

An attempt to explain Mats Minimizing Algorithm (used in TC and OC)

Bo Sundman

visiting professor at CSU, Changsha, China and professor emeritus KTH, Sweden

April 4, 2017

Content

This presentation will show

- ▶ Some background and features
- ▶ What kind of external conditions we can handle
- ▶ The formulation of the problem
- ▶ Introducing Lagrangian multipliers
- ▶ Step 1, inverting the phase matrix
- ▶ Step 2, solving the equilibrium matrix
- ▶ Changing the set of stable phases and the convergence criteria
- ▶ How to know if we have a global equilibrium?
- ▶ Snapshots of iteration results for two cases
- ▶ Conclusions and references

Background

The algorithm was first published by Mats Hillert in 1981. Bo Jansson described the implementation in TC in his PhD thesis 1984. The OC implementation was published 2015.

It is based on a simpler algorithm for ideal gases published 1975.

Background

The algorithm was first published by Mats Hillert in 1981. Bo Jansson described the implementation in TC in his PhD thesis 1984. The OC implementation was published 2015.

It is based on a simpler algorithm for ideal gases published 1975.

The algorithm can minimize the Gibbs energy for multicomponent systems with many phases, $\alpha, \beta \dots$ with different kinds of models using many different kinds of external conditions.

Background

The algorithm was first published by Mats Hillert in 1981. Bo Jansson described the implementation in TC in his PhD thesis 1984. The OC implementation was published 2015.

It is based on a simpler algorithm for ideal gases published 1975.

The algorithm can minimize the Gibbs energy for multicomponent systems with many phases, $\alpha, \beta \dots$ with different kinds of models using many different kinds of external conditions.

The algorithm does not require the chemical potentials but it requires that we can calculate many partial derivatives of $G_M^\alpha(T, P, y_i)$, where y_i is a constituent variable:

- ▶ $\frac{\partial G_M^\alpha}{\partial y_i}$
- ▶ $\frac{\partial G_M^\alpha}{\partial T}$, $\frac{\partial^2 G_M}{\partial T^2}$ and $\frac{\partial^2 G_M}{\partial T \partial y_i}$ if T is unknown.
- ▶ $\frac{\partial G_M^\alpha}{\partial P}$, $\frac{\partial^2 G_M}{\partial P^2}$ and $\frac{\partial^2 G_M}{\partial P \partial y_i}$ if P is unknown.

External conditions

The algorithm also require at least approximate values of the second derivatives $\frac{\partial^2 G_M^\alpha}{\partial y_i \partial y_j}$. For an ideal gas this is simply equal to $\frac{RT}{y_i} \delta_{ij}$.

External conditions

The algorithm also require at least approximate values of the second derivatives $\frac{\partial^2 G_M^\alpha}{\partial y_i \partial y_j}$. For an ideal gas this is simply equal to $\frac{RT}{y_i} \delta_{ij}$.

The algorithm can handle multicomponent systems with very flexible set of conditions. The simplest conditions are when T, P and the overall composition, N_A is known. Other simple conditions are when a mole fraction, chemical potential, enthalpy or volume is known.

External conditions

The algorithm also require at least approximate values of the second derivatives $\frac{\partial^2 G_M^\alpha}{\partial y_i \partial y_j}$. For an ideal gas this is simply equal to $\frac{RT}{y_i} \delta_{ij}$.

The algorithm can handle multicomponent systems with very flexible set of conditions. The simplest conditions are when T, P and the overall composition, N_A is known. Other simple conditions are when a mole fraction, chemical potential, enthalpy or volume is known.

With Mats algorithm we can also specify as a condition that one or more of phases must be stable.

External conditions

The algorithm also require at least approximate values of the second derivatives $\frac{\partial^2 G_M^\alpha}{\partial y_i \partial y_j}$. For an ideal gas this is simply equal to $\frac{RT}{y_i} \delta_{ij}$.

The algorithm can handle multicomponent systems with very flexible set of conditions. The simplest conditions are when T, P and the overall composition, N_A is known. Other simple conditions are when a mole fraction, chemical potential, enthalpy or volume is known.

With Mats algorithm we can also specify as a condition that one or more of phases must be stable.

More special is that we can have as condition the composition of a phase, x_A^α , or even a constituent fraction, y_i^α , of a phase.

External conditions

The algorithm also require at least approximate values of the second derivatives $\frac{\partial^2 G_M^\alpha}{\partial y_i \partial y_j}$. For an ideal gas this is simply equal to $\frac{RT}{y_i} \delta_{ij}$.

The algorithm can handle multicomponent systems with very flexible set of conditions. The simplest conditions are when T, P and the overall composition, N_A is known. Other simple conditions are when a mole fraction, chemical potential, enthalpy or volume is known.

With Mats algorithm we can also specify as a condition that one or more of phases must be stable.

More special is that we can have as condition the composition of a phase, x_A^α , or even a constituent fraction, y_i^α , of a phase.

Even more special is that we can have a condition that the composition of two phase should have a relation, For example $x_A^{\text{liq}} - x_A^\alpha = 0$, to calculate the congruent melting of the α phase.

External conditions

The algorithm also require at least approximate values of the second derivatives $\frac{\partial^2 G_M^\alpha}{\partial y_i \partial y_j}$. For an ideal gas this is simply equal to $\frac{RT}{y_i} \delta_{ij}$.

The algorithm can handle multicomponent systems with very flexible set of conditions. The simplest conditions are when T, P and the overall composition, N_A is known. Other simple conditions are when a mole fraction, chemical potential, enthalpy or volume is known.

With Mats algorithm we can also specify as a condition that one or more of phases must be stable.

More special is that we can have as condition the composition of a phase, x_A^α , or even a constituent fraction, y_i^α , of a phase.

Even more special is that we can have a condition that the composition of two phase should have a relation, For example $x_A^{\text{liq}} - x_A^\alpha = 0$, to calculate the congruent melting of the α phase. In fact any linear relation between state variables of the same kind can be used as a condition.

External conditions

The algorithm also require at least approximate values of the second derivatives $\frac{\partial^2 G_M^\alpha}{\partial y_i \partial y_j}$. For an ideal gas this is simply equal to $\frac{RT}{y_i} \delta_{ij}$.

The algorithm can handle multicomponent systems with very flexible set of conditions. The simplest conditions are when T, P and the overall composition, N_A is known. Other simple conditions are when a mole fraction, chemical potential, enthalpy or volume is known.

With Mats algorithm we can also specify as a condition that one or more of phases must be stable.

More special is that we can have as condition the composition of a phase, x_A^α , or even a constituent fraction, y_i^α , of a phase.

Even more special is that we can have a condition that the composition of two phase should have a relation, For example $x_A^{\text{liq}} - x_A^\alpha = 0$, to calculate the congruent melting of the α phase. In fact any linear relation between state variables of the same kind can be used as a condition.

But we cannot set conditions to calculate the top of a miscibility gap.

External conditions

The algorithm also require at least approximate values of the second derivatives $\frac{\partial^2 G_M^\alpha}{\partial y_i \partial y_j}$. For an ideal gas this is simply equal to $\frac{RT}{y_i} \delta_{ij}$.

The algorithm can handle multicomponent systems with very flexible set of conditions. The simplest conditions are when T, P and the overall composition, N_A is known. Other simple conditions are when a mole fraction, chemical potential, enthalpy or volume is known.

With Mats algorithm we can also specify as a condition that one or more of phases must be stable.

More special is that we can have as condition the composition of a phase, x_A^α , or even a constituent fraction, y_i^α , of a phase.

Even more special is that we can have a condition that the composition of two phase should have a relation, For example $x_A^{\text{liq}} - x_A^\alpha = 0$, to calculate the congruent melting of the α phase. In fact any linear relation between state variables of the same kind can be used as a condition.

A very important feature is that the algorithm can **change the set of stable phases during the iterations** to find the equilibrium.

Calculating the equilibrium state

The equilibrium at fixed temperature, pressure and composition is given by a minimum in the total Gibbs energy of the system. The thermodynamic models describe the molar Gibbs energy, G_M^α and the total energy for the system is:

$$G(T, P, N_A) = \sum_{\alpha} N^{\alpha} G_M^{\alpha}(T, P, y_{is}) \quad (1)$$

where N_A is the amount of component A, N^{α} is the amount of the α phase and y_{is} is the fraction of constituent i in sublattice s in α .

Calculating the equilibrium state

The equilibrium at fixed temperature, pressure and composition is given by a minimum in the total Gibbs energy of the system. The thermodynamic models describe the molar Gibbs energy, G_M^α and the total energy for the system is:

$$G(T, P, N_A) = \sum_{\alpha} N^{\alpha} G_M^{\alpha}(T, P, y_{is}) \quad (1)$$

where N_A is the amount of component A, N^{α} is the amount of the α phase and y_{is} is the fraction of constituent i in sublattice s in α .

If we have other conditions like fixed volume, chemical potentials or fixed phases we can modify the Gibbs energy by the use of Lagrangian multipliers.

Calculating the equilibrium state

The equilibrium at fixed temperature, pressure and composition is given by a minimum in the total Gibbs energy of the system. The thermodynamic models describe the molar Gibbs energy, G_M^α and the total energy for the system is:

$$G(T, P, N_A) = \sum_{\alpha} N^{\alpha} G_M^{\alpha}(T, P, y_{is}) \quad (1)$$

where N_A is the amount of component A, N^{α} is the amount of the α phase and y_{is} is the fraction of constituent i in sublattice s in α .

If we have other conditions like fixed volume, chemical potentials or fixed phases we can modify the Gibbs energy by the use of Lagrangian multipliers.

As we model phases with sublattices and vacancies we use the Gibbs energy per mole of formula units, denoted G_M , whereas the Gibbs energy per mole of atoms is denoted G_m .

