1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21

MinKin: A kinetic modeling program for the
precipitation, dissolution, and phase transformation
of minerals in aqueous solution

Daniel R. Hummer^{1*} and Peter J. Heaney¹

¹Department of Geosciences The Pennsylvania State University, University Park, PA
16802

* Corresponding author
Present address: Department of Earth, Planetary and Space Sciences, University of
California Los Angeles, Los Angeles, CA 90095
E-mail: dhummer@epss.ucla.edu
Phone: (814) 321-8859
Fax: (310) 825-2779

Abstract

The development of time-resolved, synchrotron scattering techniques has recently enabled the collection of *in situ* mineral abundance data during mineral-fluid reactions. However, few computational algorithms exist to analyze the kinetics of such reactions. Here we present MinKin (for “Mineral Kinetics”), a global optimization code for Matlab capable of fitting a standard chemical kinetic model to experimental mineral abundance data. MinKin allows users to specify the species and reactions of a geochemical system consisting of a fluid with up to two aqueous species and up to three mineral species, and then uses the global optimization algorithm of Differential Evolution (DE) to calculate the rate constants that minimize the error between the model and the data. Trial calculations reveal that MinKin is able to simultaneously calculate up to six rate constants on a time scale of minutes, with an accuracy of roughly the same magnitude as that of the input data.

Keywords

Kinetics, precipitation, dissolution, modeling, optimization, rate constants

1 Introduction

A common goal in chemical kinetics is to extract reaction rates from an experimental time series of concentrations or abundances. Because the abundance of each of the system’s chemical species is potentially affected by multiple reactions, however,

direct computation of rates is prohibitively complex for all but the simplest chemical systems. For this reason, rate information is normally extracted via forward kinetic models that are fit to the experimental data. However, the variety of both chemical systems and kinetic models that exist, as well as the mathematical difficulty in obtaining integrated rate laws for all but the simplest systems, makes the development of generalized kinetic modeling algorithms extremely difficult.

Systems in which natural mineral phases interact with aqueous fluids (and with each other) are of particular interest to environmental geoscientists. However, very few experimental techniques are capable of accurately resolving changes in the abundance of multiple solid phases in a fluid, especially when the chemical compositions of the solids are very similar. Recently, the development of high-flux synchrotron X-ray sources for time-resolved scattering techniques has enabled the *in situ* analysis of reacting mineral-fluid systems that were previously difficult or impossible to study in a time-resolved fashion.¹⁻¹² These techniques have allowed experimenters to quantify the amounts of multiple mineral phases during a single experiment with high time resolution, opening the door for kinetic analyses of previously unexplored geochemical systems. However, few tools exist to analyze the kinetics of such reactions. Here we present a versatile and user-friendly Matlab code, MinKin (for “Mineral Kinetics”), that can be used to fit a standard chemical kinetic model to available mineral or species abundance data for any geochemical system involving a fluid with up to two aqueous reactants co-existing with up to three mineral phases.

2 Description of MinKin

MinKin is a global optimization code in Matlab (version R2007a, MathWorks, Inc.) that uses the differential evolution (DE) algorithm of Price and Storn^{13,14} to minimize the sum of square errors between a kinetic model of a geochemical system and experimental mineral abundance data. To the authors' best knowledge, this is the first time that a global optimization algorithm such as DE has been coupled to the equations of a kinetic model in order to optimize rate constants. Differential evolution was originally developed to optimize solutions to the Chebyshev polynomial,^{13,14} but it does not require the optimization function to have any specific algebraic form, making it one of the most generally applicable optimization algorithms available. Thus, whereas most optimization algorithms would require a kinetic system to have an analytical solution (i.e., integrated rate equations) in order to optimize the rate constants, DE is able to evaluate output from a numerical ordinary differential equation (ODE) solver and compare it directly to the experimental data.

This flexibility, together with DE's balance between efficiency and accuracy, makes DE the ideal choice for solving generalized kinetic problems. MinKin's unique coupling of DE to a versatile, numerical ODE solver therefore makes it possible to calculate rate constants in a generalized kinetic system, even when that system's differential equations do not have an analytical solution. The current version of MinKin was written to allow users to define a geochemical system in which up to three mineral phases can react with up to two components of a co-existing fluid, and with each other, but future upgrades to the code are planned that will expand MinKin's capacity.

The user first identifies each mineral within the system, and then he or she either allows or disallows the possible phase transitions that could convert one mineral into another. (MinKin presumes that any precipitation or dissolution reaction is allowed, and automatically includes those rate constants in the optimization.) The program uses this input to define a set of differential equations that describes the system. Finally, the program uses an ODE solver to produce model output, compares the output to the data, and optimizes the model parameters in order to fit the model to the data. To illustrate, we present the derivation of a simple kinetic model for a generic system in which two mineral phases may precipitate and dissolve in an aqueous fluid, but do not undergo phase transitions.

3 Construction of the kinetic model

The example system can be thought of as consisting of three reservoirs of a metal M, each containing bound M atoms in various forms: a single aqueous species (consisting of hydrated M^{n+} species in solution), Mineral 1, and Mineral 2. These reservoirs will be designated M_{aq} , M_1 , and M_2 , respectively. As the system evolves, M is shuttled among these reservoirs via four different reactions:

In addition, MinKin makes the simplifying assumption that the concentration data it is given represent the activity of the component that is actually reacting. This assumption is well justified for the aqueous phase, since in general, all aqueous ionic species are available to react with the surrounding water and its constituents. However, for mineral phases, the relevant concentration is the surface area that actually reacts with the surrounding fluid, which may not change in the same manner as the mass or molar concentration and is much more difficult to measure. This makes it challenging to obtain a data set that is ideally suited for analysis with MinKin.

