A Maple script to illustrate elementary free energy minimization as discussed in:

www.perplex.ethz.ch/papers/connolly_epsl_05.pdf

This script is at:

www.perplex.ethz.ch/simplex/simplex.mws

NOTE SECTION 1 must be executed before SECTION 2 can be executed, likewise SECTION 3 must be executed before SECTION 4.

SECTION 1: G(P,T,n[CaO],n[SiO2]) Minimization with simple compounds:

```
> restart;
```

This example illustrates free energy minimization for the system CaO-SiO2 considering 3 stoichiometric phases, wollastonite, lime and quartz. For simplicity the equation of state is taken to be

```
> G[i] := G0[i] - T*S0[i] + P*V0[i];
```

a more complex equation of state (i.e., that used by Holland & Powell 1998) is programmed in the maple script www.perplex.ethz.ch/course/problem4.mws

The (fictive and arbitrary) parameters for each phase are assigned as follows

```
> G0[1] := 0.1: S0[1] := 0.015: V0[1] := 0.0018 : ncao[1] := 1:
  nsio2[1] := 1: #wo;
> G0[2] := 0.13: S0[2] := 0.01: V0[2] := 0.001 : ncao[2] := 1:
  nsio2[2] := 0: #lime;
> G0[3] := 0.15: S0[3] := 0.012: V0[3] := 0.0011: ncao[3] := 0:
  nsio2[3] := 1: #qtz;
```

the number of phases possible in the system is

```
> pi := 3;
```

the function to be minimized (total free energy of the system) is

```
> Gtot := sum(G[i]*alpha[i],i=1..pi);
```

subject to the constraints (sum of phases = system)

```
> ncatot := sum(ncao[i]*alpha[i],i=1..pi);
> nsitot := sum(nsio2[i]*alpha[i],i=1..pi);
> constraints := {ncatot = ncasys, nsitot = nsisys}:
```

instruct maple to allow linear programming routines:

```
> with(simplex):
```

$$G_i := G0_i - T\,S0_i + P\,V0_i$$

$$\pi := 3$$

$$Gtot := (0.1 - 0.015\,T + 0.0018\,P)\,\alpha_1 + (0.13 - 0.01\,T + 0.001\,P)\,\alpha_2 + (0.15 - 0.012\,T + 0.0011\,P)\,\alpha_3$$

$$ncatot := \alpha_1 + \alpha_2$$

$$nsitot := \alpha_1 + \alpha_3$$

```
Warning, the protected names maximize and minimize have been redefined and
unprotected
```

The minimization can now be done by invoking simplex for a specified pressure (0), temperature (26) and bulk composition (ncasys = 0.4, nsisys = 0.1), the maple option "NONNEGATIVE" tells maple that the optimization variables (the phase proportions) alpha[i] are >= 0. See the appendix for a script to filter the zero alpha's.

```
> minimize(subs(P=0,T=26,Gtot),
  subs(ncasys=0.4,nsisys=0.1,constraints), NONNEGATIVE );
```

$$\{\alpha_3 = 0.1000000000,\ \alpha_2 = 0.4000000000,\ \alpha_1 = 0.\}$$

for the above conditions phase 1 (wollastonite) is not stable and the system consists of 0.1 mols of phase 3 (quartz) and 0.4 mols of phase 2 (lime) if the temperature is lowered to 25 wollastonite becomes stable

```
> minimize(subs(P=0,T=25,Gtot),
  subs(ncasys=0.4,nsisys=0.1,constraints), NONNEGATIVE );
```

$$\{\alpha_3 = 0.,\ \alpha_2 = 0.3000000000,\ \alpha_1 = 0.1000000000\}$$

this problem can be visualized graphically by constructing a complete G-X diagram for the problem as illustrated in the next section (this corresponds to unconstrained minimization in Perple_X)

SECTION 2: All stable assemblages (G-X diagram), irrespective of composition, simple compounds:

to make the graphical representation clearer, we introduce to dummy phases that will never be stable because they have large positive G's compared to those of our initial phases:

```
> G0[4] := 1e7: S0[4] := 0: V0[4] := 0 : ncao[4] := 1: nsio2[4]
  := 0: #CAO;
> G0[5] := 1e7: S0[5] := 0: V0[5] := 0: ncao[5] := 0: nsio2[5] :=
  1: #SIO2;
```

and increase the phase counter accordingly:

```
> pi := 5:
```

previously our coordinates were ncao-nsio2-G, but now we need to transform the coordinates to x-gm (where x is the mole fraction of CaO, the mole fraction of SiO2 is 1 - x, and Gm is the G of the phase per mole of the systems components) so that we can represent the problem in two dimensions as follows:

```
> for i from 1 to pi do:
>    x[i] := ncao[i]/(ncao[i] + nsio2[i]):
>    gm[i] := (G0[i] - T*S0[i] + P*V0[i])/(ncao[i] + nsio2[i]);
> end do:
```