Mass balance equation

We must always be able to calculate the amount of components in the system and we define an amount M_A for the component A in α . For a CEF model this can be calculated as:

$$M_A^\alpha = \sum_s a_s \sum_i b_{iA} y_{is} \quad (2)$$

where a_s is the number of sites on sublattice s and b_{iA} is the stoichiometric factor of component A in constituent i .

Mass balance equation

We must always be able to calculate the amount of components in the system and we define an amount M_A for the component A in α . For a CEF model this can be calculated as:

$$M_A^\alpha = \sum_s a_s \sum_i b_{iA} y_{is} \quad (2)$$

where a_s is the number of sites on sublattice s and b_{iA} is the stoichiometric factor of component A in constituent i .

From the amounts M we can calculate the mole fractions:

$$x_A^\alpha = \frac{M_A^\alpha}{\sum_B M_B^\alpha} \quad (3)$$

and the total amount of component A is

$$N_A = \sum_\alpha N^\alpha M_A^\alpha \quad (4)$$

The advantage using M rather than x will be clear when we calculate derivatives as we do not have a denominator as for x .

Single equilibrium calculation, Lagrangian method

The algorithm for minimization that is implemented in OC was proposed by Hillert (1981). It uses Lagrange multipliers to take care of both external and internal constraints. For constant composition and fixed T and P an example is:

$$L = G + \sum_A (N_A - \sum_{\alpha} N^{\alpha} M_A^{\alpha}) \mu_A + \sum_{\alpha} \sum_s \eta_s^{\alpha} (\sum_i y_{is}^{\alpha} - 1) \quad (5)$$

where N_A is the total amount of component A, M_A^{α} is the amount of component A and y_{is}^{α} is the fraction of constituent i on sublattice s in phase α . μ_A and η_s^{α} are Lagrange multipliers.

Single equilibrium calculation, Lagrangian method

The algorithm for minimization that is implemented in OC was proposed by Hillert (1981). It uses Lagrange multipliers to take care of both external and internal constraints. For constant composition and fixed T and P an example is:

$$L = G + \sum_A (N_A - \sum_{\alpha} N^{\alpha} M_A^{\alpha}) \mu_A + \sum_{\alpha} \sum_s \eta_s^{\alpha} (\sum_i y_{is}^{\alpha} - 1) \quad (5)$$

where N_A is the total amount of component A, M_A^{α} is the amount of component A and y_{is}^{α} is the fraction of constituent i on sublattice s in phase α . μ_A and η_s^{α} are Lagrange multipliers.

The first constraint is the massbalance and the second that the sum of constituent fractions in each sublattice of α is unity. There can also be constraints due to electrical neutrality.

Single equilibrium calculation, Lagrangian method

The algorithm for minimization that is implemented in OC was proposed by Hillert (1981). It uses Lagrange multipliers to take care of both external and internal constraints. For constant composition and fixed T and P an example is:

$$L = G + \sum_A (N_A - \sum_{\alpha} N^{\alpha} M_A^{\alpha}) \mu_A + \sum_{\alpha} \sum_s \eta_s^{\alpha} (\sum_i y_{is}^{\alpha} - 1) \quad (5)$$

where N_A is the total amount of component A, M_A^{α} is the amount of component A and y_{is}^{α} is the fraction of constituent i on sublattice s in phase α . μ_A and η_s^{α} are Lagrange multipliers.

At equilibrium all partial derivatives must be zero and from the derivative of the Lagrange function with respect to N^{α} we get:

$$\frac{\partial L}{\partial N^{\alpha}} = G_M^{\alpha} - \sum_A M_A^{\alpha} \mu_A = 0 \quad (6)$$

as $G = \sum_{\alpha} N^{\alpha} G_M^{\alpha}$. The multiplier μ_A is thus the chemical potential.

Single equilibrium calculation module, step 1A

We need an equation to vary the constituent fractions of the phases to find the equilibrium and start with the derivative of L with respect to a constituent fractions which is zero at the minimum.

$$\frac{\partial L}{\partial y_{is}^{\alpha}} = \lambda^{\alpha} \frac{\partial G_M^{\alpha}}{\partial y_{is}^{\alpha}} - \lambda^{\alpha} \sum_A \mu_A \frac{\partial M_A^{\alpha}}{\partial y_{is}^{\alpha}} - \eta_s^{\alpha} = 0 \quad (7)$$

Single equilibrium calculation module, step 1A

We need an equation to vary the constituent fractions of the phases to find the equilibrium and start with the derivative of L with respect to a constituent fractions which is zero at the minimum.

$$\frac{\partial L}{\partial y_{is}^{\alpha}} = \lambda^{\alpha} \frac{\partial G_M^{\alpha}}{\partial y_{is}^{\alpha}} - \lambda^{\alpha} \sum_A \mu_A \frac{\partial M_A^{\alpha}}{\partial y_{is}^{\alpha}} - \eta_s^{\alpha} = 0 \quad (7)$$

We expand the first term, $\frac{\partial G_M^{\alpha}}{\partial y_{is}^{\alpha}}$ in a Taylor series for the variables T, P and constitution:

$$\frac{\partial G_M^{\alpha}}{\partial y_{is}^{\alpha}} = \left(\frac{\partial G_M^{\alpha}}{\partial y_{is}^{\alpha}} \right) + \left(\frac{\partial^2 G_M^{\alpha}}{\partial y_{is}^{\alpha} \partial T} \right) dT + \left(\frac{\partial^2 G_M^{\alpha}}{\partial y_{is}^{\alpha} \partial P} \right) dP + \sum_j \sum_t \left(\frac{\partial^2 G_M^{\alpha}}{\partial y_{is}^{\alpha} \partial y_{jt}^{\alpha}} \right) dy_{jt}^{\alpha} \quad (8)$$

The terms on the right hand side can be calculated from the current values of T, P and constitution of the phase.

Single equilibrium calculation module, step 1A

We need an equation to vary the constituent fractions of the phases to find the equilibrium and start with the derivative of L with respect to a constituent fractions which is zero at the minimum.

$$\frac{\partial L}{\partial y_{is}^{\alpha}} = \lambda^{\alpha} \frac{\partial G_M^{\alpha}}{\partial y_{is}^{\alpha}} - \lambda^{\alpha} \sum_A \mu_A \frac{\partial M_A^{\alpha}}{\partial y_{is}^{\alpha}} - \eta_s^{\alpha} = 0 \quad (7)$$

We expand the first term, $\frac{\partial G_M^{\alpha}}{\partial y_{is}^{\alpha}}$ in a Taylor series for the variables T, P and constitution:

$$\frac{\partial G_M^{\alpha}}{\partial y_{is}^{\alpha}} = \left(\frac{\partial G_M^{\alpha}}{\partial y_{is}^{\alpha}} \right) + \left(\frac{\partial^2 G_M^{\alpha}}{\partial y_{is}^{\alpha} \partial T} \right) dT + \left(\frac{\partial^2 G_M^{\alpha}}{\partial y_{is}^{\alpha} \partial P} \right) dP + \sum_j \sum_t \left(\frac{\partial^2 G_M^{\alpha}}{\partial y_{is}^{\alpha} \partial y_{jt}^{\alpha}} \right) dy_{jt}^{\alpha} \quad (8)$$

The terms on the right hand side can be calculated from the current values of T, P and constitution of the phase.

and the term on the left hand side represent a **linear extrapolation**.

Single equilibrium calculation module, step 1A

We need an equation to vary the constituent fractions of the phases to find the equilibrium and start with the derivative of L with respect to a constituent fractions which is zero at the minimum.

$$\frac{\partial L}{\partial y_{is}^{\alpha}} = \lambda^{\alpha} \frac{\partial G_M^{\alpha}}{\partial y_{is}^{\alpha}} - \lambda^{\alpha} \sum_A \mu_A \frac{\partial M_A^{\alpha}}{\partial y_{is}^{\alpha}} - \eta_s^{\alpha} = 0 \quad (7)$$

We expand the first term, $\frac{\partial G_M^{\alpha}}{\partial y_{is}^{\alpha}}$ in a Taylor series for the variables T, P and constitution:

$$\frac{\partial G_M^{\alpha}}{\partial y_{is}^{\alpha}} = \left(\frac{\partial G_M^{\alpha}}{\partial y_{is}^{\alpha}} \right) + \left(\frac{\partial^2 G_M^{\alpha}}{\partial y_{is}^{\alpha} \partial T} \right) dT + \left(\frac{\partial^2 G_M^{\alpha}}{\partial y_{is}^{\alpha} \partial P} \right) dP + \sum_j \sum_t \left(\frac{\partial^2 G_M^{\alpha}}{\partial y_{is}^{\alpha} \partial y_{jt}^{\alpha}} \right) dy_{jt}^{\alpha} \quad (8)$$

The terms on the right hand side can be calculated from the current values of T, P and constitution of the phase.

This is the big secret trick of the algorithm!

Single equilibrium calculation module, step 1B

Inserting this Taylor expansion in eq. 7 gives after some rearrangements and using finite differences:

$$\sum_j \sum_t \left(\frac{\partial^2 G_M^\alpha}{\partial y_{is}^\alpha \partial y_{jt}^\alpha} \right) \Delta y_{jt}^\alpha - \frac{\eta_s^\alpha}{N^\alpha} =$$
$$\sum_A \left(\mu_A \frac{\partial M_A^\alpha}{\partial y_{is}^\alpha} - \left(\frac{\partial G_M^\alpha}{\partial y_{is}^\alpha} \right) - \left(\frac{\partial^2 G_M^\alpha}{\partial y_{is}^\alpha \partial T} \right) \Delta T - \left(\frac{\partial^2 G_M^\alpha}{\partial y_{is}^\alpha \partial P} \right) \Delta P \right) \quad (9)$$

This is a system of linear equations for the change in the constitution of a phase α , expressing all Δy_{jt}^α as functions of the global potential variables $\Delta T, \Delta P$ and μ_A .

