Geochemistry, Geophysics, Geosystems®

AGU ADVANCING EARTH AND SPACE SCIENCE

RESEARCH ARTICLE
10.1029/2021GC010303

Key Points:

- Thermolab: a set of MATLAB codes is presented to perform equilibrium and nonequilibrium thermodynamic calculations
- Local thermodynamic equilibrium is used to study effects of nonideality of solution models on nonlinear transport processes
- Nonlinear diffusion processes are investigated with Thermolab providing a transient natural physical process toward equilibrium

Correspondence to:
J. C. Vrijmoed,
j.c.vrijmoed@fu-berlin.de

Citation:
Vrijmoed, J. C., & Podladchikov, Y. Y. (2022). Thermolab: A thermodynamics laboratory for nonlinear transport processes in open systems. *Geochemistry, Geophysics, Geosystems*, *23*, e2021GC010303. https://doi.org/10.5281/zenodo.6383253

Received 13 DEC 2021
Accepted 15 MAR 2022

Thermolab: A Thermodynamics Laboratory for Nonlinear Transport Processes in Open Systems

J. C. Vrijmoed[1] and Y. Y. Podladchikov[2,3]

[1]Institute of Geological Sciences, Freie Universität Berlin, Berlin, Germany, [2]Institute of Earth Sciences, University of Lausanne, Lausanne, Switzerland, [3]Faculty of Mechanics and Mathematics, Moscow State University, Moscow, Russia

Abstract We developed a numerical thermodynamics laboratory called "Thermolab" to study the effects of the thermodynamic behavior of nonideal solution models on reactive transport processes in open systems. The equations of the state of internally consistent thermodynamic data sets are implemented in MATLAB functions and form the basis for calculating Gibbs energy. A linear algebraic approach is used in Thermolab to compute Gibbs energy of mixing for multicomponent phases to study the impact of the nonideality of solution models on transport processes. The Gibbs energies are benchmarked with experimental data, phase diagrams, and other thermodynamic software. Constrained Gibbs minimization is exemplified with MATLAB codes and iterative refinement of composition of mixtures may be used to increase precision and accuracy. All needed transport variables such as densities, phase compositions, and chemical potentials are obtained from Gibbs energy of the stable phases after the minimization in Thermolab. We demonstrate the use of precomputed local equilibrium data obtained with Thermolab in reactive transport models. In reactive fluid flow the shape and the velocity of the reaction front vary depending on the nonlinearity of the partitioning of a component in fluid and solid. We argue that nonideality of solution models has to be taken into account and further explored in reactive transport models. Thermolab Gibbs energies can be used in Cahn-Hilliard models for nonlinear diffusion and phase growth. This presents a transient process toward equilibrium and avoids computational problems arising during precomputing of equilibrium data.

Plain Language Summary The behavior of Earth materials, rocks, minerals, melts, fluids, and gases is important to predict physical processes in the Earth with computer models. The purpose of this is to study how the changes of variables such as fluid and solid composition influence the diffusion, fluid flow, and reaction in rocks. Here, we present a set of computer codes, called Thermolab, to calculate important physical properties such as density and chemical composition of solids, fluids, and melts in chemical equilibrium. The calculations are based on the Gibbs energy that exists for every material. We use computer codes, written in MATLAB/OCTAVE language, to show how this Gibbs energy is calculated and used to compute chemical equilibrium and find the physical properties such as density and chemical composition. We discuss techniques for accurate calculation of chemical equilibrium and physical properties in real rocks. Finally, we use Thermolab to formulate a computer model of fluids reacting with rocks. We find that chemical composition of the fluid and rock strongly affects the speed and shape of the boundary between reacted and unreacted rock. Thermolab can be used in phase growth models to investigate the way in which rocks develop toward equilibrium.

1. Introduction

One of the primary goals in Earth Science is to understand the physical processes that govern natural phenomena. This will facilitate prediction of potential hazardous events, such as earthquakes, volcanic eruptions, and related threats such as tsunamis (Fagents et al., 2013; Geller, 1997; Rundle et al., 2021; Ulrich et al., 2019). Predicting causes and effects of climate change, the reduction of global warming by storing CO_2 or the safety of storing nuclear waste, strongly rely on the understanding of underlying coupled physical processes (Orr, 2018; X. Zhang et al., 2022). Furthermore, the exploration of fossil fuels and raw materials in economic ore deposits, and most importantly the transition to renewable energy rely on a good understanding of the physics of geological processes (Feng et al., 2021; Vehling et al., 2020; Weis et al., 2012). In many cases, numerical models that solve the mathematical equations governing the physics are the only way to predict such natural processes. Even though most coupled natural processes are nonequilibrium processes, the local equilibrium thermodynamic assumption is needed to close the system of mass, momentum, and energy equations and construct the model (De Groot &

Mazur, 1984; Lebon et al., 2008). Without the knowledge of thermodynamic behavior of rocks, minerals, melts, fluids, and gases involved in the process quantitative and predictive models cannot be constructed.

Transport processes such as diffusion, advection, and deformation, play a major role in all the important topics and are nearly always coupled to chemical reactions. Thus, reactive transport models have been a subject of geoscience research for many years (Steefel, 2019; Steefel et al., 2005). A large number of reactive transport codes has been developed particularly with the focus of subsurface processes (Kulik et al., 2012; Steefel et al., 2014) and applied to relevant geological problems in the subsurface (e.g., Sonnenthal et al., 2005; Wanner et al., 2014). A thorough review of the vast amount of work on reactive transport would be too extensive for the scope of this paper. For an overview of the main reactive transport codes in use for subsurface processes see Steefel et al. (2014).

Natural observations on, for example, exposed mantle rocks exhumed from great depth can be used to learn something about natural processes of CO_2 sequestration (Kelemen & Matter, 2008) and may lead to new insights for human carbon storage. In this context, knowledge of the thermodynamic behavior of minerals is also important for industrial processes and may lead to useful discoveries for example for carbon sequestration (Glasser et al., 2016).

Large-scale geodynamic processes influence processes at the Earth's surface and play a role in the global geochemical cycles, the water budget, melt generation at ocean ridges and in collision zones, and erosion in mountain belts. In subduction zones, these processes are closely interconnected as dehydration of hydrated oceanic lithosphere generates fluids that may trigger earthquakes and induce melting in the mantle wedge that may ultimately lead to volcanism at the surface, including direct effects on climate (Bebout et al., 2018). Numerical models have been useful in gaining more understanding of the physical processes of migration of fluids from the subducting slab upwards (Tian, Katz, Rees Jones, & May, 2019; Wilson et al., 2014). Migration and melt focusing due to reaction infiltration instabilities have been studied by Aharonov et al. (1995). Connolly (1997) showed how coupling of dehydration reactions produces fluids that travel up in a viscoelastic matrix creating porosity waves as a mechanism for fluid migration. Balashov and Yardley (1998) investigated the effect of reaction on porosity structure and fluid pressure variation. More recent models further investigated fluid focusing, transport, and reaction processes (Connolly & Podladchikov, 2007; Tian & Ague, 2014; Tian et al., 2018; Utkin & Afanasyev, 2021).

Recent progress in some of the important topics above has been made possible by coupling the local equilibrium thermodynamics to transport codes. Plümper et al. (2017) showed how the initialization of dehydration in subduction zones is dominated by chemical heterogeneities that control the thermodynamics on the microscale. Coupled to mass conservation and fluid flow, these local chemical heterogeneities lead from an initially formed local porosity to the development of fluid pathways.

The timescales of geological processes can vary between milliseconds to billions of years (Beinlich et al., 2020). With the use of numerical models combined with field observations, the duration of geological processes originally thought to be on the geological timescales have been shown to be much faster (Ague & Baxter, 2007; Beinlich et al., 2020; John et al., 2012). Therefore, the prediction of the speed of reaction fronts, and elemental transport hinges on quantitative models obeying mass, momentum, and energy conservation. Beinlich et al. (2020) used a reactive transport code coupled to local equilibrium thermodynamics to constrain the speed of natural CO_2 sequestration in serpentinites. The details behind the thermodynamic calculations in Beinlich et al. (2020) are given in this paper.

Complex solution models are almost exclusively used for phase diagram calculations (e.g., Connolly, 2005, 2009; de Capitani & Brown, 1987; Ghiorso et al., 2002; Holland & Powell, 1998, 2011; Johnson et al., 1992), but usually not for the transport codes. Nonideality of the solution models and its impact on the relation between equilibrium fluid and solid composition (the so-called "isotherms") has long been recognized as an important control on mass transport in porous reactive flows (Fletcher & Hofmann, 1974; Guy, 1993; Hofmann, 1972; Lichtner & Carey, 2006). However, it is underexplored mostly due to lack of transport models coupled to thermodynamically complex nonideal models including coupled substitutions, non-convex energy functions, and ordering.

We present a numerical thermodynamics laboratory which we shall call "Thermolab," to do thermodynamic calculations on rocks, minerals, melts, and fluids involving aqueous species. We further show how this can be used in reactive transport models that are based on laws of mass, momentum, and energy conservation. Code examples are written in MATLAB/OCTAVE as it provides a transparent way of translating the mathematics and

documenting the algorithms. Matlab also provides a prototype coding development platform. Short MATLAB codes serve as "flow charts" used in the past to document the algorithms. The advantage of the MATLAB/OCTAVE codes compared to the "flow charts" is that those are actually working computer programs, and can be copy-pasted from the figure and executed. Once working these codes can also be easily translated to any preferred programming language for optimization and supercomputing. The focus in this contribution is entirely on the technicalities with limited example applications. We document the nonideality of the models and the impact on nonlinear transport processes in open systems by showing examples of how this can be achieved with Thermolab. Future studies will be investigating the impact of the nonideality on mass transport.

2. Background and Motivation

A typical set of equations that governs the physics of the reactive transport based on Beinlich et al. (2020) is shown to motivate the use of equilibrium thermodynamics in transport processes. We consider the one-dimensional transport of a mobile species described by the total mass concentration balance, assuming nonmoving, nondeforming solid (v_s (x,t) = 0):

$$\frac{\partial C_{tot}(x,t)}{\partial t} = -\left(\frac{\partial q_{C_f}(x,t)}{\partial x} + \frac{\partial \rho_f(x,t)\cdot C_f(x,t)\cdot q_D(x,t)}{\partial x}\right) \tag{1}$$

with total mass concentration in the time derivative above defined as:

$$C_{\text{tot}}(x,t) = \rho_f(x,t)\cdot C_f(x,t)\cdot \varphi(x,t) + \rho_s(x,t)\cdot C_s(x,t)\cdot(1-\varphi(x,t)) \tag{2}$$

where ρ_f and ρ_s are density of fluid and solid, respectively, C_f and C_s are concentration in mass fraction of a species in fluid and solid, respectively, and ϕ is volume fraction of fluid, all varying as a function of time and space.

In Equation 1, the Darcy flux, q_D and q_{Cf} diffusion flux in fluid are given by:

$$\begin{aligned} q_D(x,t) &= -k(x,t)\cdot\frac{\partial P_f(x,t)}{\partial x} \\ q_{C_f}(x,t) &= -D_f^{\text{eff}}(x,t)\cdot\frac{\partial \mu_{C_f}(x,t)}{\partial x} \end{aligned} \tag{3}$$

where, P_f is fluid pressure, and μ_{C_f} is chemical potential of the species in the fluid. Gravitational effects in the Darcy flux have been neglected. Furthermore, the transient permeability k and effective diffusivity D_f^{eff} of the mobile species are defined as:

$$\begin{aligned} k(x,t) &= \frac{k^0\cdot\phi(x,t)^3}{\eta_f} \\ D_f^{\text{eff}}(x,t) &= \rho_f\cdot(x,t)\cdot C_f(x,t)\cdot\varphi(x,t)\cdot D_{C_f} \end{aligned} \tag{4}$$

where k^0 is background permeability, η_f is fluid viscosity, D_{C_f} is the diffusion coefficient of the species in the fluid, and the latter three parameters are assumed constant.

For this simplifying case, conservation of total mass of the system becomes:

$$\frac{\partial \rho_{\text{tot}}(x,t)}{\partial t} = -\frac{\partial \rho_f(x,t)\cdot q_D(x,t)}{\partial x} \tag{5}$$

with total mass density defined as:

$$\rho_{\text{tot}}(x,t) = \rho_f(x,t)\cdot\varphi(x,t) + \rho_s(x,t)\cdot(1-\varphi(x,t)) \tag{6}$$

Porosity evolution is governed by the time integrated mass conservation equation of an immobile species in the solid, ρ_{im} (e.g., Beinlich et al., 2020; Malvoisin et al., 2015; Plümper et al., 2017):

$$\varphi(x,t) = 1 - \frac{(1-\varphi(x,t_0))\cdot\rho_{\text{im}}(x,t_0)}{\rho_{\text{im}}(x,t)} \tag{7}$$

where $\phi(x,t_0)$ and $\rho_{\text{im}}(x,t_0)$ are the initial values of porosity and immobile species density at t_0 and may vary in space.

In Equation 7, the expression for $\rho_n(x,t)$ in Malvoisin et al. (2015) and Plümper et al. (2017), where fluid was pure water, is replaced with $\rho_{\text{im}}(x,t)$, which denotes the mass density of any immobile species in the solid:

$$\rho_{\text{im}}(x,t) = \rho_s(x,t) \cdot C_s^{\text{im}}(x,t) \tag{8}$$

The full set of equations above presents a system of three equations for eight unknowns: fluid and solid density (ρ_f), and (ρ_s), weight fraction of a mobile component in fluid and solid (C_f), and (C_s), chemical potential of the component in the fluid (μ_{Cf}), and the weight fraction of an immobile component in the solid (C_s^{im}), porosity (ϕ), and fluid pressure (P_f). All unknowns vary in time and space. Background permeability (k^0), fluid viscosity (η_f), and diffusion coefficient of mobile component C_f in the fluid (D_{C_f}) are given and kept constant. Porosity, fluid pressure, and either the fluid or the solid composition of each component in the system are not thermodynamically constrained and the three equations are solved for ϕ, P_f, and C_s. The remaining unknowns, solid and fluid compositions and densities, are found from thermodynamic relationships and in general vary as a function of temperature, pressure, and composition. The equations above imply the assumption of constant temperature, which may be justified in the case of a small modeling domain where temperature is nearly instantly homogenized. Furthermore, we neglect the effect of varying pressure on the thermodynamic properties arguably justified in the case pressure variations are kept small.

In complex mineral, melt, and fluid solutions, the nonideal mixing behavior leads to nonlinear thermodynamic relationships. Parameterization of the various variables to obtain analytical expressions to close the system of transport equations may be possible in some cases (Tian, Katz, & Rees Jones, 2019), but may be difficult or impossible due to the complex topologies and discontinuities of phase assemblage boundaries (Stixrude & Lithgow-Bertelloni, 2022). In that case these thermodynamic properties either have to be directly calculated (i.e., on-the fly) or a precomputed lookup table may be prepared to interpolate the needed values. Here, we compute these relationships with Thermolab, benchmarked with phase diagrams for rocks, fluids, and melts. Using MATLAB codes, the first part of this contribution documents the technical details of the equilibrium thermodynamic calculations and the treatment of solid solutions. This is considered essential to understand the effects on nonlinear transport processes. We include a description of methods to sufficiently resolve the equilibrium compositions and densities of the involved phases. This is needed to avoid numerical artifacts and instability in the transport codes. In the last part, we get back to a detailed example of the implementation and solution of the system of transport equations in which all quantities evolve through time. For a full list of symbols and notation see Table 1.

3. Gibbs Energy Calculation

The starting point of equilibrium thermodynamic calculations in Thermolab is the Gibbs energy of a mineral, melt, or fluid. All other required properties are derived from the Gibbs energy. The calculation of Gibbs energies requires in the first place an internally consistent thermodynamic data set for pure solids, fluids, gases, or aqueous species, often referred to as endmembers. Second, it needs solution models that describe the energy of mixing between endmembers dissolved in a phase, also referred to as mixing models or activity-composition relationships (e.g., Ganguly, 2020; Holland & Powell, 2003). Thermolab currently has built in several internally consistent thermodynamic data sets for minerals, melts, and fluids, the most extensive and up-to-date are the Holland and Powell endmember databases (Holland & Powell, 1998, 2011). The SUPCRT (dslop98) mineral database (Helgeson et al., 1978; Johnson et al., 1992) is implemented to allow calculations with additional minerals not included in the Holland and Powell data sets. For water (and/or CO_2), several equations of state (EOS) are implemented, including the EOS of the International Association for the Properties of Water and Steam (IAPWS) (Wagner & Pruss, 2002, revised version, 2018), the EOS of Johnson and Norton (1991), Z. G. Zhang and Duan (2005), C. Zhang and Duan (2009), Pitzer and Sterner (1994), and the CORK EOS, from Holland and Powell (1991). For aqueous species, we have implemented the Holland and Powell (1998) formulation, and the Tanger and Helgeson (1988) formulation as implemented in SUPCRT92 (Johnson et al., 1992) using the 1998 database including more than 1,300 aqueous species. Furthermore, it includes the Deep Earth Water (DEW) model (Sverjensky et al., 2014) with the recent updates and additional species from Huang and Sverjensky (2019), Aranovich et al. (2020), and the Miron data set for aqueous species (Miron et al., 2016). This forms

Table 1
List of Symbols and Abbreviations

Symbol	Meaning	Units	MATLAB
C_{tot}	Total mass concentration of a species	kg/m^3	**Ctot**
C_f	Concentration of species in fluid	Weight fraction	**Cf**
C_s	Concentration of species in solid	Weight fraction	**Cs**
$C_{s,\mathrm{im}}$	concentration of immobile species in solid	Weight fraction	**Cs_im**
t	Time	s	**time**
t_0	Initial t	s	**time**
x	Spatial coordinate in x dimension	m	**x**
ρ_f	Density of fluid	kg/m^3	**rhof**
ρ_s	Density of solid	kg/m^3	**rhos**
ρ_{tot}	Total mass density	kg/m^3	**rhotot**
ρ_{im}	Mass density of immobile species	kg/m^3	**rho_im**
Φ	Porosity (volume fraction of fluid)	Volume fraction	**phi**
k^0	Background permeability	m^2	**k0**
k	Permeability	m^2	**perm**
η_f	Fluid viscosity	Pa s	**rhotot**
P_f	Fluid pressure	Pa	**Pf**
D_{Cf}	Diffusion coefficient of species in fluid	m^2/s	**DC**
D_{cf}^{eff}	Effective diffusivity of species in fluid	m^2/s	**Deff**
μ_{Cf}	Chemical potential of species in fluid	J/mol	**mu**
g^0	Specific endmember Gibbs energy	J/mol	**g0**
g_m^0	Molality based endmember Gibbs energy	J/mol per kg water	**g0**
g_x^0	Mole fraction based endmember Gibbs energy	J/mol	**g0**
s^0	Specific endmember entropy	J/mol	**S**
h^0	Specific endmember enthalpy	J/mol	**H**
v^0	Specific endmember volume	J/bar	**v0**
g_{exc}^0	Specific endmember excess Gibbs energy term	J/mol	**Gexc**
T	Temperature	K	**T**
T_r	Reference temperature, 298.15°K	K	**Tr**
P	Pressure	Pa	**P**
P_r	Reference pressure, 10^5 Pa	Pa	**Pr**
R	Universal gas constant	J/mol/K	**R**
g_r^0	Specific endmember Gibbs energy at T_r, P_r	J/mol	**Gr**
g_{Pr}^0	Specific endmember Gibbs energy at T, P_r	J/mol	–
s_r^0	Entropy at T_r, P_r	J/mol	**Sr**
h_r^0	Enthalpy a T_r, P_r	J/mol	**Hr**
v_r^0	Volume at T_r, P_r	J/bar	**Vr**
v_{Pr}^0	Volume at T, P_r	J/bar	**V1_T**
a	Heat capacity coefficient	kJ/K	**a**
b	Heat capacity coefficient	kJ/K^2	**b**
c	Heat capacity coefficient	kJ K	**c**
d	Heat capacity coefficient	kJ K$^{-1/2}$	**d**
a_{MRK}	Fluid EOS coefficient	kJ^2 kbar K$^{1/2}$ mol^{-2}	**a**

Table 1
Continued

Symbol	Meaning	Units	MATLAB
b_{MRK}	Fluid EOS coefficient	kJ kbar^{-1} mol^{-1}	**b**
v_f^0	Specific volume of fluid	J/bar	**vf0**
K_{Pr}	Bulk modulus at T, P_r	kbar	**k**
K'	Derivative of bulk modulus		**kp**
c_P	Heat capacity	J/mol/K	**Cp**
$\boldsymbol{\nu}$	Stoichiometric coefficients in chemical reaction	mol	**v**
G_{sys}	Gibbs energy of system	J	**Gsys**
g	Specific Gibbs energy of a mixture	J/mol	**g**
g_{mech}	Specific Gibbs energy of a mechanical mixing	J/mol	**gmech**
g_{id}	Specific Gibbs energy of a ideal mixing	J/mol	**gid**
g_{nid}	Specific Gibbs energy of a nonideal mixing	J/mol	**gnid**
p	Vector of proportions of endmember in a mixture	mol fraction	**p**
m	Site multiplicity vector in ideal mixing energy	mol	**mtpl**
z	Vector of site fraction in a mixture	mol fraction	**z**
$\mathbf{S_t}$	Site occupancy table	mol	**st**
$\mathbf{Z_t}$	Site fraction endmember matrix	mol fraction	**zt**
M_w	Molar mass of water	kg/mol	**Mw**
K_{eq}	Equilibrium constant	–	**Keq**
ΔG^0_{rxn}	Difference in g^0 between products and reactants	J	**dGrxn**
α	Molar amount of phase from Gibbs minimization	mol	**alph**
$\boldsymbol{\alpha}$	Vector of molar amount of phases from linprog	mol	**alph**
$\mathbf{N_{phs}}$	Matrix of composition of phases in mol	mol	**Nphs**
$\mathbf{n_{sys}}$	Vector of system compositions in mol	mol	**nsys**
V^{mol}	Molar volume of phase	m^3/mol	**Vmol**
m^{mol}	Molar mass of phase	kg/mol	**Mmol**
$\mathbf{m_{mol}}$	Molar mass of phase vector	kg/mol	**Mmol**
molm	Molar mass of components vector	kg/mol	**molm**
ρ	Density of a phase	kg/m^3	**rho**
ϕ	Volume fraction of phase	–	**phi**
ϕ^{mol}	Mole fraction of phase	–	**phim**
$\boldsymbol{\phi}^{\mathbf{mol}}$	Vector of mole fraction of phases	–	**phim**
ϕ^{wt}	Weight fraction of phase	–	**phiw**
$\boldsymbol{\rho}_{\mathbf{ss}}$	Vector of densities of solids	kg/m^3	**rho(solid_id)**
$\boldsymbol{\phi}_{\mathbf{ss}}$	Vector of volume fraction of solids	–	**phi(solid_id)**
μ_s^A	Chemical potential of component A in solid	J/mol	**muA**
μ_s^B	Chemical potential of component B in solid	J/mol	**muB**
C_s^A	Concentration of component A in solid 1	mol or mass fraction	**c**
C_s^B	Concentration of component B in solid 2	mol or mass fraction	**(1-c)**
D_{Cs}	Diffusion coefficient in solid	m^2/s	**Dc**
γ	Cahn-Hilliard surface energy parameter	m^2 J/mol	**gam**
$\odot$	Element-wise multiplication operator		

Note. Symbols in italics for scalar variables, bold font for vectors, and bold capitals to denote matrices.

