Al` in `Fcc` phase.

9.3.2 Tracer Diffusivity

Tracer diffusivity of a species *k* is directly related to its atomic mobility by

$$D_k^* = RTM_k$$

(9.14)

where *R* is the gas constant and *T* the temperature. Tracer diffusivity can be obtained in Pandat™ from table with field of `DT(*@*)`. For example, the tracer diffusivity of `Al` in `Fcc` can be extracted from calculation result with the table field of `DT(Al@Fcc)`. The natural logarithm of the tracer diffusivity is available with `logDT(*@*)`.

9.3.3 Chemical Diffusivity

Chemical diffusivity of species *k*, D_{kj}^n, could be calculated by

$$D_{kj}^n = D_{kj} - D_{kn} \text{ (when j is substitutional)}$$

(9.15)

$$D_{kj}^n = D_{kj} \text{ (when j is interstitial)}$$

(9.16)

And

$$D_{kj} = \sum_{i \in S} (\delta_{ik} - u_k) u_i M_i \frac{\partial \mu_i}{\partial u_j} + \sum_{i \notin S} \delta_{ik} u_i y_{Va} M_i \frac{\partial \mu_i}{\partial u_j}$$

(9.17)

where *S* represents the set of the substitutional species, δ_{ik} is the Kronecker delta, and μ_j is the chemical potential of species *i*. u_k is defined as

$$u_k = \frac{x_k}{\sum_{i \in S} x_j}$$

(9.18)

Chemical diffusivity of species *k* in phase *p* can be obtained in Pandat™ with the table field of `DC(k,j@p:n)`, where `j` and `n` are the gradient species and the reference species, respectively. Its corresponding natural logarithm is `logDC(k,j@p:n)`.

9.4 User-Defined Properties

Pandat™ allows user to define any property of a phase or a system following some simple syntax rules. The `User-Defined Property` database can be added to the original database, either `TDB` or `PDB` files, through `Append database` function. (Please refer to Section 3.3.9 for the detailed description on `Append database` function). user can also add the `User-Defined Property` parameters into their home-developed `TDB` files to develop a combined database for `User-Defined Property` calculations.

Three methods are implemented in Pandat™ to add User- Defined- Properties, depending on the nature of the properties. Key words `Phase_ Property`; `System_ Property` and `Property` are used in the syntax respectively.

`Phase Property` is used to define a property of a phase with a similar expression which describes the Gibbs energy of a disordered solution phase. Let *U* be the user-defined phase property and it is expressed as:

$$U = \sum_{i=1}^{c} x_i U_i^o + \sum_{i=1}^{c-1} \sum_{j=i+1}^{c} x_i x_j \sum_k (x_i - x_j)^k L_{ij}^k \qquad (9.19)$$

where x_i is the molar fraction of component *i* and U_i^o is the property of the pure component *i*, L_{ij}^k is the *k*-th order interaction parameter between components i and j. The syntax used in the `TDB` file are:

```
Type_Definition z PHASE_PROPERTY U 1 !
Type_Definition v GES AMEND_PHASE_DESCRIPTION * VARIABLE_X U !
```

In this definition, "v" is the identifier, "*" means any phase, and `VARIABLE_X` is the key word indicating `X` as the variable. The meaning of this definition is that any phase with the identifier "v" will use mole fractions (*x*) as the variables for property *U*.

`System Property` is used to define a property of a system with more than one phase. For a property in a system with multi-phase mixture, the property of the system is the weighted average of that of each phase. By default, the arithmetic mean is applied to a user defined Phase Property in a multi-phase system. For example, the user defines `Phase Property` *U* in a system with α and β two-phase mixture. The property of *U* of this system is calculated by

$$U = f_\alpha U_\alpha + f_\beta U_\beta \tag{9.20}$$

If the simple arithmetic mean does not apply, more complicated expression can be defined by User through Command `System Property`. For example, the value of the `system property` can be calculated through the function:

$$U_{\alpha+\beta} = f_\alpha U_\alpha + f_\beta U_\beta - f_\alpha f_\beta \sum_{i=0} M_i (f_\alpha - f_\beta)^i \tag{9.21}$$

where $U_{\alpha+\beta}$ is the user defined property of U in $\alpha+\beta$ two phase region. U_α and U_β is the property U in α and β phase respectively; f_α and f_β are the phase fraction of α and β phase respectively. M_i are the i-th order of the additional parameters which are used to describe additional effects on the user defined property U.

The syntax for the `System Property` used in the `TDB` file is:

```
System_Property Sys_U 1 !
Parameter L(Sys_U, Alpha, Beta;0) 298.15 M0; 3000 N !
Parameter L(Sys_U, Alpha, Beta;1) 298.15 M1; 3000 N !
```

Using command **Property**, user can also define special properties associated with phases in the original database. Any phase property available from Pandat Table can be used for user-defined property, such as `G`, `H`, `mu`, and `ThF`. However, the star symbol in a property, like `mu(*)`, cannot be used.

The syntax for the Property used in the `TDB` file is:

```
Property GFcc_GLiq 298.15 G(@Fcc)-G(@Liquid); 6000 N !
```

Below are several common User-Defined Properties. The related TDB files are included in the folder: `"...\Pandat 2024\Pandat 2024 Examples\PanPhaseDiagram\User_Defined_Property"`.

9.4.1 User-Defined Molar Volume Database

The molar volume (`Vm`) is one of the pre-defined properties in Pandat™. The molar volume of a pure component is described as a function of temperature and pressure, and the excess molar volume of a phase is usually described in the format similar to that used to describe the excess Gibbs energy of the phase. Note that, if a phase is described by a compound-energy-formalism (`CEF`) with multi-sublattices, the molar volume is automatically described as a function of `T`, `P`, and site fraction (`y`) following the format of the Gibbs

energy. This will introduce many end members in the molar volume and may lead to incontinuity in the calculated molar volume. In reality, molar volume should be a function of mole fraction (x) instead of site fraction (y), it is thus recommended to define the molar volume of a phase separately from its Gibbs energy if the phase is described by the `CEF` model. This can be realized through the user-defined molar volume property.

In brief, the molar volume of element *i* with crystal structure ϕ can be expressed as:

$$V_i^\phi(T) = V_0 \exp\left(\int_{T_0}^{T} 3\alpha \, dT\right) \tag{9.22}$$

where V_0 is the molar volume under atmospheric pressure at the reference temperature T_0. And α is the coefficient of linear thermal expansion (`CLE`). The volume of a phase with crystal structure ϕ can then be obtained via the Redlich-Kister polynomial:

$$V^\phi(T) = \sum_i x_i V_i^\phi + V_m^{ex} \tag{9.23}$$

where x_i is the mole fraction of element *i* and V_m^{ex} is the excess molar volume. Using ternary system as an example, the V_m^{ex} can be expressed by:

$$V_m^{ex} = \sum_{i=1}^{c-1} \sum_{j=i+1}^{c} x_i x_j \sum_{k} (x_i - x_j)^k L_{ij}^k + x_i x_j x_p L_{i,j,p} \tag{9.24}$$

The terms $L_{i,j}^k$ and $L_{i,j,k}$ are the interaction parameters from the constituent binary and the ternary systems, respectively. The standard unit of molar volume is m^3/mol-atom.

The molar volume database also enables us to calculate the density based on the relationship of $\rho = \frac{M}{V_m}$ (ρ is the mass density, *M* is the molar mass, and V_m is the molar volume).

Here we use the `Al-Ni` binary system as an example to demonstrate the calculation of molar volume through **User-Defined Property**. Please refer to the `AlNi_Vm.tdb` for details. In the beginning of the database file, a `Type_Definition` is given as below:

```
Type_Definition v GES AMEND_PHASE_DESCRIPTION * VARIABLE_X Vm !
```

In this definition, "v" is the identifier, "*" means any phase, and VARIABLE_X is the key word indicating X as the variable. The meaning of this definition is that any phase with the identifier "v" will use mole fractions (x) as the variables for Vm. Here are examples to define the molar volume parameters for the Fcc and L12_FCC phases in the Al-Ni system:

```
Phase Fcc %(v 1 1 !
    Constituent Fcc: Al,Ni:!
            Parameter Vm(Fcc,Al;0) 298.15 +V_Al_fcc; 3000 N !
            Parameter Vm(Fcc,Ni;0) 298.15 +V_Ni_fcc; 3000 N !
            Parameter Vm(Fcc,Al,Ni;0) 298.15 -2.85e-6; 3000 N !
Phase L12_FCC %v 2 0.75 0.25 !
    Constituent L12_FCC: Al,Ni:Al,Ni:!
            Parameter Vm(L12_FCC,Al;0) 298.15 +V_Al_fcc; 3000 N !
            Parameter Vm(L12_FCC,Ni;0) 298.15 +V_Ni_fcc; 3000 N !
            Parameter Vm(L12_FCC,Al,Ni;0) 298.15 -3.2e-6; 3000 N !
```

Even though the L12_FCC phase is modeled by CEF model with two sublattices, four end members, and 13 interaction parameters, its molar volume property can be described as a function of x_i like a solution phase through User-Defined Property. In other words, the molar volume of L12_Fcc is:

$$V_m^{L12} = x_{Al} \cdot V_{Al_{fcc}} + x_{Ni} \cdot V_{Ni_{fcc}} - 3.2 * 10^{-6} x_{Al} x_{Ni} \qquad (9.25)$$

Figure 9.7 shows the calculated Al-Ni binary phase diagram with molar volume contour lines. Please refer to AlNi_Vm.pbfx and Section 3.3.5 for details on the calculation of contour diagrams.

Figure 9.7: Al-Ni binary phase diagram with the calculated contour lines of molar volume (cm³/mol)

The lattice constant, or lattice parameter, refers to the physical dimension of unit cells in a crystal lattice. For `FCC` crystal structure, the lattice constant can be calculated by:

$$a = \left(\frac{4}{6.02\times10^{23}}V_m\right)^{\frac{1}{3}} \tag{9.26}$$

An example is given for the calculation of lattice constant in Ni-Al binary at Ni-rich corner. The calculated lattice constant agrees with the experimental data well as shown in Figure 9.8. Please refer to `AlNi_lattice.pbfx` for details. It should point out that the unit of `Vm` in the above equation is `cm`³`/mol`.

Figure 9.8: Comparison of calculated and measured lattice constants of the γ and γ'
phases in Ni-Al binary alloys at room temperature

9.4.2 Thermal Resistivity and Thermal Conductivity

`Thermal conductivity` of a pure element or a stoichiometric phase at temperature
above 273 K is described as a function of temperature using the following equation:

$$\kappa = A + B\,T + C\,T^{-1} + D\,T^2 \tag{9.27}$$

where κ is the thermal conductivity and T is the temperature in Kelvin. This function can
reasonably fit most of the experimental thermal conductivity data of elements at
temperature above 273 K.

The `thermal conductivity` of a solid solution phase can be calculated from
`thermal resistivity`, which is the reciprocal of thermal conductivity. According to the
Nordheim rule, the thermal resistivity (ρ) of a solid solution phase can be described by the
following Redich–Kister polynomials:

$$\rho^{\alpha}_{AB} = x_A \rho_A + x_B \rho_B + x_A x_B \sum_{i=0} L_i (x_A - x_B)^i \tag{9.28}$$

where ρ^α_{AB} is the thermal resistivity of the α solution phase in the $\mathtt{A-B}$ system. x_j and ρ_j are the mole fraction and thermal resistivity of pure elements j, respectively. L_i are the ith order interaction parameters which are used to describe the effect of solute elements on the thermal resistivity. In general, the interaction parameter can be expressed as:

$$L_i = a_i + b_i T + c_i T^{-1}$$

$$(9.29)$$

where the parameters a_i, b_i and c_i are evaluated based on the experimental data.

The interface scattering parameters is introduced to describe the effect of the second phase on the thermal resistivity in two-phase region. The thermal resistivity of the two-phase region is described as following:

$$\rho_{\alpha+\beta} = f_\alpha \rho_\alpha + f_\beta \rho_\beta - f_\alpha f_\beta \sum_{i=0} M_i (f_\alpha - f_\beta)^i$$

$$(9.30)$$

where $\rho_{\alpha+\beta}$ is the thermal resistivity of the alloys in $\alpha+\beta$ two-phase region, f_p and ρ_p ($p = \alpha$, β) are the mole fraction and thermal resistivity of the phase p, respectively. M_i, which can be considered to be linearly temperature dependent, is the ith interface scattering parameter and and can be evaluated from the experimental data. The thermal conductivity of alloy system and the value of each phase can be obtained by using the reciprocal of the thermal resistivity values through output option.

In this example, thermal resistivity of the $\mathtt{Al-Mg}$ binary alloys is described using the $\mathtt{User\text{-}Defined\ Property}$ function.

As shown in the $\mathtt{AlMg\_ThCond.tdb}$, the thermal resistivity **ThRss** property is first defined as **User-Defined Property** since it has not been pre-defined in the current Pandat™software.

    Type_Definition z PHASE_PROPERTY ThRss 1 !

In accordance, the following definition is also needed to add this property to the original database.

    Type_Definition e GES AMEND_PHASE_DESCRIPTION * VARIABLE_X ThRss !

As is seen, the thermal resistivity of the $\mathtt{Fcc}$ phase or the $\mathtt{Hcp}$ phase follows the same format as that of Gibbs energy for a disordered solution phase.

    Parameter ThRss(Liquid,Al;0) 298.15 1/ThCond_Al_Liq; 3000 N !
    Parameter ThRss(Liquid,Mg;0) 298.15 1/ThCond_Mg_Liq; 3000 N !

```
Parameter ThRss(Fcc,Al;0) 298.15 1/ThCond_Al_Fcc; 3000 N !
Parameter ThRss(Fcc,Mg;0) 298.15 1/ThCond_Mg_Hcp; 3000 N !
Parameter ThRss(Fcc,Al,Mg;0) 298.15 0.02566-1.3333e-05*T+14.5*T^(-1);
3000 N !
Parameter ThRss(Hcp,Al;0) 298.15 1/ThCond_Al_Fcc; 3000 N !
Parameter ThRss(Hcp,Mg;0) 298.15 1/ThCond_Mg_Hcp; 3000 N !
Parameter ThRss(Hcp,Al,Mg;0) 298.15 0.0214-1.3669e-5*T+12.7158*T^(-1);
3000 N !
Parameter ThRss(Hcp,Al,Mg;1) 298.15 0; 3000 N !
Parameter ThRss(Hcp,Al,Mg;2) 298.15 0.14825-7.7706e-05*T+25.3031*T^(-
1); 3000 N !
```

Thermal resistivity of the intermetallic phases with narrow solid solubility rage in the phase diagrams is treated like that of a stoichiometric compound phase, i.e., it is composition independent and is described as below:

```
Parameter ThRss(AlMg_Beta,*;0) 298.15 1/42; 6000 N !
Parameter ThRss(AlMg_Eps,*;0) 298.15 1/42; 6000 N !
Parameter ThRss(AlMg_Gamma,*;0) 298.15 -0.03267+2.7412e-05*T+20.722*T^
(-1); 6000 N !
```

In order to describe the thermal resistivity within two-phase region, a **system property**, `Sys_ThRss`, is then defined by the command:

```
System_Property Sys_ThRss 1 !
Parameter L(Sys_ThRss, Fcc, AlMg_Beta;0) 298.15 0.005; 3000 N !
Parameter L(Sys_ThRss, Hcp, AlMg_Gamma;0) 298.15 0; 3000 N !
Parameter L(Sys_ThRss, Hcp, AlMg_Gamma;1) 298.15 0.01; 3000 N !
```

After the thermal resistivity has been properly modeled for each phase, the thermal conductivity of each phase and that of the system can be directly calculated and outputed by using extra output in Pandat™ defined as `1/ThRss(@*)` and `1/Sys_ThRss`, respectively. The comparisons between the calculated and measured thermal conductivities of the `Al-Mg` alloys are shown in Figure 9.9 and Figure 9.10. This example demonstrates that the **User-Defined Property** function is very powerful and flexible to allow users define various types of properties. The property can be a function of any phase properties that can be calculated by PanPhaseDiagram module.

(a)   (b)

Figure 9.9: Comparison between the calculated and measured thermal conductivities in

(a) the (Al) solid solution and (b) the (Mg) solid solution, in the Al-Mg binary system

Figure 9.10: Calculated thermal conductivities of Mg-Al system at 298 K

9.4.3 Calculation of T_0 Curve Using User-defined Property

T_0 curve is the trace of a series of points in a two-phase field at which the Gibbs energy of the two phases are identical. In this example, the original database is `ABC.tdb`, and the T_0-curve property is defined in the Appended database `ABC_T0.tdb`. In the `ABC_T0.tdb` file, the T_0-curves of `Bcc/Liquid` and `Fcc/Liquid` phases are defined as:

```
Property GFcc_GLiq 298.15 G(@Fcc)-G(@Liquid); 6000 N !
Property GBcc_GLiq 298.15 G(@Bcc)-G(@Liquid); 6000 N !
```

Where "**Property**" is the key word for user-defined property and `G(@Bcc)`, `G(@Fcc)`, `G (Liquid)` are the Gibbs free energies of the `Bcc`, `Fcc`, and `Liquid` phases, respectively. Note that, the `Bcc`, `Fcc` and `Liquid` are defined in the `ABC.tdb`. In this particular case, the above "**Property**" can be directly defined in the `ABC.tdb`, i.e., the `ABC.tdb` and `ABC_T0.tdb` can be combined into one database. This example demonstrates that user-defined property can be separated from the original database. This design makes it possible to obtain user-defined properties even the original database is in encrypted `pdb` format.