Single equilibrium calculation module, step 1B

Inserting this Taylor expansion in eq. 7 gives after some rearrangements and using finite differences:

$$\sum_j \sum_t \left(\frac{\partial^2 G_M^\alpha}{\partial y_{is}^\alpha \partial y_{jt}^\alpha} \right) \Delta y_{jt}^\alpha - \frac{\eta_s^\alpha}{N^\alpha} =$$
$$\sum_A \left(\mu_A \frac{\partial M_A^\alpha}{\partial y_{is}^\alpha} - \left(\frac{\partial G_M^\alpha}{\partial y_{is}^\alpha} \right) - \left(\frac{\partial^2 G_M^\alpha}{\partial y_{is}^\alpha \partial T} \right) \Delta T - \left(\frac{\partial^2 G_M^\alpha}{\partial y_{is}^\alpha \partial P} \right) \Delta P \right) \quad (9)$$

This is a system of linear equations for the change in the constitution of a phase α , expressing all Δy_{jt}^α as functions of the global potential variables $\Delta T, \Delta P$ and μ_A .

The only problem is to implement all these partial derivatives in the software!

The phase matrix

The system of equations is a matrix relating the constituent fractions of a phase to the potentials T and μ_A (for constant P):

$$\begin{pmatrix} \frac{\partial^2 G_M}{\partial y_1^2} & \frac{\partial^2 G_M}{\partial y_1 \partial y_2} & \dots & -1 & \dots \\ \frac{\partial^2 G_M}{\partial y_1 \partial y_2} & \frac{\partial^2 G_M}{\partial y_2^2} & \dots & -1 & \dots \\ \vdots & \vdots & & & \\ -1 & -1 & \dots & 0 & \dots \\ \vdots & \vdots & & & \end{pmatrix} \begin{pmatrix} \Delta y_1 \\ \Delta y_2 \\ \vdots \\ \frac{\eta_s}{N} \\ \vdots \end{pmatrix} = \begin{pmatrix} \sum_A \mu_A \frac{\partial M_A}{\partial y_1} - \frac{\partial G_M}{\partial y_1} - \frac{\partial^2 G_M}{\partial y_1 \partial T} \Delta T \\ \sum_A \mu_A \frac{\partial M_A}{\partial y_2} - \frac{\partial G_M}{\partial y_2} - \frac{\partial^2 G_M}{\partial y_2 \partial T} \Delta T \\ \vdots \\ 0 \\ \vdots \end{pmatrix}$$

We cannot solve this at present as the values of ΔT and μ_A are unknown.

The phase matrix

The system of equations is a matrix relating the constituent fractions of a phase to the potentials T and μ_A (for constant P):

$$\begin{pmatrix} \frac{\partial^2 G_M}{\partial y_1^2} & \frac{\partial^2 G_M}{\partial y_1 \partial y_2} & \dots & -1 & \dots \\ \frac{\partial^2 G_M}{\partial y_1 \partial y_2} & \frac{\partial^2 G_M}{\partial y_2^2} & \dots & -1 & \dots \\ \vdots & \vdots & & & \\ -1 & -1 & \dots & 0 & \dots \\ \vdots & \vdots & & & \end{pmatrix} \begin{pmatrix} \Delta y_1 \\ \Delta y_2 \\ \vdots \\ \frac{\eta s}{N} \\ \vdots \end{pmatrix} = \begin{pmatrix} \sum_A \mu_A \frac{\partial M_A}{\partial y_1} - \frac{\partial G_M}{\partial y_1} - \frac{\partial^2 G_M}{\partial y_1 \partial T} \Delta T \\ \sum_A \mu_A \frac{\partial M_A}{\partial y_2} - \frac{\partial G_M}{\partial y_2} - \frac{\partial^2 G_M}{\partial y_2 \partial T} \Delta T \\ \vdots \\ 0 \\ \vdots \end{pmatrix}$$

We cannot solve this at present as the values of ΔT and μ_A are unknown.

But we can invert the matrix on the left hand side and use this to formulate explicit equations for Δy_i^α as a function of T, P and μ_A .

Single equilibrium calculation module, step 1C

With the inverted “phase matrix” we can formulate Δy_i^α as functions of the potentials (here at constant P):

$$\Delta y_i^\alpha = \sum_A \sum_j \frac{\partial M_A^\alpha}{\partial y_j^\alpha} e_{ij}^\alpha \mu_A - \sum_j \frac{\partial G_M^\alpha}{\partial y_j^\alpha} e_{ij}^\alpha - \sum_j \frac{\partial^2 G_M^\alpha}{\partial T \partial y_j^\alpha} e_{ij}^\alpha \Delta T \quad (10)$$

where e_{ij}^α are the terms of the inverted phase matrix. These depend mainly of the second derivatives of the Gibbs energy with respect to the constituent fractions:

$$e_{ij}^\alpha \approx \left(\frac{\partial^2 G_M^\alpha}{\partial y_i^\alpha \partial y_j^\alpha} \right)^{-1} \quad (11)$$

Single equilibrium calculation module, step 1C

With the inverted “phase matrix” we can formulate Δy_i^α as functions of the potentials (here at constant P):

$$\Delta y_i^\alpha = \sum_A \sum_j \frac{\partial M_A^\alpha}{\partial y_j^\alpha} e_{ij}^\alpha \mu_A - \sum_j \frac{\partial G_M^\alpha}{\partial y_j^\alpha} e_{ij}^\alpha - \sum_j \frac{\partial^2 G_M^\alpha}{\partial T \partial y_j^\alpha} e_{ij}^\alpha \Delta T \quad (10)$$

where e_{ij}^α are the terms of the inverted phase matrix. These depend mainly of the second derivatives of the Gibbs energy with respect to the constituent fractions:

$$e_{ij}^\alpha \approx \left(\frac{\partial^2 G_M^\alpha}{\partial y_i^\alpha \partial y_j^\alpha} \right)^{-1} \quad (11)$$

In TC and OC the partial derivatives are calculated analytically.

Single equilibrium calculation module, step 1C

With the inverted “phase matrix” we can formulate Δy_i^α as functions of the potentials (here at constant P):

$$\Delta y_i^\alpha = \sum_A \sum_j \frac{\partial M_A^\alpha}{\partial y_j^\alpha} e_{ij}^\alpha \mu_A - \sum_j \frac{\partial G_M^\alpha}{\partial y_j^\alpha} e_{ij}^\alpha - \sum_j \frac{\partial^2 G_M^\alpha}{\partial T \partial y_j^\alpha} e_{ij}^\alpha \Delta T \quad (10)$$

where e_{ij}^α are the terms of the inverted phase matrix. These depend mainly of the second derivatives of the Gibbs energy with respect to the constituent fractions:

$$e_{ij}^\alpha \approx \left(\frac{\partial^2 G_M^\alpha}{\partial y_i^\alpha \partial y_j^\alpha} \right)^{-1} \quad (11)$$

In TC and OC the partial derivatives are calculated analytically.

In OC the inverted phase matrix is calculated in the subroutine meq_onephase and the derivatives in the GTP model package

Single equilibrium calculation module, step 1C

With the inverted “phase matrix” we can formulate Δy_i^α as functions of the potentials (here at constant P):

$$\Delta y_i^\alpha = \sum_A \sum_j \frac{\partial M_A^\alpha}{\partial y_j^\alpha} e_{ij}^\alpha \mu_A - \sum_j \frac{\partial G_M^\alpha}{\partial y_j^\alpha} e_{ij}^\alpha - \sum_j \frac{\partial^2 G_M^\alpha}{\partial T \partial y_j^\alpha} e_{ij}^\alpha \Delta T \quad (10)$$

where e_{ij}^α are the terms of the inverted phase matrix. These depend mainly of the second derivatives of the Gibbs energy with respect to the constituent fractions:

$$e_{ij}^\alpha \approx \left(\frac{\partial^2 G_M^\alpha}{\partial y_i^\alpha \partial y_j^\alpha} \right)^{-1} \quad (11)$$

In TC and OC the partial derivatives are calculated analytically.

To find the values of μ_A and ΔT we must apply the external conditions in a second step where we will use eq. 10 many times to calculate the coefficients in this “equilibrium matrix”.

Single equilibrium calculation module, step 2A

In the second step we formulate the equilibrium matrix, a system of linear equations to obtain new values of the unknown potentials and the amounts of the stable phases,

Single equilibrium calculation module, step 2A

In the second step we formulate the equilibrium matrix, a system of linear equations to obtain new values of the unknown potentials and the amounts of the stable phases, There is one equation for each stable phase to ensure they are on the same Gibbs energy plane:

$$\sum_A M_A^\alpha \mu_A = G_M^\alpha \quad (12)$$

Single equilibrium calculation module, step 2A

In the second step we formulate the equilibrium matrix, a system of linear equations to obtain new values of the unknown potentials and the amounts of the stable phases, There is one equation for each stable phase to ensure they are on the same Gibbs energy plane:

$$\sum_A M_A^\alpha \mu_A = G_M^\alpha \quad (12)$$

If a chemical potential is known that term is moved to the right hand side.

Single equilibrium calculation module, step 2A

In the second step we formulate the equilibrium matrix, a system of linear equations to obtain new values of the unknown potentials and the amounts of the stable phases, There is one equation for each stable phase to ensure they are on the same Gibbs energy plane:

$$\sum_A M_A^\alpha \mu_A = G_M^\alpha \quad (12)$$

If a chemical potential is known that term is moved to the right hand side.