Now load the x-gm coordinates into an array (and assign pressure and temperature):

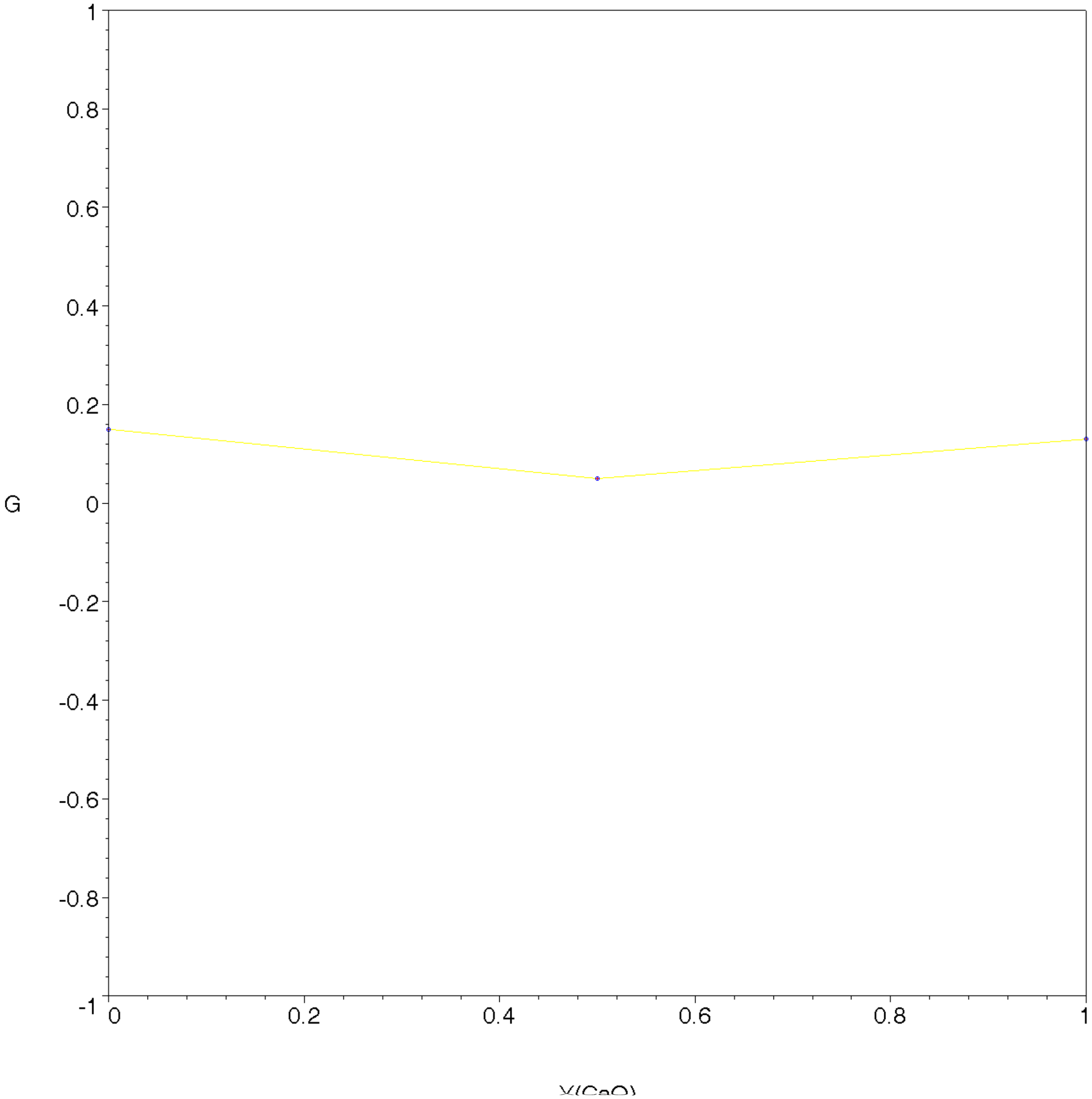
```
> set_of_points := [seq([x[i], subs(P=0, T = 0,gm[i])],i=1..pi)]:
```