the endmember basis for use in thermodynamics of mixtures and for the local equilibrium calculations. In the future, this can be extended with additional thermodynamic endmember data sets. The following documents the main code as shown in an example in Figure 1. The mathematical notation follows the convention that bold font variables present vectors, bold font capital variables represent matrices, and all others are scalars variables. Meaning of symbols and variables are summarized in Table 1.

```
1 clear
2 %% Input
3 T       = 995.3340;                                                    % T in Kelvin
4 P       = 0.2e9;                                                       % P in Pascal
5 solname = 'Amphibole';                                                 % Name of the phase
6 solfile = 'solution_models_H18';                                       % Solution model file name
7 Cname   = {'Si'  ,'Ti'  ,'Al'  ,'Ca',  'Fe',  'Mg',  'Na'  ,'K'   ,'H'   ,'O'    }; % System component names
8 Cmol    = [7.0006,0.1054,1.3603,1.6786,1.8705,2.9542,0.3382,0.0237,1.8946,24.0500]; % Composition in mol
9 %% Load thermodynamic data
10 [p_id,st,site_id,mtpl,mod_id,alp,w,n_em,chg,em_data,dGex,EOS,CEOS,mcoef,Gr] = init_phase(solfile,solname,Cname);
11 %% Water properties
12 rho_w   = rho_H2O(T,P,'ZD05');       % Water density (ZD05= Zhang & Duan (2005))
13 eps_di  = eps_H2O(T,P,rho_w,'S14');  % Dielectric constant (S14 = Sverjensky et al.2014)
14 %% Endmember Gibbs energies
15 for k = 1:length(em_data)
16     Gjp = zeros(1,size(em_data{k},1));                                % Initialize G of make endmembers
17     for jp = 1:size(em_data{k},1)                                     % For endmembers made of multiple phases
18         [SdT,S]   = intSdT(T,P,em_data{k}(jp,:),CEOS{k}(jp));         % SdT integral, (eq. 10&11)
19         [VdP,V]   = intVdP(T,P,em_data{k}(jp,:), EOS{k}(jp));         % VdP integral, (eq. 12)
20         Gexc      = g0_exc(T,P,em_data{k}(jp,:),dGex{k}(jp),rho_w,eps_di); % dG of any fitting excess energy (for eq. 12)
21         Gjp(:,jp) = Gr{k}(jp) - SdT + VdP + Gexc;                     % Eq. 12
22     end
23     g0(k) = mcoef{k}*Gjp';                                            % Eq. 17
24 end
25 %% Prepare conversion matrices
26 zt                              = st2zt(st,site_id);              % Site fraction table, (eq. 25)
27 [p_from_c_cons,icomp_indep,isite_indep] = comp2prop(n_em',zt');     % C variables, p to C conversion (Eq.41-43 & App. A)
28 %% Grid for order-disorder
29 [z_od1,z_od2]                    = ndgrid(linspace(0,1,100),linspace(0,1,100));    % all possible order-disorder states
30 z_od                             = [z_od1(:),z_od2(:)];                            % order-disorder state matrix
31 %% Proportions and site fractions
32 p  = (p_from_c_cons*[ones(length(z_od),1) repmat(Cmol(icomp_indep),length(z_od),1) z_od]')'; % Eq. 43 & App.A
33 z  = p*zt;                                                                          % Eq. 22 (site fractions)
34 %% Numerical
35 z_tol = 1e-10;z(z<1+z_tol & z>1-z_tol) = 1;z(z<  z_tol & z> -z_tol) = 1e-20;   % round off errors
36 badz  = min(z,[],2)<0|max(z,[],2)>1;p(badz,:) = [];z(badz,:) = [];             % remove unrealistic compositions
37 %% Gibbs energy of mixing
38 g_mech = p*g0';                                                                % Mechanical mixing, eq. 19
39 g_id   = (T*8.3145*(mtpl*(z'.*log(z'+double(z'==0))- (zt'.*log(zt'+double(zt'==0)))*p')))'; % Ideal mixing, eq. 20
40 g_nid  = gnid(T,P,p,mod_id,V,alp,w,rho_w,eps_di,chg);                          % Non-ideal mixing. eq. 21
41 g      = g_mech + g_id + g_nid;                                               % Gibbs energy in Joule/mol, eq. 18
42 %% Find stable configuration (homogeneous equilibrium)
43 [gmin,id] = min(g);            % minimum of Gibbs energy
44 p(id,:)                        % corresponding endmember proportions
```

Figure 1. Thermolab code example to calculate the Gibbs energies of a phase for a given composition at fixed temperature (T) and pressure (P). Here is an example for the amphibole model of Green et al. (2016). This code is a possible MATLAB translation of the equations and methods documented in the Gibbs energy calculation section of the main text. The first line in the code clears MATLAB memory and figure. In Lines 3 and 4, T and P are defined. Lines 5 and 6 specify, respectively, for which phase to calculate the Gibbs energy and the name of the Excel file in which the solution model data is stored. Line 7 specifies the name of the elements in the phase, and line 8 the corresponding composition in moles for which to calculate the Gibbs energy. Line 10 is a call to read the needed thermodynamic data. Lines 12 and 13 are needed for aqueous species and compute the density and dielectric constant of water. Lines 15–24 compute the Gibbs energy of the endmembers in the solution (e.g., Equations 9, 12 and 17). Proportions and site fractions are calculated in lines 26–33. Numerical errors are removed in lines 35 and 36. In Lines 29 and 30, a grid of all possible states of order-disorder is generated for the particular composition of the phase specified in line 8. For all these compositions the mechanical, ideal, and nonideal mixing energy is calculated in lines 38, 39, and 40, respectively. The nonideal Gibbs energy, line 40, is a call to an external function that comprises a collection of nonideal mixing functions that can be opted. Line 43 finds the Gibbs energy that is minimum and thus finds the state of order-disorder. Line 44 displays the corresponding proportions of the endmembers.

3.1. Endmembers

An internally consistent thermodynamic database contains the Gibbs energy of an endmember, for example, a mineral, gas, fluid, or melt species with a fixed composition at reference conditions (g_r^0). Alternatively, it holds enthalpy and entropy data from which the Gibbs energy of an endmember at reference conditions can be obtained:

$$g_r^0(T_r, P_r) = h_r^0(T_r, P_r) - T_r s_r^0(T_r, P_r) \tag{9}$$

where h_r^0 is enthalpy and s_r^0 is entropy at reference temperature $T_r = 298.15°\text{K}$ and pressure $P_r = 10^5$ Pa.

To calculate the Gibbs energy at elevated temperature the entropy is obtained by integration with respect to temperature at reference pressure ($P = P_r$):

$$s^0(T, P_r) = s_r^0(T_r, P_r) + \int_{T_r}^{T} \frac{c_P(T, P_r)}{T} dT \tag{10}$$

where c_P is the heat capacity at constant reference pressure (P_r).

Then, the integral of entropy at reference pressure with respect to temperature is subtracted from the Gibbs energy at reference conditions to get the Gibbs energy at reference pressure and elevated temperature:

$$g_{P_r}^0(T, P_r) = g_r^0(T_r, P_r) - \int_{T_r}^{T} s^0(T, P_r)\, dT \tag{11}$$

To obtain the Gibbs energy at elevated pressure the volume, v^0, at fixed temperature is integrated with respect to pressure and added to the Gibbs energy at reference pressure for each temperature:

$$g^0(T, P) = g_{P_r}^0(T, P_r) + \int_{P_r}^{P} v^0(T, P)\, dP + g_{\text{exc}}^0(T, P) \tag{12}$$

An additional temperature and pressure dependent excess Gibbs energy term (g_{exc}^0) is added to account for Landau phase transitions, ordering reactions, or in case of aqueous species for Gibbs energy of solvation.

3.1.1. Solids

An example of a heat capacity expression to be used in Equation 10 consists of a four-parameter (a, b, c, and d) polynomial fit to experimental heat capacity data as used in the data set of Holland and Powell (1998, 2011):

$$c_P(T, P_r) = a + bT + cT^{-2} + dT^{-1/2} \tag{13}$$

For the pressure dependence of Gibbs energy an equation of state (EOS) is used to relate volume to pressure. For example, the Murnaghan equation of state used by Holland and Powell (1998) (with K' set to 4):

$$P = P_r + \frac{K_{P_r}(T, P_r)}{K'} \left[\left(\frac{v_{P_r}^0(T, P_r)}{v^0(T, P)} \right)^{K'} - 1 \right] \tag{14}$$

where K_{P_r} is bulk modulus at reference pressure and $v_{P_r}^0$ is volume at reference pressure and given temperature.

Note that the reference pressure P_r is added to ensure the limit of standard reference conditions. The EOS can be rearranged for volume:

$$v^0(T, P) = v_r^0(T, P_r) \left[1 + (P - P_r) \left(\frac{K_{P_r}(T, P_r)}{K'} \right) \right]^{-\frac{1}{K'}} \tag{15}$$

Substituting Equations 9, 10, 11, and 15 into Equation 12, and using a suitable heat capacity formula such as Equation 13, the integrals can be evaluated analytically or numerically. Lines 18–21 in Figure 1 show a MATLAB example of using Equations 9–12. In the example, the integrals are evaluated in a separate function. To use a different thermodynamic database, the heat capacity equation and EOS can be replaced with the appropriate expression. A number of important endmembers such as quartz are treated with additional volume and entropy terms to account for first or second order phase transitions such as heat capacity anomalies or order-disorder in the crystal lattice (e.g., sillimanite Holland and Powell (1996)). This then requires an excess Gibbs energy contribution, which in principle can also be calculated according to Equation 12 using different volume and heat capacities and integration limits (e.g., Berman & Brown, 1985). Alternatively, an expression for this additional energy is given (Holland & Powell, 1998). In the example in Figure 1, this is represented by a call to a MATLAB function (line 20 in Figure 1). For the specific details, we refer to the original papers documenting the internally consistent databases (Berman, 1988; Holland & Powell, 1998, 2011; Johnson et al., 1992).

3.1.2. Fluids

For molecular fluids such as H_2O and CO_2, the EOS, for example, Equation 14, is replaced by an appropriate fluid EOS. As water is one of the most important fluids on our planet, extensive work has been done on the thermodynamic formulation. The main difference between solids is that the fluid EOS, especially for water, usually cannot easily be rearranged for volume as there are multiple volumes possible for a single pressure in the region of coexisting fluid and gas. An example of this is given by the Modified Redlich Kwong (MRK) EOS, on which Holland and Powell (1991) base the fluid Gibbs energies to be compatible with the extensively used thermodynamically consistent data set of Holland and Powell (1998):

$$P = \frac{RT}{v_f^0 - b_{MRK}} - \frac{a_{MRK}}{v_f^0\left(v_f^0 + b_{MRK}\right)\sqrt{T}} \tag{16}$$

Because this is a multivalued function a suitable algorithm must be used to calculate volume as function of *P*-*T* in the two-phase region (e.g., where gas and liquid coexist). In the single-phase region, the volume for the stable phase (gas or liquid) should be determined. Holland and Powell (1991) rearranged the equation as a cubic in volume after which the correct root must be found in each phase region. Rather than solving which phase (e.g., gas, liquid, or both) is stable, the different regions were predefined and the correct volume was selected based on the *P*-*T* conditions. The MRK formulation in Equation 16 is extended with a virial contribution and Holland and Powell (1991) give detailed instructions on how to calculate the volume as function of *P*-*T* for H_2O, CO_2, and several other COH species. The advantage of the CORK formulation is that it can be integrated analytically and details in the original paper of Holland and Powell (1991) are sufficient for reproducing the values for use in Thermolab. The more updated thermodynamically consistent data set of Holland and Powell (2011), which replaces the Holland and Powell (1998) data set, uses the EOS of Pitzer and Sterner (1994).

Calculating the Gibbs energy of fluids then follows Equations 9–12 as for solids, with an entropy integral that is consistent with the data set for solid, melt, and/or gas endmembers. The Holland and Powell (1998, 2011) data sets have parameters for heat capacity, fitted simultaneously with the minerals to ensure the Gibbs energy at room pressure and elevated temperature can be calculated consistently.

In principle, the CORK or Pitzer and Sterner (1994) EOS can be used in conjunction with the SUPCRT mineral database, however, for the entropy integral a thermodynamically consistent heat capacity formulation is needed to obtain valid thermodynamic calculations. The SUPCRT database does not contain heat capacity values for H_2O. The specific details needed for this calculation in the original papers of Johnson and Norton (1991) and Johnson et al. (1992) could not be found. Similarly, the DEW spreadsheet does not contain H_2O entries and combines Delany and Helgeson (1978) and an internal routine for use only above at least 0.1 GPa, likely as it is intended to be used for Deep Earth applications. Moreover, at low *P* and *T*, for example, for shallow processes, the CORK EOS should not be used (Holland & Powell, 1991). We found that by using the NIST Shomate heat capacity equations (Shomate, 1954), using parameters from the NIST website for liquid water and gas, the Gibbs energies retrieved from SUPCRT can be reproduced accurately. The simplicity of the formulation and up-to-date online documentation of the parameters is of advantage. For calculations at elevated *P*, the numerical integration of the

IAPWS or the EOS of Johnson and Norton (1991) give satisfactory results at low T, whereas the CORK EOS can be used above 100°C (Holland & Powell, 1991).

3.1.3. Aqueous Species

Gibbs energies of aqueous species are calculated in Thermolab following the formulation of Tanger and Helgeson (1988) as also outlined in Johnson et al. (1992). The resulting Gibbs energies have been compared to the output from SUPCRT and are reproduced within ∼1 J precision. The DEW model (Sverjensky et al., 2014) and the Miron data set (Miron et al., 2016, 2017) use the same formulation, with refitted parameters for some of the endmember species. Although fundamentally the aqueous species are also calculated according to Equation 12, there is a difference to the solid, melt, and fluid endmembers because the aqueous species need the density and dielectric constant of the solvent. The contribution to Gibbs energy of solvation therefore needs the density and dielectric constant of water for the Born equation (Figure 1, Lines 12–13). This contribution is included in the excess Gibbs energy term (Figure 1, Line 20). Furthermore, it should be noted that these Gibbs energies are usually on a molality based scale. Note that in the example for amphibole in Figure 1, the properties of water are irrelevant, but to maintain some generality they have been left in the example. When computing the Gibbs energy of aqueous species, we change the name of the phase from "Amphibole" into any desired aqueous species or fluid mixture defined in a solution model database file. In Thermolab, the user can choose from a number of EOS and dielectric constants to be used in the aqueous species Gibbs energy (Fernández et al., 1997; Johnson & Norton, 1991; Sverjensky et al., 2014). The aqueous species endmembers of Holland and Powell (1998) offer a restricted set of species compared to the SUPCRT data, but it has the advantage that it is fitted in the internally consistent data set of the mineral, melt, and gas endmembers of Holland and Powell (1998, 2011). Their formulation is based on the Anderson density equation (Anderson et al., 1991) and uses the CORK EOS for the water density. These aqueous species endmember data can also be used in fluid mixtures (Evans & Powell, 2007).

3.1.4. Dependent Endmembers

Endmembers can also be formed from a linear combination of several other endmembers (Figure 1, Line 23):

$$g^0_{\mathrm{dep}} = \nu \cdot \mathbf{g}^0 \tag{17}$$

where the $\boldsymbol{\nu}$ holds the stoichiometric reaction coefficients. A formation energy may be associated with such a reaction and this can be captured in any functional form in the excess Gibbs energy term of Equation 12.

3.2. Gibbs Energy of Mixtures

The Gibbs energy of a phase (mineral, gas, fluid, or melt species) that can form a mixture between several endmembers is represented by the sum of mechanical, ideal, and nonideal mixing energies (Figure 1, Line 41):

$$g = g_{\mathrm{mech}} + g_{\mathrm{id}} + g_{\mathrm{nid}} \tag{18}$$

The mechanical part of the mixing is the sum of the Gibbs energy of all endmembers in the mixture ($\mathbf{g^0}$), weighted by their proportions ($\mathbf{p}$) (Figure 1, Line 38).

$$g_{mech} = \mathbf{p} \cdot \mathbf{g}^0 \tag{19}$$

The ideal mixing Gibbs energy is given by a linear combination of the site fractions ($\mathbf{z}$) in the mixture multiplied with their logarithm and the site multiplicity (m) (Figure 1, Line 39):

$$g_{id} = RT\mathbf{m} \cdot (\mathbf{z} \odot \ln(\mathbf{z}) - \mathbf{Z_t} \odot \ln(\mathbf{Z_t}) \cdot \mathbf{p}) \tag{20}$$

where the symbol $\odot$ indicates elementwise multiplication of vectors and matrices.

Using the site fractions of the endmembers ($\mathbf{Zt}$), the last term in Equation 20 ensures that the ideal Gibbs energy is zero in the limit of the pure endmember. This for example occurs when a solution model is defined with certain

endmembers having site fractions between 0 and 1 (e.g., Appendix in Tajčmanová et al., 2009). How to obtain **m**, **p**, **z,** and **Zt** is explained in the following section.

Nonideal Gibbs energy can be calculated in the simplest way with a binary mixing formula:

$$g_{\text{nid}} = \mathbf{p}^{\mathbf{T}} \cdot \mathbf{W} \cdot \mathbf{p} \tag{21}$$

This is essentially a sum of multiplications of binary pairs of endmembers in the mixture multiplied with their interaction parameters. These interaction parameters can depend on composition to get an asymmetric mixing formulation, referred to as subregular mixing model (Ganguly, 2020). When **p** is replaced with a size parameter adjusted proportion and using the appropriate **W** the asymmetric formulism of Holland and Powell (2003) can be used. If the molar volumes of the endmembers are used as size parameter, it is essentially a Van Laar mixing model (Van Laar, 1906), as for example, used by Aranovich and Newton (1999). The **W** and any size parameters are in general also temperature and pressure dependent such as also the molar volume used in Van Laar mixing models. Any other regular or subregular solution models can be used in place of Equation 21 and ternary interaction terms may be added. In Thermolab, it is possible to expand the codes with a variety of mixing formulas for the nonideality by adding the appropriate formulation to the function called in line 40 in Figure 1.

3.3. Solution Models

The data for **m** and $\mathbf{Z_t}$ are retrieved from a site occupancy table, stored for example, in an Excel spreadsheet. In this spreadsheet, also data for **W** and any size parameters for asymmetric formalism models (Holland & Powell, 2003) and the model type are stored. In Figure 1, the data is loaded in the beginning of the code (Line 10, Figure 1) from a function that reads the Excel data. Site fractions **z** are obtained by multiplying the transpose of the site fraction speciation matrix with a column vector of proportions:

$$\mathbf{z} = \mathbf{Z}_\mathbf{t}^T \cdot \mathbf{p} \tag{22}$$

After specifying proportions of the endmembers in a solution, the Gibbs energy of that particular phase and composition at a given P and T can be calculated from the above.

3.3.1. Site Occupancy

The site occupancy of the mixture can be represented by a table listing the occupancy of crystallographic sites for each endmember. For example, a binary olivine solution with only one crystallographic site M, and a fixed silica tetrahedral site (which has composition SiO_4), is shown in Table 2. It can be represented by a matrix:

$$\mathbf{S_t} = \begin{bmatrix} 2 & 0 & 1 \\ 0 & 2 & 1 \end{bmatrix} \tag{23}$$

The columns represent occupancy of each species on a crystallographic site for the endmember in the rows of the table. Site fractions **z** can be retrieved from this table by dividing each site occupancy over the sum of moles of the species on that site. Unless defined otherwise by the authors of a solution model, the sum of moles of all species on each site gives the site multiplicity, and can be represented by a vector:

$$\mathbf{m} = \begin{bmatrix} 2 & 2 & 1 \end{bmatrix} \tag{24}$$

As the site multiplicity data is not always determined from $\mathbf{S_t}$ because of decisions by the authors of the solution model, we store the multiplicity values in the spreadsheet. Examples where the theoretical site multiplicity is replaced by a different value is for example the T1 site in Amphibole (Diener et al., 2007), where the authors have taken 1 instead of 4 (Appendix A).

Table 2
Binary Olivine Site Occupancy

Site	M		T
Occupancy	Mg	Fe	SiO_4
Endmember			
Forsterite	2	0	1
Fayalite	0	2	1

Table 3
Binary Olivine Site Fraction Table

Site	M		T
Occupancy	Mg	Fe	SiO_4
Site multiplicity	*2*	*2*	*1*
Endmember			
Forsterite	1	0	1
Fayalite	0	1	1

Elementwise division of each row in Table 2 or in matrix Equation 23 with this multiplicity vector leads to a table of site fractions represented by the matrix (Table 3):

$$\mathbf{Z_t} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} \tag{25}$$

Using Equation 22 and spelling this out in matrix-vector notation, it is essentially a set of equations where the coefficients in front of the proportions are directly read from the site fraction table (Table 3) and then transposed:

$$\begin{bmatrix} z^M_{\mathrm{Mg}} \\ z^M_{\mathrm{Fe}} \\ z^T_{\mathrm{SiO_4}} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} p_{fo} \\ p_{fa} \end{bmatrix} \tag{26}$$

The site fractions become:

$$\begin{aligned} z^M_{\mathrm{Mg}} &= p_{fo} \\ z^M_{\mathrm{Fe}} &= p_{fa} \\ z^T_{\mathrm{SiO_4}} &= p_{fo} + p_{fa} \end{aligned} \tag{27}$$

3.3.2. Ordering

In fact, in olivine, Mg and Fe can interchange on two molecular sites M1 and M2 (Cemič, 2005). This exchange can take place while total chemistry of the olivine is fixed. To account for variations in distribution of Mg and Fe over the M1 and M2 sites a third endmember to account for the ordering on the sites is introduced. Introducing the endmember "Ordered Olivine" which has a composition composed of 0.5 mol forsterite and 0.5 mol fayalite and having an "ordered" configuration with all Mg on M1 and all Fe on M2. We may write the site speciation as shown in Table 4.

The Gibbs energy of this endmember can be made out of two endmembers, 0.5 mol of forsterite and 0.5 mol of fayalite and an energy of reaction may be added.

After this, we get three proportions that define the Gibbs energy. To reach the configuration with all Fe on M1 and all Mg on M2, a negative proportion of the ordered olivine must be used (Powell & Holland, 1999).

Table 4
Site Occupancy Table of Olivine With Ordering of Fe-Mg on M1 and M2 Sites

Site	M1		M2		T
Occupancy	Mg	Fe	Mg	Fe	SiO_4
Endmember					
Forsterite (fo)	1	0	1	0	1
Fayalite (fa)	0	1	0	1	1
Ordered Olivine (od)	1	0	0	1	1

Using the speciation matrix and Equation 22 we get:

$$\begin{bmatrix} z^{M1}_{\mathrm{Mg}} \\ z^{M1}_{\mathrm{Fe}} \\ z^{M2}_{\mathrm{Mg}} \\ z^{M2}_{\mathrm{Fe}} \\ z^T_{\mathrm{SiO_4}} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} p_{fo} \\ p_{fa} \\ p_{od} \end{bmatrix} \tag{28}$$

Giving the site fractions:

$$
\begin{aligned}
z_{\mathrm{Mg}}^{M1} &= p_{fo} + p_{od} \\
z_{\mathrm{Fe}}^{M1} &= p_{fa} \\
z_{\mathrm{Mg}}^{M2} &= p_{fo} \\
z_{\mathrm{Fe}}^{M2} &= p_{fa} + p_{od} \\
z_{\mathrm{SiO_4}}^{T} &= p_{fo} + p_{fa} + p_{od}
\end{aligned}
\tag{29}
$$

3.3.3. Worked Example

Using the multiplicities and site fractions for the olivine example without order-disorder we get using Equation 20:

$$
g_{id} = R\,T\,2\,\left(z_{\mathrm{Mg}}^{M} \cdot \ln\left(z_{\mathrm{Mg}}^{M}\right) + z_{\mathrm{Fe}}^{M} \cdot \ln\left(z_{\mathrm{Fe}}^{M}\right)\right) \tag{30}
$$

(Note that the T-site fraction cancels due to the logarithms because the site fraction is always 1).