In this example, the `GBcc_GLiq` property is defined as the Gibbs free energy difference between the `Bcc` phase and the `Liquid` phase, and `GFcc_GLiq` property is defined as the Gibbs free energy difference between the Fcc phase and the Liquid phase.

Using the **Contour function** (Section 3.3.5), we can calculate contour maps of these user-defined properties and plot them on the calculated phase diagram. When we set the calculation condition as `GBcc_GLiq=0`, the T_0 curve for `Bcc/Liquid` is obtained as is shown by the green line in Figure 9.11. Similarly, by setting `GFcc_GLiq=0`, we obtain the T_0 curve for `Fcc/Liquid` as is shown by the red line in Figure 9.11.

Figure 9.11: Calculated isopleth of A-B-10at.%C with T_0 curves of Bcc/Liquid and Fcc/Liquid phases

User can run the `ABC_T0.pbfx` batch file to obtain Figure 9.11. Two points should be addressed for this batch file:

- Both the "`start`" and "stop" values should set to be zero for the contour mapping as following to get the T_0 line.

    ```
    <contour  name="Contour_ T0_ Fcc_ Liq"  property="GFcc_ GLiq"  start
    ="0" stop ="0" step="1"/>
    <contour  name="Contour_ T0_ BCC_ Liq"  property="GBcc_ GLiq"  start
    ="0" stop ="0" step="1"/>
    ```

- To obtain the T_0 curve, each phase needs to be considered individually, thus the equilibrium type is set to be "`individual`"

    ```
    <individual_phase value="true"/>
    <equilibrium_type type="individual"/>
    ```

Note: The T_0 function has already included in Pandat™ software as a direct output property since Pandat 2021 version.

9.4.4 Calculation of Spinodal Curve Using User-defined Property

Attention: As of Pandat 2023 and subsequent versions, a built-in function for the Hessian Matrix of Gibbs energy has been introduced. Users are no longer required to utilize the previously defined user property function. Detailed information on this feature can be found in the .

A Spinodal curve is where the determinant of the Hessian of Gibbs free energy with respect to composition is zero. For a phase with *c*-components, above condition is expressed as

$$
\begin{vmatrix}
\dfrac{\partial^2 G}{\partial x_1^2} & \dfrac{\partial^2 G}{\partial x_1 \partial x_2} & \dfrac{\partial^2 G}{\partial x_1 \partial x_3} & \cdots & \dfrac{\partial^2 G}{\partial x_1 \partial x_{n\text{-}1}} \\[2ex]
\dfrac{\partial^2 G}{\partial x_2 \partial x_1} & \dfrac{\partial^2 G}{\partial x_2^2} & \dfrac{\partial^2 G}{\partial x_2 \partial x_3} & \cdots & \dfrac{\partial^2 G}{\partial x_2 \partial x_{n\text{-}1}} \\[2ex]
\dfrac{\partial^2 G}{\partial x_3 \partial x_1} & \dfrac{\partial^2 G}{\partial x_3 \partial x_2} & \dfrac{\partial^2 G}{\partial x_3^3} & \cdots & \dfrac{\partial^2 G}{\partial x_3 \partial x_{n\text{-}1}} \\[2ex]
\vdots & \vdots & \vdots & \ddots & \vdots \\[2ex]
\dfrac{\partial^2 G}{\partial x_{n\text{-}1} \partial x_1} & \dfrac{\partial^2 G}{\partial x_{n\text{-}1} \partial x_2} & \dfrac{\partial^2 G}{\partial x_{n\text{-}1} \partial x_3} & \cdots & \dfrac{\partial^2 G}{\partial x_{n\text{-}1}^2}
\end{vmatrix}
\tag{9.31}
$$

where the molar fraction of component *c* is chosen as the dependent variable. The second derivative of *G* w.r.t. molar fractions can be calculated from the thermodynamic factors:

$$
\frac{\partial^2 G}{\partial x_j \partial x_k} =
$$

$$
\mathrm{ThF}(x_j, x_k) - \mathrm{ThF}(x_c, x_k) - \mathrm{ThF}(x_j, x_c) + \mathrm{ThF}(x_c, x_c) \quad (j, k = 1, 2, \cdots, c\text{-}1)
$$

$$
\tag{9.32}
$$

Example #1: Spinodal curve of Fcc phase in the Al-Zn binary system

In this example, the spinodal curve of the Fcc phase in the Al-Zn binary system is calculated through user-defined property. A user-defined property `d2GdxZn2` for the `Fcc` phase is defined in `AlZn_Spinodal.tdb` as:

```
   Property    d2GdxZn2    298.15    ThF (Zn,Zn@Fcc)- ThF (Al,Zn@Fcc)- ThF
(Zn,Al@Fcc)+ThF(Al,Al@Fcc); 6000 N !
```

where `ThF (Zn,Zn@Fcc)`, `ThF (Al,Zn@Fcc)`, `ThF (Zn,Al@Fcc)`, and `ThF (Al,Al@Fcc)` are the thermodynamic factors of `Fcc` phase. Since the value of `d2GdxZn2` is usually a large number, we define the `Hessian  function, HSN`, as `d2GdxZn2` multiplied by a factor of 10^{-4}:

```
   Property HSN 298.15 1e-4*d2 GdxZn2; 6000 N !
```

As shown in the `AlZn_Spinodal.pbfx`, the `AlZn_Spinodal.tdb` is appended to the `AlMgZn.tdb`. The spinodal line is calculated through contour mapping by using following conditions:

```
<contour name="Spinodal" property="HSN" start="0" stop="0" step="1"/>
<equilibrium_type type="individual"/>
```

The calculated `Fcc` spinodal is shown in Figure 9.12 with the stable Al-Zn binary phase diagram.

Figure 9.12: Calculated spinodal curve of the Fcc phase within the Al-Zn binary system

Example #2: Spinodal curve of Fcc phase in the X-Y-Z ternary system

In this example, the original database is `XYZ.tdb`, and the user-defined `HSN` property is defined in `XYZ_Spinodal.tdb` as described below:

```
Property  d2Gdx2  298.15  ThF(Y,Y@Fcc)- ThF(X,Y@Fcc)- ThF(Y,X@Fcc)+ThF
(X,X@Fcc); 6000 N !
Property  d2Gdy2  298.15  ThF(Z,Z@Fcc)- ThF(X,Z@Fcc)- ThF(Z,X@Fcc)+ThF
(X,X@Fcc); 6000 N !
Property  d2Gdxy  298.15  ThF(Y,Z@Fcc)- ThF(X,Z@Fcc)- ThF(Y,X@Fcc)+ThF
(X,X@Fcc); 6000 N !
Property HSN 298.15 1e-10*(d2Gdx2 * d2Gdy2 - d2Gdxy * d2Gdxy); 6000 N
!
```

Note that the `HSN` property within the `XYZ` ternary system is derived and described as a function of the thermodynamic factor `ThF`. A factor of 10^{-10} is used to scale the `HSN` property since the numerical value of `HSN` is very big.

Again the spinodal lines are calculated through contour mapping. Details can be found in `XYZ_Isotherm_Spinodal.pbfx` and `XYZ_Isopleth_Spinodal.pbfx`. Figure 9.13 and Figure 9.14 show the calculated spinodal curves superimposed on the stable phase diagrams.

(a)   (b)

Figure 9.13: Calculated isothermal sections of the X-Y-Z system with spinodal curves at (a) 800 K and (b) 600 K.

(a)   (b)

Figure 9.14: Calculated isopleths within the X-Y-Z system with spinodal curves for (a) 10 at.%Zn and (b) 20 at.%Zn

10　Batch Calculation and High-Throughput Calculation

Pandat™ software has a friendly graphic user interface (GUI) which allows users to setup calculation conditions easily through Pandat GUI. Moreover, Pandat™ also offers a more convenient and flexible way through which users can perform batch calculations and high-throughput calculations.

10.1 Batch Calculation

10.1.1 Batch File Structure

In Pandat™, any calculation condition defined by user is automatically saved in a batch file. A batch file, which has an extension name of .pbfx, is written in the language of XML (Extensible Markup Language). Users do not need to be familiar with the format of a batch file, as Pandat automatically generates it after a calculation. However, user can easily modify it as wishes based on the generated file as a template. The purpose here is to explain the format and contents of a batch file.

A batch file contains several sections, the "Header" section, which starts by `<Header copyright="CompuTherm, LLC">` and ends by `</Header>`, provides basic information about the Pandat™ version. The "Calculation" section, which starts with `<calculation name="Example#1" type="section">`, and ends with `</calculation>`, contains all the details about the calculation including the calculation conditions and output results setting. The keyword "type" defines the type of calculation.

Four subsections are included in the "Calculation" section. They are: **Databases**, **Units**, **System** and **output**.

10.1.1.1 Databases

This section defines the databases to be used in the calculation including the thermodynamic (or combined with mobility or property) database with name of `.tdb` or `.pdb`, and the kinetic parameter database, such as `.kdb` needed by PanEvolution module, `.sdb` by PanSolidification module, and `.pfdb` by PanPhaseField module. In the "file_name" field, if only database name is provided, the database and the batch file should locate in the same folder, otherwise the folder path where the database is located should be given. In the below example, the kdb file is located in the same folder as that of the batch file, while the tdb file is not.

```
<databases>
<database  type="tdb"  file_name="C:\..\Pandat  2024  Example  book\AlNi_
Prep.tdb" />
<database type="kdb" file_name="Ni-14Al_Precipitation.kdb" />
</databases>
```

10.1.1.2 Units

The section define the units used in the calculation and simulation. It is very important to check the units setting during the calculation and simulation. The values provided in system section are all corresponding to the units set here.

```
<units>
    <unit name="T" value="C" />
    <unit name="length" value="m" />
    <unit name="n" value="x%" />
    <unit name="time" value="hour" />
</units>
```

Note: If math operations are carried out in the output table, the output properties after math operations are in the SI units instead of the units set in the unit section . For example, the "time" unit is selected as "hour" here, but, the log10(time) value in output will be calculated using second as the unit of time.

10.1.1.3 System

Four subsections are included in the system section: **Components**; **Phases**; **Points**; and **Conditions**;

Components The components of the system are defined in this section.

```
<components>
<component name="Al" status="Selected" />
<component name="Mg" status="Selected" />
</components>
```

Phases The status of phases are defined in this section. The "*" means all phases in the system, the phase status can be set as "Entered", "Suspended" and "Dormant".

```
<phases>
   <phase name="*" status="Entered" />
   <phase name="Fcc" status="Suspended" />
   <phase name="Liquid" status="Dormant" />
</phases>
```

Points Points defines the state variables of a particular point, such temperature, pressure, and composition. The units of these state points are defined in the Units section. For point calculation, and some other calculations, such as solidification, only one point needs to be defined here. Two points at the two ends are needed for a line calculation, and three points, corresponding to the three corners of the section, are required for a 2D section calculation.

```
<points>
   <point>
    <statespace>
        <T value="300" />
        <P value="1" />
        <n component="Al" value="0.5" />
        <n component="Mg" value="0.5" />
    </statespace>
   </point>
</points>
```

Condition Some advanced features in Pandat™ can be defined in the condition section, such as contour lines in PanPhaseDiagram module. Heat treatment schedule (time and temperature) needed in PanDiffusion, PanEvolution and PanSolidification modules are also defined in this condition section.

```
<condition>
   <steps value="100" />
   <contour name="isotherm_lines" property="T" step="0" interval="50">
   <equilibrium_type value="" />
   </contour>
</condition>
```

10.1.1.4 Output

Two subsections are included in the Output section: **Tables** and **Graphs**.

Tables Here user may define the properties to be included in a table. More than one table can be defined, with each table defined separately. As the example below, the first table is imported from "`Al-Mg_Exp.dat`" and named as "`exp`". "`</table>`" means that it is the end of a table. The second table is the calculated "Default" table, including the calculated properties defined by the "column name". If the table name is with the extension name of "`.dat`", the .dat file will be output directly after the calculation. The `.dat` file will be saved in the same folder with that of the batch file if only the file name is provided, as shown by the third example table below. The `.dat` file can also save to a designated location if the file path is provided in the "Table name".

   The table "type" can be defined as "Default", "Tie-line", "Invariant", "Invariant_Tieline", "Contour" and "psd" in PanEvolution.

```
<tables>
   <table source="Al-Mg_Exp.dat" name="exp" />
   <table name="Default" source="Default" type="Default">
   <column name="T" />
   <column name="x(*)" />
   <column name="phase_name" />
   <column name="f(@*)" />
   <column name="G" />
   <column name="w(*)" />
```

```
    </table>
    <table name="Al-Mg_Calc.dat" source="Default" type="Default">
    <column name="T" />
    <column name="x(*)" />
    <column name="phase_name" />
    <column name="w(*)" />
    </table>
  <tables>
```

Graphs The graph settings are defined in Graphs section. The default graph type is Cartesian coordinates, but users can define the `type = "triangle"`to ploy Gibbs triangle diagrams, for example ternary isothermal sections: `<graph name="AlMgZn_700K" type="triangle">` . The default plot type is line, users can also define it as "`point`", "`line`", and "`linepoint`" by setting as `plot type="point"`. Below are two typical examples for Graphs.

Example 1: The graph for Al-Mg binary phase diagram

```
  <graphs>
  <graph name="2D_Al-Mg">
     <plot table_name="Default" xaxis="x(Mg)" yaxis="T" />
     <plot type="point" table_name="exp" xaxis="x(Mg)" yaxis="T" />
     <plot table_name="tieline" xaxis="x(Mg)" yaxis="T" />
     <plot table_name="invariant_tieline" xaxis="x(Mg)" yaxis="T" />
  </graph>
  </graphs>
```

Example 2: Graphs after precipitation simulation showing the number density, average size and hardness.

```
  <graphs>
  <graph name="number density">
     <plot table_name="185C_KWN" xaxis="log10(t)" yaxis="log10(nd(*))"/>
     <plot type="point" table_name="nd_exp" xaxis="logt" yaxis="nd"/>
  </graph>
  <graph name="average size">
     <plot  table_ name="185C_ KWN"  xaxis="log10 (t)"  yaxis="log10 (s
(*)*1e10)"/>
     <plot   type="point"   table_ name="size_ exp"    xaxis="logt"
yaxis="size"/>
```
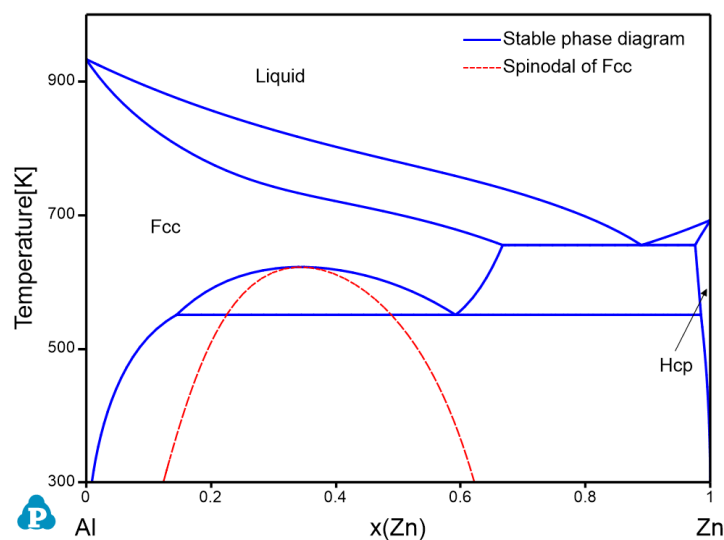
```
</graph>
<graph name="hardness">
   <plot table_name="185C_KWN" xaxis="log10(t)" yaxis="hv"/>
   <plot type="point" table_name="hv_exp" xaxis="logt" yaxis="hv"/>
</graph>
</graphs>
```