For each condition on an extensive variable we obtain an equation, for example $N_A = \text{fix} = \tilde{N}_A$ i.e.: $dN_A = \Delta N_A = N_A - \tilde{N}_A = 0$

Single equilibrium calculation module, step 2A

In the second step we formulate the equilibrium matrix, a system of linear equations to obtain new values of the unknown potentials and the amounts of the stable phases, There is one equation for each stable phase to ensure they are on the same Gibbs energy plane:

$$\sum_A M_A^\alpha \mu_A = G_M^\alpha \quad (12)$$

If a chemical potential is known that term is moved to the right hand side.

For each condition on an extensive variable we obtain an equation, for example $N_A = \text{fix} = \tilde{N}_A$ i.e.: $dN_A = \Delta N_A = N_A - \tilde{N}_A = 0$

All terms for ΔN_A (or any other extensive variable) contains differentials like Δy_{is}^α for the stable phases and these can be converted to the potential variables using eq. 10.

Single equilibrium calculation module, step 2A

We can differentiate N_A as follows:

$$N_A = \sum_{\alpha} N^{\alpha} M_A^{\alpha} \quad (13)$$

$$\Delta N_A = \sum_{\alpha} N^{\alpha} \sum_{is} \frac{\partial M_A^{\alpha}}{\partial y_{is}} \Delta y_{is} + \sum_{\alpha} M_A^{\alpha} \Delta N^{\alpha} \quad (14)$$

where Δy_{is} can be replaced using the inverted phase matrix eq. 10

Single equilibrium calculation module, step 2A

We can differentiate N_A as follows:

$$N_A = \sum_{\alpha} N^{\alpha} M_A^{\alpha} \quad (13)$$

$$\Delta N_A = \sum_{\alpha} N^{\alpha} \sum_{is} \frac{\partial M_A^{\alpha}}{\partial y_{is}} \Delta y_{is} + \sum_{\alpha} M_A^{\alpha} \Delta N^{\alpha} \quad (14)$$

where Δy_{is} can be replaced using the inverted phase matrix eq. 10 in order to obtain an equation for each component (at constant P):

$$\sum_{\alpha} N^{\alpha} \sum_i \frac{\partial M_A^{\alpha}}{\partial y_i} \sum_j \sum_B \frac{\partial M_B^{\alpha}}{\partial y_j} e_{ij}^{\alpha} \mu_B + \sum_{\alpha} M_A^{\alpha} \Delta N^{\alpha} - \sum_{\alpha} N^{\alpha} \sum_i \frac{\partial M_A^{\alpha}}{\partial y_i} \sum_j \frac{\partial^2 G_M^{\alpha}}{\partial y_j \partial T} e_{ij}^{\alpha} \Delta T = \sum_{\alpha} N^{\alpha} \sum_i \sum_j \frac{\partial M_A^{\alpha}}{\partial y_i} \frac{\partial G_M^{\alpha}}{\partial y_j} e_{ij}^{\alpha}$$

where we use the matrix elements e_{ij} in order to have as unknown variables only the potentials and the change in phase amounts, ΔN^{α} .

Single equilibrium calculation module, step 2A

We can differentiate N_A as follows:

$$N_A = \sum_{\alpha} N^{\alpha} M_A^{\alpha} \quad (13)$$

$$\Delta N_A = \sum_{\alpha} N^{\alpha} \sum_{is} \frac{\partial M_A^{\alpha}}{\partial y_{is}} \Delta y_{is} + \sum_{\alpha} M_A^{\alpha} \Delta N^{\alpha} \quad (14)$$

where Δy_{is} can be replaced using the inverted phase matrix eq. 10 in order to obtain an equation for each component (at constant P):

$$\sum_{\alpha} N^{\alpha} \sum_i \frac{\partial M_A^{\alpha}}{\partial y_i} \sum_j \sum_B \frac{\partial M_B^{\alpha}}{\partial y_j} e_{ij}^{\alpha} \mu_B + \sum_{\alpha} M_A^{\alpha} \Delta N^{\alpha} - \sum_{\alpha} N^{\alpha} \sum_i \frac{\partial M_A^{\alpha}}{\partial y_i} \sum_j \frac{\partial^2 G_M^{\alpha}}{\partial y_j \partial T} e_{ij}^{\alpha} \Delta T = \sum_{\alpha} N^{\alpha} \sum_i \sum_j \frac{\partial M_A^{\alpha}}{\partial y_i} \frac{\partial G_M^{\alpha}}{\partial y_j} e_{ij}^{\alpha}$$

where we use the matrix elements e_{ij} in order to have as unknown variables only the potentials and the change in phase amounts, ΔN^{α} . The unknown variables are marked with blue.

Single equilibrium calculation module, step 2A

We can differentiate N_A as follows:

$$N_A = \sum_{\alpha} N^{\alpha} M_A^{\alpha} \quad (13)$$

$$\Delta N_A = \sum_{\alpha} N^{\alpha} \sum_{is} \frac{\partial M_A^{\alpha}}{\partial y_{is}} \Delta y_{is} + \sum_{\alpha} M_A^{\alpha} \Delta N^{\alpha} \quad (14)$$

where Δy_{is} can be replaced using the inverted phase matrix eq. 10 in order to obtain an equation for each component (at constant P):

$$\begin{aligned} \sum_{\alpha} N^{\alpha} \sum_i \frac{\partial M_A^{\alpha}}{\partial y_i} \sum_j \sum_B \frac{\partial M_B^{\alpha}}{\partial y_j} e_{ij}^{\alpha} \mu_B + \sum_{\alpha} M_A^{\alpha} \Delta N^{\alpha} - \\ \sum_{\alpha} N^{\alpha} \sum_i \frac{\partial M_A^{\alpha}}{\partial y_i} \sum_j \frac{\partial^2 G_M^{\alpha}}{\partial y_j \partial T} e_{ij}^{\alpha} \Delta T = \sum_{\alpha} N^{\alpha} \sum_i \sum_j \frac{\partial M_A^{\alpha}}{\partial y_i} \frac{\partial G_M^{\alpha}}{\partial y_j} e_{ij}^{\alpha} \end{aligned}$$

where we use the matrix elements e_{ij} in order to have as unknown variables only the potentials and the change in phase amounts, ΔN^{α} . In OC the equilibrium matrix is formulated in the subroutine `setup_equilmatrix`.

An example of the system equilibrium matrix

For a binary A-B system (at known T, P) with two stable phases we have 4 unknown: μ_A, μ_B and the change in phase amounts $\Delta N^\alpha, \Delta N^\beta$:

$$\begin{pmatrix} M_A^\alpha & M_B^\alpha & 0 & 0 \\ M_A^\beta & M_B^\beta & 0 & 0 \\ \sum_\gamma N^\gamma \sum_i \frac{\partial M_A^\gamma}{\partial y_i} \sum_j \frac{\partial M_A^\gamma}{\partial y_j} e_{ij}^\gamma & \sum_\gamma N^\gamma \sum_i \frac{\partial M_B^\gamma}{\partial y_i} \sum_j \frac{\partial M_B^\gamma}{\partial y_j} e_{ij}^\gamma & M_A^\alpha & M_A^\beta \\ \sum_\gamma N^\gamma \sum_i \frac{\partial M_B^\gamma}{\partial y_i} \sum_j \frac{\partial M_A^\gamma}{\partial y_j} e_{ij}^\gamma & \sum_\gamma N^\gamma \sum_i \frac{\partial M_B^\gamma}{\partial y_i} \sum_j \frac{\partial M_B^\gamma}{\partial y_j} e_{ij}^\gamma & M_B^\alpha & M_B^\beta \end{pmatrix} \begin{pmatrix} \mu_A \\ \mu_B \\ \Delta N^\alpha \\ \Delta N^\beta \end{pmatrix} =$$

$$\begin{pmatrix} G_M^\alpha \\ G_M^\beta \\ \sum_\gamma N^\gamma \sum_i \sum_j \frac{\partial M_A^\gamma}{\partial y_i} \frac{\partial G_M^\gamma}{\partial y_j} e_{ij}^\gamma \\ \sum_\gamma N^\gamma \sum_i \sum_j \frac{\partial M_B^\gamma}{\partial y_i} \frac{\partial G_M^\gamma}{\partial y_j} e_{ij}^\gamma \end{pmatrix}$$

Solving this system of equations we have new values of the chemical potentials and can obtain new constitutions of all phases using eq. 10.

Correcting all phase constitutions, step 2B

If we had not inverted the “phase matrices” separately we have to solve a much larger matrix with all phase constituents as variables.

It is an essential feature of the algorithm because for multicomponent systems the number of constituent variables increases much more rapidly than the number of stable phases.

Correcting all phase constitutions, step 2B

If we had not inverted the “phase matrices” separately we have to solve a much larger matrix with all phase constituents as variables.

It is an essential feature of the algorithm because for multicomponent systems the number of constituent variables increases much more rapidly than the number of stable phases.

Inserting the chemical potentials in eq. 10 we can calculate new constitutions of all phases (in the case of unknown T):

$$\Delta y_i^\alpha = \sum_A \sum_j \frac{\partial M_A}{\partial y_j} e_{ij} \mu_A - \sum_j \frac{\partial G_M}{\partial y_j} e_{ij} - \sum_j \frac{\partial^2 G_M}{\partial T \partial y_j} e_{ij} \Delta T$$

Correcting all phase constitutions, step 2B

If we had not inverted the “phase matrices” separately we have to solve a much larger matrix with all phase constituents as variables.

It is an essential feature of the algorithm because for multicomponent systems the number of constituent variables increases much more rapidly than the number of stable phases.

Inserting the chemical potentials in eq. 10 we can calculate new constitutions of all phases (in the case of unknown T):

$$\Delta y_i^\alpha = \sum_A \sum_j \frac{\partial M_A}{\partial y_j} e_{ij} \mu_A - \sum_j \frac{\partial G_M}{\partial y_j} e_{ij} - \sum_j \frac{\partial^2 G_M}{\partial T \partial y_j} e_{ij} \Delta T$$

This equation can be used to change the constitutions of the unstable phases also.