From Equation 21, the nonideal mixing energy becomes:

$$
g_{\mathrm{nid}} = \left(w_{fo,fa} + w_{fa,fo}\right) p_{fo} p_{fa} \tag{31}
$$

In case we use a Holland and Powell (1998) model for olivine, the nonideal parameters are equal to each other and reduce to the simple binary symmetric mixing parabola:

$$
g_{nid} = 2 w_{fo,fa} \left(1 - p_{fo}\right) p_{fo} \tag{32}
$$

For the example with order-disorder the ideal mixing Gibbs energy becomes:

$$
g_{id} = R\,T\left(z_{\mathrm{Mg}}^{M1} \ln\left(z_{\mathrm{Mg}}^{M1}\right) + z_{\mathrm{Fe}}^{M1} \ln\left(z_{\mathrm{Fe}}^{M1}\right) + z_{\mathrm{Mg}}^{M2} \ln\left(z_{\mathrm{Mg}}^{M2}\right) + z_{\mathrm{Fe}}^{M2} \ln\left(z_{\mathrm{Fe}}^{M2}\right)\right) \tag{33}
$$

Using Equation 21, with three endmembers, the nonideal Gibbs energy becomes:

$$
g_{\mathrm{nid}} = \left(w_{fo,fa} + w_{fa,fo}\right) p_{fo} p_{fa} + \left(w_{fo,od} + w_{od,fo}\right) p_{fo} p_{od} + \left(w_{fa,od} + w_{od,fa}\right) p_{fa} p_{od} \tag{34}
$$

3.3.4. Proportions From Site Fractions

The proportions can be written as function of site fractions by solving the system of equations in Equation 29 using the independent equations and the fact that proportions sum up to 1. Starting from the equation that proportions sum up to one, and adding sequentially only equations from Equation 29 that increase the rank gives a closed system of equations. This leads to:

$$
\begin{aligned}
1 &= p_{fo} + p_{fa} + p_{od} \\
z_{\mathrm{Mg}}^{M1} &= p_{fo} + p_{od} \\
z_{\mathrm{Mg}}^{M2} &= p_{fo}
\end{aligned}
\tag{35}
$$

Which after solving for proportions gives:

$$
\begin{aligned}
p_{fo} &= z_{\mathrm{Mg}}^{M2} \\
p_{fa} &= 1 - z_{\mathrm{Mg}}^{M1} \\
p_{od} &= z_{\mathrm{Mg}}^{M1} - z_{\mathrm{Mg}}^{M2}
\end{aligned}
\tag{36}
$$

Hence, there are two independent site fractions together completely defining the composition and ordering of the olivine in this example.

3.3.5. Proportions From Composition

From the olivine Table 4, we can derive a list of compositions of each endmember by summing up the moles of each atom over the sites.

Again, this table can serve to set up a system of equations and may be added by the constraint that the sum of the proportions equals 1 (first equation):

$$\begin{bmatrix} 1 & 1 & 1 \\ 2 & 0 & 1 \\ 0 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} p_{fo} \\ p_{fa} \\ p_{od} \end{bmatrix} = \begin{bmatrix} 1 \\ \mathrm{Mg} \\ \mathrm{Fe} \\ \mathrm{SiO_4} \end{bmatrix} \tag{37}$$

However, this table only has two independent equations (rank = 2). We need another equation to solve for all three proportions. A simple solution may be to treat the ordered endmember as a known variable on the right hand side. Our extended system can be written:

$$\begin{bmatrix} 1 & 1 & 1 \\ 2 & 0 & 1 \\ 0 & 2 & 1 \\ 1 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_{fo} \\ p_{fa} \\ p_{od} \end{bmatrix} = \begin{bmatrix} 1 \\ \mathrm{Mg} \\ \mathrm{Fe} \\ \mathrm{SiO_4} \\ p_{od} \end{bmatrix} \tag{38}$$

Starting from the first required equation and adding equations until we have reached the rank of the matrix gives us directly the first 2 and the last equation. This means the set of independent equations becomes:

$$\begin{bmatrix} 1 & 1 & 1 \\ 2 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_{fo} \\ p_{fa} \\ p_{od} \end{bmatrix} = \begin{bmatrix} 1 \\ \mathrm{Mg} \\ p_{od} \end{bmatrix} \tag{39}$$

The matrix in Equation 39 can be inverted and used to find the expressions for the endmember proportions as function of composition:

$$\begin{bmatrix} 0 & 0.5 & -0.5 \\ 1 & -0.5 & -0.5 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ \mathrm{Mg} \\ p_{od} \end{bmatrix} = \begin{bmatrix} p_{fo} \\ p_{fa} \\ p_{od} \end{bmatrix} \tag{40}$$

Instead of varying the ordered endmember proportion, it is also possible to vary one of the site fractions. For example, we can have the site fraction of Mg on M1 as variable in addition to the compositional variables. This way, we can fix the bulk composition of the mineral while varying the site occupancies due to ordering. To this end, we add the equation for site fraction of Mg on M1 (Equation 1 in Equation 29), to the original system of equations:

$$\begin{bmatrix} 1 & 1 & 1 \\ 2 & 0 & 1 \\ 0 & 2 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_{fo} \\ p_{fa} \\ p_{od} \end{bmatrix} = \begin{bmatrix} 1 \\ \mathrm{Mg} \\ \mathrm{Fe} \\ \mathrm{SiO_4} \\ z^{M1}_{\mathrm{Mg}} \end{bmatrix} \tag{41}$$

Here only the first, second, and last equation are independent and so we have:

$$\begin{aligned} p_{fo} + p_{fa} + p_{od} &= 1 \\ 2p_{fo} + p_{od} &= \mathrm{Mg} \\ p_{fo} + p_{od} &= z^{M1}_{\mathrm{Mg}} \end{aligned} \tag{42}$$

Solved for the three unknown proportions:

$$\begin{aligned} p_{fo} &= Mg - z^{M1}_{\mathrm{Mg}} \\ p_{fa} &= 1 - z^{M1}_{\mathrm{Mg}} \\ p_{od} &= 2z^{M1}_{\mathrm{Mg}} - \mathrm{Mg} \end{aligned} \tag{43}$$

These equations guarantee a fixed olivine composition, while changing the distribution of Fe and Mg on M1 and M2. The site fraction equation that will describe this distribution, or the order-disorder, can be obtained by augmenting the system of equations in Equation 41 with the site fraction equations in Equation 29. Then from this complete set of equations, only the independent equations are selected starting from the first equation in Equation 41 and adding only equations that increase the rank of the system of equations. Defining as first equation that the sum of proportions add to one (first row in Equation 41), is convenient as it guarantees the minimum amount of compositionally independent variables. A MATLAB code that automates this procedure is used in Thermolab and detailed in Appendix A. This function gives the independent components for each solution model as well as the site fraction to vary as order parameter and the matrix of converting composition to proportion. A worked example for a more complex mineral, Amphibole with 11 endmembers is given in Appendix A. It is also possible to calculate site fractions from bulk composition as shown in the Appendix in Vrijmoed and Podladchikov (2015).

3.3.6. Aqueous Fluids

For mixtures of aqueous species, an approach similar to the ideal olivine example can be taken. Treating the fluid mixture with a one-site model in which all species mix in a similar fashion the mechanical and ideal mixture is calculated as in Equations 19 and 20, and for the nonideality, the Helgeson Kirkham Flowers (HKF) extended Debye-Hückel activity is used (Dolejš & Wagner, 2008; Helgeson et al., 1981). The Gibbs energy of the fluid mixture is on a molality scale when the formulation of Johnson et al. (1992) and the HKF activity model (Helgeson et al., 1981) is used and must be converted to a molar based scale to be compatible with the solids. This is taken care of by adding a correction term to the molality based Gibbs energy of the pure aqueous endmember as in Dolejš (2013):

$$g^0_x = g^0_m + RT \ln\left(\frac{1}{M_w}\right) \tag{44}$$

where g^0_x is the mole fraction and g^0_m is the molality based Gibbs energy, M_w is the molar mass of H_2O. We note that in Equation 12, it is unspecified what concentration scale the Gibbs energy is based on. This information must be known a priori and the conversion above must be applied appropriately.

4. Equilibrium Calculations Using Reactions

Gibbs energies calculated with the Thermolab codes are benchmarked with THERMOCALC (Holland & Powell, 2011), SUPCRT92 (Johnson et al., 1992), the DEW spreadsheet (Sverjensky et al., 2014), or data and phase diagrams from the relevant publications (Miron et al., 2016) and Perple_X (Connolly, 2005). Additionally, comparing with experimental data is essential to increase confidence in the method.

With the Gibbs energies of endmembers, equilibrium conditions for simple reactions can be calculated. Results for several solid state and fluid-solid reactions are shown in Figure 2. The curves are produced by making use of the equilibrium condition:

$$0 = \Delta G^0_{rxn} + RT \ln\left(K_{\text{eq}}\right) \tag{45}$$

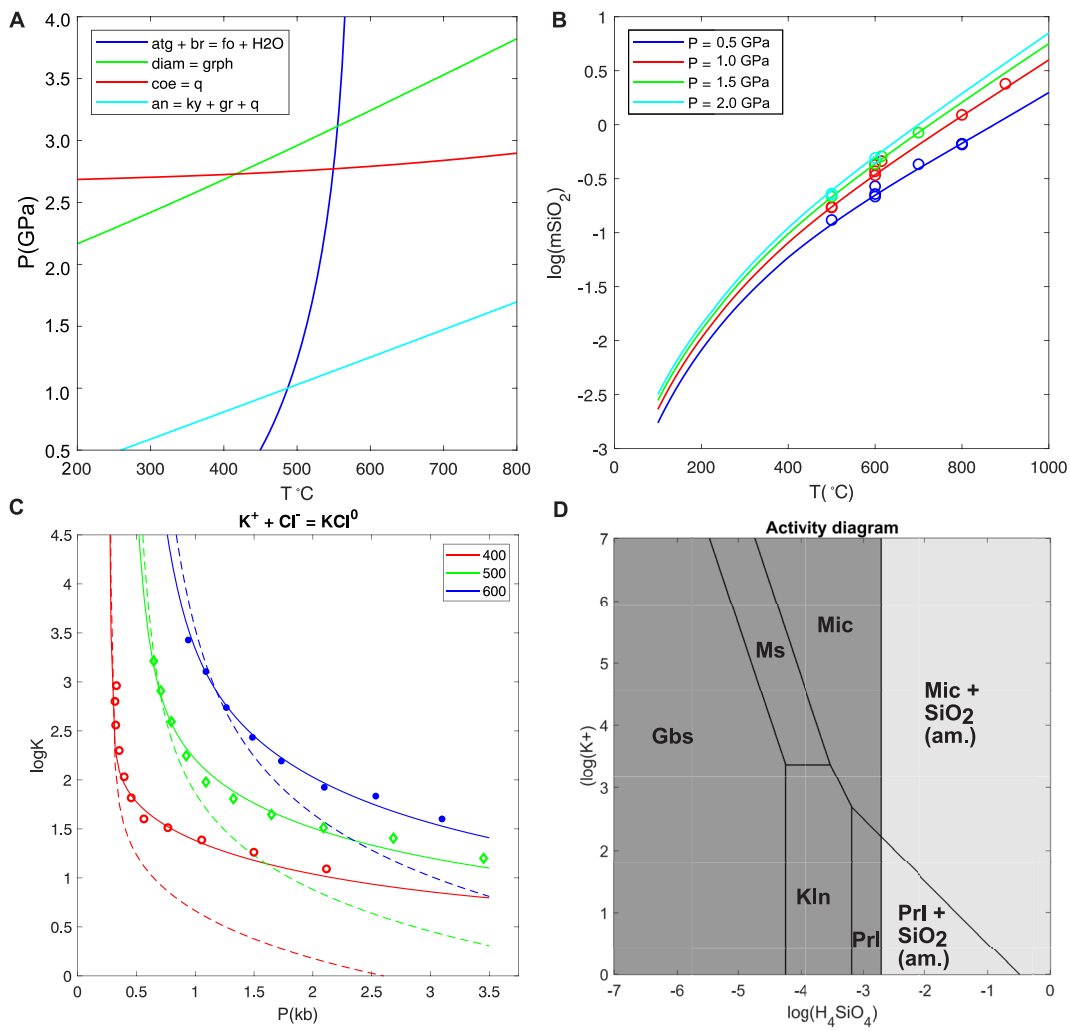


Figure 2. Example equilibrium calculations using a basic chemical reaction approach. (a) Examples of some well-known reactions between pure phases, by calculating ΔG of reaction (including any metastable parts of reactions). (b) Solubility of aqueous silica, by plotting the equilibrium constant of the dissolution of quartz into water using the Holland and Powell (1998) database. Comparison between data from Manning (1994) shows good agreement between model and experiment. (c) Comparison between experimental data and the SUPCRT (dashed lines) and Miron (solid lines) data set for the dissociation of KCl in water. This reproduces the figure shown by Miron et al. (2016), to demonstrate their improved fit to the data. (d) Activity diagram at room pressure, temperature, for basic weathering reactions using the equilibrium constants of the reaction calculated with Thermolab using the (dslop98) SUPCRT aqueous species and minerals data and Gibbs energy of water from NIST. Here a procedure to remove metastable extensions of reactions is performed in an automated way (i.e., automated Schreinemakers analysis). Gbs = Gibbsite, Ms = Muscovite, Mic = Microcline, Kln = Kaolinite, Prl = Pyrophyllite, SiO2 (am.) = amorphous silica.

with the change of Gibbs energy of the reaction given by:

$$\Delta G^0_{rxn} = \nu \cdot \mathbf{g}^0 \tag{46}$$

where the $\boldsymbol{\nu}$ hold the stoichiometric coefficients in the reaction and $\mathbf{g^0}$ are the endmember Gibbs energies of each phase in the reaction.

For pure phases (e.g., endmembers), the equilibrium constant K_{eq} will be 1 and the logarithmic term disappears leaving only $\Delta G^0_{rxn} = 0$. Using the endmember Gibbs energies and stoichiometric coefficient of reactions the contour line of $\Delta G^0_{rxn} = 0$ can be plotted to visualize the reaction in *P-T* space (Figure 2a). In a simple dissolution reaction K_{eq} will be equal to solubility and so we can use ΔG^0_{rxn} to calculate the solubility of quartz (Figure 2b). An example of a single dissociation reaction is shown in Figure 2c. By varying activities in K_{eq} and using them as axis on a diagram the contour lines where the right hand side is 0 are used to obtain an activity-activity plot (Figure 2d).

5. Gibbs Minimization in MATLAB

Using the ΔG of reaction, the metastable part of individual reactions will also be plotted (e.g., Figure 2a). For an equilibrium phase diagram, a Schreinemakers analysis or similar procedure may follow to draw the stable reactions (e.g., Figure 2d). Additionally, the phases and reactions to be calculated must be known beforehand. However, in many cases it can be desired to predict which reactions may take place. Instead of predicting the equilibrium line of a chemical reaction we calculate the stable phase assemblage at a point in *P-T* by determining which mineral has the minimum Gibbs energy. In the example code (Figure 3a), the calculation of Gibbs energy of the endmembers is done in line 9, a call to the main Thermolab function, which includes the code in Figure 1 to focus on the minimization algorithm. If all phases have the same composition (i.e., they are polymorphs), then the "min" function can be used (Figures 3a and 3b). Instead of locating the coexisting phases in a reaction in *P-T-X* space, the Gibbs minimization delivers the stable phases everywhere except for on the reaction line. Therefore, the reaction lines reflect the resolution of the *P-T* grid for which the Gibbs energies of each phase are calculated. At sufficiently high resolution, the reaction lines become smooth.

Natural chemical systems, for example, rocks, generally, are multicomponent systems. In this case, mass balance must be considered while finding the minimum of Gibbs energy. For a more thorough thermodynamic analysis of this, see for example, Connolly (2017). Constrained Gibbs energy minimization is employed in the majority of phase diagram calculation software (Connolly, 2009; Gordon & McBride, 1994). In MATLAB, the simplest way this can be achieved is using linear programming (function "linprog," e.g., Dantzig et al., 1955). With this function, minimization can be done under the constraints of mass balance equalities, which for completeness is shortly outlined in the following.

The function "linprog" is used to find the minimum Gibbs energy of the system by solving the following optimization problem,

$$\min\left(G_{\text{sys}}\right) = \mathbf{g} \cdot \alpha \tag{47}$$

where $\boldsymbol{\alpha}$ and $\mathbf{g}$ are vectors of the molar amount and the Gibbs energies of all phases, respectively. The "linprog" algorithm then searches for the components in $\boldsymbol{\alpha}$ that make the minimum Gibbs energy of the system, while respecting the following mass balance equality:

$$\mathbf{n_{sys}} = \mathbf{N_{phs}} \cdot \alpha \tag{48}$$

where $\mathbf{n_{sys}}$ is a vector of total moles of each component in the system and $\mathbf{N_{phs}}$ holds the molar composition of the phases as detailed below.

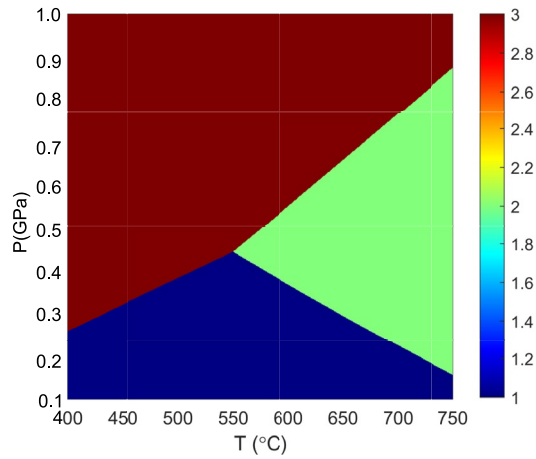
Equation 48 states that the sum of the amount of each phase multiplied by the composition of a component should equal the sum of that component in the system. A final requirement is that no negative amount $\boldsymbol{\alpha}$ is allowed as that is physically meaningless:

$$\alpha \geq 0 \tag{49}$$

A

```
1 clear,clf
2 % Input
3 T          = linspace(300,800,200)+ 273.15;              % Temperature (K)
4 P          = linspace(0.1e9,1e9,200);                    % Pressure (Pa)
5 phase      = {'ky,tc-ds55','sill,tc-ds55','and,tc-ds55'}; % Phase name
6 % Numerics
7 [T2d,P2d]  = ndgrid(T,P);                                % Create P-T grid
8 % Gibbs energy function
9 g          = tl_gibbs_energy(T2d(:),P2d(:),phase);       % G (J/mol)
10 % Find minimum of Gibbs energy and stable phase index (ind)
11 [gmin,ind] = min(g);
12 % Visualize stable phase
13 pcolor(T2d,P2d,reshape(ind,length(T),length(P))),colorbar,shading flat
```

B



C

```
1 clear,clf
2 % Input
3 T     = 1000+ 273.15;     % Temperature (K)
4 P     = 0.1e9;            % Pressure    (Pa)
5 phase = {'q,tc-ds55','per,tc-ds55','en,tc-ds55','fo,tc-ds55'};  % Phase na
6 g     = tl_gibbs_energy(T,P,phase);
7 Nsys = [1.2  % Si in systen
8         2.0  % Mg in systen
9         4.4];% O  in system
10 Nphs = [1 0 2 1  % Si in phases
11          0 1 2 2  % Mg in phases
12          2 1 6 4];% O  in phases
13 LB   = zeros(1,length(g));           % lower bounds for linprog minimiza
14 alph = linprog(g,[],[],Nphs,Nsys,LB); % constrained Gibbs energy minimiza
```

D

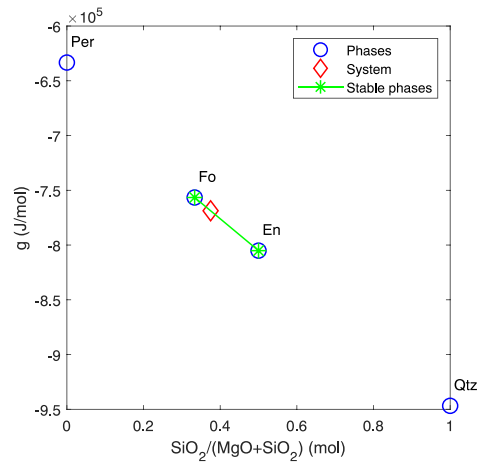


Figure 3. Two examples of Gibbs minimization to calculate phase equilibrium with Thermolab. The Gibbs energy calculation shown in Figure 1 is used here as a MATLAB function to improve clarity of the algorithm. (a) Code showing unconstrained Gibbs minimization used to calculate the aluminosilicate phase diagram. The Gibbs energies of the phases are obtained in line 9, using a unique identifier to distinguish them from the same phases in another database. Andalusite from the tc-ds55 THERMOCALC (Holland & Powell, 1998) data set (and, tc-ds55), kyanite (ky,tc-ds55), and sillimanite (sill,tc-ds55) in the order in which they are specified in line 5. The result is a 3D array with *T* in first dimension, *P* in second dimension, and phase in third dimension. Line 11 then finds the minimum in the third dimension, which results in a value (val) and index (ind) for each *P*-*T*. The index corresponds to the list of phases in line 5. (b) Resulting plot from code in panel (a). A color coded visualization of the phase index that gives the lowest Gibbs energy is shown using the pcolor function in MATLAB (code: line 13). The color bar indicates that the blue area has index 1, which then means it is the first phase in the list (line 5) that is stable (i.e., 1 = andalusite, 2 = kyanite, and 3 = sillimanite). The resolution of phase diagram can be increased by decreasing the step size of temperature and pressure (line 3 and 4). (c) MATLAB example to calculate chemical equilibrium using constrained Gibbs energy minimization. For the first lines see panel (a). In lines 7–9, the system composition is specified using total moles of SiO_2 and MgO. Then the phase compositions in moles is expressed in the matrix N_{phs} with in each column the phase corresponding to the list in line 5. Each row holds the components (SiO_2 and MgO). This information is used as equality constraints in the function linprog which is called in line 14. It then finds the minimum Gibbs energy of the system out of the four possible phases, while satisfying the equality constraints. The second and third input arguments for "linprog" are empty as they are reserved for inequalities and the fourth and fifth arguments are for the matrix of coefficients and right-hand side vector, respectively. To restrict the linear programming search for the amount of each phase to positive values we input also the lower bound vector of zeros (LB), for example, Equation 49. Then alph will hold the amount of each phase that will make minimum system Gibbs energy while obeying system composition. The phases that have an alph above zero represent the equilibrium assemblage for the given *P*-*T*-*X* conditions. (d) *G*-*X* diagram in which the Gibbs energy of the phases are plotted against system composition (for plotting the Gibbs energy is normalized over the sum of the moles of component in each phase: i.e., the sum over the rows of N_{phs}).

As a worked example we may consider the binary system SiO_2-MgO. Possible phases that can be built from these components are quartz, periclase, enstatite, and forsterite. Spelled out for this case the equations read in matrix form:

$$\min\left(G_{sys}\right) = \begin{bmatrix} g_{qtz} & g_{per} & g_{en} & g_{fo} \end{bmatrix} \begin{bmatrix} \alpha_{qtz} \\ \alpha_{per} \\ \alpha_{en} \\ \alpha_{fo} \end{bmatrix} \tag{50}$$

And

$$\begin{bmatrix} N_{sys,\mathrm{SiO_2}} \\ N_{sys,\mathrm{MgO}} \end{bmatrix} = \begin{bmatrix} N_{qtz,\mathrm{SiO_2}} & N_{per,\mathrm{SiO_2}} & N_{en,\mathrm{SiO_2}} & N_{fo,\mathrm{SiO_2}} \\ N_{qtz,\mathrm{MgO}} & N_{per,\mathrm{MgO}} & N_{en,\mathrm{MgO}} & N_{fo,\mathrm{MgO}} \end{bmatrix} \begin{bmatrix} \alpha_{qtz} \\ \alpha_{per} \\ \alpha_{en} \\ \alpha_{fo} \end{bmatrix} \tag{51}$$

Figure 3c shows the MATLAB translation and usage of "linprog" (Figure 3c, Line 14) to calculate the stable phase assemblage. The required known values for the minimization are the vector of Gibbs energies (Figure 3c, Line 6), a vector of system compositions (Figure 3c, Line 7) with matrix of mineral compositions (Figure 3c, Lines 10–12) as in Equation 51. Note that in Figure 3c single elements have been used as components instead of the oxides, which leads to the same result. After defining the lower bounds of alpha's as zero, linprog finds the alpha's that make the minimum of Gibbs energy. The results are plotted in Figure 3d and shows that for a system composition between enstatite and forsterite, alpha's of enstatite and forsterite are nonzero, which means that those are the stable phases and the magnitude of the alpha's give the molar amount of each stable phase in order of definition of the list of phases in Figure 3c, Line 5.

Performing a minimization at each point in a grid of pressure and temperature and evaluating at each *P-T* point the stable phase assemblage leads to a phase diagram in which also the amount of each phase is obtained from the mass balance constraint. Figure 4a shows that this produces the same result in case of the pure Al_2SiO_5 system, as in the unconstrained minimization, but that it can now be used to do multicomponent systems such as dehydration of antigorite in SiO_2, MgO, and H_2O (Figure 4b).