10.1.2 Create and Edit Batch File

A batch file can be easily created by saving the calculation conditions either before the calculation or after the calculation is completed. User can save the calculation conditions as a batch file before performing the calculation as shown in Figure 10.1, or after the calculation as shown in Figure 10.2. In fact, Pandat™ creates a batch file for each calculation automatically and save it in the folder where the workspace is saved.
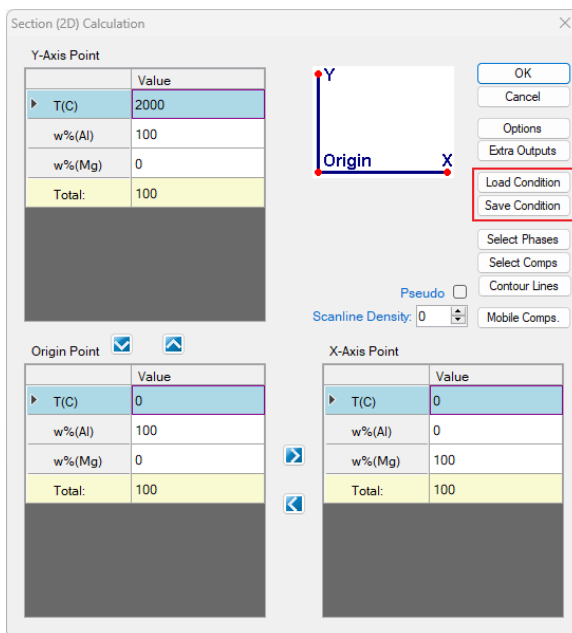
Figure 10.1: Save condition from Calculation condition setting interface
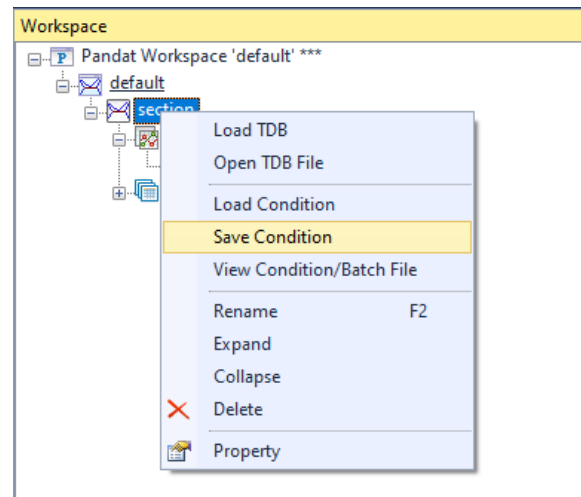
Figure 10.2: Save condition from calculated result

For more examples of batch files, please refer to Pandat example folder: "..\Program Files\CompuTherm LLC\Pandat 2024\Pandat 2024 Examples"

or Pandat Example book folder: "..\Program Files\CompuTherm LLC\Pandat 2024\Pandat 2024 Example book"

Since a batch file is a text file using xml language, it can be viewed and edited by any text editor. However, all keywords and value in the batch file are highlighted in Pandat™ text editor as shown in the Figure 10.3, which make it more efficient for the user to edit the batch file. User can open a batch file in Pandat™ text editor by using "open file", or simply drag and drop the batch file into Pandat™ main window. Again, it should point out that user does not need to remember the batch file formats, instead one may create a batch file of certain type of calculation through user interface, then modify the calculation condition through editing a batch file as needed.
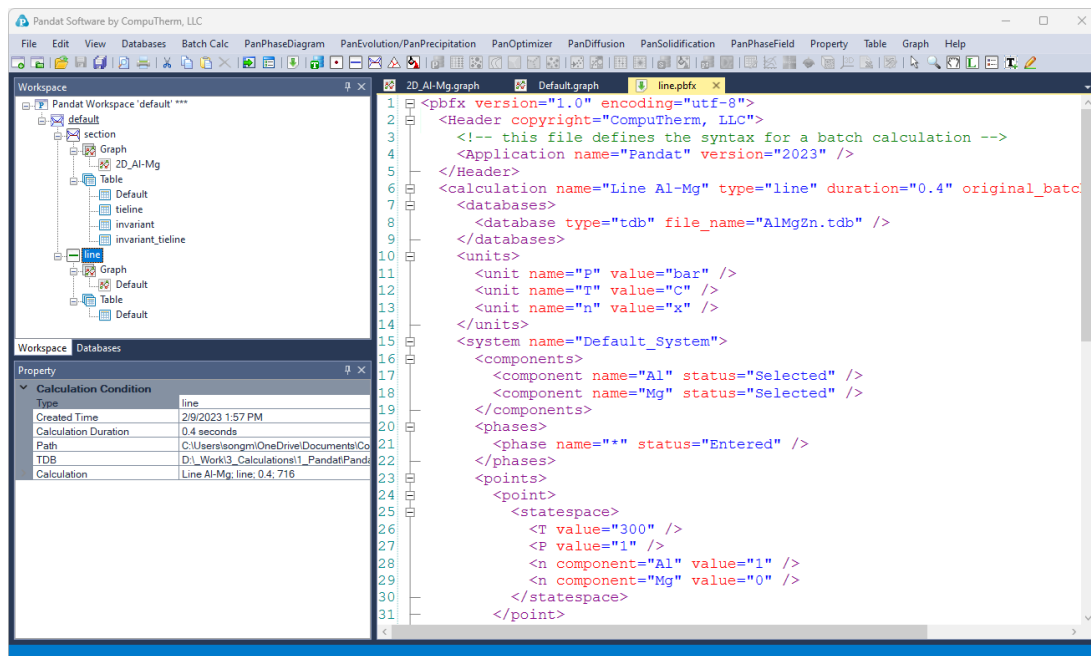
Figure 10.3: View and edit batch file in the Pandat™ text editor

10.1.3 Run Calculation from Batch File

After the required batch files are created, the user can run the batch files from the menu **Batch Calc**. Below are several functions of Batch calculations.

10.1.3.1 Run one calculation in a batch file

User can run a calculation promptly by using a batch file directly without going through the user interface. This can be accomplished by clicking the icon ⬇ on the tool bar or through the menu "**Batch Calc → Batch Run**" as shown in Figure 10.4.

Reminder: Before you start Batch run, please make sure the active project in Pandat™ workspace is the same module as the calculation type in the batch file. In other words, if the calculation type in the batch file needs to use PanPhaseDiagram module, then PanPhaseDiagram module must be the activated module in the Pandat™ workspace/project. Figure 10.5 shows the warning message when one tries to run a batch calculation of a phase diagram when PanDiffusion module is activated.
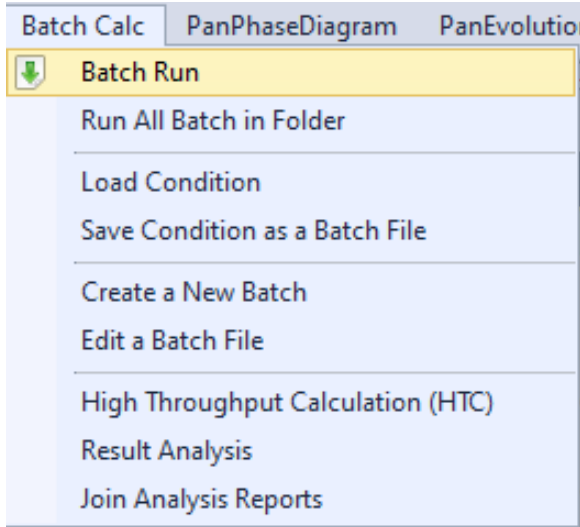
Figure 10.4: The `Batch run` function under the menu `Batch Calc`
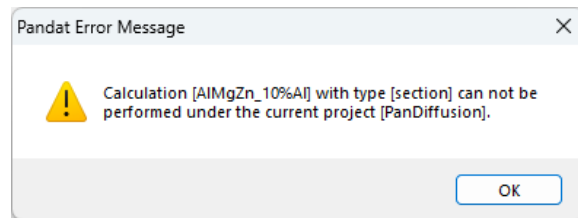
Figure 10.5: Warning message when run batch file under a wrong Pandat™ module

10.1.3.2 Run a group of calculations using batch files

In addition to the "`Batch run`" function that allows user to quickly run a calculation using a batch file, the "`Run All batch in folder`" function under the `Batch Calc` menu allows user to do a batch of calculations using a group of batch files in the same folder. Moreover, if the "`Include Sub-Folders`" function is selected, all batch files in the subfolders will also be included in the batch calculation.

　　This function is very useful if you want to do a lot of similar calculations with different composition or temperature. You may first prepare a group of batch files using a template batch file and then use "`Run All batch Files in Folder`" function to do all the calculations. Using ShellScript, Python code or some other program languages, one can easily generate a lot of batch files by replacing the composition and temperature conditions from a template batch file.

Reminder: when using "**Run All Batch Files in Folder**", please be aware that the calculations in this group of batch files should require the same Pandat™ module. In other words, if in the same folder you have some batch files require PanPhaseDiagram module to do the calculations and some others require PanDiffusion module to do the calculations, you can not perform all these calculations together using "**Run All Batch Files in Folder**". Instead, you need to perform two separate "**Run All Batch Files in Folder**", one group using PanPhaseDiagram module and the other using PanDiffusion module. Figure 10.6 shows that user can select or deselect the batch file to be included in the calculation.
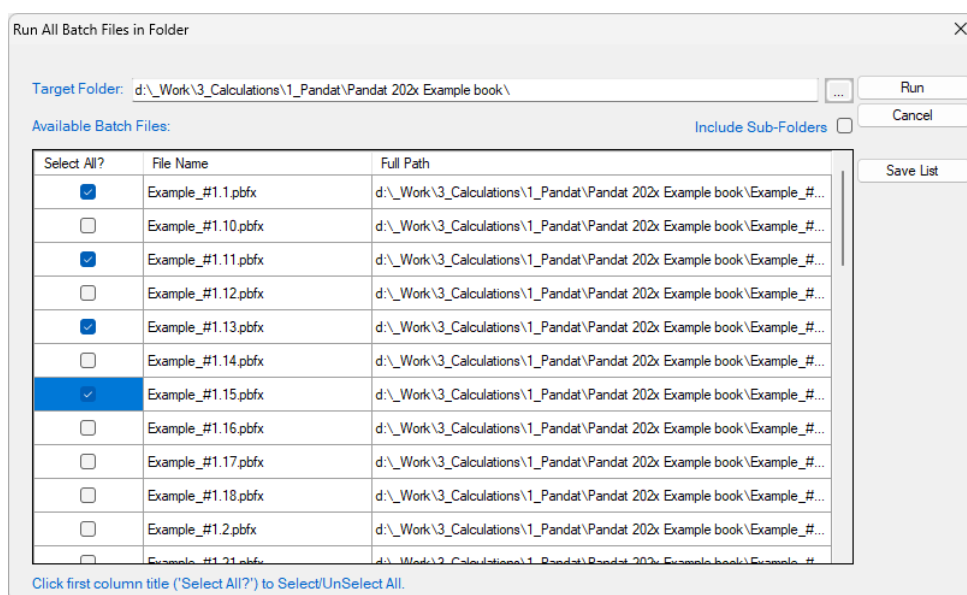
Figure 10.6: Pop-up window for Run All Batch Files in Folder

10.1.3.3 Save and run calculation list

In the above **Run All Batch File in Folder** interface as show in Figure 10.6, click the button "**Save List**", the selected calculations can be saved as a Batch File List file (.pbl), as shown in Figure 10.7. The pbl files can be run using "**Batch Run**" function. After click "**Batch Run**" in Pandat, select the file type as Batch File List (*.pbl) from the drop-down list as shown in Figure 10.8, Pandat™ will then run all the calculations listed in the .pbl file. The *.pbl file is also a simple xml format text file as shown in Figure 10.9. Users can also manually revise the *.pbl file by adding or deleting calculations.
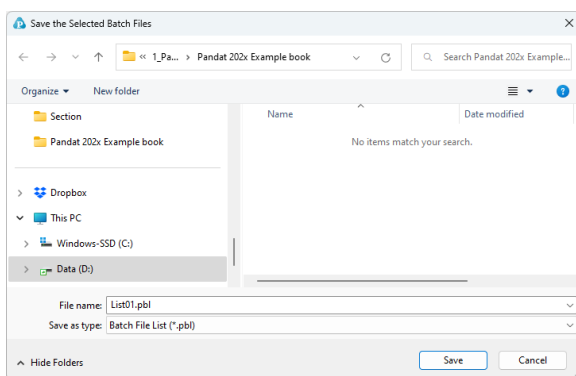
Figure 10.7: Save batch file list (.pbl) file
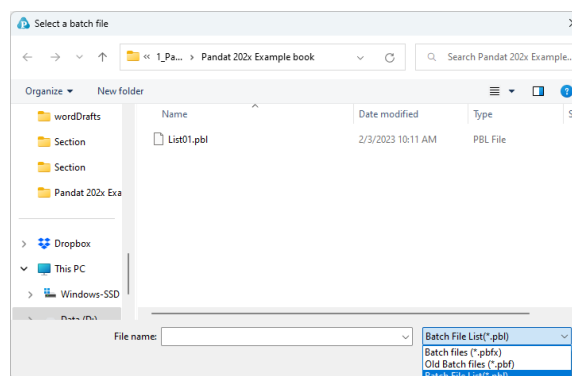
Figure 10.8: Run pbl file from Batch Run function

Figure 10.9: An example of pbl file

10.2 High-Throughput calculation

The High-Throughput-Calculation (HTC) function has been implemented in Pandat™ for the PanPhaseDiagram, PanEvolution, and PanSolidification modules. It can perform thousands of calculations in a user defined compositional space by a simple setting. Alloy compositions that satisfy user defined criteria can then be identified through mining the thousands of simulated results. This function allows user to develop alloys with certain properties through design. There are two methods to define alloy compositions for HTC: one is to setup the composition range of each component and its steps through user-interface; the other one is to load the user-defined composition data file. These two methods will be explained in detail in this section. Note that, the current HTC function can carry out 0D-point calculation and solidification simulation using both the Scheil and Lever-rule models in the PanPhaseDiagram module. In the PanEvolution module, HTC can be carried out for all defined/imported alloys under one or multiple heat-treatment conditions.

The HTC function is also available in the PanSolidification module for alloys under different cooling rates. The tutorial of the HTC function in different modules will be given in this section as well. Other types of HTC simulations will be continuously developed in the future.

10.2.1 Alloy Composition Setup

User can set up the alloy compositions in two ways: Setting Composition Range and Steps through PanGUI, or Import Alloy Compositions from a prepared composition data file.

10.2.1.1   Setting Composition Range and Steps

The composition setup dialog for the HTC function in different modules is the same. As shown in Figure 10.10, the composition range of each component can be defined by using the `Start` and `End` values. The #steps define the number of calculations between the `Start` value and the `End` value. In this example, the composition of Ti is set as the balance by right-click the row of Ti, and all the values will be shown as "`-1`" as shown in Figure 10.10. Similarly, the temperature can also be set at constant temperature with #steps is 0, or change temperature by setting proper #steps value.
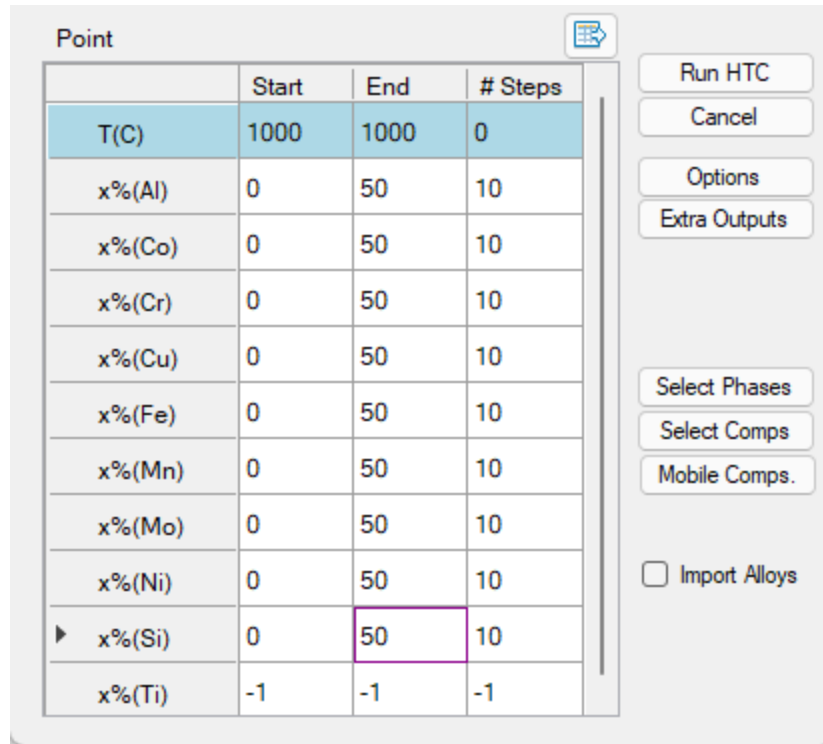
Figure 10.10: HTC composition setup dialog

The Export alloy composition function is introduced in Pandat™2022. After setting the composition range, then click ▣ button to export the composition conditions into a text file, which can be imported directly as described below. If the temperature is constant, the exported file only contains alloy composition, if the temperature #steps is not zero, then the exported file also contains temperature conditions. User can also prepare other alloy composition files based on the exported composition file as a template.

10.2.1.2 Import Alloy Compositions

Pandat™ also allows user to import alloy compositions for HTC through a data file. This is a more flexible and efficient way to perform HTC for a group of multi-component alloys which may contain different alloying elements. The format of a data file (.txt or .dat) is simple. Figure 10.11 shows an example which contains a group of Al-rich alloys. The first row shows all alloying elements involved in these alloys. The first column of the second row defines the unit of the alloy composition. Then alloy compositions are given from the third row forward. The last three columns are for temperature settings. There are a few points to emphasize here:

- The element in the first column is automatically treated as "balance" element, such as Al in this example. No matter what values are given in the first column, those values will be recalculated after the compositions of all other components are read in.