and obtain an array of the stable points by invoking the maple function convexhull (essentially unconstrained optimization in Perple_X)

```
> stable_points := convexhull(  set_of_points  ):
```

finally plot the results highlighting the stable phases by a red cross and showing the minimum free energy surface of the system (i.e., the piecewise linear curve connecting the stable points) by a yellow curve.

```
> plot([set_of_points,stable_points,stable_points], x=0..1,-1..1,
  style=[point,point,line],symbol=[circle,cross],color=[blue,red,
  yellow],axes=boxed,labels=["X(CaO)","G"]);
```



by raising the temperature to 26, the gm-x coordinate of wollastonite (x=0.5) can be made to move off the minimum free energy surface of the system.

SECTION 3: G(P,T,n[CaO],n[SiO2]) Minimization with a solution phase:

```
> restart;
```

restart and reinitialize both maple and the data used in the initial section of this script:

```
> with(simplex):
```

```
> G[j] := G0[j] - T*S0[j] + P*V0[j]:
> G0[1] := 0.1: S0[1] := 0.015: V0[1] := 0.0018 : ncao[1] := 1:
  nsio2[1] := 1: #wo;
> G0[2] := 0.13: S0[2] := 0.01: V0[2] := 0.001 : ncao[2] := 1:
  nsio2[2] := 0: #lime;
> G0[3] := 0.15: S0[3] := 0.012: V0[3] := 0.0011: ncao[3] := 0:
  nsio2[3] := 1: #qtz;
> pi := 5:
```

we now introduce the complication that we have a solution whose free energy is a non-linear, but continuous function of its composition. as is common practice we describe this solution as a mixture of two endmember compositions defined here as:

```
> G0[4] := 0.14: S0[4] := 0.014: V0[4] := 0.0017 : ncao[4] := 0:
  nsio2[4] := 1: #sio2 endmember;
> G0[5] := 0.12: S0[5] := 0.011: V0[5] := 0.00105 : ncao[5] := 1:
  nsio2[5] := 0: #cao endmember;
```

and we assume an equation of state of the form

G = G0 - T*S0 + P*V0

where

G0 = xcao*G0[5] + (1-xcao)*G0[4]
S0 = xcao*(S0[5]-R*ln(xcao)) + (1-xcao)*(S0-R*ln(1-xcao))
V0 = xcao*V0[5] + (1-xcao)*V0[4]

and xcao is the mole fraction of endmember 5 in the solution and R is the universal gas constant in (arbitrary) units of S0.

The non-linear free energy minimization problem is reduced to a linear optimization problem by creating pseudocompounds to represent the non-linear free energy composition surface of the solution. The pseudocompounds are generated in the following loop with a compositional spacing of dx, the counter of possible phases (actually possible "pseudocompound" and/or compounds) is incremented accordingly, and the value of R is also assigned:

```
> dx := 0.1: xcao := dx: R := 0.08314:
> while xcao < 1 do:
>    pi := pi + 1:
>    ncao[pi] := xcao*ncao[5] + (1-xcao)*ncao[4]:
>    nsio2[pi] := xcao*nsio2[5] + (1-xcao)*nsio2[4]:
>    G0[pi] := xcao*G0[5] + (1-xcao)*G0[4]:
```

```
>    S0[pi] := xcao*(S0[5]-R*ln(xcao)) +
  (1-xcao)*(S0[4]-R*ln(1-xcao)):
>    V0[pi] := xcao*V0[5] + (1-xcao)*V0[4]:
>    xcao := xcao + dx:
> end do:
Warning, the protected names maximize and minimize have been redefined and
unprotected
```