Phase set change, step 2C

We have also new amounts of the phases, N^α . **If any phase amount becomes negative the phase will be removed from the stable set.**

Phase set change, step 2C

We have also new amounts of the phases, N^α . **If any phase amount becomes negative the phase will be removed from the stable set.**

As we will update the constitutions of all unstable phases at each iteration we can calculate their Gibbs energy, G_M^δ , and compare this with the stable Gibbs energy plane, i.e. their driving force, ΔG^δ :

$$\Delta G^\delta = \sum_A M_A^\delta \mu_A - G_M^\delta \quad (15)$$

If this is positive it means the phase is stable and added to the set of stable phases.

Phase set change, step 2C

We have also new amounts of the phases, N^α . **If any phase amount becomes negative the phase will be removed from the stable set.**

As we will update the constitutions of all unstable phases at each iteration we can calculate their Gibbs energy, G_M^δ , and compare this with the stable Gibbs energy plane, i.e. their driving force, ΔG^δ :

$$\Delta G^\delta = \sum_A M_A^\delta \mu_A - G_M^\delta \quad (15)$$

If this is positive it means the phase is stable and added to the set of stable phases.

In OC the subroutine `meq_phaseset` organizes the overall solution of the equilibrium. It calls `meq_onephase` to invert phase matrices, `setup_equilmatrix` to formulate the equilibrium matrix and `meq_sameset` to handle the iterations as long as there is no phase change. A change of the stable set of phases is handled carefully inside `meq_phaseset`.

Convergence criteria

If there is no phase change there is a check after each iteration if the change in chemical potentials and phase amounts are small and all external conditions are fulfilled the calculation has converged.

Convergence criteria

If there is no phase change there is a check after each iteration if the change in chemical potentials and phase amounts are small and all external conditions are fulfilled the calculation has converged.

Otherwise we go back to step 1 and invert the new phase matrices.

How to know if we have a global equilibrium?

An iterative algorithm can only find a local minimum. To ensure we find a global minimum we must first make a search of the Gibbs energy (hyper-)surface of all phases.

How to know if we have a global equilibrium?

An iterative algorithm can only find a local minimum. To ensure we find a global minimum we must first make a search of the Gibbs energy (hyper-)surface of all phases. A few important things to note:

- ▶ A global minimizer must have T, P known and preferable also the overall composition.
- ▶ Some stable phases may have a composition very far from the overall composition.
- ▶ A phase may be stable with different compositions (a miscibility gap)
- ▶ With many components (> 10) the number of gridpoints in each phase will be vary limited. With more than 300 solution phases and limiting to ternary combinations we must calculate more than 10^5 gridpoints and that will take longer time than the iterative algorithm. However, it can be done in parallel!
- ▶ For unknown T, P the global minimizer can be used after an iterative algorithm to check for gridpoints below the calculated equilibrium.

An example of the OC grid minimizer

As already mentioned any iterative algorithm must have a reasonable set of starting values, i.e. a guess of the stable phases and their approximate constitution.

An example of the OC grid minimizer

As already mentioned any iterative algorithm must have a reasonable set of starting values, i.e. a guess of the stable phases and their approximate constitution.

To ensure the calculated equilibrium is a global minimum the TC and OC software has an initial step 0 which calculates the Gibbs energy of all phases over a grid of compositions.

An example of the OC grid minimizer

As already mentioned any iterative algorithm must have a reasonable set of starting values, i.e. a guess of the stable phases and their approximate constitution.

To ensure the calculated equilibrium is a global minimum the TC and OC software has an initial step 0 which calculates the Gibbs energy of all phases over a grid of compositions.

These are treated as stoichiometric phases in a preliminary minimization to find a set of gridpoints representing a global minimum and fulfilling the mass balance equations.

An example of the OC grid minimizer

As already mentioned any iterative algorithm must have a reasonable set of starting values, i.e. a guess of the stable phases and their approximate constitution.

To ensure the calculated equilibrium is a global minimum the TC and OC software has an initial step 0 which calculates the Gibbs energy of all phases over a grid of compositions.

These are treated as stoichiometric phases in a preliminary minimization to find a set of gridpoints representing a global minimum and fulfilling the mass balance equations.

This set of gridpoints for the minimum are then used as initial guess of the set of stable phases and their constitutions are inserted as start values for the iterative algorithm.

An example of the OC grid minimizer

As already mentioned any iterative algorithm must have a reasonable set of starting values, i.e. a guess of the stable phases and their approximate constitution.

To ensure the calculated equilibrium is a global minimum the TC and OC software has an initial step 0 which calculates the Gibbs energy of all phases over a grid of compositions.

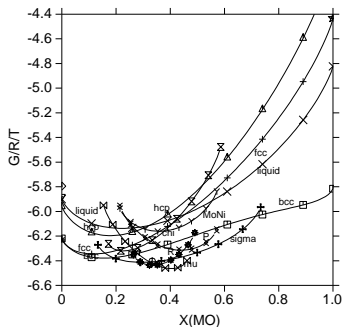
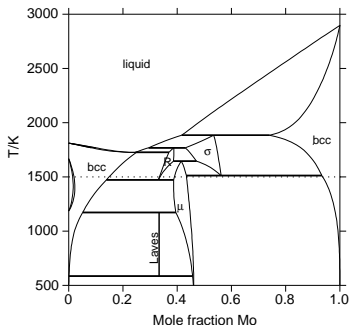
These are treated as stoichiometric phases in a preliminary minimization to find a set of gridpoints representing a global minimum and fulfilling the mass balance equations.

This set of gridpoints for the minimum are then used as initial guess of the set of stable phases and their constitutions are inserted as start values for the iterative algorithm.

The grid minimizer will automatically find miscibility gaps as the same phase may have several gridpoints in this set.

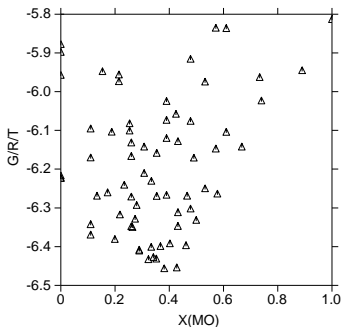
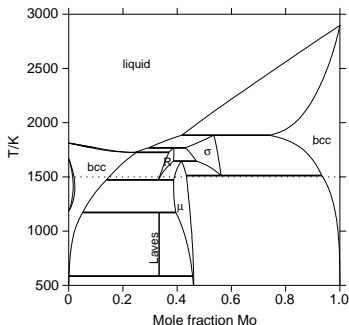
Grid minimizer, an example

In the Fe-Mo system at 1500 K the grid minimizer calculates a number of points along the Gibbs energy curves of the different phases.



Grid minimizer, an example

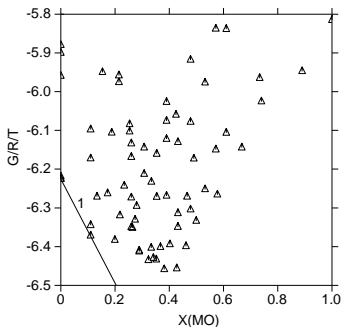
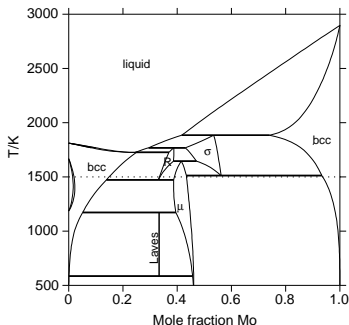
In the Fe-Mo system at 1500 K the grid minimizer calculates a number of points along the Gibbs energy curves of the different phases.



The gridpoints are treated as individual stoichiometric phases by the grid minimizer.

Grid minimizer, an example

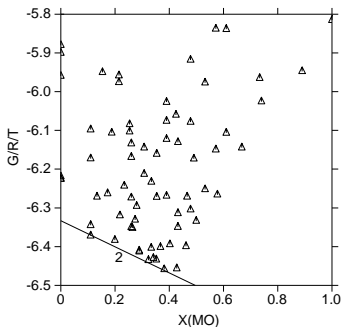
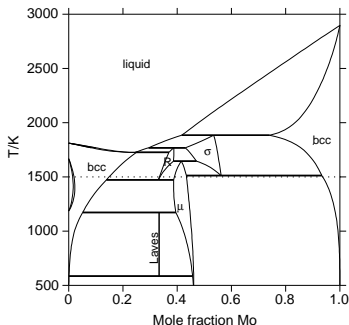
In the Fe-Mo system at 1500 K the grid minimizer calculates a number of points along the Gibbs energy curves of the different phases.



For low Mo content these two gridpoints will be selected as starting constitutions. They represent the FCC and BCC solution phases.

Grid minimizer, an example

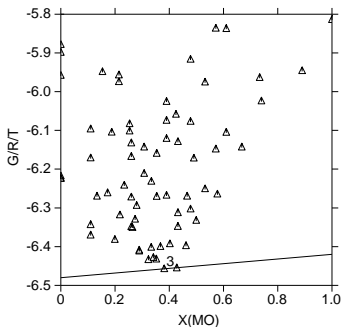
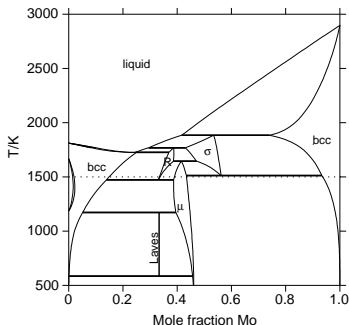
In the Fe-Mo system at 1500 K the grid minimizer calculates a number of points along the Gibbs energy curves of the different phases.



For higher Mo content these two gridpoints will be selected. They represent the BCC and μ phases. The iterative algorithm when then replace either BCC or μ with the R phase.