- The first row lists all the elements used in the alloys. The compositions of some of them can be zero in a certain alloy.

- If an alloying element is not available in the database used to perform HTC, this element will be kicked out in the calculation and its composition will be automatically added to that of the "balance" element.

- Pandat™ will read the temperature conditions from the imported file if they are included. If the imported file only contains the alloy compositions, the temperature can be set from GUI for all alloy composition.

Al	Cr	Cu	Fe	Mg	Mn	Pb	Si	Sr	Ti	Zn	T_start	T_End	Steps
w%										C			
91.8	0	5	0	0.5	0.3	0	0.3	0	0.3	0.1	500	1000	1
92.49	0.2	0	0	0.3	0	0	7	0	0	0	500	1000	1
90.149	0	0.05	0.1	0.25	0.1	0	9	0	0.2	0.15	500	1000	1
88.18	0	0.03	0.15	0.3	0.65	0	10.5	0.015	0.095	0.08	500	1000	1
87.499	0	0.1	1	0.35	0.55	0.001	10	0	0.2	0.15	500	1000	1
88.25	0	0.05	0.1	0.25	0.1	0	11	0	0.2	0.05	500	1000	1
86.92	0.03	0	0.03	0.01	0	0.02	13	0	0	0	500	1000	1
86	0	0.1	1	0	0.55	0	12	0	0.15	0.15	500	1000	1
89.5	0	0.1	0.65	0.1	0.5	0	9	0	0	0.15	500	1000	1

Figure 10.11: Example of composition file for multi-component Al-rich alloys

10.2.2 HTC with PanPhaseDiagram

In this example, we will demonstrate how to calculate liquidus, solidus, and solidification ranges for a number of `Al-Mg-Zn` alloys via the `HTC` function by setting composition range and steps.

1. Load proper database and choose the Al-Mg-Zn system.

2. Choose the `HTC` function from the **Batch Calc → High Throughput Calculation (HTC)** (as shown in Figure 10.12).
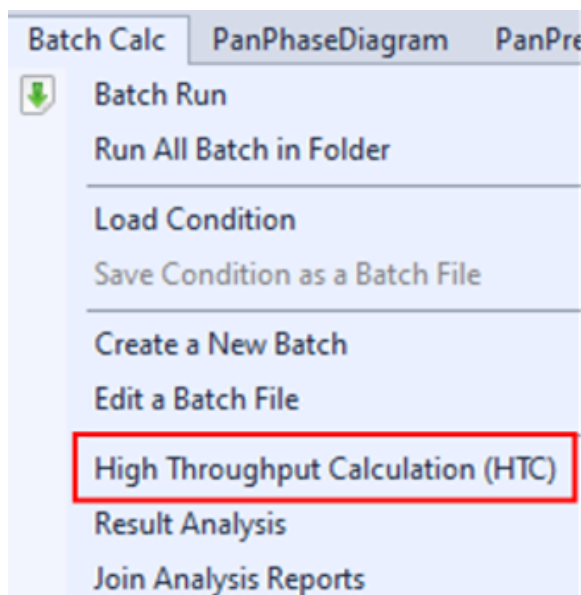
Figure 10.12: `HTC` function under the "`Batch Calc`" menu

3. Choose the calculation type from the drop-down list of `HTC` pop-up window and select "`Solidification`"(as shown in Figure 10.13)
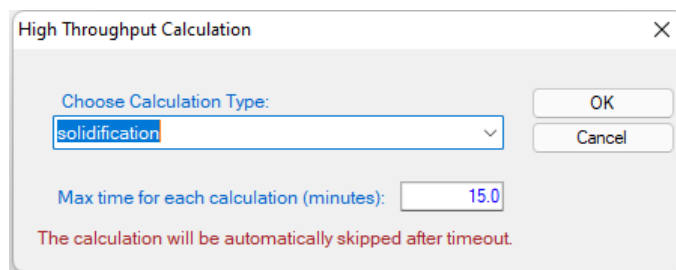
Figure 10.13: Dialog to choose calculation type of `HTC`

4. Define the compositional space for HTC simulation.

After step 3, a window pops out as shown in Figure 10.14. In this setting, the compositions of both `Mg` and `Zn` vary from 1 to 30 wt.% in a double composition loops. The "`Steps`" is set to 29, which means the composition increases by 1 wt% at each step. The total number of calculations is 30x30=900 in this setting. The composition of `Al` is set as balance by typing "-1" for steps or right-click the row of Al. No "`Start`" or "`End`" values are required for the balance component, which is `Al` in this case. After

setup the compositional space for HTC and choose the proper solidification model, user can click "**Run HTC**" button to perform HTC simulations, which is 900 calculations in this case.
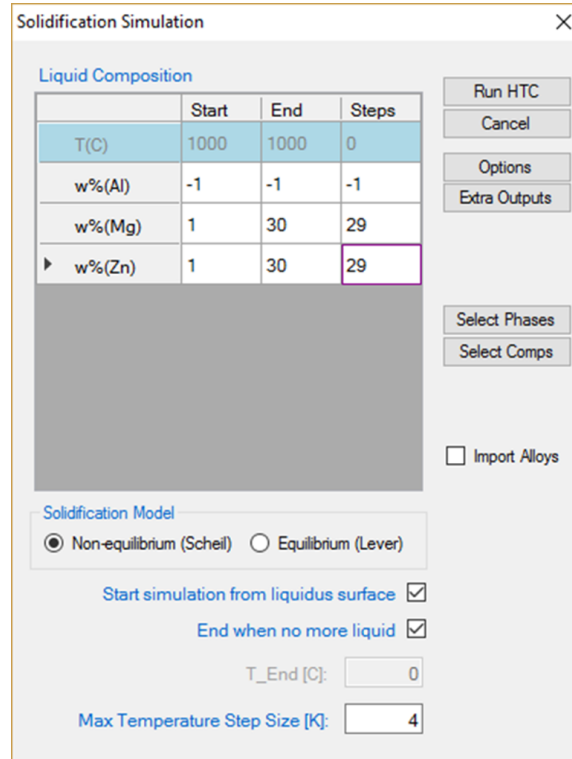
Figure 10.14: Dialog to setup compositional space for HTC

5. Save the current workspace after all calculations are finished. It is suggested that user saves the current workspace immediately after the HTC calculation. This allows all the calculated results saved in the workspace for future use.

6. Choose "**Result Analysis**" from the "**Batch Calc**" menu. User can use this command to analyze the calculated results for a group of alloys and choose a certain property from each calculation for comparison.
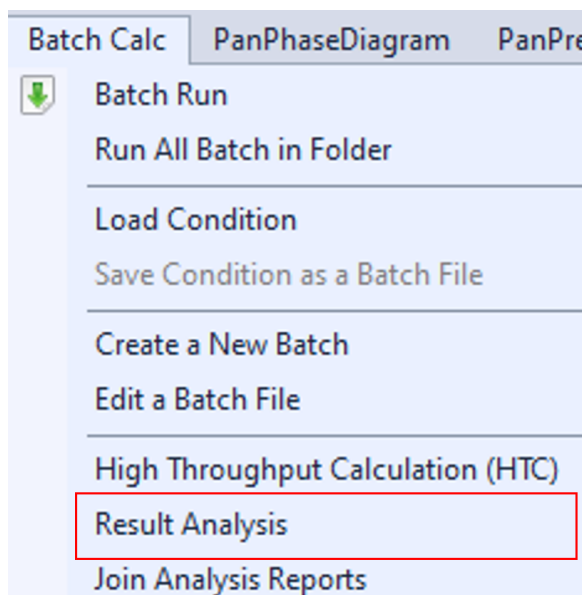
Figure 10.15: "`Result Analysis`" function under the "`Batch Calc`" menu

7. Open the workspace saved previously for "`Result Analysis`". User can perform several `HTC` calculations and save all the workspaces. User can then analyze the results of the selected `HTC` calculation by opening the corresponding workspace as shown in Figure 10.16.
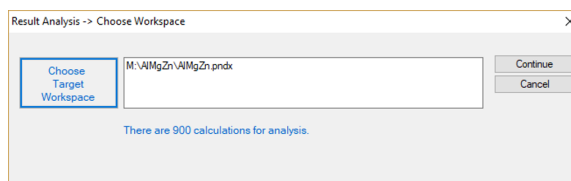
Figure 10.16: "`Result Analysis`" popup dialog to choose target workspace

8. Define the criteria of the properties as filters for result analysis.

In Figure 10.17, the "`Target Workspace`" shows the workspace selected by the user for results analysis. It should point out that there can be more than one table in each calculation, the "`Common Tables for All Calculations`" allows user to choose the table for analysis. In the "`Common Columns for the Target Table`" window, names for all the output properties available in the selected table are listed. User can choose the properties to be listed in the "`Analysis Report`". In this case. Since the purpose of HTC is to compare a special target property for the several hundred/thousands of calculations, the "`Set an Expression to Select Rows`"

at the bottom of the window allows user to define the criteria. This criterion is $f_L = 1$, i.e., the fraction of liquid is 1. With this filter, only the row satisfies this criterion will be listed in the "`Analysis Report`". It should point out that several criteria can be set in the "`Set an Expression to Select Rows`".
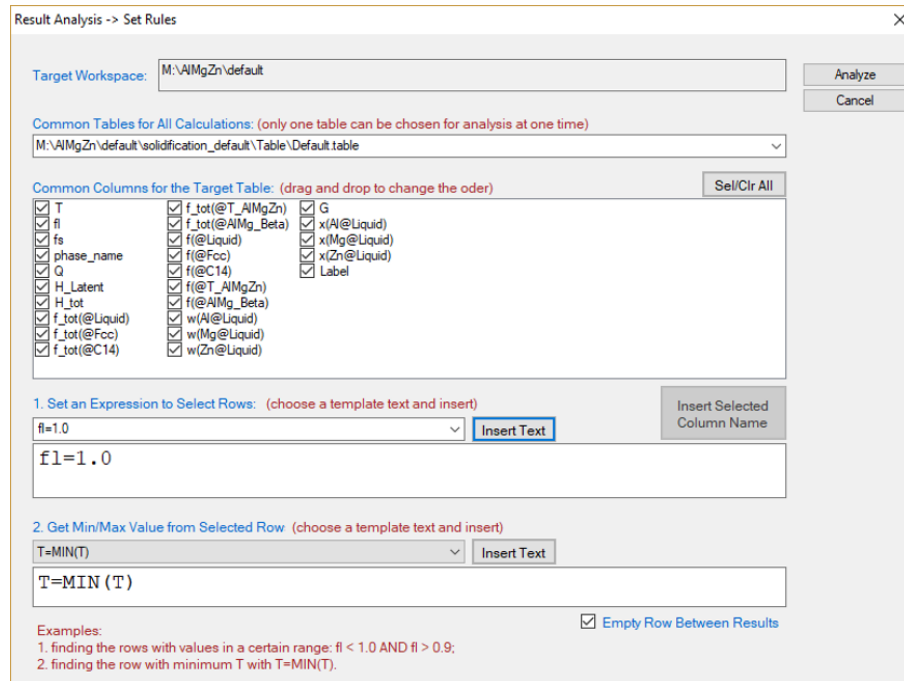
Figure 10.17: "`Result Analysis`" popup dialog to define the criteria of the properties

9. Present analyzed results on liquidus temperature as "`Analysis Report`" table and graphs.

Click "`Analyze`" to create the "`Analysis Report`" as shown in Figure 10.18. In this table, each row lists the liquidus temperature ($f_L = 1$) for the corresponding alloy composition. The liquidus temperatures for 900 alloys are listed in the same report which allows a quick comparison of liquidus temperature for different alloy composition. User can also plot 3D colormap and surface diagrams using the data in this report as shown in Figure 10.19.

	CalculationName	AlloyChemistry	T	fl	fs	phase
			C	mole/mole	mole/mole	
1	solidification_de...	98Al+1Mg+1Z...	653.3770	1.000000	1.110223E-016	Liquid+Fcc
2						
3	solidification_de...	97Al+1Mg+2Z...	651.4880	1.000000	1.110223E-016	Liquid+Fcc
4						
5	solidification_de...	96Al+1Mg+3Z...	649.5910	1.000000	0.000000	Liquid+Fcc
6						
7	solidification_de...	95Al+1Mg+4Z...	647.6870	1.000000	2.220446E-016	Liquid+Fcc
8						
9	solidification_de...	94Al+1Mg+5Z...	645.7750	1.000000	0.000000	Liquid+Fcc
10						
11	solidification_de...	93Al+1Mg+6Z...	643.8550	1.000000	0.000000	Liquid+Fcc
12						
13	solidification_de...	92Al+1Mg+7Z...	641.9280	1.000000	0.000000	Liquid+Fcc
14						
15	solidification_de...	91Al+1Mg+8Z...	639.9940	1.000000	0.000000	Liquid+Fcc
16						
17	solidification_de...	90Al+1Mg+9Z...	638.0520	1.000000	2.220446E-016	Liquid+Fcc
18						
19	solidification_de...	89Al+1Mg+10...	636.1030	1.000000	0.000000	Liquid+Fcc
20						
21	solidification_de...	88Al+1Mg+11...	634.1460	1.000000	2.220446E-016	Liquid+Fcc
22						
23	solidification_de...	87Al+1Mg+12...	632.1820	1.000000	1.110223E-016	Liquid+Fcc
24						
25	solidification_de...	86Al+1Mg+13...	630.2110	1.000000	1.110223E-016	Liquid+Fcc
26						
27	solidification_de...	85Al+1Mg+14...	628.2320	1.000000	0.000000	Liquid+Fcc
28						
29	solidification_de...	84Al+1Mg+15...	626.2460	1.000000	0.000000	Liquid+Fcc

Figure 10.18: Analysis report file of liquidus temperature T_L
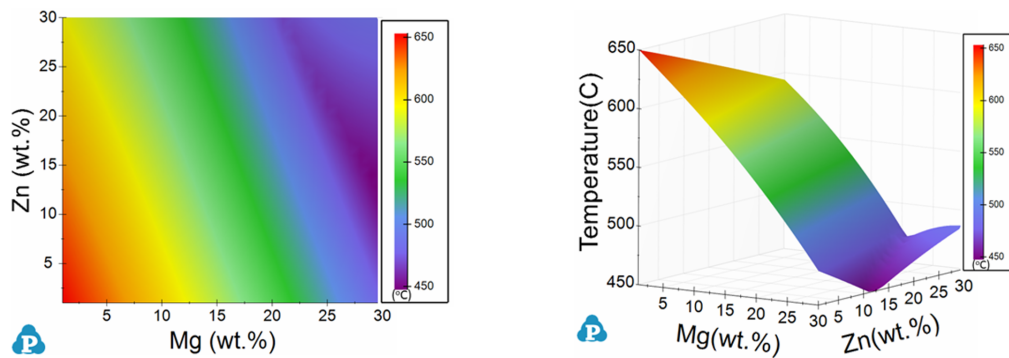
Figure 10.19: 3D diagrams of the liquidus temperatures: colormap (left) and 3D surface (right)

10. Save the obtained T_L report file via "`File → Save Current File As`

11. Now repeat the steps of 1-9 to obtain the T_S report file and save it.

TS.report		
CalculationName	AlloyChemistry	T
		C
1 solidification_default	98AL+1MG+1ZN in w%	440.4160
2		
3 solidification_default_1	97AL+1MG+2ZN in w%	480.1710
4		
5 solidification_default_2	96AL+1MG+3ZN in w%	474.9480
6		
7 solidification_default_3	95AL+1MG+4ZN in w%	474.9480
8		
9 solidification_default_4	94AL+1MG+5ZN in w%	474.9510
10		
11 solidification_default_5	93AL+1MG+6ZN in w%	459.2860
12		
13 solidification_default_6	92AL+1MG+7ZN in w%	341.2350
14		
15 solidification_default_7	91AL+1MG+8ZN in w%	341.2160

Figure 10.20: Analysis report file of solidus temperature T_S

12. Combine the two analysis reports. Note that user can also easily export the obtained report file to excel (**Table → Export to Excel**) for further editing and then import the modified file back to Pandat™ to create plot. For example, we can export the T_L and T_S reports to excel files and then combine them to obtain the solidification range ($\Delta T = T_L - T_S$). By importing the modified table data, one can plot the contour mapping of the solidification range as shown in Figure 10.21.
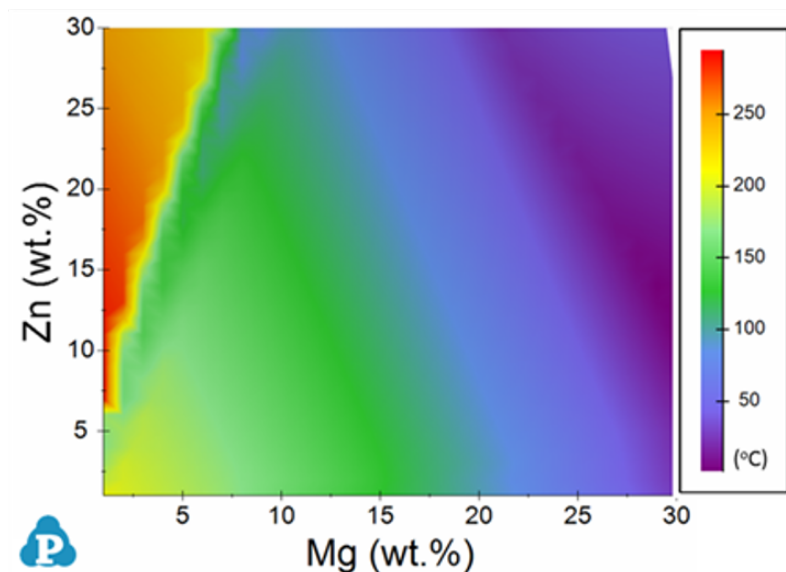
Figure 10.21: The color map of solidification range for Al-rich alloys using the Scheil model

10.2.3 HTC with PanEvolution/PanPrecipitation

In this example, we will demonstrate how to find the peak yield strength of the AA6005 alloy with varying both composition and heat-treatment temperature.