Most minerals, fluids, and melts are not just pure phases, but can form mixtures of endmembers. The basic example above is shown because essentially the Gibbs minimization approach used in Thermolab is the same

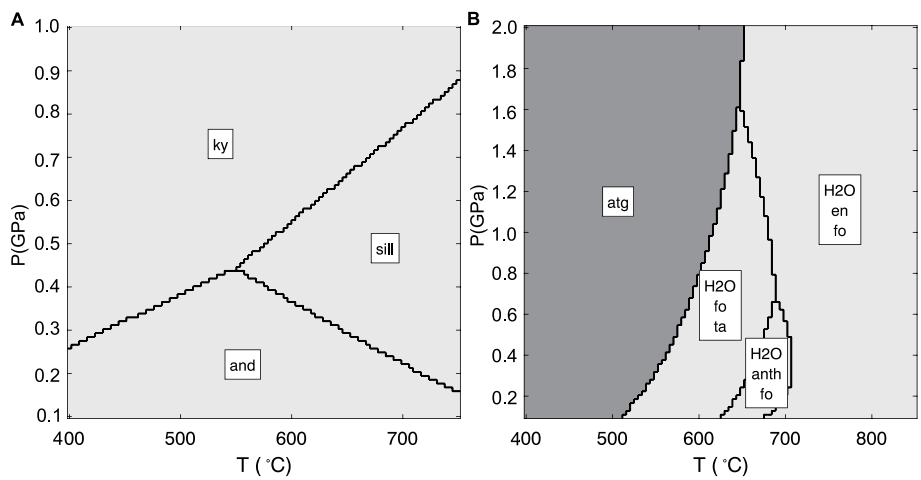


Figure 4. (a) One component system phase diagram, using constrained Gibbs minimization, results in a similar diagram to the unconstrained minimization result for the aluminosilicate phase diagram in Figure 3 (ky = kyanite, sill = sillimanite, and = andalusite). (b) Multicomponent system phase diagram, with pure phases, using components SiO_2, MgO, and H_2O. The same code is used as in panel (a), only the composition and system components need modification (atg = antigorite, fo = forsterite, ta = talc, and anth = anthophyllite).

for systems in which phases occur that have a variable composition. In the following we shortly outline how we approximate the equilibrium in Thermolab to a reasonable degree by using the same linear programming method.

5.1. Linearization of Mixtures

We compute the Gibbs energy of a set of discrete compositions of the mixture and add them to the list of endmembers as discrete entities, being treated as a fixed composition phase. In the following, all these entities that are the consequence of discretizing a real phase that can form a thermodynamic mixture will be arbitrarily called phase-compounds. With this approach the minimization algorithm remains unchanged from the above case for pure phases in Figure 3c. Figure 5 shows a complete code example to do a calculation with mineral solid solutions using the approach described here.

An important step in preparing the mixtures for Gibbs minimization is to generate a grid holding a set of discrete phase compositions representing the mineral, fluid, melt, or gas. The compositions need to cover the full range of possible site fractions or proportions that can exist for a mixture. Fixing the pressure and temperature, Gibbs energy can be calculated either by specifying the site fractions or the proportions. The simplest is to use the site fractions for grid generation because they range from zero to one. Using proportions is more complex because, although proportions sum up to one for each mixture, they can also be negative. Thus, it requires knowledge of the range of values for proportions in each different solution. The grid generation is done in the function call in Line 12, Figure 5, by varying the site fractions by default from 0 to 1, and generating a multidimensional Cartesian grid over the correct number of independent site fractions, using the MATLAB function "ndgrid" inside. With this brutal way, many nonphysical site fractions are generated, but these are removed inside the grid generation function. For example, on a crystallographic site with three site fractions, always one of them is dependent because the sum should equal one. So, there are two independent site fractions that both can vary between 0 and 1. When creating a Cartesian grid with "ndgrid," it is possible to have a combi-

```
1 clear
2 runname = 'example_soap';
3 T       = 300 + 273.15;                                          % T in Kelvin
4 P       = 0.3e9;                                                 % P in Pascal
5 solfile = 'solution_models_soapstone';                           % Name of solution model file
6 Cname   = {'Si'  ,'Al'  ,'Mg'  ,'Fe',  'Ca',  'C',   'H'   ,'O'   ,'e'}; % Components in the system
7 Xsys    = [0.7392;0.0451;1.0000;0.0644;0.0113;0.0232;1.3895;3.3629; 0];  % moles of each component in system
8 fluid   = 'Fluid-CO2-H2O(DEW)';                                  % Name of fluid model
9 phases  = {fluid,'Chlorite','Antigorite','Clinopyroxene','Talc','Magnesite','Dolomite','Brucite','Olivine',...
10           'per,tc-ds55','q,tc-ds55','ky,tc-ds55','sill,tc-ds55','and,tc-ds55'}; % Names of phases to consider
11 td      = init_thermo(phases,Cname,solfile);                     % Initialize phase data
12 p       = props_generate(td);                                    % Grid of compositions for mixtures
13 rho_w   = rho_H2O(T,P,'ZD05');                                   % Density of water
14 eps_w   = eps_H2O(T,P,rho_w,'S14');                              % Dielectric constant
15 [g0,v0] = tl_g0(T,P,td,rho_w,eps_w);                             % Endmember g and v in mixture
16 % Compute Gibbs energy
17 [g,Npc,pc_id] = tl_gibbs_energy(T,P,phases,td,p,g0,v0,rho_w,eps_w);  % Call Gibbs energy function
18 % Gibbs minimization
19 LB     = zeros(1,length(g));                                    % stable phase amount cannot be negative.
20 alph   = linprog(g,[],[],Npc,Xsys,LB);                          % The Gibbs energy minimization
21 % Save results
22 save([runname '_linprog']);
```

Figure 5. Complete example code for the calculation of chemical equilibrium using Gibbs energy minimization including solid solutions, aqueous species, and pure phases in Thermolab for the case of serpentinite with addition of carbon (Beinlich et al., 2020). After defining a *P* and *T* of interest, the solution model Excel spreadsheet is selected (line 5). The chemical components in the system are defined in line 6 and the composition of the system is specified in line 7. The names of the phases to be considered in the calculation are specified in lines 8–10, where the tc-ds55 identifies the thermodynamic data set to be used (here THERMOCALC data set 55). The static thermodynamic data (td) for all phases are loaded in line 11 and stored in one structure array (this replaces the command in Figure 1, line 10). The grid of compositions to linearize the solutions is obtained in line 12. Lines 13–14 compute the density (ZD05 = Z. G. Zhang & Duan, 2005) and dielectric constant of water (S14 = Sverjensky et al., 2014). Line 15 computes the Gibbs energy and volumes of the endmembers in each solution. Line 17 is the call to the main function outlined in Figure 1, and calculates for all phases in the list the Gibbs energy (*g*), the composition of each phase compound (Npc), an identifier for each phase compound (pc_id). Line 20 performs the constrained Gibbs minimization and in line 22 the data is stored for postprocessing.

nation of the two independent site fractions being for example, 0.9 and 0.5. This is not a feasible composition because the sum of those fractions exceeds 1. So, all combinations in the grid that produced values with a sum of the site fractions above one are excluded. In summary, we just generate Gibbs energies for the full range of possible site fractions in the solution including all states of ordering and internal speciation. For amphibole or clinopyroxene from for example, Green et al. (2016), only maximal 3–6 discrete compositions per dimension can be used, when executed on a single PC. The resulting set of Gibbs energies and compositions is used in place of the example in Figure 3c. The call to the main code "tl_gibbs_energy," in Figure 5 Line 17, now includes the complete Gibbs energy code example Figure 1.

As an example of including solutions, we show a variety of phase diagrams, all produced with the same code for a discrete set of *P-T* values. The first example in Figure 6a, applies the olivine solution model as described above together with a binary melt mixing model, replacing the solid endmembers with their liquid equivalents

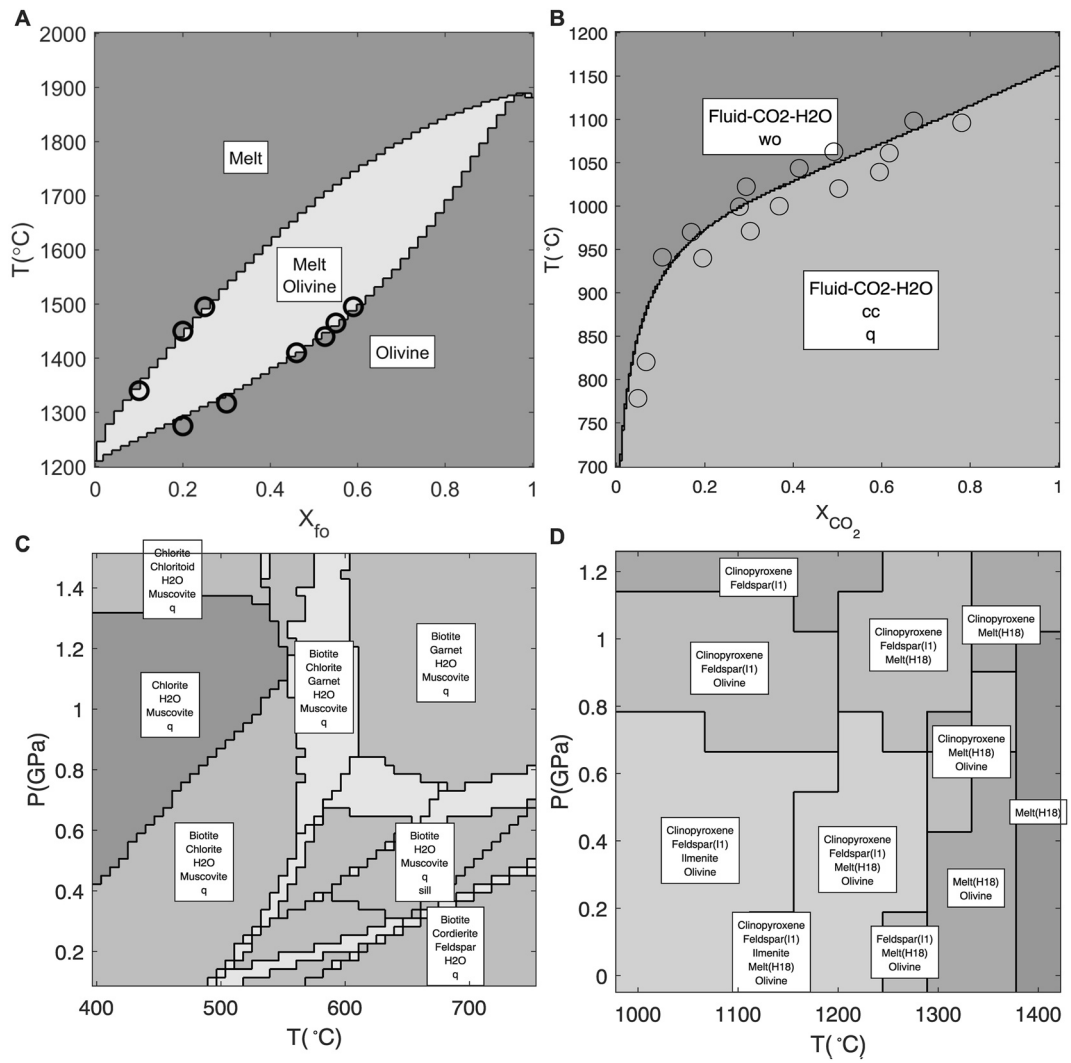


Figure 6. (a) *T-X* diagram for the binary system fo-fa, with olivine and melt as mixtures using the Holland and Powell (2011) data set and solution model for olivine from Holland et al. (2018) and melt using the example from the text and by adapting the Margules parameters to fit the Bowen and Schairer (1935) experiments. (b) T-X_{CO_2} diagram for CaO-SiO_2-H_2O-CO_2 system at 14 kbar. Data points from Aranovich and Newton (1999) as benchmark to the Gibbs energy and minimization approach in Thermolab. (c) Benchmark KFMASH metapelite example using the composition from the code in Figure 5. Solution models: Chlorite (Holland et al., 1998), Chloritoid, Cordierite (Holland & Powell, 1998), Muscovite (Coggon & Holland, 2002), Garnet, and Biotite (R. W. White et al., 2007). (d) Basalt melting to benchmark the Holland et al. (2018) melting model by reproducing main topology in Figure 5 of Holland et al. (2018), here calculated in SiO_2-TiO_2-Al_2O_3-MgO-FeO-CaO-Na_2O.

(liquid forsterite and fayalite). For olivine, a grid is created where z_{Mg}^{M1} and z_{Mg}^{M2} vary independently and for the melt, the site fraction of forsterite liquid is varied. The Margules parameters in the melt are adjusted until they fit the experimental data (Bowen & Schairer, 1935). The *T*-*X* diagram including CO_2 fluid using the mixing model of Aranovich and Newton (1999) is produced by changing the bulk CO_2 in the system. When the amount of fluid components (H_2O) in the calculation is orders of magnitude higher than the solid components, the diagram approximates the T-X_{CO2} diagram as the system is largely dominated by fluid. The results show good agreement with the experimental data from Aranovich and Newton (1999). A more complex calculation for a metapelite in KFMASH is done as a benchmark with the Perple_X and THERMOCALC diagrams and shows good agreement, improving the confidence of the method. Note that in complex systems, it becomes increasingly more demanding to get high-resolution phase diagrams. For example, a computation with the melt model of Holland et al. (2018) yields a low resolution diagram. Additionally, Cr_2O_3, Fe_2O_3, and K_2O were omitted. However, the topology of the diagram is overall similar. In particular, the melt and pyroxene models used in the reproduction of phase diagram for basalt melting are computationally challenging with the method of linearizing the solutions with discrete hypothetical phase compounds.

5.2. Postprocessing the Minimization Results

To precompute a lookup table for the local equilibrium the unknown variables need to be retrieved from the minimization results. The results of multiple computations can be saved in one data set to be postprocessed separately. An example for one "linprog" result is given in Figure 7.

Using the discrete phase compound linearization approach, mixtures (solid solutions, fluids, melts, and gas mixtures) are split into individual phases having a fixed composition. The minimization algorithm will determine which of those discrete phases are stable, and for mixtures this often results in multiple discrete phase compounds with different composition of that mixture being stable. This is a mathematical consequence of solving the optimization program and the thermodynamic meaning is that we have a divariant field in which the composition of mixtures may change. Thermodynamically, it can happen that two or three phases are stable as a result of a miscibility gap. However, this needs to be determined by an algorithm that distinguishes the discrete phase compound compositions from each other. If they are significantly different for a given resolution of the discretization, then they are true separate phases. In the other case, the properties are obtained from a weighted average into one composition for the true stable phase. Afterwards, only the stable phase amount and the composition are retained. A clustering algorithm is used to determine if we have multiple or single phases stable for each particular mixture (Figure 7, Line 10).

Considering a given equilibrium calculation with linprog, the main result is a vector alpha, which holds the total amount of mole of each phase that has been considered in the system. The clustering algorithm first takes care of all distinct phases and removes any phases with zero alpha (i.e., those phases are found not stable by the linprog algorithm).

Mole fraction amount of the *i*th stable phase in the system is found by normalizing the molar amount of stable phase by the total:

$$\varphi_i^{\mathrm{mol}} = \frac{\alpha_i}{\sum_{i=1}^{np} \alpha_i} \tag{52}$$

where *np* is the number of stable phases in this case.

Volume fraction ϕ of phase *i* is then found from molar volume and mole fraction amount:

$$\varphi_i = \frac{\varphi_i^{\mathrm{mol}} \cdot V_i^{\mathrm{mol}}}{\boldsymbol{\varphi}_{\mathbf{mol}} \cdot \mathbf{V}_{\mathbf{mol}}} \tag{53}$$

where the molar volume of each phase can be found using the numerical derivative of the Gibbs energy:

$$V_i^{\mathrm{mol}} = \left(\frac{\partial g_i}{\partial P} \right) \Big|_T \tag{54}$$

```
clear,addpath ../
runname = 'example_soap';
load([runname '_linprog']);                                          % load linprog run data
molm = molmass_fun(Cname);                                           % get molar mass of the components
% Numerics
delP     = 1e5;                                                      % for numerical differentiation
delc     = 1e-5;                                                     % for numerical differentiation
solv_tol = 2;                                                        % Tolerance for recognizing exsolved phases
% Cluster analysis to find exsolved phases
[alph,Nphs,p_out,pc_id] = cluster_p(alph,Npc,p,pc_id,solv_tol,phases);    % Compute exsolved, or numerically equivalent phases
% For numerical derivatives
rho_w         = rho_H2O(T,P,'ZD05');                % Calculate density of water
eps_w         = eps_H2O(T,P,rho_w,'S14');           % Calculate dielectric constant of water
rho_w_dP      = rho_H2O(T,P+delP,'ZD05');           % Calculate density of water at P + dP
eps_w_dP      = eps_H2O(T,P+delP,rho_w_dP,'S14');   % Calculate dielectric constant of water at P + dP
[g0,v0]       = tl_g0(T,P,td,rho_w,eps_w);          % Calculate endmember Gibbs energies of each solution
[g0_dP,v0_dP] = tl_g0(T,P+delP,td,rho_w_dP,eps_w_dP); % Calculate endmember Gibbs energies of each solution at dP
g             = tl_gibbs_energy(T,P     ,phases,td,p_out,g0,v0,rho_w,eps_w);           % get Gibbs energy at P
g_P           = tl_gibbs_energy(T,P+delP,phases,td,p_out,g0_dP,v0_dP,rho_w_dP,eps_w_dP);% get Gibbs energy at P+dP
% Postprocessing
Vmol     = (g_P-g)/delP;                                             % Equation 54
Mmol     = Nphs'*molm;                                               % Equation 56
rho      = Mmol./Vmol;                                               % Equation 57
phim     = alph/sum(alph);                                           % Equation 52
phi      = phim.*Vmol./(Vmol'*phim);                                 % Equation 53
phiw     = phim.*Mmol./(Mmol'*phim);                                 % Equation 55
Cwt      = Nphs.*repmat(molm,1,size(Nphs,2))./repmat(Mmol',size(Nphs,1),1);  % Equation 58
fluid_id =  strcmp(phases(pc_id),fluid);                             % Find index of fluid
solid_id = ~fluid_id;                                                % Find index of solids
rhos     = rho(solid_id)'*phi(solid_id)/sum(phi(solid_id));          % Equation 59
rhof     = rho(fluid_id)'*phi(fluid_id)/sum(phi(fluid_id));          % Fluid density
% Chemical potentials
for iphase = 1:numel(g)
    Gphase  = g(iphase);
    mu_last = Gphase;
    nc      = sum(p_out{pc_id(iphase)}>0);
    if nc > 1
        for ic = 1:nc-1
            c_ind                       = find(p_out{pc_id(iphase)}>0);
            p1                          = p_out;
            p1{pc_id(iphase)}(c_ind(ic)) = p_out{pc_id(iphase)}(c_ind(ic)) + delc;
            p1{pc_id(iphase)}(c_ind(nc)) = p_out{pc_id(iphase)}(c_ind(nc)) - delc;
            g_dc                        = tl_gibbs_energy(T,P,phases,td,p1,g0,v0,rho_w,eps_w);
            Gphase1                     = g_dc(iphase);
            dGdc1                       = (Gphase1-Gphase)/delc;
            mu_last                     = mu_last - p_out{pc_id(iphase)}(c_ind(ic)).*dGdc1; % Equation 60
            mu{iphase}(ic)              = dGdc1;
        end
        for ic = 1:nc-1
            mu{iphase}(ic) = mu{iphase}(ic) + mu_last;                                  % Equation 61
        end
    end
    mu{iphase}(nc) = mu_last;
    chk_mu(iphase) = mu{iphase}*p_out{pc_id(iphase)}(c_ind)'-g(iphase);  % Check that g = sum(mu*p)
end
% Display some results
disp(phases(pc_id))                                                  % Display stable phase assemblage
disp(phi')                                                           % Display volume fraction
disp(Cwt)                                                            % Display phase composition
disp(rhos)                                                           % Display solid density
disp(max(abs(chk_mu)))                                               % Check consistency of chemical potential and g
```

Figure 7.

The weight fraction of each phase is found from:

$$\varphi_i^{wt} = \frac{\varphi_i^{\text{mol}} m_i^{\text{mol}}}{\boldsymbol{\varphi}_{\mathbf{mol}} \cdot \mathbf{m}_{\mathbf{mol}}} \tag{55}$$

in which the molar mass of all phases can be found by:

$$\mathbf{m}_{\mathbf{mol}} = \mathbf{molm} \cdot \mathbf{N}_{\mathbf{phs}} \tag{56}$$

where **molm** holds the molar masses of the elements.

The density of phase i can then be computed:

$$\rho_i = \frac{m_i^{\text{mol}}}{V_i^{\text{mol}}} \tag{57}$$

Composition in weight is found from:

$$C_{wt} = \frac{N_{phs}\text{molm}}{\mathbf{N}_{\mathbf{phs}} \cdot \mathbf{molm}} \tag{58}$$

Properties for total rock and fluid can be found from

$$\rho_s = \boldsymbol{\rho}_{\mathbf{ss}} \cdot \boldsymbol{\varphi}_{\mathbf{ss}} \tag{59}$$

where the vectors hold the densities of all solid minerals including both pure phases and solid solutions, and their volume fractions. Total solid concentrations are obtained from summing for each component the weight of all solids and dividing over the total weight. Note that Figure 7 presents a vectorized version of Equation 52 through Equation 59.

Chemical potential is calculated following:

$$\mu_i = \mu_{nc} + \frac{\partial g}{\partial p_i} \tag{60}$$

where μ_{nc} is the chemical potential of the last dependent proportions (as proportions sum up to 1) and is calculated from:

$$\mu_{nc} = g - \sum_{i=1}^{nc-1} p_i \frac{\partial g}{\partial p_i} \tag{61}$$

6. Improving the Calculations

Increasing the resolution of the *P-T* grid improves the smoothness of the lines in the diagram, but it does not ultimately lead to a better calculation. This is illustrated in Figure 8, where the resolution of the *T-X* grid for the olivine diagram is kept constant, while the number of discrete compositions for which the olivine and melt are calculated is increased. This means that the mixture is better approximated, and the result is that the compositions and modal abundances become better resolved and the step-like behavior that is present in the calculation with lower compositional grid gradually disappears. Although this is feasible for calculations that involve one or two simple mixtures, like binary olivine and melt, the computational demand required for more complex chemical systems and minerals makes such high-resolution compositional grids unpractical for computation. Even if

Figure 7. Code example of postprocessing results to prepare a lookup table for use in nonequilibrium processes, such as reactive transport. After loading the result of the minimization in lines 2–3, the molar masses of the components needed in the postprocessing are obtained from an external function in line 4. The cluster function in line 10 checks for each stable phase if the phase compounds are numerically distinct to distinguish exsolved phases. Lines 12–19 then calculate the densities and dielectric constants and Gibbs energy for the stable phase compositions at *P* of interest and a small deviation from *P* of interest for numerical derivatives. Molar volume using a numerical derivative of Gibbs energy is calculated in line 21. Lines 21–31 implement Equations 52–59 in the text to calculate phase abundances, composition and densities. Lines 33–51 calculate the chemical potentials for the components in the phases (Equations 60 and 61). From lines 56–59, the results are displayed on the screen as an example.