The functions and constraints for the optimization are, as before (i.e., section 1):

```
> Gtot := sum(G[j]*alpha[j],j=1..pi):
> ncatot := sum(ncao[j]*alpha[j],j=1..pi):
> nsitot := sum(nsio2[j]*alpha[j],j=1..pi):
> constraints := {ncatot = ncasys, nsitot = nsisys}:
```

and assigning the pressure, temperature and bulk composition we obtain:

```
> minimize( subs(P=0,T=25.00,Gtot),
  subs(ncasys=0.4,nsisys=0.12,constraints), NONNEGATIVE );
```

$$\{\alpha_{12} = 0.1600000001, \alpha_{13} = 0.3599999999, \alpha_4 = 0., \alpha_5 = 0., \alpha_7 = 0., \alpha_8 = 0., \alpha_9 = 0., \alpha_{10} = 0., \alpha_{11} = 0., \alpha_{14} = 0., \alpha_6 = 0., \alpha_3 = 0., \alpha_1 = 0., \alpha_2 = 0.\}$$

thus the stable assemblage is a mixture of ~0.36 moles of pseudocompound 13 and a ~0.16 moles pseudocompound 14, since these correspond to adjacent pseudocompounds the true solution is that the system consists of a single real phase whose composition and properties can be obtained by averaging the properties of pseudocompounds 13 and 14.

See the appendix for a script to filter the zero alpha's.

SECTION 4: All stable assemblages (G-X diagram), irrespective of composition, with a solution phase:

As before (section 2) we make the transformation to G-X coordinates:

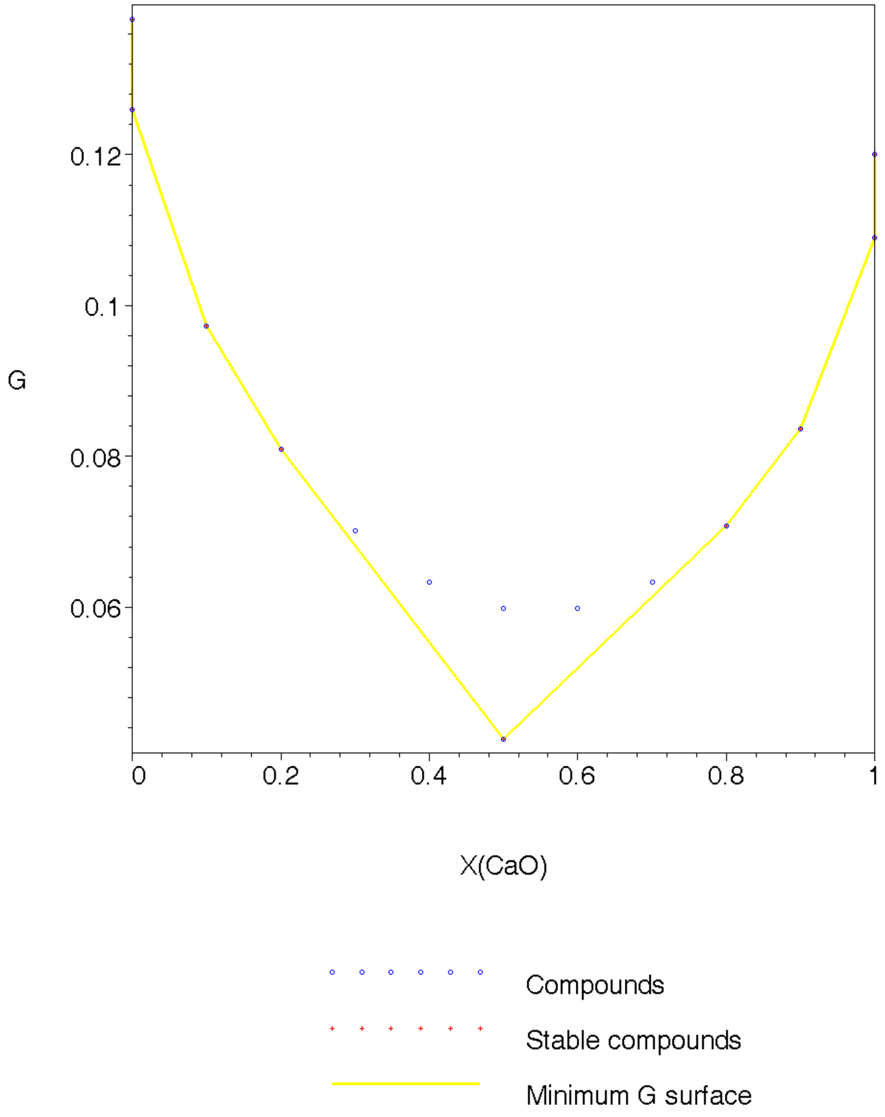
```
> for i from 1 to pi do:
>    x[i] := ncao[i]/(ncao[i] + nsio2[i]);
>    gm[i] := (G0[i] - T*S0[i] + P*V0[i])/(ncao[i] + nsio2[i]);
> end do:
> set_of_points := [seq([x[i], subs(P=0, T = 1,gm[i])],i=1..pi)]:
```