1. Load proper thermodynamic + mobility database and select Al, Mg, Si three components.

2. Load proper kinetic-parameter database to select the matrix phase and precipitates.

3. Choose the HTC function from the menu `Batch Calc → High Throughput Calculation (HTC)`.

4. Choose the calculation type from the drop-down list of `HTC` pop-up window and select "`Precipitation`".
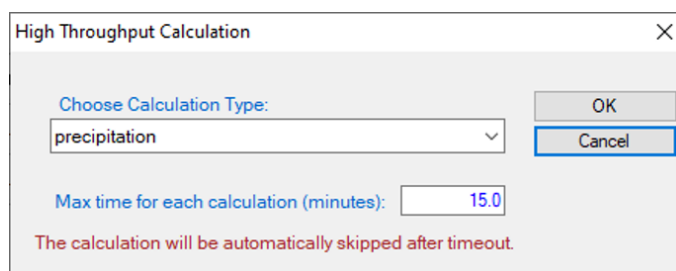
Figure 10.22: Dialog to choose calculation type of HTC

5. Define the compositional space for HTC simulation. As shown in Figure 10.23, the compositions of Mg and Si varying within the ranges of 0.4-0.6, and 0.6-0.9 (wt.%), respectively.

6. Define the thermal history for HTC simulation. User can define/import one thermal history by using the "**Thermal History**" dialog in Figure 10.23 or select the "**Load C. Curve**" and click the "**Import CC**" button to browse and load the predefined cooling curves (.txt or .dat). The example format of cooling curve file can be found in Figure 10.32 in Section 10.2.5 . In this example, isothermal aging for 20 hours at three temperatures 190, 182, and 170°C are defined individually in the loaded curve file. It is noted that only one thermal history can be typed in through GUI, while more than one thermal history can be considered through using the cooling curves file.
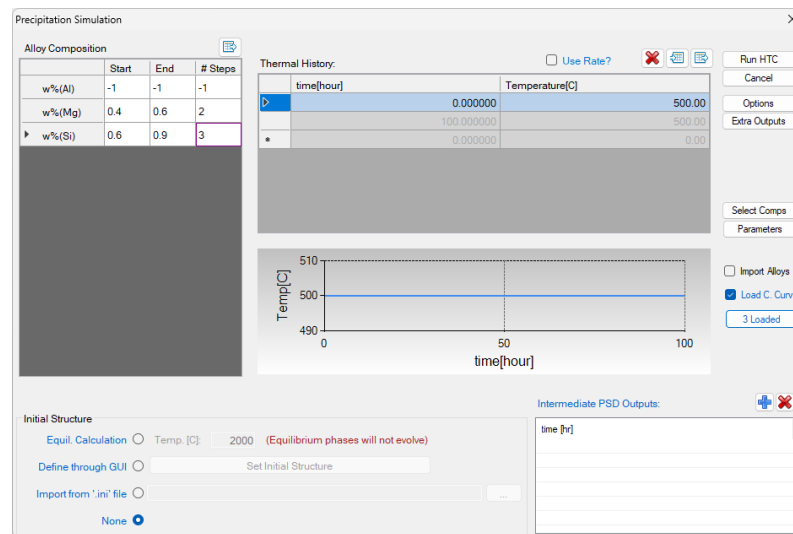
Figure 10.23: Dialog to setup compositional space and thermal history for precipitation HTC

7. In addition to use the default output, user can also customize the outputs using the "**Extra Outputs**" function shown in Figure 10.23. Then another window (Figure 10.24) will pop-up for setting the extra outputs. In this example, the extra output with time, T - temperature, w(*) - alloy composition, and sigma_y (yield strength) is generated.

Figure 10.24: Set properties in the extra table

8. After setup the compositional space for HTC and define/import the proper thermal history, user can click "`Run HTC`" button to perform HTC simulations.

9. Save the current workspace after all calculations are finished and run "`Result Analysis`" from the "`Batch Calc`" menu. User can use this commend to analyze the calculated results for a group of alloys and pick a certain property from each calculation for comparison. As shown in the following Figure 10.25, the following rule is used to obtain the maximum yield strength of each alloy under thee different heat-treatment conditions.

Figure 10.25: Criteria for precipitation results analysis

Figure 10.26 shows the obtained peak yield strength distribution within the defined composition space considering all three heat treatment conditions. User can run "`Result Analysis`" on this save workspace and dig out more information using other rules. For example, user can plot peak yield strength distribution in the composition space for one particular heat treatment condition.
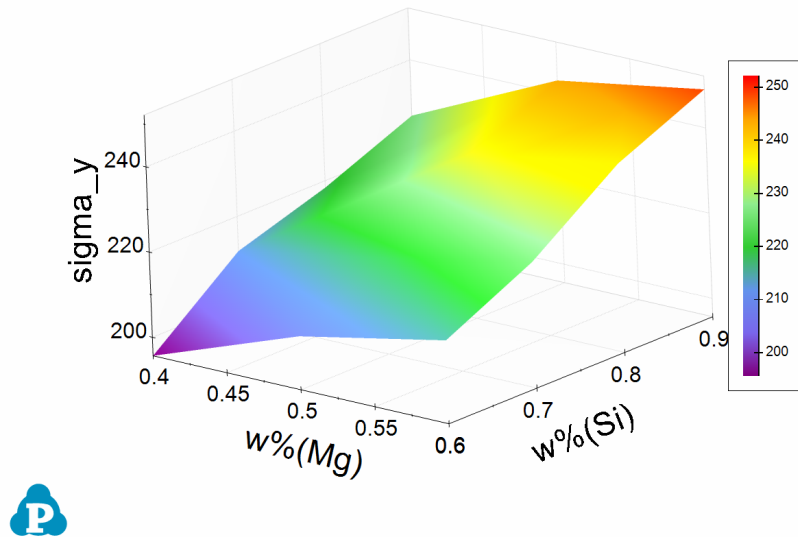
Figure 10.26: The maximum yield strength distribution within the defined compositional space

10.2.4 HTC with PanSolidification

Hot tearing or hot cracking is a serious defect occurred in welding and casting solidification. Cracking usually generated at the end stage of solidification along grain boundaries. Prof. Kou [2015Kou] proposed a criterion to describe the crack susceptibility by using a simple `crack susceptibility index` (CSI), which is the maximum value of $|dT/d(f_s)^{1/2}|$ at $f_s^{1/2} < 0.99$. The `CSI` criterion has been successfully applied to several Al-based alloy systems. In this example, we will demonstrate how to use `HTC` function in PanSolidification module to produce a susceptibility map in the `Al-Cu-Mg` ternary system.

1. Create a workspace and select PanSolidification module. Save the workspace in a user assigned folder different from that of the default workspace. The HTC calculation results will be saved automatically under this folder.

2. Load proper thermodynamic + mobility database and select Al, Cu, Mg three components.

3. Load proper solidification kinetic- parameter database (`.sdb`) to select the solidification alloy system.

4. Choose the HTC function from the `Batch Calc → High Throughput Calculation (HTC)`.

5. Choose the calculation type from the drop-down list of HTC pop-up window and select "`pan_solidification`".
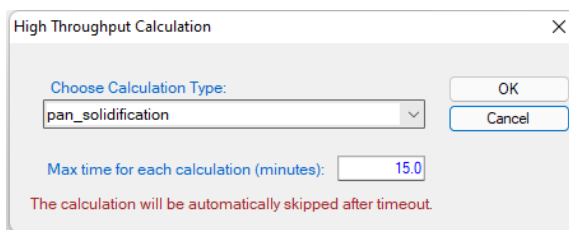
Figure 10.27: Dialog to choose calculation type of HTC in PanSolidification

6. Define the compositional space for HTC simulation. As shown in Figure 10.28, the compositions of both Cu and Mg vary from 0 to 5wt% in a double composition loops.

7. Define the solidification conditions for HTC simulation. User can define the cooling rate (the example is 20 K/s) by using the "`Thermal History`" dialog in Figure 10.28 or select the "`Load C. Curve`" and click the "`Import CC`" button to browse and load the predefined cooling curves (`.txt` or `.dat`). The example format of cooling curve file can be found in Figure 10.32 in Section 10.2.5 . Users can also simply output the Thermal History from GUI by click the icon 📝, then use the output file as a template to revise the cooling curve. Besides the cooling rate, the solidification rate or temperature gradient is also needed to be defined.
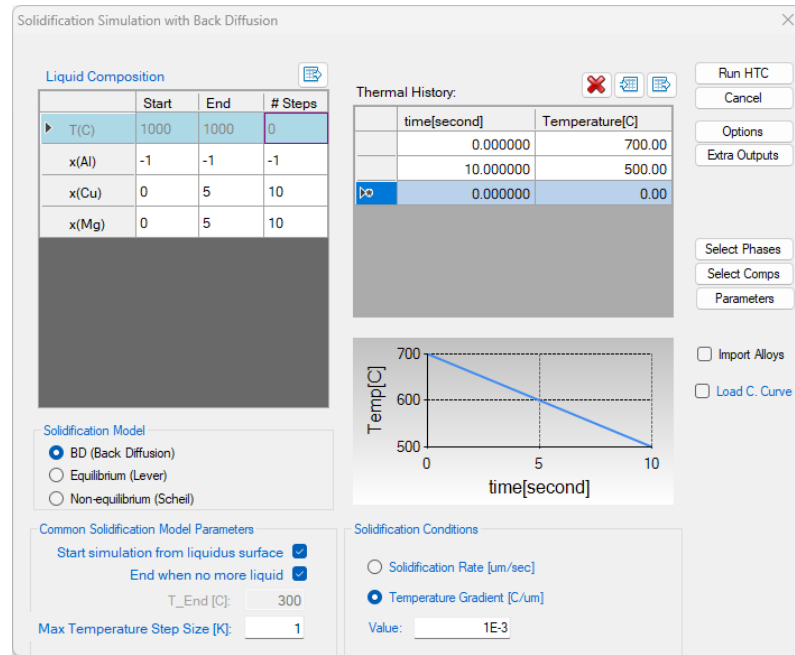
Figure 10.28: Dialog to setup compositional space and solidification conditions for PanSolidification HTC

8. In addition to use the default output, user can also customize the outputs using the "**Extra Outputs**" function. In this example, we defined the following extra output: `time`; `T` – temperature; `w(*)` – alloy composition; `fs` – solid phase fraction; `sqrt (fs)` – the square root value of the solid phase fraction; and `CSI` – the CSI index $|dT/d(fs)^{1/2}|$. (Note: The CSI index is introduced in Pandat™ 2022 version. For earlier versions, users can use the expression of `-T//sqrt(fs)` to obtain the CSI values.)
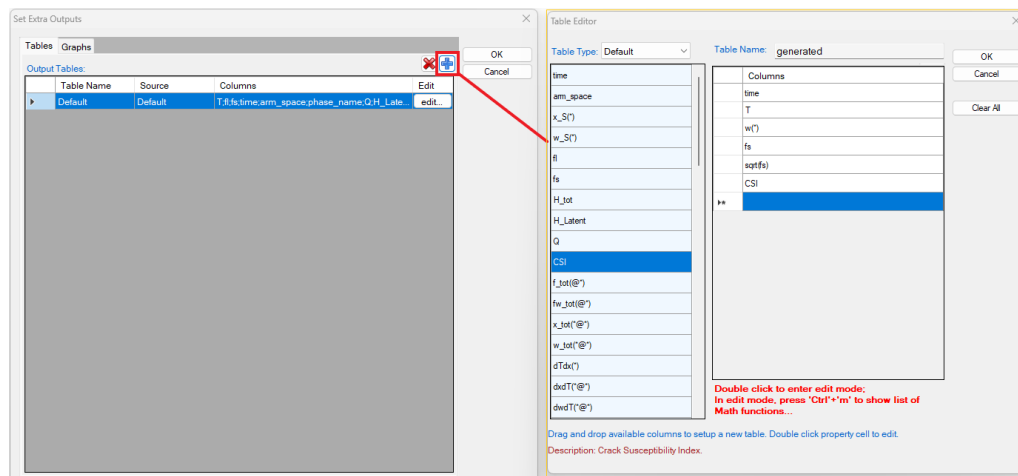
Figure 10.29: Set properties in the extra table

9. After setup the compositional space for HTC and define the proper thermal history, user can click "**Run HTC**" button to perform HTC simulations.

10. Run "**Result Analysis**" from the "**Batch Calc**" menu. User can use this commend to analyze the calculated results for a group of alloys and pick a certain property from each calculation for comparison. As shown in the following Figure 10.30, the criterion is to output the `CSI=MAX(CSI)` point with the condition of `sqrt (fs) < 0.99` at each composition point.
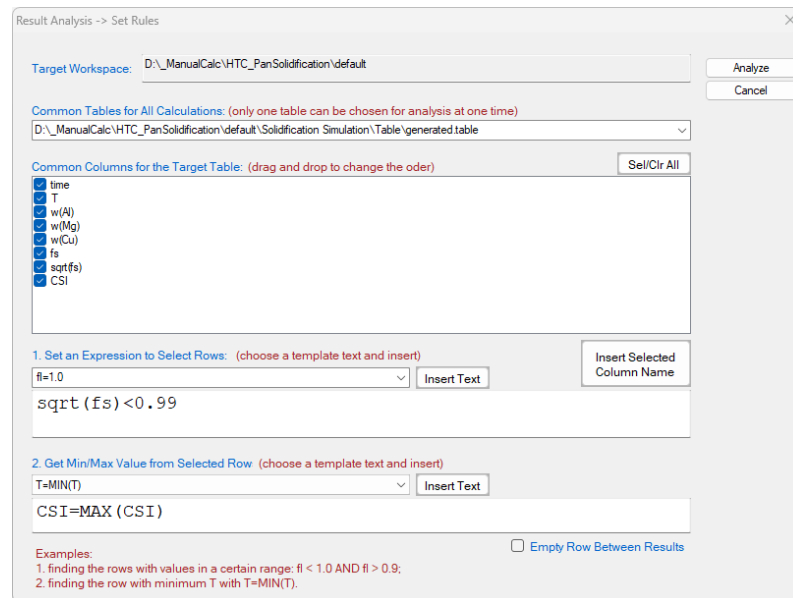
Figure 10.30: Criteria for Cracking Susceptibility Index setting from solidification results analysis

11. Figure 10.31 shows the obtained crack susceptibility map for Al-Cu-Mg alloys with cooling rate of 20 K/s. User can run "**Result Analysis**" on this save workspace and dig out more information using other rules.
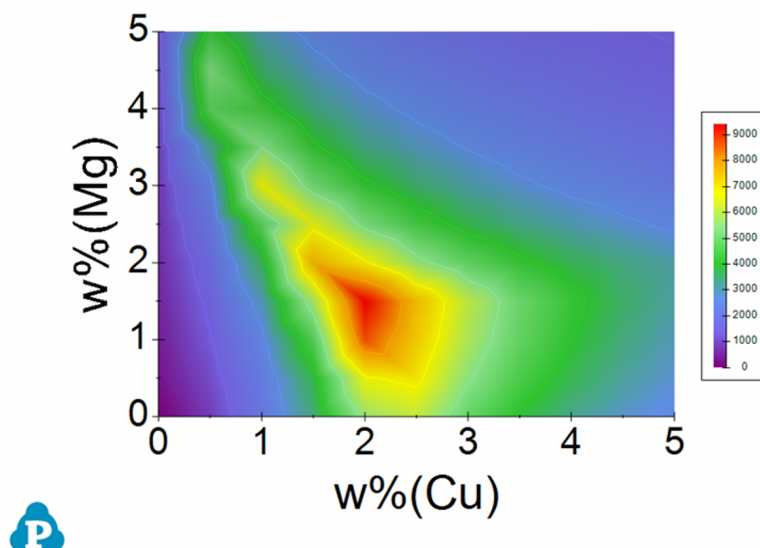
Figure 10.31: Al-Cu-Mg crack susceptibility map with cooling rate of 20 K/s

10.2.5 HTC in Console Mode

In order to facilitate the integration of PanEvolution with a third-party software package such as `iSight` or `DEFORM`, a new feature is developed so that the software can call PanEvolution for multiple simulations with different conditions. In this case, PanEvolution run in a console mode rather than in a regular GUI mode. This would significantly reduce the overhead from creating and maintaining many GUI components. In this mode, the simulation is performed through a script file or Pandat™ batch file (`.pbfx` file). After the simulation is done, the results are saved as `ASCII` files, which can then be loaded by third-party software package for subsequent simulations.