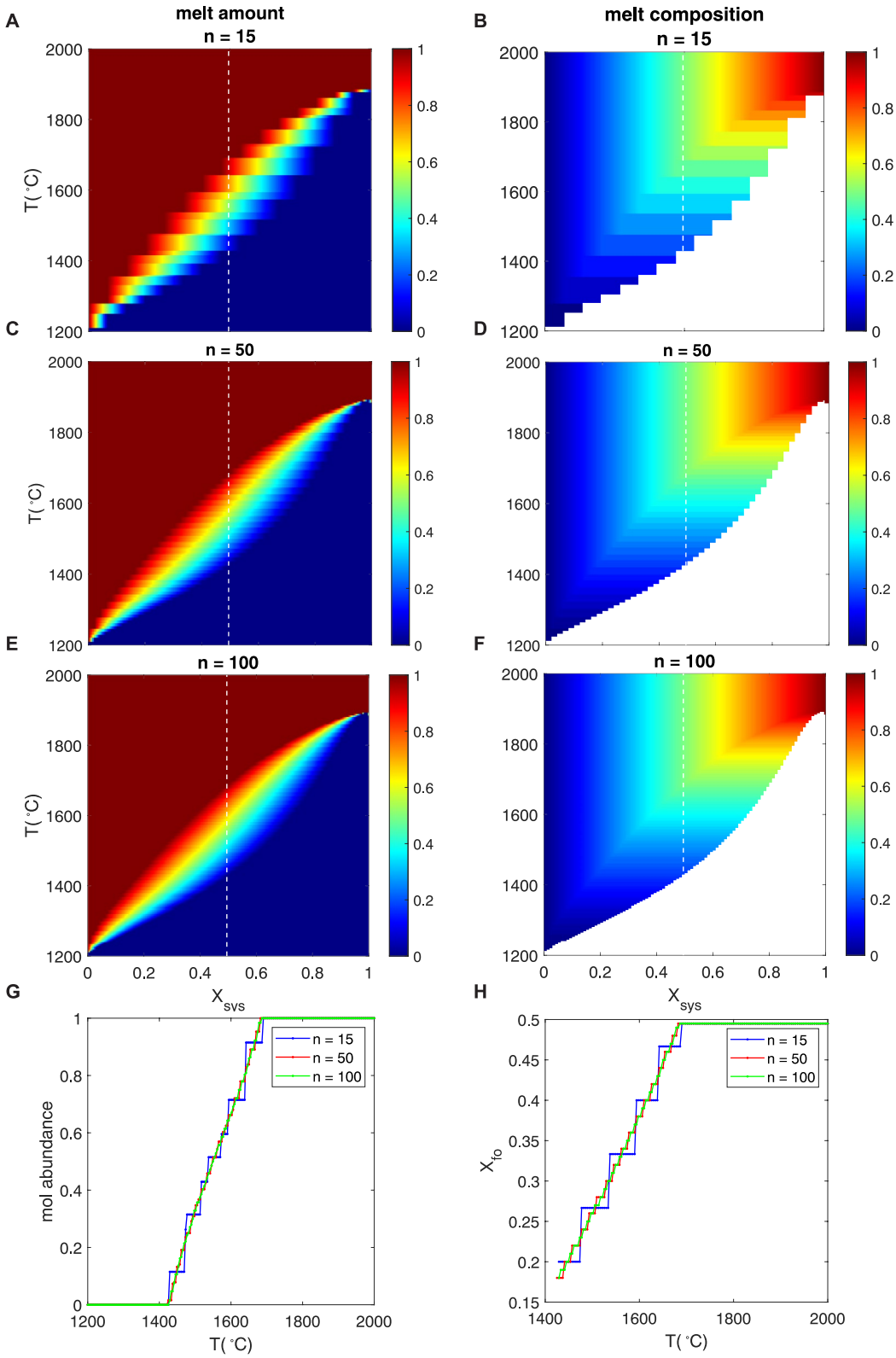

A
melt amount
n = 15
B
melt composition
n = 15
C
n = 50
D
n = 50
E
n = 100
F
n = 100
T(°C)
X_sys
G
mol abundance
n = 15
n = 50
n = 100
T(°C)
H
X_fo
n = 15
n = 50
n = 100
T(°C)

Figure 8.

computers with abundant memory are employed, we found that linprog will not find a solution for systems with more than 2 million discrete phases in the minimization.

Another option to improve the calculation is therefore to refine compositions in an iterative approach as also suggested and employed since decades (Connolly, 2009; Rossi et al., 2009; W. B. White et al., 1958). In Thermolab, this is open for the user to implement or improve. Here, we describe a simple approach that currently is employed as a working example. First, an initial minimization is done with a suitable compositional starting grid. This initial starting grid is also subject to possible modification and improvement; however, it is possible to simply set up a coarse resolution initial grid so that each minimization is fast. Each discrete phase which has an alpha above zero is recognized as a stable phase. And, the index in the list of nonzero alpha is used to find the composition of the mixture in terms of the endmember proportions and site fractions. In the next step, the proportions are used to generate a denser (i.e., higher resolution) compositional grid around the stable proportions found in the first minimization. For a binary, a simple graphical explanation of this procedure in Thermolab is illustrated in Figure 9. The new higher resolution compositional grid is added to the already existing grid and taken to the next iteration. We checked that the Gibbs energy of the system is always decreasing during the iterations. The iterations are stopped when the Gibbs energy of the system is not changing anymore within limits of machine-precision, or when the compositional spacing in the refined grid reaches this tolerance. A number of parameters are manually set by the user to control the initial refinement window, the number of refined discrete compounds generated, and the factor with which this window is decreased during each iteration. This leads to improved calculation of compositions, modal abundances, as shown in Figure 10 for the metapelite (KFMASH) *P-T* diagram from Figure 6c.

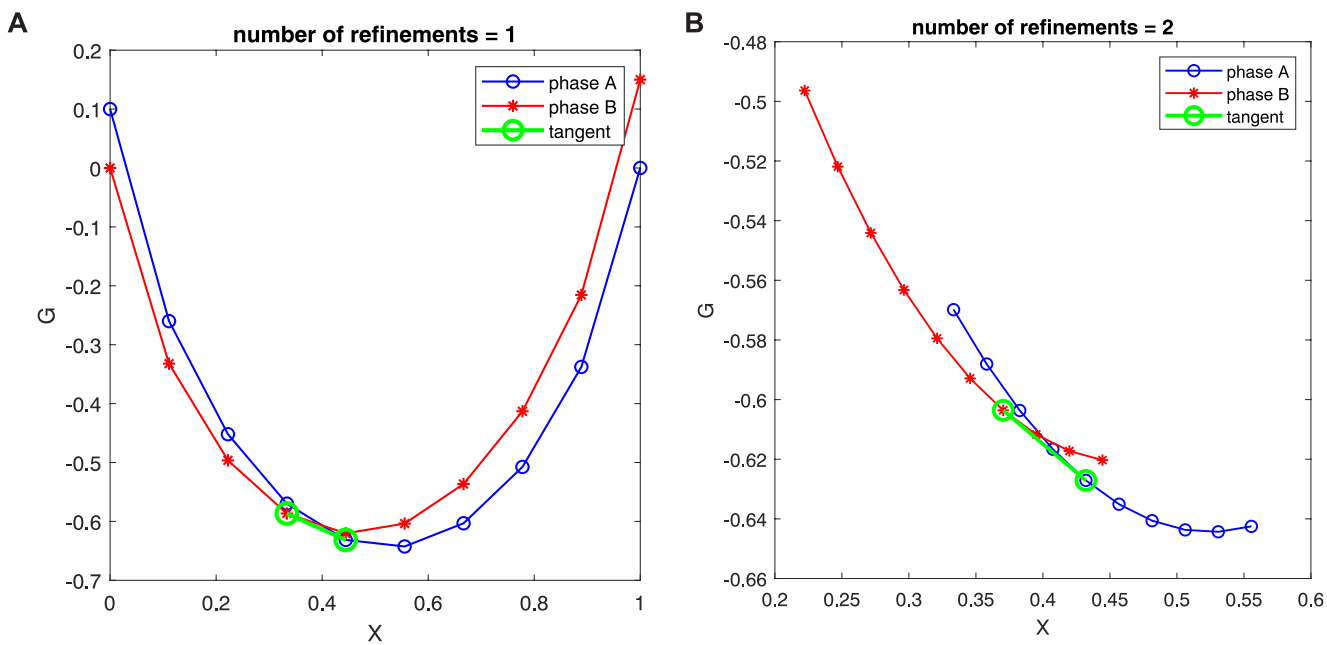


Figure 9. Conceptual diagrams showing two steps in the refinement of the composition of mixtures. (a) Shows an initial minimization result which is then zoomed in and refined around the solution in step one. (b) Here the refined compositions are used for a second minimization with smaller compositional spacing between the discretized mixture, without increasing the number of discrete compounds, thereby avoiding using too much computer memory.

Figure 8. Influence of resolution of the discretization of solutions on the quality of the thermodynamic calculations. *T-X* resolution is kept constant at 300 × 300 steps. By increasing the number of compounds to discretize the mixtures, melt and olivine, the quality of the data is improved and the stepwise behavior disappears. The melt and olivine binary mixing curves are approximated with 15, 50, and 100 discrete compounds (n). (a) Fraction of melt with the melt and solid discretized with 15 compounds. (b) Melt composition (X_{fo}) corresponding to panel (a). Panel (c) as in panel (a), for $n = 50$ compounds. (d) Melt composition corresponding to panel (c). Panel (e) as in panel (a) with $n = 100$ compounds. (f) Melt composition corresponding to panel (e). (g and h) One-dimensional profiles for melt fraction and composition at $X_{sys} = 0.5$, (dashed lines in panels (a–f)). Improvement of calculation with higher number of compounds clearly visible by smoothening of the lines as resolution increases.

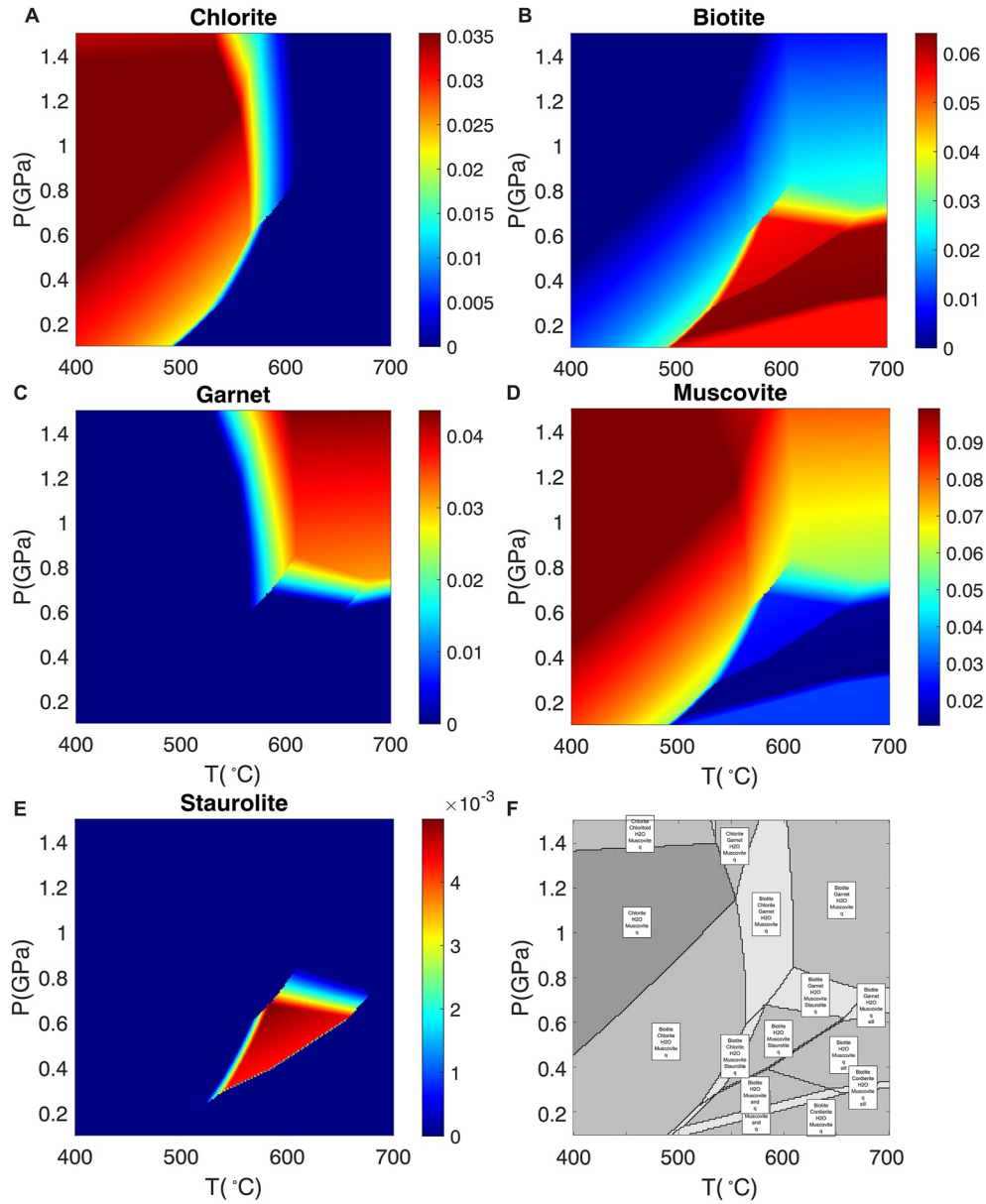


Figure 10. Example of a phase diagram after refinement. The mineral modes in panels (a–e) show smooth variations within the phase fields due to sufficiently resolving the compositions of mixtures in an iterative refinement procedure. The final phase diagram in panel (f) shows also improvement, but the general topology was also captured in Figure 6c.

7. Investigating Nonlinear Transport Processes in Open Systems

Coupling local equilibrium thermodynamics to transport processes such as the system of equations introduced above has been discussed previously (e.g., Malvoisin et al., 2015). The method described here follows the approach employed in recent studies of Malvoisin et al. (2015), Plümper et al. (2017), and Beinlich et al. (2020). For the transport models a complete set of equilibria for all possible external conditions, T, P, X, may be precomputed and stored in a lookup table. To this end, loops over P, T, and X can be programmed around the linprog minimization and stored in a database. As an example, we show the soapstone formation in serpentinite from Beinlich et al. (2020). To the initial bulk serpentinite composition, carbon (C) is added to produce a lookup table of 150 different bulk compositions at fixed T and P. Employing the refinement method the Beinlich et al. (2020) thermodynamic calculations can be improved in smoothness and computation speed (Figures 12b, 12d and 12f).

However, the main results are the same. The important aspect is that the Beinlich et al. (2020) calculation was more robust, because no iterative refinement was used and hence any chance of reaching a local-minimum rather than the true minimum was avoided. Thus, the results of Beinlich et al. (2020) served as benchmark for the refinement algorithm described above.

7.1. Reactive Transport Example

Regardless of the assumptions or the numerical method used to solve the equations, the thermodynamic properties need to be calculated using either a computation on the fly, a precomputed lookup table (Malvoisin et al., 2015), or by parameterization (Tian, Katz, & Rees Jones, 2019). After retrieving the density of solid, fluid, and concentration of fluid and solid, and the chemical potential, the system of equations described above can be solved. Here, a possible numerical implementation in MATLAB is shown (Figure 11), using explicit finite differences and a lookup table approach. The purpose is to focus on the concept of coupling the transport processes to the precomputed local equilibrium table to show how the effect of the nonideal solution models can be taken into account in reactive transport. The precomputed thermodynamic relationships are loaded in the beginning of the code. After setting up the physical and numerical parameters and defining a Cartesian grid the initial conditions. The simulation starts with a constant initial porosity distribution throughout the domain. Fluid pressure is initialized such that flow would start from the left boundary toward the right into the model. Initial solid concentration is chosen such that everywhere in the domain a serpentinite is stable and only the boundary on the left has a different composition. Such that a fluid with a composition out of equilibrium with the rock on the right will enter the model. The fluid reacts with the initial rock to form new mineral assemblages progressively transforming the rock from the left to the right in the model.

The unknowns, ρ_f, ρ_s, C_s^{im}, C_f, and μ_{Cf} are found from interpolation in the lookup table at each time step (Figure 11, Lines 36–40) at a given P-T (0.3 GPa and 300°C). Here, the fluid only contains CO_2 and for example, Mg can be used as immobile species in the solid C_s^{im} to find porosity according to Equation 7. From this, transient permeability and effective diffusion coefficient can be computed and used in the Darcy and diffusion fluxes. These are used in the total mass balance (5) and total mass concentration balance (1) to update concentrations and fluid pressure. In the example, the fluid pressure is found from finding the steady state solution of the total mass balance using an iterative pseudotransient method. The total mass concentration C_{tot} is found from an explicit finite different formulation. From this new C_{tot}, Equation 2 is rearranged to find the updated C_s. In the postprocessing stage the thermodynamic lookup table can also be used to find the modal abundances of the minerals to plot the evolution of the mineralogy through time.

Details of the coding example are found in the caption of Figure 11 and results of the transport code are shown in Figure 12. The nonlinearity of the thermodynamic relationship between solid and fluid composition is demonstrated in Figures 12c, 12d and 12f. The shape of the curves is a result of both the nonideality of solution models as well as changes in the phase assemblage. Depending on the function of solid composition versus fluid composition, the reaction fronts behave very differently in terms of shape and velocity (Figures 12a, 12c and 12e). In the case that the incoming fluid composition is low in CO_2 in equilibrium with antigorite, talc and magnesite no sharp fronts develop (Figure 12a) but modal abundance may change. When incoming CO_2 fluid composition is higher and antigorite disappears from the system, a sharp reaction front propagates, and when the CO_2 in the fluid increases further, talc also disappears and a second reaction front appears. The shape of the reaction fronts is controlled by the curvature of the C_f-C_s relation as well as the advection velocity versus speed of diffusion.

7.2. Cahn-Hilliard Exsolution

Natural processes usually are not in equilibrium and do not strictly follow the path that can be followed from a phase diagram. Instead, there will be a process that develops toward the state of global equilibrium depicted on a phase diagram although it may never reach it (e.g., due to low temperature effectively arresting diffusion, or other kinetic effects). With Thermolab, the Gibbs energies can be directly used instead of first calculating thermodynamic equilibrium and the process toward equilibrium can be modeled. A demonstration of this is given by solving the Cahn-Hilliard equations in which driving force for diffusion is chemical potential and an uphill diffusion process causes an initial random homogeneous system to develop into an equilibrium phase assemblage (Cahn &

```
clear,figure(1),clf,colormap(jet),addpath ../
% Physics
load lookup_soapstone_2022_02_14_n1500_nz15_full
mu_tab   =   mu_tab/max(abs(  mu_tab));
rhos_tab = rhos_tab/max(abs(rhos_tab));
rhof_tab = rhof_tab/max(abs(rhof_tab));
%independent
Lx     = 1;                                                    % Length of model domain (m)
Dc     = 1;                                                    % Diffusion coefficient (m^2/s)
P1     = 1;                                                     % Boundary fluid pressure (Pa)
%nondim
phi0   = 0.2;                                                   % Background porosity (fluid volume fraction)
C0     = 0.004;                                                 % Background CO2 concentration in solid (wt%)
C1     = 0.04715;                                               % Csolid soapstone
npow   = 3;                                                     % Non-linearity in permeability-porosity relation
%dependent
k_muf0 = 1*Dc/P1;                                               % Background permeability/fluid viscosity (m^2/(Pa*s))
t_tot  = 1e+1*Lx^2/Dc;                                          % Total time (s)
% Numerics
nx     = 200;                                                   % Number of grid points in x direction
nt     = 1e9;                                                   % Number of time steps
niter  = 1e5;
nout   = 10000;                                                 % Maximum number of Pf iterations
tol    = 1e-6;                                                  % Exit tolerance from Pf iterations
% preprocessing
dx     = Lx/(nx-1);                                             % Grid step in x
x      = 0:dx:Lx;                                               % X-coordinates of the nodes
% Initialisation;
phi    = x*0 + phi0;                                            % Initial porosity
Pf     = P1*(1-x/Lx);                                           % Initial fluid pressure
Cs     = x*0 + C0;                                              % Initial solid concentration
Cs(1)  = C1;                                                    % Left solid CO2 concentration boundary
time   = 0;
% Processing
for it = 1:nt   % Time loop
    mu      = interp1(Cs_tab,mu_tab,Cs);                        % Local equilibrium mu CO2 in fluid from lookup table
    Cf      = interp1(Cs_tab,Cf_tab,Cs);                        % Local equilibrium CO2 in fluid from lookup table
    Cs_im   = interp1(Cs_tab,Mg_tab,Cs);                        % Local equilibrium Mg  in solid from lookup table
    rhos    = interp1(Cs_tab,rhos_tab,Cs);                      % Local equilibrium density in solid from lookup table
    rhof    = interp1(Cs_tab,rhof_tab,Cs);                      % Local equilibrium density in fluid from lookup table
    if it == 1,phi_srho_sCs_im0 = (1 - phi).*rhos.*Cs_im;end    % Store initial values
    phi     = 1 - phi_srho_sCs_im0./(Cs_im.*rhos);              % mass balance of immobile species in solid (eq.7)
    Ctot    = Cf.*rhof.*phi + Cs.*rhos.*(1-phi);                % Shorthand notation (eq. 2)
    rhotot  =     rhof.*phi +     rhos.*(1-phi);                % Shorthand notation (eq. 6)
    if it == 1,rhotot0         = rhotot;end                     % Store initial values
    drhotot = rhotot(2:end-1) - rhotot0(2:end-1);               % Shorthand notation
    % averaging
    rhofc   = 0.5*(rhof(1:end-1)             + rhof(2:end)           );  % rhof values between the nodes
    rhofCfc = 0.5*(rhof(1:end-1).*Cf(1:end-1) + rhof(2:end).*Cf(2:end));  % rhof*Cf values between the nodes
    phic    = 0.5*( phi(1:end-1)             +  phi(2:end)           );  % phi values between the nodes
    % transport properties
    perm    = k_muf0*phic.^npow;                                % Permeability
    Deff    = Dc*rhofCfc.*phic;                                 % Effective diffusivity
    % Updates
    for iter = 1:niter
        dt_dif       = 0.45*dx^2/Dc;                            % maximum time step of diffusion
        dt_adv       = 0.45*dx/max(abs(rhofc.*perm.*diff(Pf)/dx));  % maximum time step of advection
        dt           = min([dt_adv,dt_dif, t_tot-time]);        % maximum time step for numerical stability
        qD           = -perm.*diff(Pf)/dx;                      % Darcy flux (eq. 3)
        Pres         = -drhotot/dt - diff(rhofc.*qD)/dx;        % Residual of the total mass balance (eq. 5)
        Pf(2:end-1)  = Pf(2:end-1) + dx^2/max(perm.*rhofc)/4*Pres;  % Pf iterative update
        if max(abs(Pres)) < tol,break,end                       % Exit criteria from Pf iterations
    end
    iters(it)        = iter;                                    % store for plotting
    qC               = -Deff.*diff(mu)/dx;                      % Diffusion flux in fluid (first term in rhs eq. 3)
    Ctot(2:end-1)    = Ctot(2:end-1) - dt*(diff(qC + rhofCfc.*qD)/dx);  % mass concentration balance of mobile species (eq.1)
    Cs               = (Ctot - rhof.*Cf.*phi)./rhos./(1-phi);   % calculate new solid concentration (from eq. 2)
    time             = time + dt;
    rhotot0          = rhotot;                                  % for the next time step
    if mod(it,nout)==1 || time == t_tot  %Postprocessing        % plot every nout time step
        phs_modes   = interp1(Cs_tab,vol_frac_solids,Cs);       % find local equilibrium stable phase volume fraction
        area(x,phs_modes,'FaceColor','flat'),axis tight
        legend(solid_names),title([mean(iters)/nx,iter/nx])
        xlabel('x')
        drawnow
    end
    if time >= t_tot,break,end
end
```

Figure 11.

Hilliard, 1958). A simplified version of the Cahn-Hilliard equation consists of a balance of mass concentration and a flux equation for the diffusion. Assuming a system without fluid, and assuming constant densities a basic mass concentration balance can be written (Figure 13, Line 36):

$$\frac{\partial C_s^A(x,t)}{\partial t} = -\frac{\partial q_{c_s}(x,t)}{\partial x} \tag{62}$$

where C_s^A is the concentration of species A in the solid.

Flux of concentration in solid of a component can be defined as a function of gradients in chemical potential differences or a single chemical potential in conjunction with the Gibbs-Duhem relation (e.g., p. 80 in Lebon et al., 2008; Nauman & He, 2001), here the former is used (Figure 13, Line 35):

$$q_{c_s}(x,t) = -D_{c_s} \cdot C_s^A(x,t) \cdot \left(1 - C_s^A(x,t)\right) \cdot \frac{\partial \left(\mu_s^A(x,t) - \mu_s^B(x,t)\right)}{\partial x} \tag{63}$$

The chemical potential difference between species A and B in the solid, $\mu_s^A - \mu_s^B$, can be conveniently expressed by:

$$\mu_s^A(x,t) - \mu_s^B(x,t) = \frac{\partial g}{\partial C_s(x,t)} - \gamma \frac{\partial^2 C_s(x,t)}{\partial x^2} \tag{64}$$

where the last term in Equation 64 is the interfacial free energy contribution as introduced by Cahn and Hilliard (1958). Similar to Figure 7, the chemical potential can be calculated with numerical differentiation of the Gibbs energy (see Figure 13, Lines 30–32).