and compute the stable points

```
> stable_points := convexhull(  set_of_points  ):
```

and plot the results

```
> plot([set_of_points,stable_points,stable_points], x=0..1,
  style=[point,point,line],symbol=[circle,cross],color=[blue,red,
  yellow],axes=boxed,thickness=3,legend=["Compounds","Stable
  compounds","Minimum G surface"],labels=["X(CaO)","G"]);
```



from this plot it is evident that at the chosen conditions the intermediate compositions of the solution are metastable with respect to "wollastonite".

SECTION 5: More exotic potentials, Omega(V,T,n[CaO],mu[SiO2]) Minimization:

minimization for an isochoric-isothermal system that is closed with respect to CaO (and volume, V) and open with respect to SiO2 (and entropy, S) (this approach is detailed in Connolly 1990).

this section carries on from the previous section.

first do the legendre transform to get the function to be minimized:

because our model G function gives the value of G0 at T=0 and P=0 and V is not a function of pressure or temperature the function Omega[i] = G0[i] - T*S0[i] - mu[SiO2]*nsio2[i] = mu[CaO]*ncao[i] + P*V0[i]

first restore T as a symbolic variable:

```
> T := 'T':
```

next create omega-composition coordinates, bearing in mind that the composition X must now define the proportions of the conservative extenstive properties of the system, i.e., V and nCaO, e.g., x[i] = ncao[i] / (V0[i]+ncao[i])

```
> for i from 1 to pi do:
>    x[i] := ncao[i]/(ncao[i] + V0[i]);
>    omega[i] := (G0[i] - T*S0[i] - mu[SiO2]*nsio2[i]);
>    omegam[i] := (G0[i] - T*S0[i] - mu[SiO2]*nsio2[i])/(ncao[i]
  + V0[i]);
> end do:
```

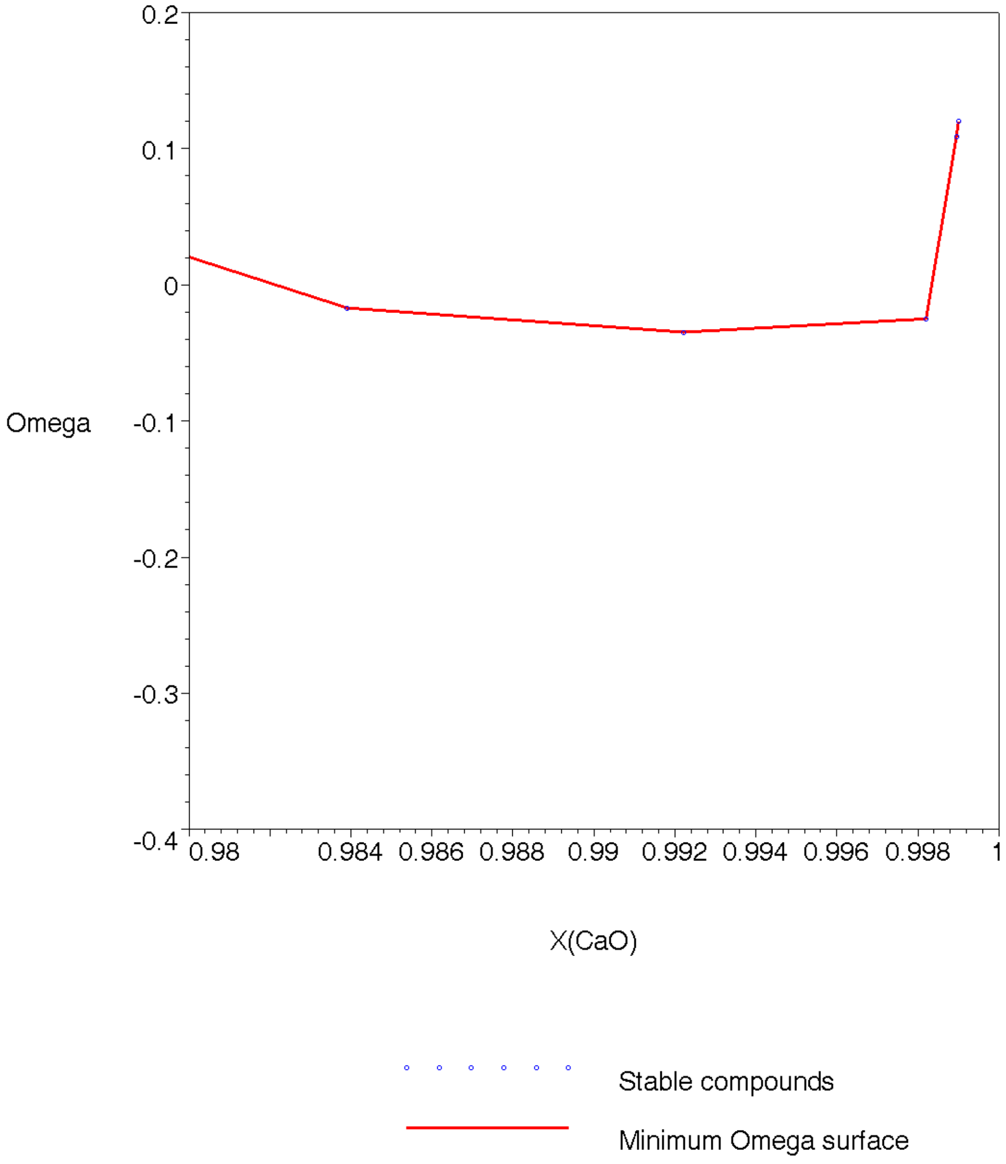
in distinction to the previous section, where P and T were specified for the minimization, now it is necessary to specify mu[SiO2] and T.