A typical application of this function is to run HTC of an alloy at various cooling profiles. The command to run this type of precipitation HTC is:

```
Pandat.exe Ni-14Al.pbfx "D:\ConsoleMode\results" cooling_curve.txt
1
```

There are four arguments passed to `Pandat.exe` in order to run HTC:

a. `Ni-14Al.pbfx:` batch file name, which defines all the simulation conditions such as unit, alloy chemistry, output format, etc.; The heat treatment schedule will be replaced by the 3rd argument if there is cooling curve file attached;

b. `D:\ConsoleMode\results`: working folder for Pandat™ HTC. A default workspace will be created automatically when running Pandat™ each time. The old workspace will be removed in this folder. If the user wants to keep the workspace and its results, all the files in this folder should be backed up before running HTC each time. Or the user may specify a different working folder for each HTC calculation;

c. `cooling_curve.txt`: the files defining cooling curves for different points. The file format is Tab Delimited text file.

time	P1	P2	P3
hour	C	C	C
0	550	550	550
10	500	550	550
100	500	500	550
1000	450	500	550

Figure 10.32: The example file format defining cooling curves for different points

The following is the structure of the working folder (see Figure 10.33):

1. `Workspace folder`: contains all the results for each simulation;

2. `cooling_curve_summary.txt`: the summary file which contains the results for the final step of each simulation; if there are multiple tables in `pbfx` file, only the results from the last table is summarized; the file format is Tab Delimited text file;

3. `pandat.log`: which logs the simulation progress; the level can be controlled by the last argument as shown above.

results
cooling_curve_summary.txt
pandat.log
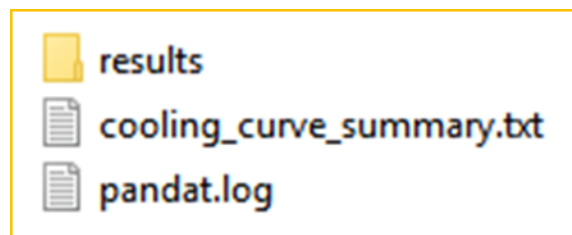
Figure 10.33: The structure of the working folder

Please refer to `...\Pandat 2024\Pandat 2024 Examples\ConsoleMode` folder for the related files.

11   Appendix

11.1 Mathematical Functions and Operations

Table 11.1 lists the common mathematical functions which can be used in databases (`tdb` file) as well as in Pandat™ GUI table column expressions. Table 11.2 and Table 11.3 lists the mathematical operations which could be used in Pandat™ table column expressions.

Table 11.1: Mathematical Functions for Databases and Table Column Expressions

Name	Comments
`sin(x), cos(x), tan(x), tan2(y,x)`	Trigonometric functions. `tan2(y,x)=tan(y/x)`
`asin(x), acos(x), atan(x), atan2(y,x)`	Arcus functions. `atan2(y,x)=atan(y/x)`
`sinh(x), cosh(x), tanh(x)`	Hyperbolic functions
`asinh(x), acosh(x), atanh(x)`	Arcus hyperbolic functions
`log2(x), log10(x), ln(x)`	Hyperbolic functions
`exp(x)`	Exponential function
`abs(x)`	Absolute value
`sqrt(x)`	Square root
`rint(x)`	Round to integral value
`sign(x)`	Sign function

Table 11.2: Mathematical Operations for Pandat™ Table Column Expressions

Name	Operation	Examples
`+, -,` `*, /, ^`	Addition, subtraction, multiplication, division and exponentiation	`H-T*S` `x(Al)+x(Ni)` `1/T`
`//`	Numerical derivative	`H//T` (See Section 3.3.8 for an example)

Table 11.3: Logical Operators for Pandat™ Table Column Expressions

Name	Meaning	Examples
`=`	equal to	`#phases=2` `phase_name=Liquid+Fcc+Hcp` `tieline=5` (specifying tieline density)
`!=`	not equal to	`phase_name!=Fcc`
`>`	larger than	`T>1200`
`<`	less than	`x(Al)<0.5`
`>=`	larger than or equal to	`f(@Liquid)>=0`
`<=`	less than or equal to	`a(Al)<=0.3`

11.2 Pandat Table Format Syntax

Table 11.4 lists the special words and formats used in Pandat™ table columns.

Table 11.4: Table Format Syntax (commonly used properties)

Syntax	Meaning	Note and Example
`T`	Temperature	Temperature can be in Celsius, Kelvin, or Fahrenheit. The unit can be changed/selected in the row below symbol T and the values updated instantly
`phase_name`	Names of phases that are in equilibrium	`Liquid+Fcc` means two phases are in equilibrium, one is `Liquid` and the other is `Fcc`
`#phases`	Number of phases in equilibrium	
`f(@*)` `fw(@*)`	Molar fraction(s) and weight fraction(s) of a phase or phases	`f(@Fcc)`: molar fraction of `Fcc` phase `f(@*)`: molar fraction of every phase in equilibrium
`x(*)` `w(*)`	Nominal composition of an alloy in molar fraction or weigh fraction	Fraction can be easily converted to percentage by selecting % from the unit row in the table
`x(*@*)` `w(*@*)`	Composition of a phase or every phase in equilibrium in molar fraction or weight fraction	`w(*@Liquid)`:composition of the Liquid phase in weight fraction `w(*@*)`:composition of every phase in equilibrium in weight fraction
`y(*@*)`	Site fraction of species in every sublattice for a specified phase or for every phase in equilibrium. If a phase has only one sublattice, it lists the composition of the phase; if	`y(AL@*)`:site fraction of AL in every sublattice of every phase `y(*@Beta)`:site fraction of every component in every sublattice in Beta phase

Syntax	Meaning	Note and Example
	a phase has two or more sublattices, it lists the site fraction of every component in every sublattice. If a component does not occupy a certain sublattice, the site fraction of this component will not be listed in that sublattice	`y(*@*)`:site fraction of every component in every sublattice in every phase in equilibrium
`G, H, S, Cp`	Gibbs energy, enthalpy, entropy and heat capacity of the system in the equilibrium state. The equilibrium state may include one phase or a mixture of phases	The listed Gibbs energy, enthalpy, and entropy are the properties of one mole atoms refer to the default reference state defined in the database. If the Gibbs energies of pure components are from SGTE substance database, the default reference state is `GHS298`
`G(:ref_ph [*])` `H(:ref_ph [*])` `S(:ref_ph [*])`	Gibbs energy, enthalpy and entropy of the system per mole of atoms referring to the given reference state.	`G(:Fcc[*])` is the Gibbs energy of the system per mole of atoms referring to `FCC` of every element. `H(:Fcc[Al], Hcp[Mg])` is the enthalpy per mole of atoms referring to `Fcc Al` and `Hcp Mg`.
`mu(*)`	Chemical potential of a specified component or every component when the system reach equilibrium	`mu(Al)` is chemical potential of Al in the equilibrium system referring to the default reference state
`mu(*:ref_ph [*])`	Chemical potential of a specified component or every component when the system reach equilibrium referring to the given reference state	`mu(Al:Fcc[Al])` is chemical potential of `Al` in the equilibrium system referring to the `Fcc Al`
`a(*)`	activity of a specified component or every component when the system reach equilibrium	`a(Al)` is the activity of Al in the equilibrium system referring to the default reference state

Syntax	Meaning	Note and Example
`a(*:ref_ph [*])` `r(*:ref_ph [*])`	activity or activity coefficient of a specified component or every component when the system reach equilibrium referring to the given reference state	`a(Al:Fcc[Al])` is the activity of `Al` in the equilibrium system referring to the `Fcc Al`
`fs, fl`	Fraction of solid (accumulated) and liquid during solidification	
`H_tot`	Total enthalpy of the system per mole of atoms. During the solidification process, `H_tot` is listed at each temperature. It is the total enthalpy of one mole atoms of the system at that temperature. It refers to the default reference state defined in the database	For example, at 500°C there listed two phases: `Liquid+Fcc`, the fraction of Liquid is 0.9 and that of `Fcc` is 0.1, then `H_tot` is the enthalpy of 0.9 mole of `Liquid` plus the enthalpy of 0.1 mole of `Fcc` at 500°C.
`Q`	Heat evolved during solidification from the beginning temperature to the current temperature	The beginning temperature can be set as a temperature above the liquidus if the user chooses to do so. The default setting for solidification starts at liquidus temperature
`H_Latent`	Latent heat. It is the heat released due to phase transformation only. During solidification, a small amount of liquid transformed to solid at each small temperature decrease. Latent heat listed at a certain temperature is accumulated from the beginning of solidification to the current temperature	For example, when temperature decreases from `T1` to `T2`, a small fraction of liquid, `dfL`, transformed to solid, the latent heat of this small step is `dfL*[H_Liquid (T2) - H_Solid (T2)]`. In other words, the heat released of the `Liquid` due to the temperature decrease from `T1` to `T2` is not included in Latent heat
`P`	External pressure	Unit can be changed in the second row of the table, and P values

Syntax	Meaning	Note and Example
		updated instantly
`P(*)`	Partial pressure of species	`P(O2)` is the partial pressure of `O2`
`P(@gas)`	The pressure of gas phase when the system reaches equilibrium	
`G(@*)` `H(@*)` `S(@*)` `Cp(@*)`	Gibbs energy, enthalpy, entropy and heat capacity of a specified phase or every phase involved in the calculation	The listed value is for per mole of atoms. The reference state is the default reference state defined in the database
`G(@*:ref_ph [*])` `H(@*:ref_ph [*])` `S(@*:ref_ph [*])` `Cp(@*:ref_ph [*])`	Gibbs energy, enthalpy, entropy and heat capacity of a specified phase or every phase involved in the calculation referring to the given reference state	The listed value is for per mole of atoms. If the calculation (line calculation) is for the system, these properties for a phase are listed only in the range where the phase is stable; if the calculation is for individual phase, then these properties are listed in the entire range
`H(*@*:ref_ ph) S (*@*:ref_ph)`	Partial molar enthalpy and entropy of a component in a phase with a given reference phase	If reference phase is not given, the default reference state in database is used
`G_id(@*)` `H_id(@*)` `S_id(@*)`	Gibbs energy, enthalpy and entropy due to ideal mixing for the given phase	`S_id(@Fcc)` is the entropy of the `Fcc` phase per mole of atoms due to ideal mixing.
`G_ex(@*)` `H_ex(@*)` `S_ex(@*)`	Excess Gibbs energy, enthalpy and entropy other than ideal mixing part for the given phase	`G_ex(@Fcc)` is the excess Gibbs energy of the `FCC` phase per mole of atoms.
`G_Mag(@*)` `H_Mag(@*)` `S_Mag(@*)`	Magnetic contribution part on the Gibbs energy, enthalpy and	`G_Mag(@Bcc)` is the magnetic contribution on the Gibbs energy of the `BCC` phase per mole of atoms.

Syntax	Meaning	Note and Example
	entropy for the given phase	
`mu(*@*)`	Chemical potential of component(s) in a specified phase or in every phase involved in the calculation	`mu(Al@Fcc)` is chemical potential of `Al` in `Fcc` phase. If the calculation (line calculation) is for the system, these properties for a phase are listed only in the range where the phase is stable; if the calculation is for individual phase, then these properties are listed in the entire range
`mu(*@*:ref_ph[*])`	Chemical potential of component(s) in a specified phase or in every phase involved in the calculation referring to the given reference state	`mu(Al@Fcc:Fcc[*])` is chemical potential of `Al` in `FCC` phase referring to `FCC` state of every component.

`a(*@*)`
`r(*@*)`
	Activity and activity coefficient of component(s) in a phase referring to the default reference state in database	`a(Cu@fcc)=exp{mu (Cu@fcc)/RT}`
		`r=a/x`
		see `mu(*@*)` for an example

`a(*@*:ref_ph[*])` `r(*@*:ref_ph[*])`	Activity and activity coefficient of component(s) in a phase referring to the given reference state	`a(Cu@fcc:liquid)=exp{ (mu (Cu@fcc)-mu(pure liquid Cu at same T))/RT}` see `mu(*@*:ref_ph[*])` for an example
`DF(@|*)`	Driving force of each phase entered the calculation referring to the equilibrium state of the system. Driving force can only apply to point calculation or line calculation	For example `P1`, `P2`, and `P3` phases are selected in a point calculation (given an overall composition and temperature), and the result shows P1 and P2 are in equilibrium at this point, then `DF(@|P1)=0,` `DF(@|P2)=0,` and `DF(@|P3)<0.` P3

Syntax	Meaning	Note and Example
		is not stable, and the absolute value of `DF(@|P3)` is the minimum energy needed to make `P3` stable. It should be pointed out that the equilibrium compositions of `P1` and `P2` are not at the overall composition, and the driving force of `P3` is most likely not at this overall composition as well unless `P3` is a line compound and its composition is the same as the overall composition. For a line calculation, such as at a fix temperature with varying composition, the equilibrium at each point along the line is calculated, and the driving force of each phase at each point can be listed in the table using `DF(@|*)`. Again, notice that driving force of a phase at a certain composition point is not the energy difference between this phase and the equilibrium state at this composition, it is the minimum energy needed to form this phase (most likely at another composition).
`DF(@!*)`	Driving force for each dormant phase referring to the equilibrium state of the system. Again, it only applies to point calculation or line calculation. A dormant phase does not enter a calculation, but its driving force is calculated. This is different from a suspended phase, which does not involve in the calculation at all.	For example `P1`, `P2`, and `P3` phases are selected in a point calculation (given an overall composition and temperature), `P4` is set as dormant phase. Again, the result shows `P1` and `P2` are in equilibrium at this point, then `DF(@|P1)=0,` `DF(@|P2)=0`, and `DF(@|P3)<0`. However, if you list DF(@!P4), it may be greater than zero. This

Syntax	Meaning	Note and Example
		means `P4` will be stable if it is selected to enter the calculation.
`tieline`	This column will list the names of phases in equilibrium and the table is the selected tieline properties.	See Section 3.3.8.2 Tutorial for an example
`f_tot(@*)`	Accumulated fraction of each solid phase during solidification. By Scheil model, it is accumulated from each solidification step. By Lever rule (equilibrium) model, it is the fraction of each phase in equilibrium at the current temperature.	`f_tot(@Liquid) = fl` total summation of the solid phases: `sum(f_tot(@Solid`$_i$`))` `= fs`
`Vm, alpha_` `Vm, density`	Molar volume, expansion coefficient, density	
`Vm(@*),` `density(@*)`	Molar volume and density of each phase involved in the calculation	
`n_mole` `n_kg`	amount in mole and in kg	`n_mole, n_kg, n_kg(*), n_` `mole (@*), n_kg(*@*)`
`surface_` `tension` `(@liquid),` `viscosity` `(@liquid)`	Surface tension and viscosity of liquid phase	
`M(*@*)`	Atomic mobility of species in a phase	
`DC(*,J@*:N)`	Chemical diffusivity of species in a phase	`J` = gradient species, `N` = reference species (`N` cannot be `*`)
`DT(*@*)`	Tracer diffusivity of species in a	

Syntax	Meaning	Note and Example
	phase	
struct(@*)	This is for a phase with multiple sublattice structure. It gives the structure in the form such as "[2011]", which means the first two sublattice have the same site fractions.	
HSN(@*)	determinant of Hessian matrix of Gibbs free energy of a phase	HSN(@Fcc)
eVal(#*@*)	eigenvalues of Hessian matrix of Gibbs free energy of a phase. * after # represents the eigenvalue index	eVal(#Al@Fcc), eVal (#Cu@Fcc)
eVec(*#*@*)	eigenvectors for the eigenvalues of Hessian matrix of Gibbs free energy of a phase. * before # represents the component of eigenvector. * after # represents the eigenvalue index.	eVec(Al#1@Fcc), eVec (Cu#1@Fcc) eVec(Al#2@Fcc), eVec (Cu#2@Fcc)