These equations describe the spinodal decomposition of a phase with strong nonideal behavior, such as feldspar, due to nonlinear diffusion driven by chemical potential gradients. A useful review is found in Nauman and He (2001). See also caption Figure 14 for details.

8. Discussion

The motivation behind the development of Thermolab is to study the effects of the nonideality of solution models in reactive transport processes in open systems. Most natural materials including minerals, rocks, melt, fluids, and gases display some degree of nonideal mixing behavior (e.g., Ganguly, 2020). We showed that for the case study of the soapstone formation studied by Beinlich et al. (2020) both the shape and the velocity of the reaction front vary strongly depending on the nonlinearity of the partitioning of carbon between fluid and solid (Figure 12). For evaluating potential risks during transport of nuclear waste material understanding this nonlinear behavior is crucial and a study of Shao et al. (2009) is line with this conclusion.

For mathematical analysis, the formulation of complex solution models is important to show in a transparent manner such that nonlinearities in transport processes can be studied. Therefore we presented the linear algebraic approach that is used in Thermolab to compute Gibbs energy of mixing for arbitrary multicomponent phases (Figure 1). The starting point was a crystallographic or structural model of a solution and a predefined set of endmembers as developed by previous workers (e.g., Green et al., 2016; Palin et al., 2016). More in-depth

Figure 11. Reactive transport code using precomputed equilibrium approach. After loading the precomputed thermodynamic equilibrium data (line 3), the physical parameters are defined in lines 8–18, the numerical parameters (e.g., number of time steps and nodes in the model) are defined (lines 20–24). Preprocessing and initialization including setup of the grid and initial conditions are done in lines 26–33. Here an incoming fluid and fluid pressure increase is setup as left boundary to initialize reactive fluid flow from left to right. In Lines 36–40, all properties retrieved from local equilibrium are found by interpolating on the precomputed thermodynamic equilibrium data. Line 41 stores the initial condition for the nonmobile solid and the initial porosity distribution is set to a constant value (phi0 in line 12). Line 43 is the definition of total concentration and mass density in the system (Equations 2 and 6). Averaging values used in fluxes defined between the nodes are calculated in lines 48–50. Lines 52–53 are the dynamic permeability and effective diffusivity, from Equation 4 used in driving forces for the reactive transport, flux of diffusion (qC), and Darcy flux (qD), Equation 3. Adaptive time steps, following Courant Friedrichs-Lewy stability criterion, are calculated in lines 55–57 to constrain numerical error propagation and maintain the stability of the numerics. Lines 59–64 are an iteration loop to solve the steady state of mass balance Equation 5. Lines 66–67 are the explicit finite difference formulation of Equation 1. The solid concentration is found from the definition of total system mass concentration, Equation 2. Plotting takes place during model run every "nout" time steps until maximum time is reached. The phase abundances can be visualized when desired as they are interpolated from the lookup table (line 72) and do not enter any of the physics equations. Plotting of the mineralogical evolution is done in lines 73–76.

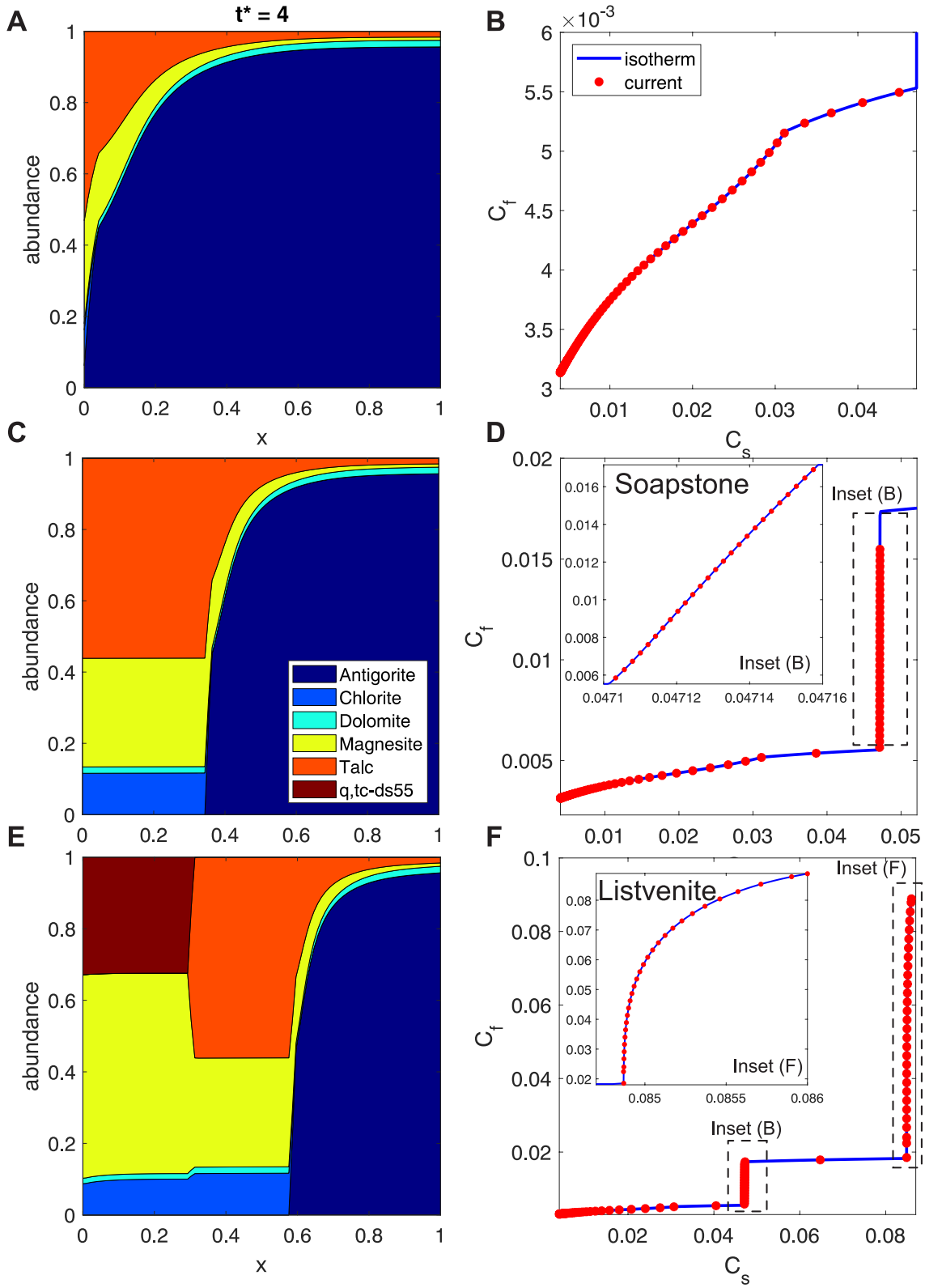

A
t* = 4
abundance
x
B
×10^-3
isotherm
current
C_f
C_s
C
Antigorite
Chlorite
Dolomite
Magnesite
Talc
q,tc-ds55
D
Soapstone
Inset (B)
E
F
Listvenite
Inset (F)

Figure 12.

discussion on how to define new crystallographic and speciation models for solid solutions is given by Myhill and Connolly (2021), after which the model should then be fitted to experimental data.

To ensure the reliability of the Gibbs energy calculation and equilibrium calculation methods, Thermolab is benchmarked versus phase diagrams and exemplified in use for reactive transport projects. Constrained Gibbs energy minimization is discussed following existing approaches (Connolly, 2005; W. B. White et al., 1958) and the results are then used to calculate equilibrium compositions and retrieve the so-called isotherm that can be used in a transport code.

The isotherm, introduced in chromatographic studies of metasomatism, is a relation between fluid and solid composition at fixed *P-T* (Hofmann, 1972). An approach in which the system is divided in solid and fluid and for which the mass conservations equations have been summed up to eliminate reaction source terms has proven very useful for reaction front propagation studies (Orr, 2005). Shape and velocity of propagating reaction fronts strongly depend on the nonlinearity of solutions models (Guy, 1993). The approach of precomputing the equilibrium compositions and using the results as an isotherm in the transport codes is useful to study the effects of the nonlinearity of solution models on reactive transport. As the studies of Hofmann (1972) and Guy (1993) focus on single phases, our results show this is functioning similarly on multiphase systems such as rocks. The steep reaction fronts act similar to isotherms with miscibility gaps (Fletcher & Hofmann, 1974), but the solid solutions are responsible for the continuity of the isotherm across the "jump-like" curves. Previous studies usually introduced simple isotherms to study the behavior during transport. Thermolab has been motivated by the need to generate realistic flow functions using complex solution models.

Solid solutions with strong nonideal behavior result in phase separation due to nonlinear diffusion when combined with transport models. The Cahn-Hilliard model (Cahn & Hilliard, 1958) is used to stabilize the numerical solutions by introducing an energy penalty for generating surfaces between phases introduced by a surface energy parameter. This method is widely applied in material science (Nauman & He, 2001) and also some geological applications have been studied (Abart et al., 2009; Petrishcheva & Abart, 2009). For geological materials other than feldspar, the behavior of such nonlinear diffusion systems can be further investigated with Thermolab.

Mixing of databases is possible in a flexible framework like Thermolab, however, it may not be recommended since internally consistent databases are not necessarily consistent among each other. Nevertheless, to use aqueous species in fluids together with most up-to-date solution models of minerals, the only possibility to date is to combine for example SUPCRT databases with Holland and Powell (1998, 2011). It has been argued that such a combination may serve as a good approximation (Dolejš, 2013), when appropriate EOS and dielectric constant for water are used. However, refitting the aqueous species databases from SUPCRT in conjunction with mineral database of Holland and Powell (1998) is likely the more reliable approach to combine data sets (Miron et al., 2016, 2017).

Figure 12. Snapshot of reaction fronts produced by influx of three different incoming CO_2 fluid compositions from the left boundary into a serpentinite using the code in Figure 11. Concentration units are in elemental carbon weight fraction. For comparison all runs were stopped after the same duration. In all cases, the starting rock consists primarily of serpentine (here antigorite), with minor amounts of talc, magnesite, and dolomite. (a) If the incoming fluid composition lies below ~0.005 antigorite never completely disappears and a gradual reaction front forms transforming the serpentinite to ophimagnesite (magnesite-talc-serpentinite). (b) The equilibrium relation between C_f and C_s (i.e., the isotherm) shown in blue with the transient (current) C_f and C_s compositions plotted as red dots. (c) With an incoming fluid composition between 0.01 and 0.015, the pristine serpentinite to the right transforms into soapstone with a sharp reaction front when antigorite has reacted out. (d) The equilibrium relation between C_f and C_s (i.e., the isotherm). The steep slope at C_s between ~0.045 and 0.052 is enlarged in the inset. Here antigorite disappears (the soapstone on the left in panel (c)), and the steepness of the slope on the isotherm causes a sharp soapstone front. The shallow slopes below $C_s = 0.045$ correspond to the transitional front where serpentinite is partially transformed to soapstone. (e) Development of an additional reaction front, forming listvenite (quartz-magnesite rock), as incoming fluid composition is now higher, ~0.086 weight fraction dissolved carbon. (f) As the incoming fluid composition lies on another steep part of the isotherm, the listvenite front is also sharp. Inset shows the curvature of the isotherm corresponding to the listvenite stable assemblage. Endmember data from the tc-ds55 data set (Holland & Powell, 1998), solution models: CO_2-H_2O Fluid (Aranovich & Newton, 1999), Antigorite (Padróon-Navarta et al., 2013), Chlorite (Holland et al., 1998), Talc (Holland & Powell, 1998), Magnesite, and Dolomite (R. W. White et al., 2003). t^* = dimensionless time.

```
1 clear,figure(1),clf,colormap(jet(256))
2 % Physics
3 Lx = 1;                                                    % Model domain length
4 Dc = 1;                                                    % Diffusion coefficient
5 % Thermodynamics
6 T        = 500 + 273.15;                                   % Temperature (K)
7 P        = 1e8;                                            % Pressure (Pa)
8 R        = 8.31;
9 phs_name = {'Feldspar(C1)'};                               % phase name
10 Cname    = {'Si','Al','Na','K','O'};                       % system components
11 td       = init_thermo(phs_name,Cname,'solution_models_H18');  % load static thermodynamic data
12 [g0,V0]  = tl_g0(T,P,td);                                  % get endmember gibbs energies of the solution model
13 % Numerics
14 nx   = 100;                                               % Grid resolution
15 nt   = 1e6;                                               % number of time steps
16 nout = 5000;                                              % plotting each nout time step
17 gam  = 2*(Lx/(nx-1))^2;                                   % Cahn Hilliard surface energy parameter
18 delC = 1e-3;                                              % delta concentration for numerical derivative
19 % Preprocessing
20 dx = Lx/(nx-1);                                           % grid step
21 dt = min(dx^2/Dc/2.1,dx^4/gam/4.1)/8;                     % time step
22 x  = 0:dx:Lx;                                             % x coordinates
23 % Initialization
24 c    = 0.5 + 0.01*rand(1,nx);                             % initial concentration of albite
25 cini = c;                                                 % store initial concentration for plot
26 % Processing
27 for it = 1:nt
28     p{1}        = [c; 1-c]';                              % proportions of endmembers Ab-Or
29     p_dC{1}     = [c+delC; 1-(c+delC)]';                  % proportions of endmembers Ab-Or plus delta C
30     g_p         = tl_gibbs_energy(T,P,phs_name,td,p_dC,g0,V0); % Gibbs energy + delta G, for numerical derivative
31     g           = tl_gibbs_energy(T,P,phs_name,td,p   ,g0,V0); % Gibbs energy
32     mu          = (g_p-g)'/delC/R/T;                      % mu1 - mu2; numerical derivative dg/dc, eq. 64
33     mu(2:end-1) = mu(2:end-1) - gam*diff(c,2)/dx^2;       % Cahn-Hilliard (1958), eq. 64
34     cc          = (c(1:end-1)+c(2:end))/2;                % Centered concentrations between gridpoints
35     qc          = -Dc*(1-cc).*cc.*diff(mu)/dx;            % Non-Fickean Diffusion flux, eq. 63
36     c(2:end-1)  = c(2:end-1) - dt*diff(qc)/dx;            % Concentration balance with assumptions, eq. 62
37     c(1:3)      = mean(c(1:3));                           % no flux left boundary
38     c(end-2:end)= mean(c(end-2:end));                     % no flux right boundary
39     % Postprocessing
40     if mod(it,nout) == 1
41         plot(x,c,x,cini),title([it,mean(c)]),axis([0 1 0 1])   % plot concentration versus distance
42         drawnow
43         irec          = (it-1)/nout + 1;
44         cevol(:,irec) = c;
45         time(irec)    = it*dt;
46     end
47 end
48 pcolor(log(time),x,cevol),shading flat,colorbar
49 xlabel('log(time)'),ylabel('x'),title('C(t,x)'),xlim([-4 0.1])
```

Figure 13. Cahn-Hilliard example code for 1D binary exsolution. The length of the domain and diffusion coefficient, assumed constant for simplicity here, are defined in lines 3–4, after which the thermodynamic data is loaded for a given *P-T*. The phase for which Gibbs energy is calculated is input in line 9, and the components in the system can be given in line 10. Line 11 initializes static thermodynamic data as in Figure 5. Line 12 calculates the Gibbs energy of the endmembers in the solution (this is done to increase performance as they do not vary here). Numerical parameters are setup in lines 14–18, followed by preprocessing for generating the grid and the time step. An initial setup consists of a homogeneous concentration with small random perturbations. The physical process is modeled in a time loop from lines 27 to 38. Lines 28–31 show how Thermolab is used without precomputed lookup tables. A small increment of proportions of the endmembers is used to make a numerical derivative to calculate the chemical potential in line 32, with Cahn-Hilliard addition of energy regularization added in line 33, corresponding to Equation 64. The chemical potential gradients drive diffusion in the system via the diffusion flux in line 35, Equation 63 and line 36 is the mass concentration balance in simplified form, Equation 62. Lines 36–37 are boundary conditions that ensure total concentration in the system remains constant. Here, making the concentrations equal in a mass conservative way (using the mean of the three grid points on the boundary), the chemical potentials will be the same at these nodes and hence it will lead to no flux boundary condition. Lines 41–45 are plotting the results as the model runs. Line 48 visualizes the coarsening of the separate exsolved phases through time.

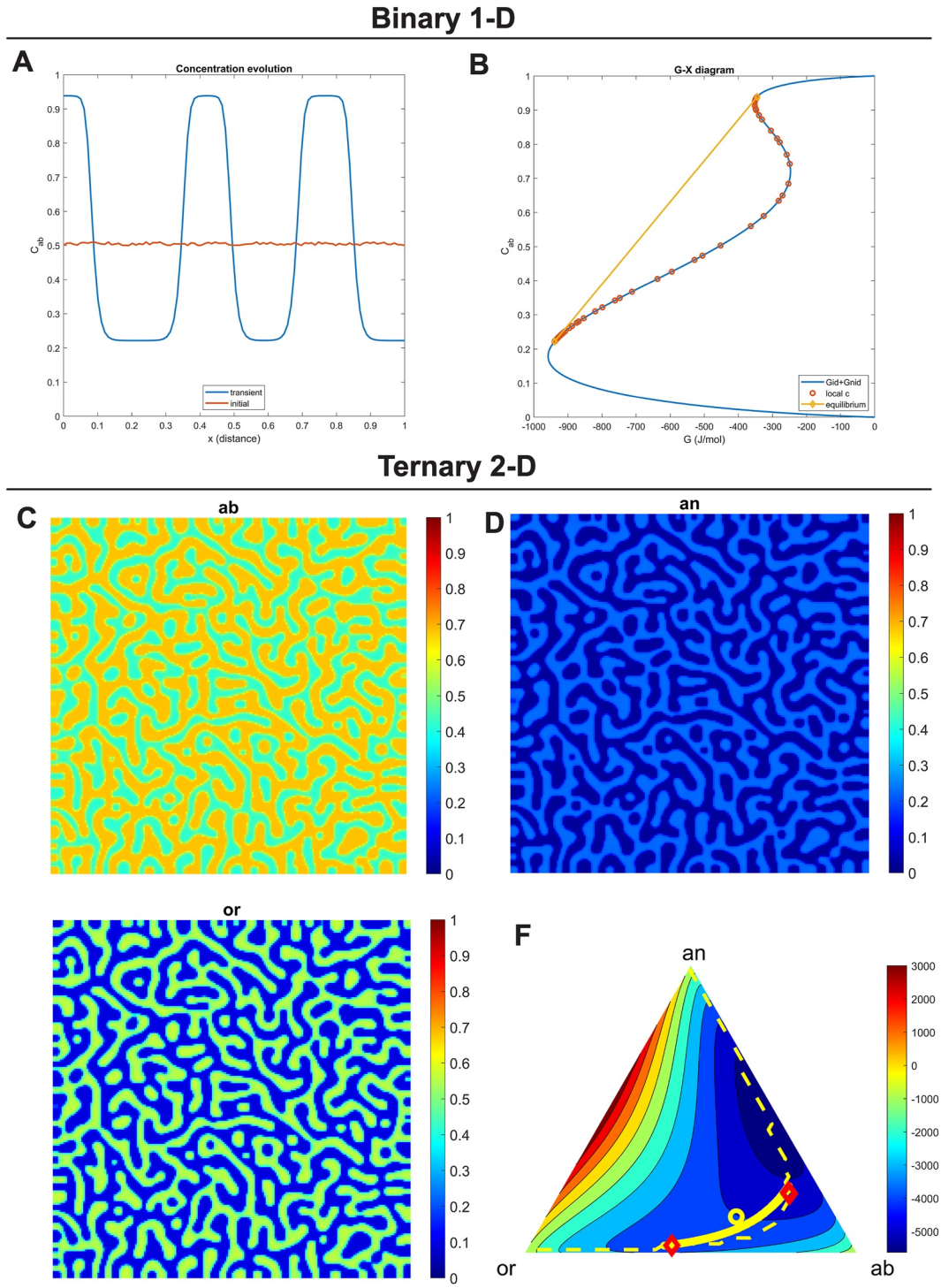


Figure 14.

9. Conclusions

With Thermolab it is possible to reproduce published phase diagrams involving complex solution models and these solution models can then be used in transport codes to investigate the effects of nonlinearity on the open system processes. Complex flow functions can be retrieved from Thermolab and used in mathematical analysis

and numerical models of reactive transport. Gibbs energy can be directly used to construct chemical potentials as a driving force for nonlinear diffusion leading to phase separation.

The modeling framework provided by Thermolab allows users to add custom functions, include new databases, design solution models and also improve procedures to calculate equilibrium. Limits of the discretization of solutions approach are due to computer memory restrictions, which for complex solution models increases computation time to unpractical durations. Computers with increased memory would be a solution, however, we found that the linprog algorithm will stop to converge when about 2 million discrete phases or more in systems with more than six components are used. Refinement strategies are a potential solution to this problem; however, these compromise the robustness of the result by possibly missing the global minimum of the Gibbs energy. Thus, Thermolab leaves the door open for the development of faster and more robust future algorithms by a transparent open source coding environment, which due to the compactness of MATLAB example code can be translated to other programming languages with minimum effort.

Appendix A

An example of the automated selection of compositional or site fraction variables to calculate proportions for a mineral having 11 endmembers is given for amphibole:

From the site occupancy table, we obtain the following site fraction table by dividing each occupancy over the sum of moles on the site. For example, the M2 site reaches a sum of 2 mol when the Mg, Fe Al Fe^3, and Ti are summed. Then the moles of each occupancy are normalized by dividing by this sum to get fractions on the site. See Table A2 for the result.

Table A1
Site Occupancy Table of Amphibole With Ordering of Fe-Mg on M13, M2 Sites

Site	A			M13		M2					M4				T1		V	
Occupancy	v	Na	K	Mg	Fe	Mg	Fe	Al	Fe3	Ti	Ca	Mg	Fe	Na	Si	Al	OH	O
Endmember																		
Tremolite	1	0	0	3	0	2	0	0	0	0	2	0	0	0	4	0	2	0
Tschermakite	1	0	0	3	0	0	0	2	0	0	2	0	0	0	2	2	2	0
Pargasite	0	1	0	3	0	1	0	1	0	0	2	0	0	0	2	2	2	0
Glaucophane	1	0	0	3	0	0	0	2	0	0	0	0	0	2	4	0	2	0
Cummingtonite	1	0	0	3	0	2	0	0	0	0	0	2	0	0	4	0	2	0
Grunerite	1	0	0	0	3	0	2	0	0	0	0	0	2	0	4	0	2	0
Ordered Amphibole A	1	0	0	3	0	0	2	0	0	0	0	0	2	0	4	0	2	0
Ordered Amphibole B	1	0	0	0	3	2	0	0	0	0	0	0	2	0	4	0	2	0
Magnesio-riebekite	1	0	0	3	0	0	0	0	2	0	0	0	0	2	4	0	2	0
K-Pargasite	0	0	1	3	0	1	0	1	0	0	2	0	0	0	2	2	2	0
Ti-Tschermakite	1	0	0	3	0	0	0	0	0	2	2	0	0	0	2	2	0	2

Figure 14. Example of using Thermolab without precomputed thermodynamic data. Physical processes develop toward thermodynamic equilibrium using the Cahn-Hilliard model at 500°C, 0.1 GPa. The 1D example code for a binary system is given in Figure 13. An initially homogenous concentration distribution in a single phase develops into a two phase system due to uphill diffusion as a result of the nonlinearity of solution models (the nonideality that leads to exsolution of phases). (a) Snapshot of a 1D model during exsolution of binary feldspar (concentration corresponds to albite component in albite-sanidine mixture). Red line shows initial concentration distribution developing into separate phases represented by the blue line. (b) Comparing the concentrations in the physical domain to the Gibbs energy-composition (X) curve (for clarity only the mixing energy, ideal + nonideal is shown). Plotting the transient concentrations on the equilibrium curve shows that the system develops toward the tangent construction (i.e., equilibrium condition) shown with the yellow line and calculated with "linprog" following methods described in the main text. (c) Albite concentration in a 2D feldspar model after exsolving for some time from a homogeneous initial distribution. Panels (d) and (e) represent the corresponding compositions of respectively, components anorthite, orthoclase (using sanidine endmember here). (f) Transient compositions of the feldspar for all nodes in the 2D domain plotted on the equilibrium mixing Gibbs-energy ternary represented by the contours ($g_{id} + g_{nid}$). These compositions form a continuous yellow line (as there are many overlapping symbols). Dashed line corresponds to solvus computed with "linprog," and yellow open circle corresponds to average concentration in the system. Red symbols are equilibrium compositions of the two stable phases from Gibbs minimization.