Grid minimizer, an example

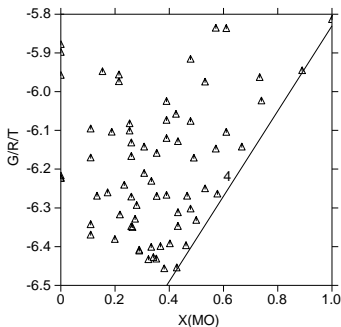
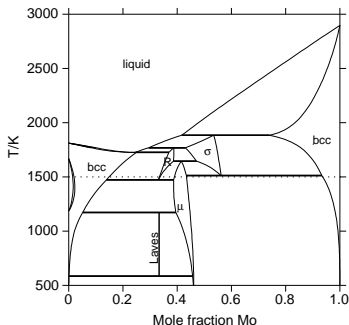
In the Fe-Mo system at 1500 K the grid minimizer calculates a number of points along the Gibbs energy curves of the different phases.



For Mo content around 40% these two gripoints will be selected. They are both in the μ phase and the iterative algorithm will merge them into a single composition.

Grid minimizer, an example

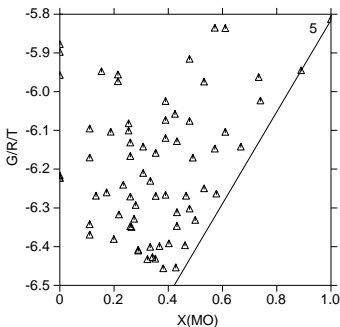
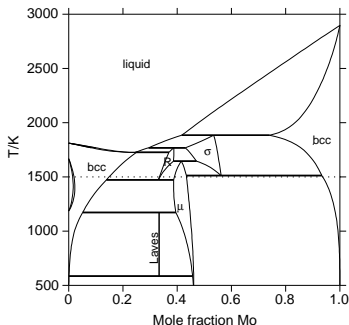
In the Fe-Mo system at 1500 K the grid minimizer calculates a number of points along the Gibbs energy curves of the different phases.



At higher Mo content one gridpoint in the μ and another in the Mo-rich BCC phase will be selected.

Grid minimizer, an example

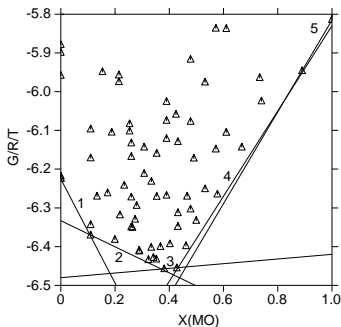
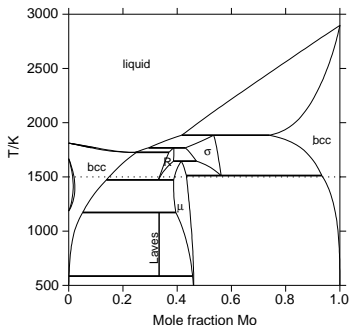
In the Fe-Mo system at 1500 K the grid minimizer calculates a number of points along the Gibbs energy curves of the different phases.



For very high Mo content these two gridpoints in the BCC phase will be used as start points. They will merge to a single composition during the iterative minimization.

Grid minimizer, an example

In the Fe-Mo system at 1500 K the grid minimizer calculates a number of points along the Gibbs energy curves of the different phases.



Here is the convex hull represented by the gridpoints.

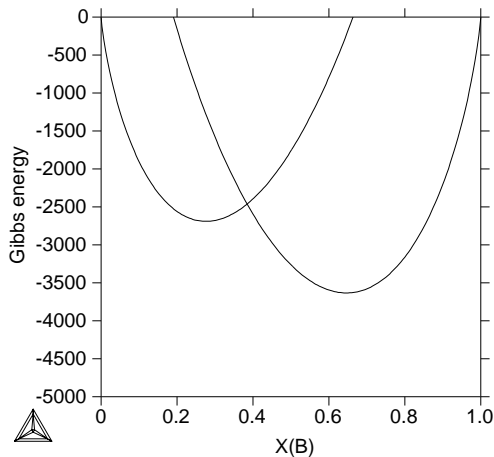
The density of the grid can be selected but more gridpoints require longer computational times. When calculating diagrams OC uses the previous calculated equilibrium as startpoint but occasionally global tests can be made.

A simple example how Mats algorithm work

To better understand Mats algorithm we use a binary A-B system

A simple example how Mats algorithm work

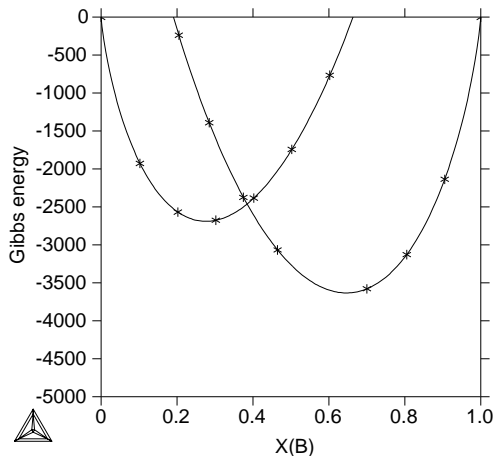
To better understand Mats algorithm we use a binary A-B system



The Gibbs energy curves for two phases at constant T, P . We will use as conditions that the system is closed and we have a value of x_B .

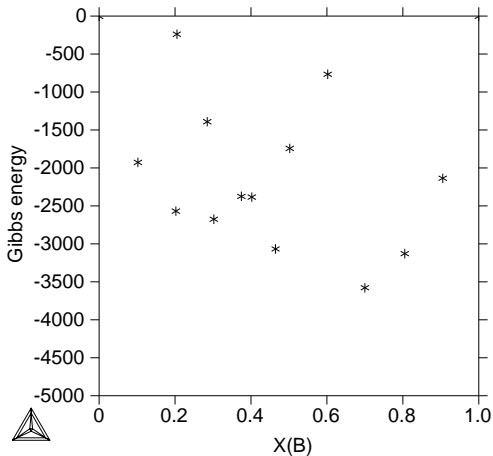
The global gridminimizer generates a number of points

With the global gridminimizer we calculate some points on the curves



The global gridminimizer generates a number of points

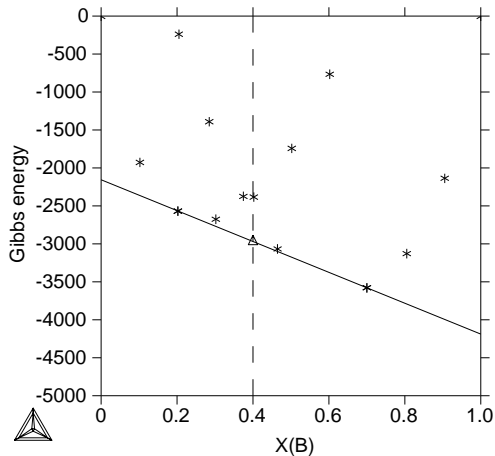
With the global gridminimizer we calculate some points on the curves



and treat them as individual phases with fixed composition (or as molecules of a gas phase).

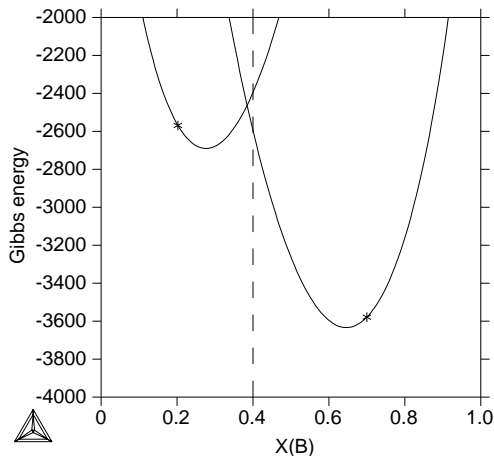
The global gridminimizer generates a number of points

With the global gridminimizer we calculate some points on the curves



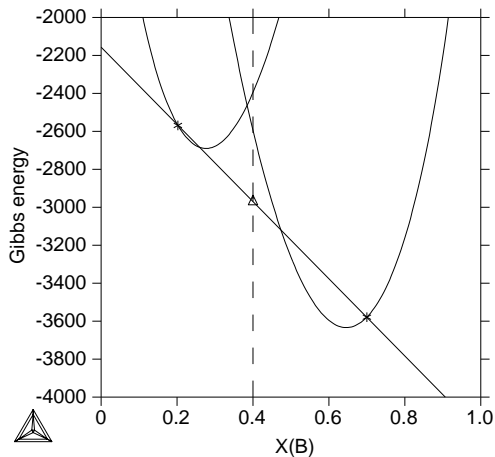
and we find two points for the condition $x_B = 0.4$. The triangle at the overall composition is the first guess of the total Gibbs energy.

Snapshots during the iterations



These are the start points and the overall composition. They must fulfill the mass balance conditions.

Snapshots during the iterations



This is the first Gibbs plane.

Step 1 of the iteration

Using the compositions of the two phases we calculate G and all derivatives for the phase matrix:

$$\begin{pmatrix} \frac{\partial^2 G_M}{\partial y_1^2} & \frac{\partial^2 G_M}{\partial y_1 \partial y_2} & \cdots & -1 & \cdots \\ \frac{\partial^2 G_M}{\partial y_1 \partial y_2} & \frac{\partial^2 G_M}{\partial y_2^2} & \cdots & -1 & \cdots \\ \vdots & \vdots & & & \\ -1 & -1 & \cdots & 0 & \cdots \\ \vdots & \vdots & & & \end{pmatrix} \begin{pmatrix} \Delta y_1 \\ \Delta y_2 \\ \vdots \\ \frac{\eta_s}{N} \\ \vdots \end{pmatrix} = \begin{pmatrix} \sum_A \mu_A \frac{\partial M_A}{\partial y_1} - \frac{\partial G_M}{\partial y_1} \\ \sum_A \mu_A \frac{\partial M_A}{\partial y_2} - \frac{\partial G_M}{\partial y_2} \\ \vdots \\ 0 \\ \vdots \end{pmatrix}$$

and invert the matrix on the left hand side to obtain:

$$\Delta y_i^\alpha = \sum_A \sum_j \frac{\partial M_A^\alpha}{\partial y_j^\alpha} e_{ij}^\alpha \mu_A - \sum_j \frac{\partial G_M^\alpha}{\partial y_j^\alpha} e_{ij}^\alpha \quad (16)$$

where e_{ij}^α are the terms of the inverted phase matrix.