11.3 PanEvolution tables

Table 11.5: Summary of Equations for Nucleation Models

Name	Operation	Examples
J	$$J = N_\nu Z \beta^* \exp\left(-\frac{\Delta G^*}{k_B T}\right) \exp\left(-\frac{\tau}{t}\right)$$	The transient nucleation rate
N_ν	$$N_\nu = N_f N_i \text{ and}$$ $$N_i = \rho_i \left(\frac{N_A}{V_\alpha}\right)^{\frac{i}{3}}$$	Potential nucleation sites
Z	$$Z = \frac{V_\alpha}{2\pi N_A R^{*2}} \sqrt{\frac{\sigma_{\alpha\beta}}{k_B T}}$$	Zeldovich factor
β^*	$$\beta^* = \frac{4\pi R^{*2}}{a^4} D_{eff}$$	Atomic attachment rate
τ	$$\tau = \frac{1}{2Z^2 \beta^*}$$	Incubation time
ΔG^*	$$\Delta G^* = \frac{16\pi}{3} \frac{\sigma_{\alpha\beta}^3}{(\Delta G_V + \Delta G_S)^2}$$	Nucleation barrier energy
R^*	$$R^* = -\frac{2\sigma_{\alpha\beta}}{\Delta G_{vol}}$$	Critical nucleation radius
ΔG_{vol}	$\Delta G_V + \Delta G_S$	Volume energy change during nucleation. ΔG_V is the chemical driving force per volume and ΔG_S is the elastic strain energy
ΔG_S	$$\Delta G_S = \frac{2}{3}\mu\Delta^2 f(A_R)$$	Elastic strain energy. The volume misfit Δ and particle aspect ratio A_R are given in `kdb` file
σ_{eff}	$$\sigma_{eff} = \frac{(b\,\sigma_{\alpha\beta} - a\,\sigma_{\alpha\alpha})^3}{c^{\frac{2}{3}}} \sqrt[3]{\frac{1}{36\pi}}$$	Effective interfacial energy

Name	Operation	Examples
$\sigma_{\alpha\beta}$	$\sigma_{\alpha\beta} = 0.329\, \alpha_{SCF}\beta_{diff}(N_A V_\alpha^2)^{-\frac{1}{3}}\Delta H_{sol}$	The estimated interfacial energy based on GBB method
D_{eff}	$D_{eff} = \left[\sum_{i=1}^{n}\frac{(C_{pi}-C_{0i})^2}{C_{0i}D_{0i}}\right]^{-1}$	Effective diffusivity for multi-component alloys

Table 11.6: Model Options for precipitation in kdb

Name	Options	Description
model	KWN, Fast-Acting(FA)	Refer to Figure 5.2
morphology	Sphere; Cylinder	Refer to Figure 5.3 . The aspect ratio A_R and shape factors are set to be 1 automatically when "Sphere" is selected
nucleation	Modified_Homogeneous; Grain_Boundary; Grain_Edge; Grain_Corner; Dislocation	Refer to Table 11.5. Both homogeneous and heterogeneous nucleation can be considered by "Modified_Homogeneous". In this case, the values of N_v and $\Delta G*$ must be manually adjusted through Nf, ΔG_v and ΔG_S as discussed in Table 11.8.
growth	Simplified; SFFK; SFFK_Shape_Evolution	A constant value of The aspect ratio A_R can be assigned for "Simplified" and "SFFK". Choose "SFFK_Shape_Evolution" for shape evolution, which means A_R varies during particle growth
phase_name		Each "PrecipitatePhase" has a "name" and "phase_name". "phase_name" must be consistent with the name in tdb/pdb. The "phase_name" tag can be empty if "name" and "phase_name" are same.

Table 11.7: Model Options for grain growth in kdb

Name	Options	Description
model	Grain_OneSize, Grain_MultiSize	Refer to Figure 5.2
morphology	Sphere	Only spherical approximation is considered
nucleation	N/A	Recystallization process is not included
growth	Grain_Simplified	The simplified grain growth model is implemented as shown in Equations Eq. 5.38 and Eq. 5.41. User's grain growth model can be defined in KDB (see example Section 5.3.8)

Table 11.8: Kinetic model parameters for precipitation in kdb

Name	Unit	Description	Equation
Molar_Volume	m^3/mole	Molar volume of matrix or precipitate phase `<Parameter type="Molar_ Volume" value="6E- 6" description="Molar Volume" />`	V_m in Eq. 5.2 and V_α in Eq. 5.3
Grain_Size	m	The grain size of the matrix phase `<Parameter type="Grain_ Size" value="1e- 4" description="Grain size, default value = 1e-4m" />`	D in Eq. 5.12
Dislocation_ Density	m^{-2}	The dislocation density in the matrix phase `<Parameter type="Dislocation_ Density" value="1e13" description = "Dislocation density, Default value =1.0e12/m^-2" />`	ρ_i in Eq. 5.10
Grain_Aspect_ Ratio	N/A	The aspect ratio for the matrix grain `<Parameter type="Grain_ Aspect_ Ratio" value="1.0" description="grain aspect ratio,`	A in Eq. 5.12