Table A2
Site Fraction Table of Multisite Amphibole With Ordering on Two Sites

Site	A			M13		M2					M4				T1		V	
Occupancy	v	Na	K	Mg	Fe	Mg	Fe	Al	Fe3	Ti	Ca	Mg	Fe	Na	Si	Al	OH	O
Endmember																		
Tremolite	1	0	0	1	0	1	0	0	0	0	1	0	0	0	1	0	1	0
Tschermakite	1	0	0	1	0	0	0	1	0	0	1	0	0	0	0.5	0.5	1	0
Pargasite	0	1	0	1	0	0.5	0	0.5	0	0	1	0	0	0	0.5	0.5	1	0
Glaucophane	1	0	0	1	0	0	0	1	0	0	0	0	0	1	1	0	1	0
Cummingtonite	1	0	0	1	0	1	0	0	0	0	0	1	0	0	1	0	1	0
Grunerite	1	0	0	0	1	0	1	0	0	0	0	0	1	0	1	0	1	0
Ordered Amphibole A	1	0	0	1	0	0	1	0	0	0	0	0	1	0	1	0	1	0
Ordered Amphibole B	1	0	0	0	1	1	0	0	0	0	0	0	1	0	1	0	1	0
Magnesio-riebekite	1	0	0	1	0	0	0	0	1	0	0	0	0	1	1	0	1	0
K-Pargasite	0	0	1	1	0	0.5	0	0.5	0	0	1	0	0	0	0.5	0.5	1	0
Ti-Tschermakite	1	0	0	1	0	0	0	0	0	1	1	0	0	0	0.5	0.5	0	1

The site fraction table can be transposed to form a set of equations in matrix form:

$$\begin{bmatrix} 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0.5 & 0 & 1 & 0 & 0 & 1 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0.5 & 1 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0.5 & 0.5 & 1 & 1 & 1 & 1 & 1 & 1 & 0.5 & 0.5 \\ 0 & 0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0.5 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_{tr} \\ p_{ts} \\ p_{par} \\ p_{gl} \\ p_{cm} \\ p_{grun} \\ p_{a} \\ p_{b} \\ p_{mrb} \\ p_{kpar} \\ p_{tts} \end{bmatrix} = \begin{bmatrix} z_{v}^{A} \\ z_{\mathrm{Na}}^{A} \\ z_{K}^{A} \\ z_{\mathrm{Mg}}^{M13} \\ z_{\mathrm{Fe}}^{13} \\ z_{\mathrm{Mg}}^{M2} \\ z_{\mathrm{Fe}}^{M2} \\ z_{\mathrm{Al}}^{M2} \\ z_{\mathrm{Fe3}}^{M2} \\ z_{\mathrm{Ti}}^{M2} \\ z_{\mathrm{Ca}}^{M4} \\ z_{\mathrm{Mg}}^{M4} \\ z_{\mathrm{Fe}}^{M4} \\ z_{\mathrm{Na}}^{M4} \\ z_{\mathrm{Si}}^{T1} \\ z_{\mathrm{Al}}^{T1} \\ z_{\mathrm{OH}}^{V} \\ z_{\mathrm{O}}^{V} \end{bmatrix} \quad \text{(A1)}$$

The system of equations describing the relation between proportion and site fractions that is obtained automatically using the MATLAB script from Equation A1 is given in matrix form by:

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0.5 & 0 & 1 & 0 & 0 & 1 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0.5 & 1 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} p_{tr} \\ p_{ts} \\ p_{par} \\ p_{gl} \\ p_{cm} \\ p_{grun} \\ p_{a} \\ p_{b} \\ p_{mrb} \\ p_{kpar} \\ p_{tts} \end{bmatrix} = \begin{bmatrix} 1 \\ z_{v}^{A} \\ z_{\mathrm{Na}}^{A} \\ z_{\mathrm{Mg}}^{M13} \\ z_{\mathrm{Mg}}^{M2} \\ z_{\mathrm{Fe}}^{M2} \\ z_{\mathrm{Al}}^{M2} \\ z_{\mathrm{Fe3}}^{M2} \\ z_{\mathrm{Ca}}^{M4} \\ z_{\mathrm{Mg}}^{M4} \\ z_{\mathrm{Fe}}^{M4} \end{bmatrix} \tag{A2}$$

Figure A1 shows the MATLAB script used to retrieve the matrix in Equation A2 above.

And from the inverse of the matrix in Equation A2 above the proportions can be found as function of site fractions:

$$\begin{bmatrix} -0.5 & 0.5 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & -1 & -1 \\ -1.5 & 0.5 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & -1 \\ -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & -1 & -1 & -1 & -1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ z_{v}^{A} \\ z_{\mathrm{Na}}^{A} \\ z_{\mathrm{Mg}}^{M13} \\ z_{\mathrm{Mg}}^{M2} \\ z_{\mathrm{Fe}}^{M2} \\ z_{\mathrm{Al}}^{M2} \\ z_{\mathrm{Fe3}}^{M2} \\ z_{\mathrm{Ca}}^{M4} \\ z_{\mathrm{Mg}}^{M4} \\ z_{\mathrm{Fe}}^{M4} \end{bmatrix} = \begin{bmatrix} p_{tr} \\ p_{ts} \\ p_{par} \\ p_{gl} \\ p_{cm} \\ p_{grun} \\ p_{a} \\ p_{b} \\ p_{mrb} \\ p_{kpar} \\ p_{tts} \end{bmatrix} \tag{A3}$$

```
clear
%% Amphibole solution model data
sitenames   = {'A',  'A' , 'A', 'M13','M13','M2', 'M2', 'M2','M2' , 'M2', 'M4','M4','M4','M4','T1','T1','V' ,'V'};
occupancy   = {'v',  'Na', 'K', 'Mg' ,'Fe' ,'Mg', 'Fe', 'Al','Fe3', 'Ti', 'Ca','Mg','Fe','Na','Si','Al','OH','O'};
st          = [
                1,    0,    0,   3,    0,    2,    0,    0,    0,    0,    2,   0,   0,   0,   4,   0,   2,   0;
                1,    0,    0,   3,    0,    0,    0,    2,    0,    0,    2,   0,   0,   0,   2,   2,   2,   0;
                0,    1,    0,   3,    0,    1,    0,    1,    0,    0,    2,   0,   0,   0,   2,   2,   2,   0;
                1,    0,    0,   3,    0,    0,    0,    2,    0,    0,    0,   0,   0,   2,   4,   0,   2,   0;
                1,    0,    0,   3,    0,    2,    0,    0,    0,    0,    0,   2,   0,   0,   4,   0,   2,   0;
                1,    0,    0,   0,    3,    0,    2,    0,    0,    0,    0,   0,   2,   0,   4,   0,   2,   0;
                1,    0,    0,   3,    0,    0,    2,    0,    0,    0,    0,   0,   2,   0,   4,   0,   2,   0;
                1,    0,    0,   0,    3,    2,    0,    0,    0,    0,    0,   0,   2,   0,   4,   0,   2,   0;
                1,    0,    0,   3,    0,    0,    0,    0,    2,    0,    0,   0,   0,   2,   4,   0,   2,   0;
                0,    0,    1,   3,    0,    1,    0,    1,    0,    0,    2,   0,   0,   0,   2,   2,   2,   0;
                1,    0,    0,   3,    0,    0,    0,    0,    0,    2,    2,   0,   0,   0,   2,   2,   0,   2];
Cname       = {'Si','Ti','Al','Ca','Fe','Mg','Na','K','H','O'};
comp_table = [
                8,   0,   0,   2,   0,   5,   0,   0,  2,  24;
                6,   0,   4,   2,   0,   3,   0,   0,  2,  24;
                6,   0,   3,   2,   0,   4,   1,   0,  2,  24;
                8,   0,   2,   0,   0,   3,   2,   0,  2,  24;
                8,   0,   0,   0,   0,   7,   0,   0,  2,  24;
                8,   0,   0,   0,   7,   0,   0,   0,  2,  24;
                8,   0,   0,   0,   4,   3,   0,   0,  2,  24;
                8,   0,   0,   0,   5,   2,   0,   0,  2,  24;
                8,   0,   0,   0,   2,   3,   2,   0,  2,  24;
                6,   0,   3,   2,   0,   4,   0,   1,  2,  24;
                6,   2,   2,   2,   0,   3,   0,   0,  0,  24]';
%% Find the site indices
sites   = unique(sitenames,'stable'); % List of all sites
site_id = zeros(1,length(sitenames'));
for i_site = 1:length(sites)
    site_id(strcmp(sitenames,sites(i_site))) = i_site; % index for the site (rather than a string with names)
end
%% Make site fraction table
max_st = zeros(size(st));
for i_site = 1:max(site_id)
    max_st(:,site_id==i_site) = repmat(sum(st(:,site_id==i_site),2),1,sum(site_id==i_site));% Find maximum site occupancy moles
end
zt = st./max_st;   zt(isnan(zt)) = 0;
%% Site fraction as function of proportion
z_equations          = [ones(1,size(zt,1)); zt'];% The proportions summing up to one, and z equations in one matrix
[z_eqns,ieq_z_indep] = indep_eqns(z_equations);  % Find independent site fraction equations
site_var             = ieq_z_indep(2:end) - 1; % Find independent site variable index (this assumes first equation is sum(p)=1)
p_from_z_cons        = inv(z_eqns);              % Proportion from site fraction matrix
%% Bulk chemistry composition as function of proportion
p_equations          = [ones(1,size(comp_table,2)); comp_table; zt'];         % Augmented matrix of sum(p)=1, comp from p, and zt'
eq_id                = [1 2*ones(1,size(comp_table,1)) 3*ones(1,size(zt,2))]; % Identifier for equation type
[p_eqns,ieq_p_indep] = indep_eqns(p_equations);                               % Find independent equations
icomp_indep          = find(eq_id(ieq_p_indep)==2)-1;                         % Find independent compositional variables
isite_indep = ieq_p_indep(eq_id(ieq_p_indep)==3)-sum(eq_id==2)-sum(eq_id==1); % Find site fraction variables determining order-disorder
p_from_c_cons = inv(p_eqns);                                                 % Proportion from composition matrix
%% Function for automated search for independent equations
function [eqns,ieq_indep] = indep_eqns(all_equations)
% Find independent equations in p_constraints
eqns = all_equations;
ieq_indep    = 1:size(all_equations,1);
for icons = 1:size(all_equations,1)
    itry       = 1:size(eqns,1);
    ieqn       = size(all_equations,1) + 1 - icons;
    itry(ieqn) = [];
    if rank(eqns) == rank(eqns(itry,:))
        eqns           = eqns(itry,:);
        ieq_indep(end+1-icons) = -1;
    end
end
ieq_indep(ieq_indep<0) = [];
end
```

Figure A1.

Table A3
Table of Bulk Chemistry of Endmembers in Amphibole

Element	Si	Ti	Al	Ca	Fe	Mg	Na	K	H	O
Endmember										
Tremolite	8	0	0	2	0	5	0	0	2	24
Tschermakite	6	0	4	2	0	3	0	0	2	24
Pargasite	6	0	3	2	0	4	1	0	2	24
Glaucophane	8	0	2	0	0	3	2	0	2	24
Cummingtonite	8	0	0	0	0	7	0	0	2	24
Grunerite	8	0	0	0	7	0	0	0	2	24
Ordered Amphibole A	8	0	0	0	4	3	0	0	2	24
Ordered Amphibole B	8	0	0	0	5	2	0	0	2	24
Magnesio-riebekite	8	0	0	0	2	3	2	0	2	24
K-Pargasite	6	0	3	2	0	4	0	1	2	24
Ti-Tschermakite	6	2	2	2	0	3	0	0	0	24

Endmember proportions can also be obtained from the bulk composition using the table of moles of elements of the endmembers:

First, the composition table and transposed site fraction table (i.e., the matrix in Equation A1) are augmented along with the equation that proportions sum up to one (which forms the first equation in the set):

Figure A1. Example of automated method for determining independent compositional variables, site fractions, and order-disorder variables for an amphibole with 11 endmembers (Green et al., 2016). The script produces matrices that are used to convert site fractions to proportions, or vice versa, and from bulk chemical composition to proportion. In lines 3 and 4 the names and occupancy of the crystallographic sites are defined. Lines 5–29 are definition of the input site occupancy and bulk chemistry table (Tables A1 and A3). In lines 31–35 an identifier for each site is made, such that site A, will get the index 1, site M13, index 2, etc. This is used in lines 37–40 to retrieve the maximum moles per site to use in producing the site fraction table in zt (Table A2). Matrices Equations A1–A3 are formed in lines 43–46, where a generalized function defined in lines 55–69 is used. This function finds for any input set of equations the independent set, by starting from the first equation in the matrix and adding only equations that increase the rank of the system. Lines 48–53 also make use of this function to find the matrices in Equations A4–A6.

$$
\begin{bmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
8 & 6 & 6 & 8 & 8 & 8 & 8 & 8 & 8 & 6 & 6 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 \\
0 & 4 & 3 & 2 & 0 & 0 & 0 & 0 & 0 & 3 & 2 \\
2 & 2 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 2 \\
0 & 0 & 0 & 0 & 0 & 7 & 4 & 5 & 2 & 0 & 0 \\
5 & 3 & 4 & 3 & 7 & 0 & 3 & 2 & 3 & 4 & 3 \\
0 & 0 & 1 & 2 & 0 & 0 & 0 & 0 & 2 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 0 \\
24 & 24 & 24 & 24 & 24 & 24 & 24 & 24 & 24 & 24 & 24 \\
1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\
1 & 0 & 0.5 & 0 & 1 & 0 & 0 & 1 & 0 & 0.5 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0.5 & 1 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
1 & 0.5 & 0.5 & 1 & 1 & 1 & 1 & 1 & 1 & 0.5 & 0.5 \\
0 & 0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0.5 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
=
\begin{bmatrix}
p_{tr} \\ p_{ts} \\ p_{par} \\ p_{gl} \\ p_{cm} \\ p_{grun} \\ p_{a} \\ p_{b} \\ p_{mrb} \\ p_{kpar} \\ p_{tts}
\end{bmatrix}
=
\begin{bmatrix}
1 \\ Si \\ Ti \\ Al \\ Ca \\ Fe \\ Mg \\ Na \\ K \\ H \\ O \\ z_{v}^{A} \\ z_{\mathrm{Na}}^{A} \\ z_{K}^{A} \\ z_{\mathrm{Mg}}^{M13} \\ z_{\mathrm{Fe}}^{13} \\ z_{\mathrm{Mg}}^{M2} \\ z_{\mathrm{Fe}}^{M2} \\ z_{\mathrm{Al}}^{M2} \\ z_{\mathrm{Fe3}}^{M2} \\ z_{\mathrm{Ti}}^{M2} \\ z_{Ca}^{M4} \\ z_{Mg}^{M4} \\ z_{Fe}^{M4} \\ z_{Na}^{M4} \\ z_{Si}^{T1} \\ z_{Al}^{T1} \\ z_{OH}^{V} \\ z_{O}^{V}
\end{bmatrix}
\tag{A4}
$$

The automatic selection of independent compositional variables and independent site fractions to describe ordering in the mineral starts from taking the first equation and adding from the top down only the equations that increase the rank of the system until the rank of the selected system of equations equals the number of endmember proportions. For Equation A4 above, this results in the following system of equations:

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 8 & 6 & 6 & 8 & 8 & 8 & 8 & 8 & 8 & 6 & 6 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 \\ 0 & 4 & 3 & 2 & 0 & 0 & 0 & 0 & 0 & 3 & 2 \\ 2 & 2 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 2 \\ 0 & 0 & 0 & 0 & 0 & 7 & 4 & 5 & 2 & 0 & 0 \\ 5 & 3 & 4 & 3 & 7 & 0 & 3 & 2 & 3 & 4 & 3 \\ 0 & 0 & 1 & 2 & 0 & 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0.5 & 0 & 1 & 0 & 0 & 1 & 0 & 0.5 & 0 \end{bmatrix} \begin{bmatrix} p_{tr} \\ p_{ts} \\ p_{par} \\ p_{gl} \\ p_{cm} \\ p_{grun} \\ p_{a} \\ p_{b} \\ p_{mrb} \\ p_{kpar} \\ p_{tts} \end{bmatrix} = \begin{bmatrix} 1 \\ \mathrm{Si} \\ \mathrm{Ti} \\ \mathrm{Al} \\ \mathrm{Ca} \\ \mathrm{Fe} \\ \mathrm{Mg} \\ \mathrm{Na} \\ \mathrm{K} \\ z_{\mathrm{Mg}}^{M13} \\ z_{\mathrm{Mg}}^{M2} \end{bmatrix} \quad \text{(A5)}$$

$$\begin{bmatrix} -4.0 & 0.5 & 0.0 & 0.0 & 0.5 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 19.0 & -1.5 & -1.5 & -1.0 & -1.0 & -1.0 & -1.0 & -1.0 & -1.0 & 0.0 & 0.0 \\ -15.0 & 1.0 & 1.0 & 1.0 & 1.0 & 1.0 & 1.0 & 1.0 & 0.0 & 0.0 & 0.0 \\ -15.5 & 1.5 & 1.0 & 1.0 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.5 & 0.0 & 0.0 & -1.5 & -1.0 \\ -10.5 & 1.0 & 0.5 & 0.5 & 1.0 & 0.5 & 1.0 & 0.5 & 0.5 & -2.5 & -2.0 \\ -7.5 & 0.5 & 0.5 & 0.5 & 0.0 & 0.5 & 0.0 & 0.0 & 0.0 & 2.5 & 1.0 \\ 11.5 & -1.0 & -0.5 & -0.5 & -1.0 & -0.5 & -1.0 & -0.5 & -0.5 & 1.5 & 2.0 \\ 23.0 & -2.0 & -1.5 & -1.5 & -1.0 & -1.0 & -1.0 & -0.5 & -0.5 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 1.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.5 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \end{bmatrix} \begin{bmatrix} 1 \\ \mathrm{Si} \\ \mathrm{Ti} \\ \mathrm{Al} \\ \mathrm{Ca} \\ \mathrm{Fe} \\ \mathrm{Mg} \\ \mathrm{Na} \\ \mathrm{K} \\ z_{\mathrm{Mg}}^{M13} \\ z_{\mathrm{Mg}}^{M2} \end{bmatrix} = \begin{bmatrix} p_{tr} \\ p_{ts} \\ p_{par} \\ p_{gl} \\ p_{cm} \\ p_{grun} \\ p_{a} \\ p_{b} \\ p_{mrb} \\ p_{kpar} \\ p_{tts} \end{bmatrix} \quad \text{(A6)}$$

Conflict of Interest

The authors declare no conflicts of interest relevant to this study.

Data Availability Statement

Thermolab is made available at https://hansjcv.github.io/Thermolab/ and in the Zenodo repository under https://doi.org/10.5281/zenodo.6334839 (Vrijmoed & Podladchikov, 2022).

Acknowledgments
The Deutsche Forschungsgemeinschaft (DFG) financially supported this research for J. C. Vrijmoed through grant CRC 1114 "Scaling Cascades in Complex Systems," Project Number 235221301, Project (C09)—"Dynamics of rock dehydration on multiple scales." Y. Y. Podladchikov was supported by the Russian Ministry of Science and Higher Education (Project No. 075-15-2019-1890). Open access funding enabled and organized by Projekt DEAL. The authors are grateful for constructive reviews by Meng Tian and Nicolas Riel which improved the manuscript.

References

Abart, R., Petrishcheva, E., Wirth, R., & Rhede, D. (2009). Exsolution by spinodal decomposition II: Perthite formation during slow cooling of anatexites from Ngoronghoro, Tanzania. *American Journal of Science*, *309*(6), 450–475. https://doi.org/10.2475/06.2009.02

Ague, J. J., & Baxter, E. F. (2007). Brief thermal pulses during mountain building recorded by Sr diffusion in apatite and multicomponent diffusion in garnet. *Earth and Planetary Science Letters*, *261*(3–4), 500–516. https://doi.org/10.1016/j.epsl.2007.07.017

Aharonov, E., Whitehead, J. A., Kelemen, P. B., & Spiegelman, M. (1995). Channeling instability of upwelling melt in the mantle. *Journal of Geophysical Research*, *100*(B10), 20433–20450. https://doi.org/10.1029/95jb1307

Anderson, G. M., Castet, S., Schott, J., & Mesmer, R. E. (1991). The density model for estimation of thermodynamic parameters of reactions at high-temperatures and pressures. *Geochimica et Cosmochimica Acta*, *55*(7), 1769–1779. https://doi.org/10.1016/0016-7037(91)90022-w

Aranovich, L. Y., Akinfiev, N. N., & Golunova, M. (2020). Quartz solubility in sodium carbonate solutions at high pressure and temperature. *Chemical Geology*, 550. https://doi.org/10.1016/j.chemgeo.2020.119699

Aranovich, L. Y., & Newton, R. C. (1999). Experimental determination of CO_2-H_2O activity-composition relations at 600–1000°C and 6–14 kbar by reversed decarbonation and dehydration reactions. *American Mineralogist*, *84*(9), 1319–1332. https://doi.org/10.2138/am-1999-0908

Balashov, V. N., & Yardley, B. W. D. (1998). Modeling metamorphic fluid flow with reaction-compaction-permeability feedbacks. *American Journal of Science*, *298*(6), 441–470. https://doi.org/10.2475/ajs.298.6.441

Bebout, G. E., Scholl, D. W., Stern, R. J., Wallace, L. M., & Agard, P. (2018). Twenty years of subduction zone science: Subduction top to bottom 2 (ST2B-2). *Geological Society of America Today*, 4–10. https://doi.org/10.1130/gsatg354a.1

Beinlich, A., John, T., Vrijmoed, J. C., Tominaga, M., Magna, T., & Podladchikov, Y. Y. (2020). Instantaneous rock transformations in the deep crust driven by reactive fluid flow. *Nature Geoscience*, *13*(4), 307–311. https://doi.org/10.1038/s41561-020-0554-9

Berman, R. G. (1988). Internally-consistent thermodynamic data for minerals in the system Na_2O-K_2O-CaO-MgO-FeO-Fe_2O_3-Al_2O_3-SiO_2-TiO_2-H_2O-CO_2. *Journal of Petrology*, *29*(2), 445–522. https://doi.org/10.1093/petrology/29.2.445

Berman, R. G., & Brown, T. H. (1985). Heat-capacity of minerals in the system Na_2O-K_2O-CaO-MgO-FeO-Fe_2O_3-Al_2O_3-SiO_2-TiO_2-H_2O-CO_2 – Representation, estimation, and high-temperature extrapolation. *Contributions to Mineralogy and Petrology*, *89*(2–3), 168–183. https://doi.org/10.1007/bf00379451

Bowen, N. L., & Schairer, J. F. (1935). The system MgO-FeO-SiO_2. *American Journal of Science*, *s5–29*(170), 151–217. https://doi.org/10.2475/ajs.s5-29.170.151

Cahn, J. W., & Hilliard, J. E. (1958). Free energy of a Nonuniform system. 1. Interfacial free energy. *The Journal of Chemical Physics*, *28*(2), 258–267. https://doi.org/10.1063/1.1744102

Cemič, L. (2005). *Thermodynamics in mineral sciences: An introduction*. Springer Berlin/Heidelberg.