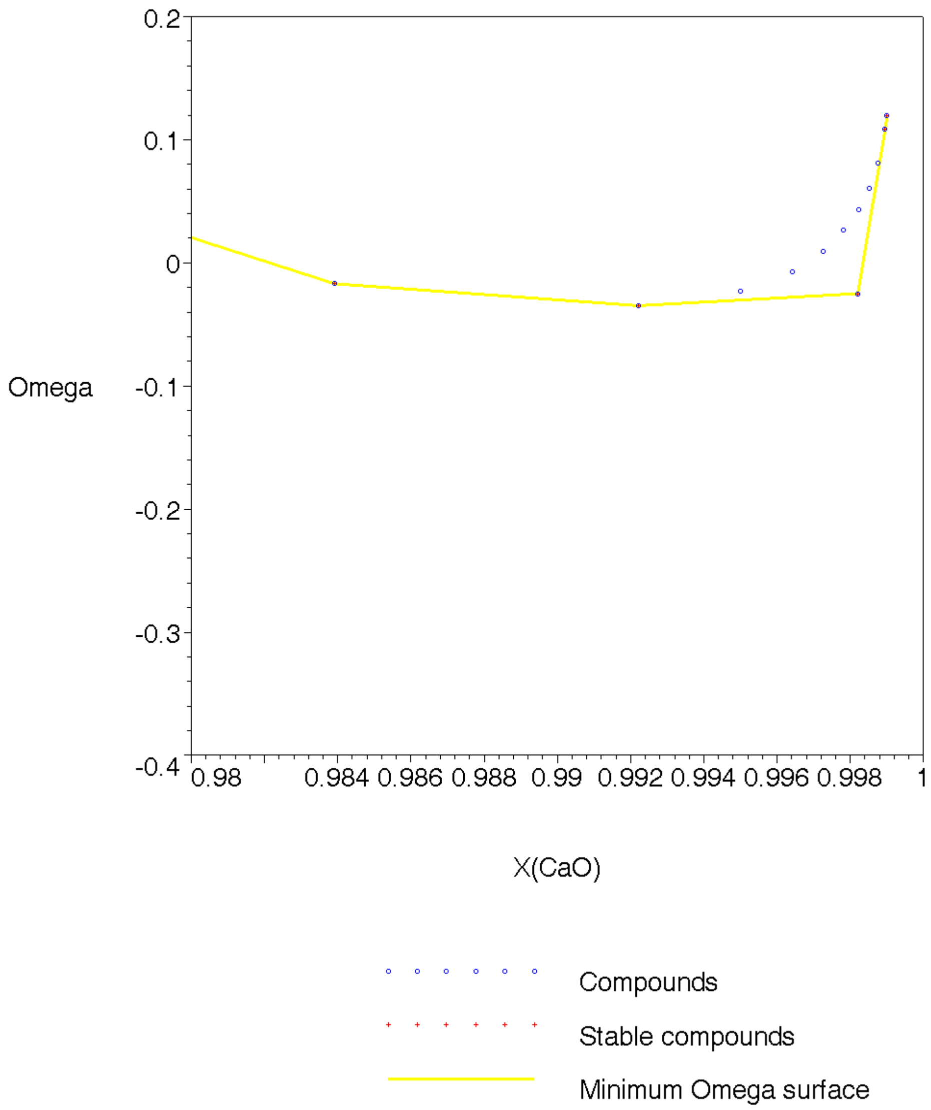
```
> set_of_points := [seq([x[i], subs(mu[SiO2]= .11, T =
  1,omegam[i])],i=1..pi)]:
Compute stable points:

> stable_points := convexhull(  set_of_points  ):

plot the results:
```