Name	Unit	Description	Equation
		```default value = 1.0" />```	
Contact_Angle	degree	Contact angle of nucleus on grain boundary, default value = 90 degree	$\sigma_{\alpha\alpha} = 2\cos\theta\sigma_{\alpha\beta}$ in Eq. 5.8
Aspect_Ratio	N/A	The aspect ratio of the precipitate phase. The value of A_R is evolving if "SFFK_Shape_Evolution" is chosen as growth model. ```<Parameter type="A_R" value="1" description ="Initial aspect ratio" />```	A_R in Eq. 5.19

Interfacial_Energy	J/m^2	Interfacial energy
		```<Parameter type="Interfacial_Energy" value="0.2" description ="Interfacial Energy" />```
		User keyword "IFE_CAC(*)" to get the calculated interfacial energy:
		```<Parameter type="Interfacial_Energy" value="IFE_CALC(*)" description="Interfacial Energy" />```
			$\sigma_{\alpha\beta}$ in Eq. 5.2 and Eq. 5.13

Interfacial_Energy_L	J/m^2	Interfacial energy in L direction ```<Parameter type="Interfacial_Energy_L" value="0.05" description="Interfacial Energy in L direction" />```	Used in "SFFK_Shape_Evolution" model
Antiphase_Boundary_Energy	J/m^2	Antiphase boundary energy	γ_{APB} in Eq. 5.34 and Eq. 5.36
Atomic_Spacing	m	Usually use lattice constant ```<Parameter type="Atomic_Spacing" value="7.6E-10" description="Atomic Spacing" />```	a in Eq. 5.4

Name	Unit	Description	Equation
Nucleation_Site_Parameter	N/A	Homogeneous: choose a value close to solute concentration; Heterogeneous: choose a value close to nucleation density when "Modified_Homogeneous" option is chosen for nucleation model. Otherwise, use the model automatically estimate the nucleation density and default value of 1.0 can be used. Such an example is given in Section 5.3.5	N_f in Eq. 5.7 and Eq. 5.11
Driving_Force_Factor	N/A	A factor adjusting chemical driving force obtained by thermodynamic calculation	A pre-factor applied to ΔG_V in Eq. 5.2
Strain_Energy		The elastic strain energy per volume of precipitate offsetting the calculated value by equation Eq. 5.5.	
Volume_Misfit	N/A	The volume misfit	Δ in Eq. 5.6
Kinetic_Parameter_Factor	N/A	A factor adjusting kinetic parameter obtained by thermodynamic and mobility calculation	A pre-factor applied to adjust K in equation Eq. 5.14
Effective_Diffusivity_Factor	N/A	A factor adjusting effective diffusivity for nucleation obtained by mobility calculation	A pre-factor applied to adjust D_{eff} in Eq. 5.4
Steady_State_Nucleation_Rate	N/A	0: transient nucleation rate; 1: steady state nucleation rate;	$exp(\frac{-\tau}{t})$ in Eq. 5.1

Table 11.9: Kinetic model parameters for grain growth in kdb

Name	Unit	Description	Equation
`Interfacial_ Energy`	J/m^2	High angle grain boundary energy	γ in equations Eq. 5.40, Eq. 5.41 and Eq. 5.42
`Grain_ Boundary_Width`	m	Grain boundary width. A suggested value is equal to twice the atomic radius.	δ in Eq. 5.39
`Grain_ Boundary_ Mobility_ Factor`	N/A	A pre-factor to adjust grain boundary mobility.	A$_2$ in Eq. 5.39
`Zener_Drag_ Force_Factor`	N/A	A factor for Zener drag force, default is 0 for not considering Zener pinning effect.	β in Eq. 5.41

Table 11.10: User-defined models for grain growth in kdb

Name	Unit	Description	Equation
`Grain_ Boundary_ Mobility`	m/(s·J/mol)	User-defined grain boundary mobility. It can be an expression, for example: 1e-18/2.86e-10 `<VariableTable name="Variables replacing built- in variables"> Parameter type="Grain_ Boundary_ Mobility" value="KP (@Grain)/2.86e-10" description="Grain_ Boundary_ Mobility" /> </VariableTable>`	`KP(@Fcc)` ($\bar{M}$ in Eq. 5.39 is the effective atomic mobility for multi-component alloys calculated from mobility database.
`Zener_Drag_`	J/mol	User-defined drag force for Zener	

Name	Unit	Description	Equation
Force		pinning effect. It can be an expression, for example: -1.5*7.1E-6*0.75*0.15/(0.3 * 1e-6), which is "– `factor*Vm*Sigma*fv/size`"	
Growth_Rate	m/s	User-defined grain growth rate. It can be an expression, for example: KP (@Grain)/2.86e-10*2*0.75*7.1e-6/s (@Grain) with `KP(@Grain)` is the effective mobility and `s(@Grain)` is the mean grain size	

Table 11.11: Mechanical model parameters defined in kdb

Name	Unit	Description	Equation
Shear_Modulus	Pa	The shear modulus of the matrix phase	μ in Eq. 5.6 and Eq. 5.36
Burgers_Vector	m	The Burgers vector of the matrix phase	b in Eq. 5.36
Taylor_Factor	N/A	The Taylor factor of the matrix phase	M in Eq. 5.37
Solution_ Strengthening_ Factor	N/A	scaling factor of alloying element for solution strengthening	a_j in Eq. 5.31
Strength_ Parameter	N/A	Strengthening parameter due to precipitation hardening	k_P in Eq. 5.27
Shearing_ Critical_Radius	m	Critical radius shifting from shearing to looping mechanism	R_C in Eq. 5.27
Intrinsic_ Strength	MPa	The baseline contribution including lattice resistance, work-hardening and grain boundaries hardening.	σ_0 in Eq. 5.30

Name	Unit	Description	Equation
Hardness_Factor	N/A	The yield strength in MPa can be converted to hardness in VPN based on Eq. 5.31	A in Eq. 5.32
Hardness_ Constant	VPN	The yield strength in MPa can be converted to hardness in VPN based on Eq. 5.31	B in Eq. 5.32

Table 11.12: Model parameters for RX and Dislocation Density in kdb

Name	Unit	Description	Equation
f_WH	N/A	Work hardening coefficient for dislocation density evolution model	f_w in Eq. 5.44
f_DRV	N/A	Dynamic recovery coefficient for dislocation density evolution model	f_v in Eq. 5.45
f_SRV	N/A	Static recovery coefficient for dislocation density evolution model	
f_RX	N/A	Recrystallization coefficient for dislocation density evolution model	f_x in Eq. 5.46
Avrami_ Exponent	N/A	Exponent for JMAK	n in Eq. 5.49
time_half	s	The time for 50% RX for JMAK	$t_{0.5}$ in Eq. 5.49
Nucleation_ Site_ Parameter	N/A	Parameter for potential nucleation sites for the Fast-Acting nucleation model	N_f in Eq. 5.57
Kinetic_ Parameter_ Factor	N/A	Kinetic parameter factor for the Fast-Acting growth model, which is a pre-factor to adjust the effective self-diffusivity D_{eff}	Pre-factor of D_{eff} in Eq. 5.60

Table 11.13: User-Defined variables for RX and Dislocation Density in kdb

Name	Description
Dislocation_Density_Rate	User-defined dislocation density rate to replace the built-in K-M model defined by Eq. 5.43
Nucleation_Rate	User-defined nucleation rate to replace the built-in model defined by Eq. 5.58
Nucleation_Barrier_Energy	User-defined Nucleation barrier energy to replace the built-in model defined by Eq. 5.56
Critical_Grain_Size_RX	User-defined Critical nucleus size for RX to replace the built-in model defined by Eq. 5.55
Dislocation_Mean_Free_Path	User-defined Dislocation mean free path for critical density to replace the built-in model defined by Eq. 5.54
Growth_Rate	User-defined Growth rate to replace the built-in RX growth model defined by Eq. 5.59

Table 11.14: Symbol and syntax for retrieving system quantities

Name	Unit (SI)	Comments
time	second	Time
T	K	Temperature
vft		Total Transformed Volume Fraction: $vft = \sum_p vf_p$ where vf_p is the transformed volume fraction of phase
x(comp), w(comp)		Overall alloy composition

Table 11.15: Symbol and syntax for retrieving quantities of precipitate phases or grain

Name	Unit (SI)	Comments
s(@phase)	m	Average size/radius of equivalent sphere particles or grain

Name	Unit (SI)	Comments
D(@phase)	m	Diameter of cylinder
L(@phase)	m	Length/Height of cylinder
A_R(@phase)	m	Aspect ratio of cylinder
nd(@phase)	$\#m^{-3}$	Number density
nr(@phase)	$m^{-3}sec^{-1}$	Nucleation rate
vf(@phase)		Volume fraction of specified phase
x(comp@phase), w (comp@ phase)		Instant composition of the matrix or precipitate phases
IFE_CALC(@phase)	J/m^2	Model calculated interfacial energy
dgm(@phase)	J/mole	Nucleation driving force of phase(s)
vf_range (@phase,lb,ub)		The volume fraction for different particle groups defined by a size range [lb, ub] such as primary, secondary and tertiary in Ni-based super alloys, for example vf_range(@L12_FCC,0.5e-8, 0.5e-7)
s_range (@phase,lb,ub)		average size for different particle groups defined by a size range [lb, ub], for example s_range(@L12_FCC,0.5e-8, 0.5e-7)
Grain_Size	m	Mean grain size for the deformed matrix grains and the recrystallized grains
dislocation_ density(@*)	$1/m^2$	Dislocation density in the deformed matrix grains or the recrystallized grains. Example: dislocation_density(@Grain)

Table 11.16: Symbol and syntax for retrieving quantities of particle size distribution (PSD)

Name	Unit	Comments
time	s	The PSDs are saved for the user-specified times; the PSD for the last time step is automatically saved. Using `time = t` to get the PSD for time "t".
psd_id	N/A	The PSD consists of a certain number of cells (size classes); `psd_id` gets the cell id.
psd_s (@phase)	m	The characteristic size of a precipitate phase or grain for each cell.
psd_nd (@phase)	#m⁻³	The number density of a precipitate phase or grain for each cell.
psd_gr (@phase)	m/sec	The growth rate of a precipitate phase or grain for each cell.
psd_ns (@phase)		Normalized size of the cell $$psd\_ns(@phase) = \frac{psd\_s(@phase)}{s(@phase)}$$
psd_nnd (@phase)		Normalized number density of the cell $$psd\_nnd(@phase) = \frac{psd\_nd(@phase)}{nd(@phase)}$$
psd_df (@phase)		The distribution function: $$psd\_df(@phase) = \frac{s(@phase)}{psd\_w(@phase)}psd\_nnd(@phase)$$ with $psd\_w(@phase)$ being the cell width
psd_cvf (@phase)		Cumulative volume fraction of phase(s). Example: `psd_cvf (@L12_FCC)`.

Table 11.17: Symbol and syntax for retrieving mechanical properties

Name	Unit (SI)	Comments
sigma_y	MPa	Overall yield strength. Example: `sigma_y`
Hv	vpn	Overall microhardness. Example: `hv`
sigma_i	MPa	Intrinsic yield strength. Example: `sigma_i`.
sigma_ss	MPa	Yield strength due to solution strengthening. Example: `sigma_ss`.
sigma_p(@*)	MPa	Yield strength due to precipitation hardening. Example: `sigma_p(@Mg5Si6)`.
sigma_m (@Grain)	MPa	Yield strength due to dislocation interactions in grain structure.

Table 11.18: Constants of mathematics and physics

Name	Comments
_K	Boltzmann constant
_PI	Archimedes' constant.
_R	Molar gas constant.
_NA	Avogadro constant.
_E	Natural Logarithmic Base.

11.4 Keywords and Syntax in POP File

One of the commonly used experimental data file formats is POP. The POP format has been widely accepted in CALPHAD community. PanOptimizer accepts most of the keywords in POP forma and adds a few special keywords. The keywords and syntax accepted in PanOptimizer are summarized below in Table 11.19. PanOpimizer is case insensitive to the following keywords.

Table 11.19: Keywords and Syntax in POP File

Command	Syntax
Create_New_Equilibrium	Create_New_Equilibrium [integer1], integer2 This command defines specific phase equilibrium. Each experimental equilibrium must start with the CREATE command. The first integer is a unique identifier, and the second integer is an initialization code, for which 0 means that all components and phases are initially suspended, 1 means that all components are entered but all phases suspended, 2 means all components and phase are initially entered.
ENTER_SYMBOL CONSTANT	ENTER_SYMBOL CONSTANT dT=200
ENTER_SYMBOL Function	ENTER_ SYMBOL FUNCTION SXS= -100+8.314*T
COMMENT	COMMENT [string] this will be used for legend text when merged with the phase diagram
Change_Status	Change_Status [Phase] [Name] = [Status] [Value] This command defines the status of selected phases. Four statuses are allowed: Fixed, Suspended, Dormant, and Entered. The value is an initial phase amount for Entered phases

Command	Syntax
	(PanOptimizer will use this value as a start value for the Entered phase to solve the phase equilibrium). The exact value of phase amount must be provided for Fixed phases.
Set_Condition	Set_Condition [state variable 1 = value] [state variable 2 = value]... This command defines the calculation conditions at which the degree of freedom of the system is zero and the phase equilibrium can be calculated. Examples of conditions are temperature, pressure, and composition, or a constraint function. Uncertainty can be added to each condition.
Set_Reference_State	Set_Reference_State [component] [reference states] This keyword defines the reference state of a component. The field of reference state includes phase name, temperature and pressure in sequence.
Experiment	Experiment[state variable 1 = value : uncertainty] [state variable 2 = value : uncertainty] This keyword specifies the quantity to be fitted by the optimization. The syntax of this command is similar to the command of Set_Condition. It is followed by a state variable or a function name and a value and an uncertainty.
Set_Start_Value	Set_Start_Value [state variable 1 = value]…… This keyword is mainly used if the phase described by complicated compound energy model. It is also used for phases with fixed status. The most frequently used variable is site fraction of sublattice. PanOptimizer can automatically search the site fractions of components and species when the phases are in entered status

Command	Syntax
Table_Head Table_Values Table_End	Table_Head, Table_Values and Table_End The three keywords define a set of phase equilibria under similar conditions. The keyword Table_Head must be followed by Create_New_Equilibrium.

Table 11.20: Recognized variables used in defining experimental data

Function Name	Syntax
`T`	Temperature in the system (in K).
`P`	Pressure in the system (in Pascal).
`N`	System size (mole number in moles)
`X`	`X(comp)`: overall molar fraction of a component in a system. `X(phase, comp)`: molar fraction of a component in a phase.
`W`	`W(comp).`: overall weight fraction of a component in a system. `W(phase, comp)`: weight fraction of a component in a phase.
`Y`	`Y(phase,constituent # sublattice)`: The site fraction of the constituent in certain sublattice of the phase. For example: `Y(Sigma,Fe#3)` is the site fraction Fe in the third sublattice of the Sigma phase.
`GM, HM, SM`	`GM(phase), HM(phase), SM(phase)` Gibbs energy, enthalpy and entropy of phase per mole of atoms.
`GMR, HMR, SMR`	`GMR(phase), HMR(phase), SMR(phase)` Gibbs energy, enthalpy and entropy of phase per mole of atoms with given reference state.
`Cp`	`Cp(phase)`: The heat capacity of a phase.
`ACR`	`ACR(comp, phase)` Activity of component in a phase with a given reference state.

Function Name	Syntax
MU	`MU(comp), MU(phase,comp)` Chemical potential of component in system or phase.
MUR	`MUR(comp), MUR(phase,comp)` Chemical potential of component in system or phase with given reference state.
M, LOGM	`M(phase, K), LOGM (phase, K)` Mobility coefficient or log10 of the mobility coefficient where K = diffusing species.
DT, LOGDT	`DT(phase, K), LOGDT(phase, K)` Tracer diffusion coefficient or log10 of the tracer diffusion coefficient where K = diffusing species.
DC, LOGDC	`DC(phase, K, J, N), LOGDC(phase, K, J, N)` Chemical diffusion coefficient or log10 of the chemical diffusion coefficient where J = gradient species and N = reference species. Note: the reference species N must be given for DC calculation.
NPF	`NPF(phase)`: mole phase fraction of the given phase.
DGM	`DGM(phase)`: The driving force of the given phase.
TC	Curie temperature for magnetic ordering
BMAGN	Bohr magneton number for magnetic ordering
VM, DS, MW, VS, ST	`VM(phase), DS(phase), MW(phase), VS (Phase), ST(phase)` Molar volume, density, molecular weight, viscosity and surface tension of phase.

12   References

[1931Ons1] L. Onsager, "Reciprocal Relations in Irreversible Processes I", Phys. Rev., 37 (1931): 405-426.

[1931Ons2] L. Onsager, "Reciprocal Relations in Irreversible Processes II", Phys. Rev., 38 (1931): 2265-2279.

[1932But] J.A.V. Butler, "The Thermodynamics of the Surfaces of Solutions", Proc. R. Soc. Lond. A, 135 (1932): 348-375.

[1940Nab] F.R.N. Nabarro, "The strains produced by precipitation in alloys", Proc. R. Soc. Lond. A, 175 (1940): 519-538.

[1937Kol] A.N. Kolmogorov, Statistical theory of crystallization of metals (in Russian). Bull. Acad. Sci. USSR Ser. Math., 1937. 1: p. 355-359.

[1939Joh] W.A. Johnson and R.F. Mehl, Reaction Kinetics in Processes of Nucleation and Growth. Transactions of the American Institute of Mining and Metallurgical Engineers, 1939. 135: p. 416-442.

[1939Avr] M. Avrami, Kinetics of Phase Change. I. General Theory. Journal of Chemical Physics, 1939. 7: p. 1103-1112.

[1940Avr] M. Avrami, Kinetics of Phase Change. II Transformation‐Time Relations for Random Distribution of Nuclei. The Journal of Chemical Physics, 1940. 8(2): p. 212-224.

[1941Gla] S. Glasstone et al., "The theory of rate processes", New York, NY: McGraw-Hill; 1941.

[1944Hill] H.N. Hill, Determination of stress‐strain relations from the offset yield strength values, Technical Note No. 927, National Advisory Committee for Aeronautics, Washington, D.C., USA, 1944.

[1955Cle] P.J. Clemm and J.C. Fisher, "The influence of grain boundaries on the nucleation of secondary phases", Acta Metall. 3 (1955): 70-73.

[1964Fri] J. Friedel, "Dislocations", 1964, Oxford: Pergamon.

[1974Fle] M.C. Flemings, Solidification Processing, McGraw-Hill, 1974.

[1975Koc] U.F. Kocks et al., "Thermodynamics and Kinetics of Slip", Prog. Mater. Sci., 19 (1975), Oxford: Pergamon Press.

[1975Mug] Y.M. Muggianu et al., "Enthalpies of formation of liquid alloys bismuth-gallium-tin at 723k - choice of an analytical representation of integral and partial thermodynamic functions of mixing for this ternary-system", J. Chim. Phys., 72 (1975): 83-88.

[1979Ans] I. Ansara, "Comparison of methods for thermodynamic calculation of phase diagrams", Int. Metal Rev., 24 (1979): 20-53.

[1979Ger] V. Gerold, "Precipitation hardening. in Dislocations in Solids". 1979. North Holland, Amsterdam.

[1979Sel] C.M. Sellars and J.A. Whiteman, Recrystallization and grain growth in hot rolling. Metal Science, 1979. 13(3-4): p. 187-194.

[1982Ågr] J. Ågren, "Diffusion in phases with several components and sublattices", J. Phys. Chem. Solids, 43 (1982): 421-430.

[1985Ard] A.J.Ardell, "Precipitation hardening", Metall. Trans. A,16 (1985): 2131-2165.

[1985Hil] M. Hillert et al., "A two- sublattice model for molten solutions with different tendency for ionization", Metall. Trans. A, 16 (1985): 261-266.

[1985Kur] W. Kurz and D.J. Fisher, Fundamentals of Solidification, Trans Tech Publications, Switzerland, 1985.

[1985Nes] Nes, E., N. Ryum, and O. Hunderi, On the Zener drag. Acta Metallurgica, 1985. 33(1): p. 11-22.

[1986Roo] A. Roósz, E. Halder, and H.E. Exner. "Numerical calculation of microsegregation in coarsened dendritic microstructures", Mater. Sci. & Tech. 2 (1986): 1149–1155.

[1988Ans] I. Ansara et al., "Thermodynamic modeling of ordered phases in the Ni-Al system", Acta Metall., 36 (1988): 977-982.

[1989Cho] K.C. Chou et al., "A study of Ternary Geometrical Models", Ber. Bunsenges. Phys. Chem., 93 (1989): 735-741.

[1989Yeu] K. S. Yeum et al., "Estimation of the surface tensions of binary liquid alloys", Metall. Trans. B, 20 (1989): 693-703.

[1990Roo] A. Roósz, H.E. Exner, "Numerical modelling of dendritic solidification in aluminium-rich Al-Cu-Mg alloys", Acta Metall. Mater., 38 (1990): 375-380.

[1991Sun] B. Sundman, "Modification of the two-sublattice model for liquids", Calphad, 15 (1991): 109-119.

[1992And] J.O. Andersson et al., "Models for numerical treatment of multicomponent diffusion in simple phases", J. Appl. Phys., 72 (1992): 1350-1355.

[1992Oat] W.A. Oates et al., "The bond-energy model for ordering in a phase with sites of different coordination numbers", Calphad, 16 (1992): 73-78.

[1993Pec] P. Peczak and M. Luton, The effect of nucleation models on dynamic recrystallization I. Homogeneous stored energy distribution. Philosophical Magazine B 1993. 68(1): p. 115-144.

[1994Mor] J.E. Morral et al., "Particle coarsening in binary and multicomponent alloys", Scr. Metall. Mater., 30 (1994): 905-908.

[1994See] S. Seetharaman et al., "Estimation of the viscosities of binary metallic melts using Gibbs energies of mixing". Metall. Trans. B, 25 (1994): 589-595.

[1995Che] S.L. Chen et al., "A generalized quasi-chemical model for ordered multicomponent, multi-sublattice intermetallic compounds with anti-structure defects", Intermetallics, 3 (1995): 233-242.

[1996Ste] I. Steinbach et al., "A phase field concept for multiphase systems", Phys. D Nonlinear Phenom., 94 (1996): 135–147.

[1996Zen] K. Zeng and R. Schmid-Fetzer, "Critical Assessment and Thermodynamic Modeling of the Ti-N System", Z. Metallkd, 87 (1996): 540-554.

[1997Kat] U.R. Kattner, "The thermodynamic modeling of multicomponent phase equilibria", JOM, 49 (1997): 14-15.

[1997Zha] F. Zhang et al., in "Proceedings of the 1997 International Symposium on Liquid Metal Processing and Casting," Eds. A. Mitchell and P. Auburtin, American Vacuum Society, 1997: 35-56.

[1998Des] A.Deschamps et al., "Influence of predeformation and agEing of an Al-Zn-Mg alloy-II. Modeling of precipitation kinetics and yield stress", Acta Mater., 47 (1998): 293-305.

[1998Hil] M. Hillert. Phase Equilibria, "Phase Diagrams and Phase Transformations: Their Thermodynamic Basis". United Kingdom: Cambridge University Press, 1998.

[1999Kim] S.G. Kim et al., Phase-field model for binary alloys, Phys. Rev. E., 60 (1999): 7186–7197.

[2001Myh] O.R Myhr et al., "Modelling of the age hardening behaviour of Al–Mg–Si alloys", Acta Mater., 49 (2001): 65–75.

[2002Che] S.-L.Chen et al., "The PANDAT software package and its applications", Calphad, 26 (2002): 175-188.

[2002Chen] L. Q. Chen, "PHASE-FIELD MODELS FOR MICROSTRUCTURE EVOLUTION", Annu. Rev. Mater. Res. 2002. 32:113–40.

[2003Che] S.-L.Chen et al., "Calculating phase diagrams using PANDAT and PanEngine", JOM, 55 (2003): 48-51.

[2003Esm] S.Esmaeili et al., "A yield strength model for the Al-Mg-Si-Cu alloy AA6111", Acta Mater., 51 (2003): 2243-2257.

[2003Koc] U.F. Kocks and H. Mecking, Physics and phenomenology of strain hardening: the FCC case. Progress in Materials Science, 2003. 48(3): p. 171-274.

[2004Cha] Y.A. Chang et al., "Phase Diagram Calculation: Past, Present and Future", Prog. Mater. Sci., 49 (2004): 313-345.

[2004Che] Q. Chen et al., Quantitative phase field modeling of diffusion- controlled precipitate growth and dissolution in Ti–Al–V, Scr. Mater., 50 (2004): 471–476.

[2004Svo] J. Svoboda et al., "Modelling of kinetics in multi-component multi-phasesystems with spherical precipitatesI: Theory", Mater. Sci. Eng. A, 385 (2004): 166–174.

[2004Fis] F.D.. Fischer and Helmut J. Böhm, "On the role of the transformation eigenstrain in the growth or shrinkage of spheroidal isotropic precipitations", Acta Mater., 53 (2004): 367-374.

[2005Jan] A. Janz, R. Schmid-Fetzer, Impact of ternary parameters, Calphad, 29 (2005) 37-39. doi:10.1016/j.calphad.2005.01.003.

[2006Koz] E. Kozeschnik et al., "Shape factors in modeling of precipitation", Mater. Sci. Eng. A, 441 (2006): 68–72.

[2006Iva] Ivasishin, O.M. et al., A 3-D Monte-Carlo (Potts) Model for Recrystallization and Grain Growth in Polycrystalline Materials. Mater. Sci. Eng. A 2006, 433, 216–232.

[2006Ste] I. Steinbach et al., "Multi phase field model for solid state transformation with elastic strain", Phys. D Nonlinear Phenom., 217 (2006): 153–160.

[2007Luk] H. Lukas et al., "Computational Thermodynamics: The Calphad Method", Cambridge University Press, 2007.

[2008Lar] H. Larsson et al., "On the numerical simulation of diffusion-controlled reactions under local equilibrium conditions", Acta Mater., 56 (2008): 3754-3760.

[2008Svo] J. Svoboda et al., "A model for evolution of shape changing precipitates in multicomponent systems", Acta Mater., 56 (2008): 4896–4904.

[2009Cao] W.S. Cao et al., "PANDAT software with PanEngine, PanOptimizer and PanPrecipitation for multi- component phase diagram calculation and materials property simulation", Calphad, 33 (2009): 328-342.

[2009Lar] H. Larsson, L. Höglund, Multiphase diffusion simulations in 1D using the DICTRA homogenization model, Calphad, 33 (2009) 495-501.

[2009Pay] Payton, E.J., Characterization and Modeling of Grain Coarsening in Powder Metallurgical Nickel-Based Superalloys, in Materials Science and Engineering. 2009, Ohio State University.

[2009Sun] B. Sundman, et al., "An assessment of the entire Al-Fe system including D03 ordering", Acta Mater., 57 (2009): 2896-2908.

[2010Wan] Y. Wang and J. Li, "Phase field modeling of defects and deformation", Acta Mater., 58 (2010): 1212–1235.

[2011Guo] W. Guo et al., "An analytical study of the static state of multi-junctions in a multi-phase field model", Phys. D Nonlinear Phenom., 240 (2011): 382–388.

[2013Kha] A.G. Khachaturyan, "Theory of structural transformations in solids", Courier Corporation, 2013.

[2015Che] S.L. Chen et al., "Calculation of 2D and 3D Phase Diagrams", JOM, 67 (8) (2015): 1876-1880.

[2015Gal] E.I. Galindo-Nava et al, On the prediction of the yield stress of unimodal and multimodal γ′ Nickel-base superalloys, Acta Mater., 98 (2015): 377-390.

[2015Kou] S. Kou, "A criterion for cracking during solidification", Acta Mater., 88 (2015): 366-374.

[2016Lin] Y.C. Lin et al., A novel unified dislocation density-based model for hot deformation behavior of a nickel-based superalloy under dynamic recrystallization conditions, Appl. Phys. A, 122 (2016): 805

[2018Shi] R. Shi et al., "Growth behavior of γ'/γ'' coprecipitates in Ni-Base superalloys", Acta Mater., 164 (2019): 220–236.

[2019Shi] Rongpei Shi et al., Integrated Simulation Framework for Additively Manufactured Ti-6Al-4V: Melt Pool Dynamics, Microstructure, Solid-State Phase Transformation, and Microelastic Response. JOM, 71 (2019): 3640.

[2020Cai] Cai, Yun, et al., "Phase field modeling of discontinuous dynamic recrystallization in hot deformation of magnesium alloys." International Journal of Plasticity 133 (2020): 102773.

[2021Mor] J.E. Morral, S. Chen, Stability of High Entropy Alloys to Spinodal Decomposition, J. Phase Equilib. Diffus., 42 (2021): 673–695

[2022Kon] S.R. Koneru, K. Kadirvel, Y. Wang, High-Throughput Design of Multi-Principal Element Alloys with Spinodal Decomposition Assisted Microstructures, J. Phase Equilib. Diffus., 43 (2022): 753-763.

[2022Che] S. Chen, R. Schmid-Fetzer, J.E. Morral, Calculation of Critical Points, J. Phase Equilib. Diffus., 43 (2022): 968-978

[2019Tao] Z. Tao, X.-Q. Wang, M.K. Hassan, T.-Y. Song, L.-A. Xie, Behaviour of three types of stainless steel after exposure to elevated temperatures, J. Constr. Steel Res., 152 (2019) 296-311.

[2023Yan] J.-B. Yan, B. Zhang, Y. An, Stress-strain models and numerical simulations of high-strength steel Q690E and Q960E at low temperatures, J. Constr. Steel Res., 207 (2023) 107957.