Coggon, R., & Holland, T. J. B. (2002). Mixing properties of phengitic micas and revised garnet-phengite thermobarometers. *Journal of Metamorphic Geology*, *20*(7), 683–696. https://doi.org/10.1046/j.1525-1314.2002.00395.x

Connolly, J. A. D. (2005). Computation of phase equilibria by linear programming: A tool for geodynamic modeling and its application to subduction zone decarbonation. *Earth and Planetary Science Letters*, *236*(1–2), 524–541. https://doi.org/10.1016/j.epsl.2005.04.033

Connolly, J. A. D. (1997). Devolatilization-generated fluid pressure and deformation-propagated fluid flow during prograde regional metamorphism. *Journal of Geophysical Research*, *102*(B8), 18149–18173. https://doi.org/10.1029/97jb00731

Connolly, J. A. D. (2009). The geodynamic equation of state: What and how. *Geochemistry, Geophysics, Geosystems*, *10*. https://doi.org/10.1029/2009gc002540

Connolly, J. A. D. (2017). A primer in Gibbs energy minimization for geophysicists. *Petrology*, *25*(5), 526–534. https://doi.org/10.1134/s0869591117050034

Connolly, J. A. D., & Podladchikov, Y. Y. (2007). Decompaction weakening and channeling instability in ductile porous media: Implications for asthenospheric melt segregation. *Journal of Geophysical Research*, *112*(B10), B10205. https://doi.org/10.1029/2005jb004213

Dantzig, G. B., Orden, A., & Wolfe, P. (1955). The generalized simplex method for minimizing a linear form under linear inequality restraints. *Pacific Journal of Mathematics*, *5*(2), 183–195. https://doi.org/10.2140/pjm.1955.5.183

de Capitani, C., & Brown, T. H. (1987). The computation of chemical equilibrium in complex systems containing non-ideal solutions. *Geochimica et Cosmochimica Acta*, *51*(10), 2639–2652. https://doi.org/10.1016/0016-7037(87)90145-1

De Groot, S., & Mazur, P. (1984). *Non-equilibrium thermodynamics*. Dover Publications, Inc.

Delany, J. M., & Helgeson, H. C. (1978). Calculation of thermodynamic consequences of dehydration in subducting oceanic-crust to 100 kb and greater-than 800°C. *American Journal of Science*, *278*(5), 638–686. https://doi.org/10.2475/ajs.278.5.638

Diener, J. F. A., Powell, R., White, R. W., & Holland, T. J. B. (2007). A new thermodynamic model for clino- and orthoamphiboles in the system Na_2O-CaO-FeO-MgO-Al_2O_3-SiO_2-H_2O-O. *Journal of Metamorphic Geology*, *25*(6), 631–656. https://doi.org/10.1111/j.1525-1314.2007.00720.x

Dolejš, D. (2013). Thermodynamics of aqueous species at high temperatures and pressures: Equations of state and transport theory. *Reviews in Mineralogy and Geochemistry*, *76*(1), 35–79. https://doi.org/10.1515/9781501508295-003

Dolejš, D., & Wagner, T. (2008). Thermodynamic modeling of non-ideal mineral–fluid equilibria in the system Si–Al–Fe–Mg–Ca–Na–K–H–O–Cl at elevated temperatures and pressures: Implications for hydrothermal mass transfer in granitic rocks. *Geochimica et Cosmochimica Acta*, *72*(2), 526–553. https://doi.org/10.1016/j.gca.2007.10.022

Evans, K. A., & Powell, R. (2007). DES-code: A metacode to aid calculation of the chemical potential of aqueous solutions at elevated temperatures and pressures. *Computers & Geosciences*, *33*(6), 789–807. https://doi.org/10.1016/j.cageo.2006.09.010

Fagents, S., Gregg, T., & Lopes, R. (Eds.). (2013). *Modeling volcanic processes: The Physics and Mathematics of Volcanism*. Cambridge University Press. https://doi.org/10.1017/CBO9781139021562

Feng, G., Wang, Y., Xu, T., Wang, F., & Shi, Y. (2021). Multiphase flow modeling and energy extraction performance for supercritical geothermal systems. *Renewable Energy*, *173*, 442–454. https://doi.org/10.1016/j.renene.2021.03.107

Fernández, D. P., Goodwin, A. R. H., Lemmon, E. W., Levelt Sengers, J. M. H., & Williams, R. C. (1997). A formulation for the static permittivity of water and Steam at temperatures from 238 K to 873 K at pressures up to 1200 MPa, including derivatives and Debye–Hückel coefficients. *Journal of Physical and Chemical Reference Data*, *26*(4), 1125–1166. https://doi.org/10.1063/1.555997

Fletcher, R. C., & Hofmann, A. W. (1974). Simple models of diffusion and combined diffusion-infiltration metasomatism. *Geochemical Transport and Kinetics*, 243–259.

Ganguly, J. (2020). *Thermodynamics in Earth and Planetary Sciences* (2nd ed.). Springer International Publishing. https://doi.org/10.1007/978-3-030-20879-0

Geller, R. J. (1997). Earthquake prediction: A critical review. *Geophysical Journal International*, *131*(3), 425–450. https://doi.org/10.1111/j.1365-246x.1997.tb06588.x

Ghiorso, M. S., Hirschmann, M. M., Reiners, P. W., & Kress, V. C. (2002). The pMELTS: A revision of MELTS for improved calculation of phase relations and major element partitioning related to partial melting of the mantle to 3 GPa. *Geochemistry, Geophysics, Geosystems*, *3*(5), 1–35. https://doi.org/10.1029/2001gc000217

Glasser, F. P., Jauffret, G., Morrison, J., Galvez-Martos, J.-L., Patterson, N., & Imbabi, M. S.-E. (2016). Sequestering CO_2 by mineralization into useful nesquehonite-based products. *Frontiers in Energy Research*, *4*. https://doi.org/10.3389/fenrg.2016.00003

Gordon, S., & McBride, B. J. (1994). Computer program for the calculation of complex chemical equilibrium compositions with applications; I. Analysis. *NASA Reference Publication*, 1311.

Green, E. C. R., White, R. W., Diener, J. F. A., Powell, R., Holland, T. J. B., & Palin, R. M. (2016). Activity-composition relations for the calculation of partial melting equilibria in metabasic rocks. *Journal of Metamorphic Geology*, *34*(9), 845–869. https://doi.org/10.1111/jmg.12211

Guy, B. (1993). Mathematical revision of Korzhinskii's theory of infiltration metasomatic zoning. *European Journal of Mineralogy*, *5*(2), 317–340. https://doi.org/10.1127/ejm/5/2/0317

Helgeson, H. C., Delany, J. M., Nesbitt, H. W., & Bird, D. K. (1978). Summary and critique of the thermodynamic properties of rock-forming minerals. *American Journal of Science*, *278*, 1–229.

Helgeson, H. C., Kirkham, D. H., & Flowers, G. C. (1981). Theoretical prediction of the thermodynamic behavior of aqueous-electrolytes at high-pressures and temperatures. 4. Calculation of activity-coefficients, osmotic coefficients, and apparent molal and standard and relative partial molal properties to 600-°C and 5 Kb. *American Journal of Science*, *281*(10), 1249–1516. https://doi.org/10.2475/ajs.281.10.1249

Hofmann, A. (1972). Chromatographic theory of infiltration metasomatism and its application to feldspars. *American Journal of Science*, *272*(1), 69–90. https://doi.org/10.2475/ajs.272.1.69

Holland, T. J. B., Baker, J., & Powell, R. (1998). Mixing properties and activity-composition relationships of chlorites in the system MgO-FeO-Al_2O_3-SiO_2-H_2O. *European Journal of Mineralogy*, *10*(3), 395–406. https://doi.org/10.1127/ejm/10/3/0395

Holland, T. J. B., Green, E. C. R., & Powell, R. (2018). Melting of peridotites through to granites: A simple thermodynamic model in the system KNCFMASHTOCr. *Journal of Petrology*, *59*(5), 881–899. https://doi.org/10.1093/petrology/egy048

Holland, T. J. B., & Powell, R. (1991). A Compensated-Redlich-Kwong (CORK) equation for volumes and fugacities of CO_2 and H_2O in the range 1-bar to 50-kbar and 100-1600-°C. *Contributions to Mineralogy and Petrology*, *109*(2), 265–273. https://doi.org/10.1007/bf00306484

Holland, T. J. B., & Powell, R. (1996). Thermodynamics of order-disorder in minerals. 1. Symmetric formalism applied to minerals of fixed composition. *American Mineralogist*, *81*(11–12), 1413–1424. https://doi.org/10.2138/am-1996-11-1214

Holland, T. J. B., & Powell, R. (1998). An internally consistent thermodynamic data set for phases of petrological interest. *Journal of Metamorphic Geology*, *16*(3), 309–343. https://doi.org/10.1111/j.1525-1314.1998.00140.x

Holland, T. J. B., & Powell, R. (2003). Activity-composition relations for phases in petrological calculations: An asymmetric multicomponent formulation. *Contributions to Mineralogy and Petrology*, *145*(4), 492–501. https://doi.org/10.1007/s00410-003-0464-z

Holland, T. J. B., & Powell, R. (2011). An improved and extended internally consistent thermodynamic dataset for phases of petrological interest, involving a new equation of state for solids. *Journal of Metamorphic Geology*, *29*(3), 333–383. https://doi.org/10.1111/j.1525-1314.2010.00923.x

Huang, F., & Sverjensky, D. A. (2019). Extended Deep Earth Water Model for predicting major element mantle metasomatism. *Geochimica et Cosmochimica Acta*, *254*, 192–230. https://doi.org/10.1016/j.gca.2019.03.027

John, T., Gussone, N., Podladchikov, Y. Y., Bebout, G. E., Dohmen, R., Halama, R., et al. (2012). Volcanic arcs fed by rapid pulsed fluid flow through subducting slabs. *Nature Geoscience*, *5*(7), 489–492. https://doi.org/10.1038/ngeo1482

Johnson, J. W., & Norton, D. (1991). Critical phenomena in hydrothermal systems – State, thermodynamic, electrostatic, and transport-properties of H_2O in the critical region. *American Journal of Science*, *291*(6), 541–648. https://doi.org/10.2475/ajs.291.6.541

Johnson, J. W., Oelkers, E. H., & Helgeson, H. C. (1992). Supcrt92 – A software package for calculating the standard molal thermodynamic properties of minerals, gases, aqueous species, and reactions from 1-bar to 5000-bar and 0 to 1000°C. *Computers & Geosciences*, *18*(7), 899–947. https://doi.org/10.1016/0098-3004(92)90029-q

Kelemen, P. B., & Matter, J. (2008). In situ carbonation of peridotite for CO_2 storage. *Proceedings of the National Academy of Sciences*, *105*(45), 17295–17300. https://doi.org/10.1073/pnas.0805794105

Kulik, D. A., Wagner, T., Dmytrieva, S. V., Kosakowski, G., Hingerl, F. F., Chudnenko, K. V., & Berner, U. R. (2012). GEM-Selektor geochemical modeling package: Revised algorithm and GEMS3K numerical kernel for coupled simulation codes. *Computational Geosciences*. https://doi.org/10.1007/s10596-012-9310-6

Lebon, G., Jou, D., & Casas-Vázquez, J. (2008). *Understanding non-equilibrium thermodynamics: Foundations, applications, frontiers*. Springer. https://doi.org/10.1007/978-3-540-74252-4

Lichtner, P. C., & Carey, J. W. (2006). Incorporating solid solutions in reactive transport equations using a kinetic discrete-composition approach. *Geochimica et Cosmochimica Acta*, *70*(6), 1356–1378. https://doi.org/10.1016/j.gca.2005.11.028

Malvoisin, B., Podladchikov, Y. Y., & Vrijmoed, J. C. (2015). Coupling changes in densities and porosity to fluid pressure variations in reactive porous fluid flow: Local thermodynamic equilibrium. *Geochemistry, Geophysics, Geosystems*, *16*(12), 4362–4387. https://doi.org/10.1002/2015gc006019

Manning, C. E. (1994). The solubility of quartz in H_2O in the lower crust and upper-mantle. *Geochimica et Cosmochimica Acta*, *58*(22), 4831–4839. https://doi.org/10.1016/0016-7037(94)90214-3

Miron, G. D., Wagner, T., Kulik, D. A., & Heinrich, C. A. (2016). Internally consistent thermodynamic data for aqueous species in the system Na-K-Al-Si-O-H-Cl. *Geochimica et Cosmochimica Acta*, *187*, 41–78. https://doi.org/10.1016/j.gca.2016.04.026

Miron, G. D., Wagner, T., Kulik, D. A., & Lothenbach, B. (2017). An internally consistent thermodynamic dataset for aqueous species in the system Ca-Mg-Na-K-Al-Si-O-H-C-Cl to 800°C and 5 kbar. *American Journal of Science*, *317*(7), 755–806. https://doi.org/10.2475/07.2017.01

Myhill, R., & Connolly, J. A. D. (2021). Notes on the creation and manipulation of solid solution models. *Contributions to Mineralogy and Petrology*, *176*(10), 86. https://doi.org/10.1007/s00410-021-01825-1

Nauman, E. B., & He, D. Q. (2001). Nonlinear diffusion and phase separation. *Chemical Engineering Science*, *56*(6), 1999–2018. https://doi.org/10.1016/s0009-2509(01)00005-7

Orr, F. M. (2005). *Theory of gas injection processes*. Stanford University.

Orr, F. M. (2018). Carbon capture, utilization, and storage: An update. *SPE Journal*, *23*(06), 2444–2455. https://doi.org/10.2118/194190-pa

Padron-Navarta, J. A., Sanchez-Vizcaino, V. L., Hermann, J., Connolly, J. A. D., Garrido, C. J., Gomez-Pugnaire, M. T., & Marchesi, C. (2013). Tschermak's substitution in antigorite and consequences for phase relations and water liberation in high-grade serpentinites. *Lithos*, *178*, 186–196. https://doi.org/10.1016/j.lithos.2013.02.001

Palin, R. M., White, R. W., Green, E. C. R., Diener, J. F. A., Powell, R., & Holland, T. J. B. (2016). High-grade metamorphism and partial melting of basic and intermediate rocks. *Journal of Metamorphic Geology*, *34*(9), 871–892. https://doi.org/10.1111/jmg.12212

Petrishcheva, E., & Abart, R. (2009). Exsolution by spinodal decomposition I: Evolution equation for binary mineral solutions with anisotropic interfacial energy. *American Journal of Science*, *309*(6), 431–449. https://doi.org/10.2475/06.2009.01

Pitzer, K. S., & Sterner, S. M. (1994). Equations of state valid continuously from zero to extreme pressures for H_2O and CO_2. *The Journal of Chemical Physics*, *101*(4), 3111–3116. https://doi.org/10.1063/1.467624

Plümper, O., John, T., Podladchikov, Y. Y., Vrijmoed, J. C., & Scambelluri, M. (2017). Fluid escape from subduction zones controlled by channel-forming reactive porosity. *Nature Geoscience*, *10*(2), 150–156. https://doi.org/10.1038/ngeo2865

Powell, R., & Holland, T. (1999). Relating formulations of the thermodynamics of mineral solid solutions: Activity modeling of pyroxenes, amphiboles, and micas. *American Mineralogist*, *84*(1–2), 1–14. https://doi.org/10.2138/am-1999-1-201

Rossi, C. C. R. S., Cardozo, L., & Guirardello, R. (2009). Gibbs free energy minimization for the calculation of chemical and phase equilibrium using linear programming. *Fluid Phase Equilibria*, *278*(1–2), 117–128. https://doi.org/10.1016/j.fluid.2009.01.007

Rundle, J. B., Stein, S., Donnellan, A., Turcotte, D. L., Klein, W., & Saylor, C. (2021). The complex dynamics of earthquake fault systems: New approaches to forecasting and nowcasting of earthquakes. *Reports on Progress in Physics*, *84*(7). https://doi.org/10.1088/1361-6633/abf893

Shao, H. B., Dmytrieva, S. V., Kolditz, O., Kulik, D. A., Pfingsten, W., & Kosakowski, G. (2009). Modeling reactive transport in non-ideal aqueous-solid solution system. *Applied Geochemistry*, *24*(7), 1287–1300. https://doi.org/10.1016/j.apgeochem.2009.04.001

Shomate, C. H. (1954). A method for evaluating and correlating thermodynamic data. *Journal of Physical Chemistry*, *58*(4), 368–372. https://doi.org/10.1021/j150514a018

Sonnenthal, E., Ito, A., Spycher, N., Yui, M., Apps, J., Sugita, Y., et al. (2005). Approaches to modeling coupled thermal, hydrological, and chemical processes in the Drift Scale Heater Test at Yucca Mountain. *International Journal of Rock Mechanics and Mining Sciences*, *42*(5–6), 698–719. https://doi.org/10.1016/j.ijrmms.2005.03.009

Steefel, C. I. (2019). Reactive transport at the crossroads. *Reviews in Mineralogy and Geochemistry*, *85*(1), 1–26. https://doi.org/10.1515/9781501512001-002

Steefel, C. I., Appelo, C. A. J., Arora, B., Jacques, D., Kalbacher, T., Kolditz, O., et al. (2014). Reactive transport codes for subsurface environmental simulation. *Computational Geosciences*, *19*(3), 445–478. https://doi.org/10.1007/s10596-014-9443-x

Steefel, C. I., DePaolo, D. J., & Lichtner, P. C. (2005). Reactive transport modeling: An essential tool and a new research approach for the Earth sciences. *Earth and Planetary Science Letters*, *240*(3–4), 539–558. https://doi.org/10.1016/j.epsl.2005.09.017

Stixrude, L., & Lithgow-Bertelloni, C. (2022). Thermal expansivity, heat capacity and bulk modulus of the mantle. *Geophysical Journal International*, *228*(2), 1119–1149. https://doi.org/10.1093/gji/ggab394

Sverjensky, D. A., Harrison, B., & Azzolini, D. (2014). Water in the deep Earth: The dielectric constant and the solubilities of quartz and corundum to 60 kb and 1200°C. *Geochimica et Cosmochimica Acta*, *129*, 125–145. https://doi.org/10.1016/j.gca.2013.12.019

Tajčmanová, L., Connolly, J. A. D., & Cesare, B. (2009). A thermodynamic model for titanium and ferric iron solution in biotite. *Journal of Metamorphic Geology*, *27*(2), 153–165. https://doi.org/10.1111/j.1525-1314.2009.00812.x

Tanger, J. C., & Helgeson, H. C. (1988). Calculation of the thermodynamic and transport-properties of aqueous species at high-pressures and temperatures – Revised equations of state for the standard partial molal properties of ions and electrolytes. *American Journal of Science*, *288*(1), 19–98. https://doi.org/10.2475/ajs.288.1.19

Tian, M., & Ague, J. J. (2014). The impact of porosity waves on crustal reaction progress and CO_2 mass transfer. *Earth and Planetary Science Letters*, *390*, 80–92. https://doi.org/10.1016/j.epsl.2013.12.044

Tian, M., Ague, J. J., Chu, X., Baxter, E. F., Dragovic, N., Chamberlain, C. P., & Rumble, D. (2018). The potential for metamorphic thermal pulses to develop during compaction-driven fluid flow. *Geochemistry, Geophysics, Geosystems*, *19*(1), 232–256. https://doi.org/10.1002/2017gc007269

Tian, M., Katz, R. F., & Rees Jones, D. W. (2019). Devolatilization of Subducting Slabs, Part I: Thermodynamic parameterization and open system effects. *Geochemistry, Geophysics, Geosystems*, *20*(12), 5667–5690. https://doi.org/10.1029/2019gc008488

Tian, M., Katz, R. F., Rees Jones, D. W., & May, D. A. (2019). Devolatilization of Subducting Slabs, Part II: Volatile fluxes and storage. *Geochemistry, Geophysics, Geosystems*, *20*(12), 6199–6222. https://doi.org/10.1029/2019gc008489

Ulrich, T., Vater, S., Madden, E. H., Behrens, J., van Dinther, Y., van Zelst, I., et al. (2019). Coupled, physics-based modeling reveals earthquake displacements are critical to the 2018 Palu, Sulawesi tsunami. *Pure and Applied Geophysics*, *176*(10), 4069–4109. https://doi.org/10.1007/s00024-019-02290-5

Utkin, I., & Afanasyev, A. (2021). Decompaction weakening as a mechanism of fluid focusing in hydrothermal systems. *Journal of Geophysical Research*, *126*(9). https://doi.org/10.1029/2021jb022397

Van Laar, J. J. (1906). *Sechs Vorträge über das thermodynamische Potential und seine Anwendungen auf chemische und physikalische Gleichgewichtsprobleme: eingeleitet durch zwei Vorträge über nichtverdünnte Lösungen und über den osmotischen Druck*. Vieweg.

Vehling, F., Hasenclever, J., & Rüpke, L. (2020). Brine formation and mobilization in submarine hydrothermal systems: Insights from a Novel multiphase hydrothermal flow model in the system H_2O–NaCl. *Transport in Porous Media*, *136*(1), 65–102. https://doi.org/10.1007/s11242-020-01499-6

Vrijmoed, J. C., & Podladchikov, Y. Y. (2015). Thermodynamic equilibrium at heterogeneous pressure. *Contributions to Mineralogy and Petrology*, *170*(1), 10. https://doi.org/10.1007/s00410-015-1156-1

Vrijmoed, J. C., & Podladchikov, Y. Y. (2022). *Thermolab version 23.03.2022* (Version v2022.03.23) [Computer software]. Zenodo. https://doi.org/10.5281/ZENODO.6383253

Wagner, W., & Pruss, A. (2002). The IAPWS formulation 1995 for the thermodynamic properties of ordinary water substance for general and scientific use. *Journal of Physical and Chemical Reference Data*, *31*(2), 387–535. https://doi.org/10.1063/1.1461829

Wanner, C., Peiffe, L., Sonnenthal, E., Spycher, N., Iovenitti, J., & Kennedy, B. M. (2014). Reactive transport modeling of the Dixie Valley geothermal area: Insights on flow and geothermometry. *Geothermics*, *51*, 130–141. https://doi.org/10.1016/j.geothermics.2013.12.003

Weis, P., Driesner, T., & Heinrich, C. A. (2012). Porphyry-copper ore shells form at stable pressure-temperature fronts within dynamic fluid plumes. *Science*, *338*(6114), 1613–1616. https://doi.org/10.1126/science.1225009

White, R. W., Powell, R., & Holland, T. J. B. (2007). Progress relating to calculation of partial melting equilibria for metapelites. *Journal of Metamorphic Geology*, *25*(5), 511–527. https://doi.org/10.1111/j.1525-1314.2007.00711.x

White, R. W., Powell, R., & Phillips, G. N. (2003). A mineral equilibria study of the hydrothermal alteration in mafic greenschist facies rocks at Kalgoorlie, Western Australia. *Journal of Metamorphic Geology*, *21*(5), 455–468. https://doi.org/10.1046/j.1525-1314.2003.00454.x

White, W. B., Johnson, S. M., & Dantzig, G. B. (1958). Chemical equilibrium in complex mixtures. *The Journal of Chemical Physics*, *28*(5), 751–755. https://doi.org/10.1063/1.1744264

Wilson, C. R., Spiegelman, M., van Keken, P. E., & Hacker, B. R. (2014). Fluid flow in subduction zones: The role of solid rheology and compaction pressure. *Earth and Planetary Science Letters*, *401*, 261–274. https://doi.org/10.1016/j.epsl.2014.05.052

Zhang, C., & Duan, Z. H. (2009). A model for C-O-H fluid in the Earth's mantle. *Geochimica et Cosmochimica Acta*, *73*(7), 2089–2102. https://doi.org/10.1016/j.gca.2009.01.021

Zhang, X., Ma, F., Dai, Z., Wang, J., Chen, L., Ling, H., & Soltanian, M. R. (2022). Radionuclide transport in multi-scale fractured rocks: A review. *Journal of Hazardous Materials*, *424*, 127550. https://doi.org/10.1016/j.jhazmat.2021.127550

Zhang, Z. G., & Duan, Z. H. (2005). Prediction of the PVT properties of water over wide range of temperatures and pressures from molecular dynamics simulation. *Physics of the Earth and Planetary Interiors*, *149*(3–4), 335–354. https://doi.org/10.1016/j.pepi.2004.11.003