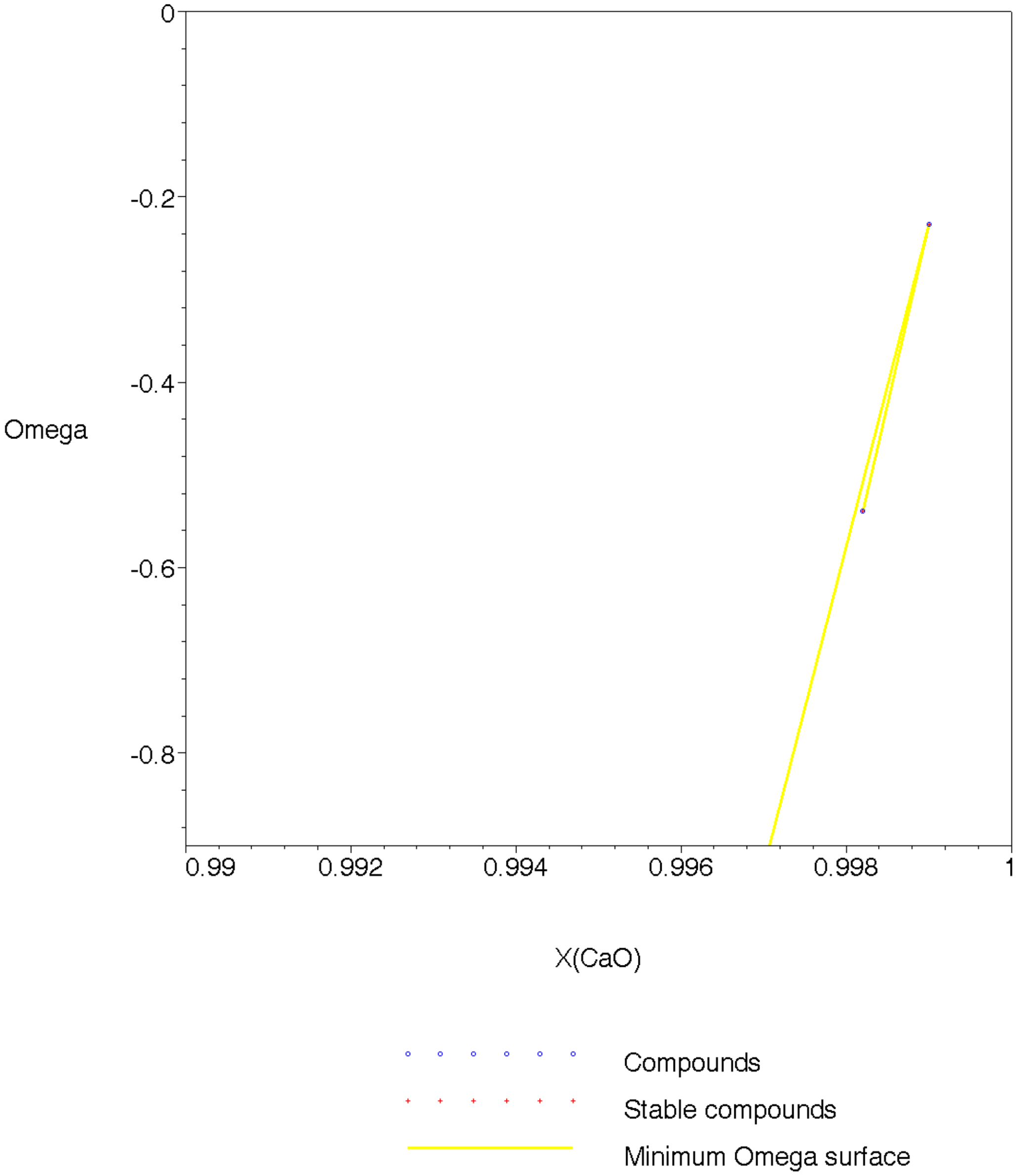
```
> plot([stable_points,stable_points],x=00.98..1,-0.4..0.2,style=[
  point,line],symbol=[circle,cross],color=[blue,red,yellow],axes=
  boxed,thickness=3,legend=["Stable compounds","Minimum Omega
  surface"],labels=["X(CaO)","Omega"]);
> plot([set_of_points,stable_points,stable_points],x=0.98..1,-0.4
  ..0.2,style=[point,point,line],symbol=[circle,cross],color=[blu
  e,red,yellow],axes=boxed,thickness=3,legend=["Compounds","Stabl
  e compounds","Minimum Omega
  surface"],labels=["X(CaO)","Omega"]);
> 
```