Formulating the equilibrium matrix

We use the Δy_{is} to calculate the terms in the equilibrium matrix where we have just 4 unknowns: μ_A, μ_B and the change in phase amounts $\Delta N^\alpha, \Delta N^\beta$:

$$\begin{pmatrix} M_A^\alpha & M_B^\alpha & 0 & 0 \\ M_A^\beta & M_B^\beta & 0 & 0 \\ \sum_\gamma N^\gamma \sum_i \frac{\partial M_A^\gamma}{\partial y_i} \sum_j \frac{\partial M_A^\gamma}{\partial y_j} e_{ij}^\gamma & \sum_\gamma N^\gamma \sum_i \frac{\partial M_A^\gamma}{\partial y_i} \sum_j \frac{\partial M_B^\gamma}{\partial y_j} e_{ij}^\gamma & M_A^\alpha & M_A^\beta \\ \sum_\gamma N^\gamma \sum_i \frac{\partial M_B^\gamma}{\partial y_i} \sum_j \frac{\partial M_A^\gamma}{\partial y_j} e_{ij}^\gamma & \sum_\gamma N^\gamma \sum_i \frac{\partial M_B^\gamma}{\partial y_i} \sum_j \frac{\partial M_B^\gamma}{\partial y_j} e_{ij}^\gamma & M_B^\alpha & M_B^\beta \end{pmatrix} \begin{pmatrix} \mu_A \\ \mu_B \\ \Delta N^\alpha \\ \Delta N^\beta \end{pmatrix} =$$

$$\begin{pmatrix} G_M^\alpha \\ G_M^\beta \\ \sum_\gamma N^\gamma \sum_i \sum_j \frac{\partial M_A^\gamma}{\partial y_i} \frac{\partial G_M^\gamma}{\partial y_j} e_{ij}^\gamma \\ \sum_\gamma N^\gamma \sum_i \sum_j \frac{\partial M_B^\gamma}{\partial y_i} \frac{\partial G_M^\gamma}{\partial y_j} e_{ij}^\gamma \end{pmatrix}$$

Even in multicomponent system this matrix is quite small as we only use the chemical potentials and phase amounts as variables.

New constitutions

Solving this system of equations we have new values of the chemical potentials, μ_A and can obtain new constitutions of all phases, y_i^α using the same inverted phase matrices:

$$\Delta y_i^\alpha = \sum_A \sum_j \frac{\partial M_A^\alpha}{\partial y_j^\alpha} e_{ij}^\alpha \mu_A - \sum_j \frac{\partial G_M^\alpha}{\partial y_j^\alpha} e_{ij}^\alpha \quad (17)$$

We also have new amounts of the phases.

New constitutions

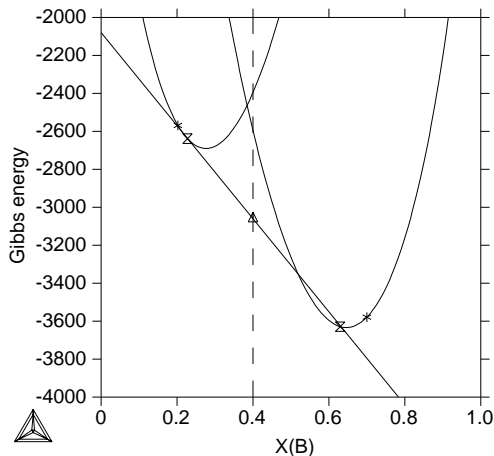
Solving this system of equations we have new values of the chemical potentials, μ_A and can obtain new constitutions of all phases, y_i^α using the same inverted phase matrices:

$$\Delta y_i^\alpha = \sum_A \sum_j \frac{\partial M_A^\alpha}{\partial y_j^\alpha} e_{ij}^\alpha \mu_A - \sum_j \frac{\partial G_M^\alpha}{\partial y_j^\alpha} e_{ij}^\alpha \quad (17)$$

We also have new amounts of the phases.

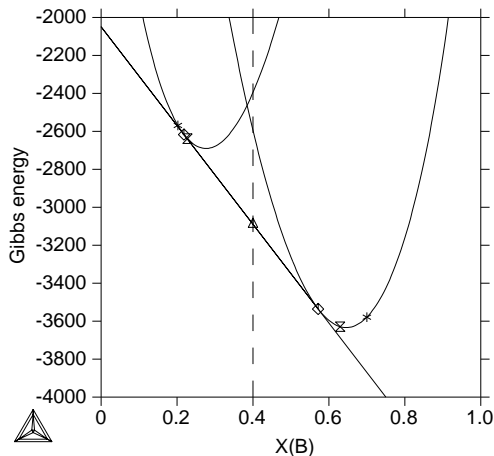
And we may have to handle a change of the set of stable phases.

Snapshots during the iterations



This is the change in the phase constitution and chemical potentials and the Gibbs plane at the first iteration. So we invert the new phase matrices and continue.

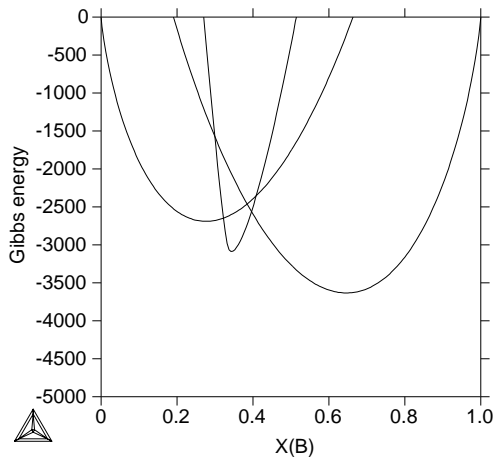
Snapshots during the iterations



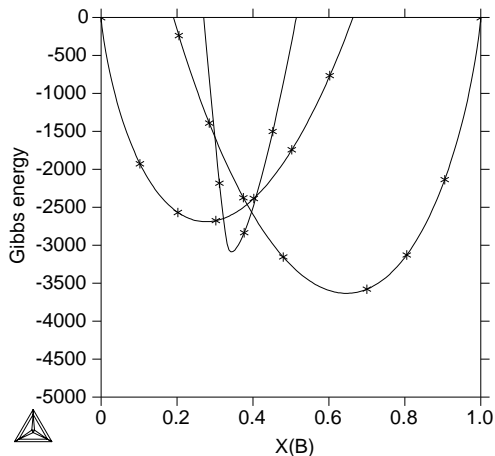
This is the final equilibrium with the common tangent, the equilibrium chemical potentials and the composition and amounts of the phases.

Consider adding a third phase to this system

The Gibbs energy curve for the third phase is very narrow.

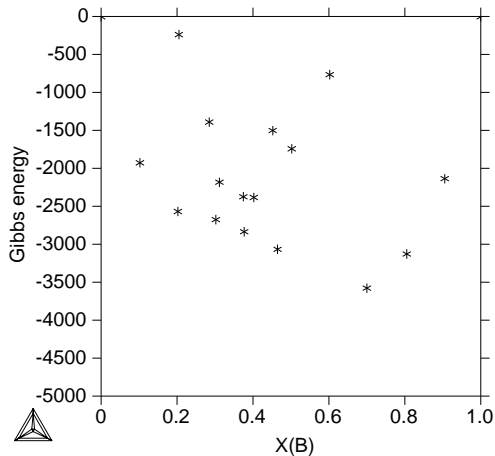


The global gridminimizer again



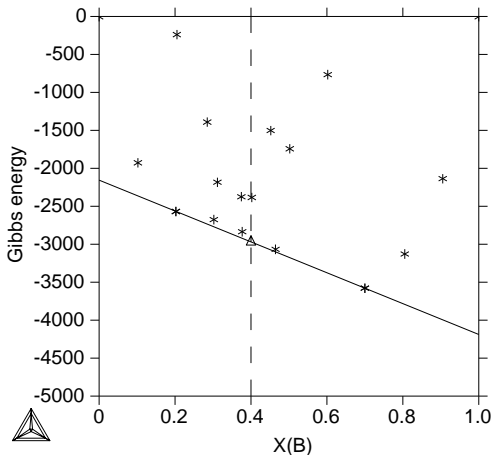
Assume these points are calculated by the grid minimizer

The global gridminimizer again



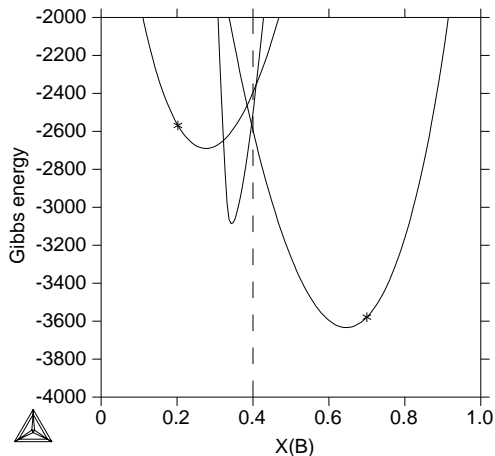
Treating these points as stoichiometric phases