and now without the solution: find the T of the equilibrium lime+qtz+wo for mu[SiO2]= .1

```
> set_of_points := [seq([x[i], subs(mu[SiO2]= .1, T =
  36,omegam[i])],i=1..3)]:
> stable_points := convexhull(  set_of_points  ):
> plot([set_of_points,stable_points,stable_points],x=0.99..1,-.9.
  .0.,style=[point,point,line],symbol=[circle,cross],color=[blue,
  red,yellow],axes=boxed,thickness=3,legend=["Compounds","Stable
  compounds","Minimum Omega surface"],labels=["X(CaO)","Omega"]);
```



lastly to compute the reaction stoichiometry for the reaction lime+wo+qtz:

```
> with(linalg):
> wo := vector (2,[V0[1],ncao[1]]);
  lime := vector (2,[V0[2],ncao[2]]);
  qtz := vector (2,[V0[3],ncao[3]]);

> a := concat (qtz,lime);

> linsolve(a,wo);
Warning, the names basis and pivot have been redefined
```

```
Warning, the protected names norm and trace have been redefined and
unprotected
```

$$wo := [0.0018, 1]$$

$$lime := [0.001, 1]$$

$$qtz := [0.0011, 0]$$

$$a := \begin{bmatrix} 0.0011 & 0.001 \\ 0 & 1 \end{bmatrix}$$

$$[0.7272727273, 1.]$$

So the reaction is 0.471 Qtz + 1 lime = 1 wo

and what is the pressure of coexisting lime + wo if mu[SiO2] = .1, T = 36?

```
> at := transpose(concat(wo,lime));
> b := subs (mu[SiO2]= .1, T = 36, vector
  (2,[omega[1],omega[2]]));
> linsolve(at,subs(mu[SiO2]= .1, T = 36,b));
```

$$at := \begin{bmatrix} 0.0018 & 1 \\ 0.001 & 1 \end{bmatrix}$$

$$b := [-0.540, -0.23]$$

$$[-387.5000000, 0.1575000000]$$

i.e., P = 387.5, mu[CaO] = 0.1575

APPENDIX: A loop to filter non-zero values of an array

a practical problem arising in section 3 is that a large number of phases may have zero alphas, if we are interested in only the stable phase (e.g., for output) we can filter the alphas as follows:

```
> ans := minimize( subs(P=0,T=25,Gtot),
  subs(ncasys=0.4,nsisys=0.12,constraints), NONNEGATIVE );
> assign(ans);printf("\n");
> no_stable := 0:
  for i from 1 to pi do
     if (alpha[i]>0) then
       no_stable := no_stable +1;
       indice[no_stable] := i;
>      printf ("%s %g %s \n","pseudocompound #",i," is stable");
>    end if;
  end do;

> printf("\n %g %s",no_stable," stable pseudocompounds");
> unassign(alpha);
```

$$ans := \{\alpha_{13} = 0.3599999999, \alpha_{12} = 0.1600000001, \alpha_{7} = 0., \alpha_{11} = 0., \alpha_{8} = 0., \alpha_{14} = 0., \alpha_{9} = 0., \alpha_{2} = 0., \alpha_{4} = 0., \alpha_{3} = 0., \alpha_{10} = 0., \alpha_{1} = 0., \alpha_{5} = 0., \alpha_{6} = 0.\}$$

```
pseudocompound # 12  is stable
pseudocompound # 13  is stable

 2  stable pseudocompounds
```

```
>
```