The global gridminimizer again



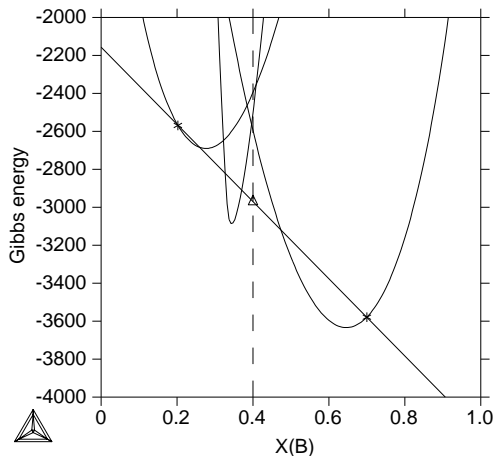
The same two gridpoints are found by the grid minimizer. (In a binary case we can calculate very many points but we must consider cases when we have more than 15 components and 300 phases)

Snapshots during the iterations



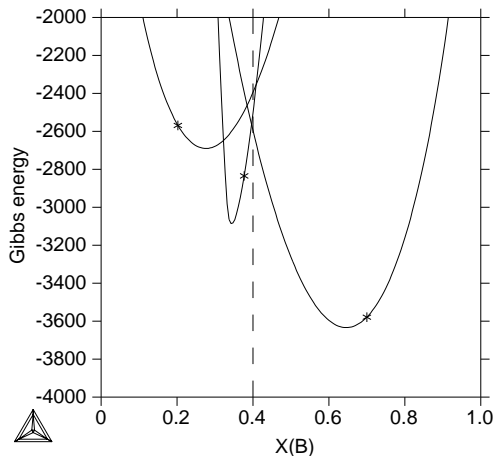
We thus have the same two start points as in the previous case.

Snapshots during the iterations



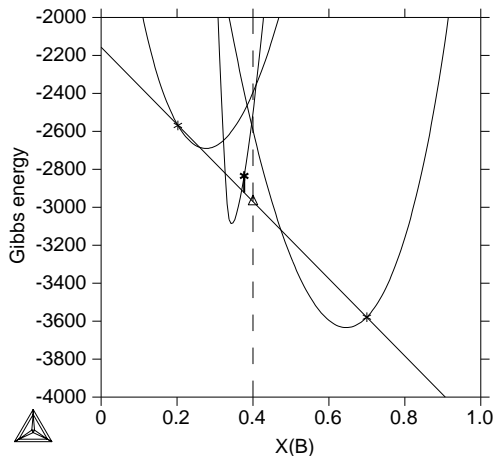
and the same Gibbs energy plane. We can see the third phase has a Gibbs energy curve below this plane.

Snapshots during the iterations



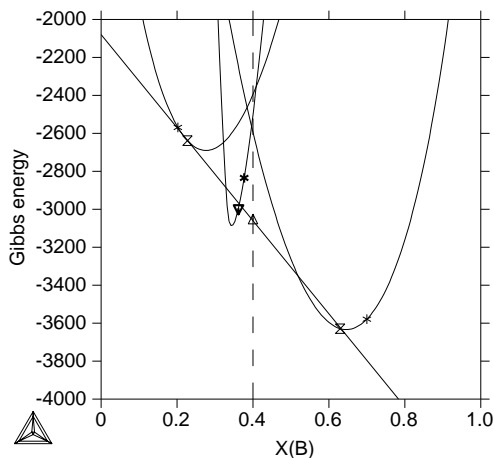
The grid minimizer has set the composition of the third unstable phase to a gridpoint in this phase closest to the Gibbs energy plane.

Snapshots during the iterations



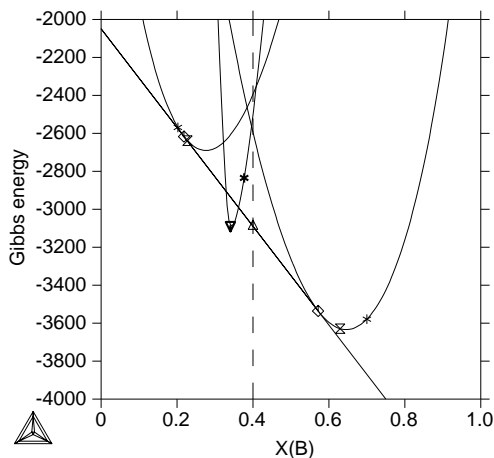
The grid minimizer has set the composition of the unstable phase to its gridpoint closest to the Gibbs energy plane. As it is above this plane it has a negative driving force.

Snapshots during the iterations



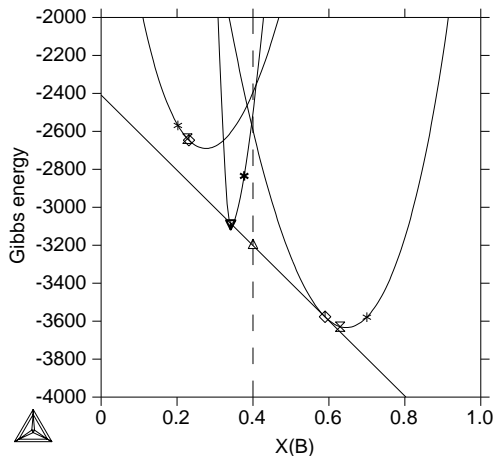
After the first iteration Mats algorithm will also calculate a correction of the composition of the unstable phases. This means the Gibbs energy of this phase is just above the stable Gibbs energy plane.

Snapshots during the iterations



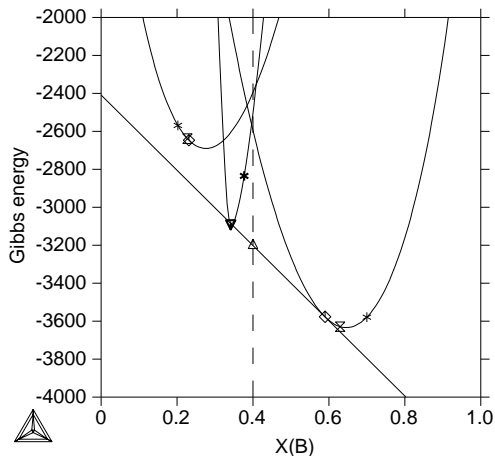
At the next iteration the unstable phase has a positive driving force and it will be added to the stable phase set, ensuring that the mass balance is still fulfilled.

Snapshots during the iterations



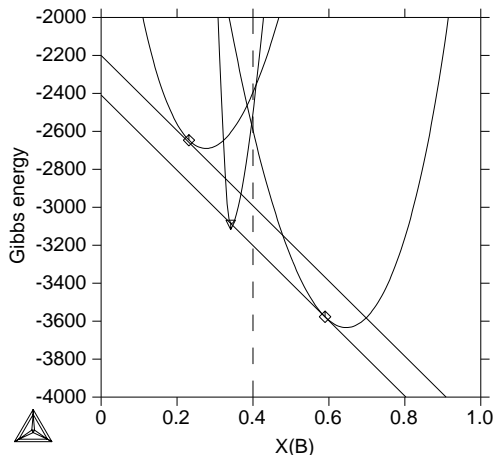
At a following iteration one of the stable phases will have negative amount and it will be removed from the stable phase set

Snapshots during the iterations



After a small correction of the phase compositions and amounts the final equilibrium with the third phase is found.

Snapshots during the iterations



Mats algorithm will modify the composition of all unstable phases to ensure they are as close as possible to the stable Gibbs energy plane, i.e. where they have a tangent parallel to the stable plane.

Conclusions

The easiest way to explain how to calculate a two-phase equilibrium (in a binary system) is to set the chemical potential equal for two selected phases. In a binary system we can use almost any algorithm to find the equilibrium but there are few that works when we have more than 10 components and 300 phases.

Conclusions

The easiest way to explain how to calculate a two-phase equilibrium (in a binary system) is to set the chemical potential equal for two selected phases. In a binary system we can use almost any algorithm to find the equilibrium but there are few that works when we have more than 10 components and 300 phases. For many CEF models it is not possible to calculate the chemical potential of the system components for a phase, only the chemical potential for the endmembers of a phase.

Conclusions

The easiest way to explain how to calculate a two-phase equilibrium (in a binary system) is to set the chemical potential equal for two selected phases. In a binary system we can use almost any algorithm to find the equilibrium but there are few that works when we have more than 10 components and 300 phases. For many CEF models it is not possible to calculate the chemical potential of the system components for a phase, only the chemical potential for the endmembers of a phase.

All algorithms require some global search of the Gibbs energy surface of all phases. There are more complex search methods than the grid method and if anyone can provide a better global algorithm I am happy to try that in OC.

Conclusions

The easiest way to explain how to calculate a two-phase equilibrium (in a binary system) is to set the chemical potential equal for two selected phases. In a binary system we can use almost any algorithm to find the equilibrium but there are few that works when we have more than 10 components and 300 phases. For many CEF models it is not possible to calculate the chemical potential of the system components for a phase, only the chemical potential for the endmembers of a phase.

All algorithms require some global search of the Gibbs energy surface of all phases. There are more complex search methods than the grid method and if anyone can provide a better global algorithm I am happy to try that in OC.

Mats algorithm has a unique facility to change the set of stable phase “on the fly” while iterating, both to remove and add phases.

Mats algorithm has also a unique flexibility for the user to specify external conditions. For example the condition that a specific phase should be stable is used to calculate phase diagrams and transition points.

That is all

References and more reading

- ▶ Eriksson G., *Chem Scr* **8** (1975) 100–104
- ▶ Hillert M., *Physica* **103B** (1981) 31–40
- ▶ Jansson B., PhD Thesis, KTH (1984)
- ▶ Sundman B., Lu X-G. and Ohtani H., *Comp Mat Sci*, **101** (2015) 127–137
- ▶ OC software documentation.