The issue of how best to describe reaction rates of solid phases as a function of surface area has been debated in the literature for many years, and many detailed studies and reviews have been written on the subject.¹⁶⁻²³ The issue of surface area is particularly problematic in describing dissolution reactions, since for these reactions the solid is the reactant and therefore the rate is dependent not only on the amount of surface available to react, but also on the nature of that surface. First, there is the question of whether reaction rates are proportional to geometric surface area, total surface area, or specific surface area, which each have different definitions.²² In addition, recent work suggests that there is a great deal of intrinsic variability of surface reactivity due to the presence of surface sites with differing energies.²³ Since each surface site has a different reactivity, and the distribution of surface sites evolves in a non-linear fashion that is difficult to predict, it may be difficult or impossible to describe dissolution rates using a single rate constant.

Given that this problem is an active area of research still under debate, in this work we do not present a comprehensive solution to the problem of surface area, but merely a new computational tool for extracting kinetic information from the experimental

data sets that are available. We must therefore leave it to users to consider recent literature on this subject¹⁹⁻²³ in order to construct data sets that best represent the reactivity of solid phases. In particular, the user should keep in mind that rate constants calculated with MinKin, especially dissolution rate constants, actually represent an integration of rates for a distribution of surface sites that are likely specific to the experimental material. We therefore urge users to interpret rate constants with some caution, taking into account the issues discussed here. In the present example of a model kinetic system, we assume that the concentrations provided in the equations represent the actual activity of the reactive surface.

With these simplifications, we can write the rates of each reaction as:

$$1) \text{ Rate}_1 = k_1 [M_{aq}]^A \quad (\text{Eqs. 3})$$

$$2) \text{ Rate}_2 = k_2 [M_1]^B$$

$$3) \text{ Rate}_3 = k_3 [M_{aq}]^C$$

$$4) \text{ Rate}_4 = k_4 [M_2]^D$$

where Rate_i is the rate of reaction i , k_i is the rate constant of reaction i , $[X]$ is the molar concentration of phase/species X , and A - D are the reaction orders of reactions 1-4, respectively (Eqs. 1).¹⁵ MinKin uses rate laws of the type described above in constructing models for any geochemical system.

The rates of the individual reactions can now be used to express the rate of change of M in each M reservoir, by summing the rates of each reaction that affects the reservoir:

$$M_{aq}: \frac{d[M_{aq}]}{dt} = -k_1[M_{aq}]^A + k_2[M_1]^B - k_3[M_{aq}]^C + k_4[M_2]^D \quad (\text{Eqs. 4})$$

$$M_1: \frac{d[M_1]}{dt} = k_1[M_{aq}]^A - k_2[M_1]^B$$

$$M_2: \frac{d[M_2]}{dt} = k_3[M_{aq}]^C - k_4[M_2]^D$$

For example, in the case of the “M_{aq}” reservoir, reactions 2 and 4 add M, while reactions 1 and 3 remove M. Mass balance in this model can be verified by noting that $\frac{d[M]_{total}}{dt} =$

$$\frac{d[M_{aq}]}{dt} + \frac{d[M_1]}{dt} + \frac{d[M_2]}{dt}.$$

The three ordinary differential equations (ODE) in Eqs. 4, together with the mass balance equation above, constitute a kinetic model that can be fit to experimental data by finding optimal values of the four model parameters using the four system equations. MinKin automatically constructs these equations using the input species and allowed reactions, and recognizes the model parameters as the rate constants of the four reactions (k_1 - k_4). In constructing the model, terms are added to the differential equations in pairs (one of which subtracts material from the reactant reservoir and the other of which adds material to the product reservoir), such that mass balance is always maintained regardless

of which reactions are allowed. The user can specify stoichiometric coefficients for each species in a reaction so that the terms of the differential equation are scaled properly. Because reaction orders are typically small whole numbers (0, 1, or 2), the user is asked to input the order of each reaction and test one 'reaction order model' at a time, rather than allowing reaction orders to be optimized from a continuum of possible values.

The user is also asked to place an upper bound on the values that rate constants may adopt. The rate constant maximum sets the size of the parameter space that will be searched, and will vary with the system and may require trial and error on the part of the user. The goal should be to set a maximum value somewhat greater than the value of the largest rate constant. A maximum that is set too large will cause slow convergence, since the program will search a much larger parameter space than is necessary. A maximum that is set too low will cause mis-convergence, since at least one of the true values is outside the bounds of parameter space. Since MinKin lets users include the reverse reaction for any given reaction in the system, negative rate constants are not meaningful, and the program uses a default minimum of zero for all parameter values.

4 Data input

The user is next allowed to input experimental data from a delimited text file that contains the abundances of one or more of the system's minerals as a function of time. It is not necessary to input data for all of the system's mineral phases, although the best results are obtained when experimental data for as many minerals as possible are included. The text file must have time data in the first column, and abundance data for

each available species in subsequent columns consistent with the previously entered order of minerals. The file should not contain column headers, since the program prompts the user to specify whether data for each mineral are included in the file. A sample input file is shown in Table 1. *(Example: Suppose the experimental system is defined to contain one aqueous component and minerals 1 and 2, as discussed above, entered in that order, but time series data are available only for mineral 2. The program allows the user to indicate that aqueous data are absent, mineral 1 data are absent, and mineral 2 data are present. Therefore, the program knows to interpret column 1 as time and column 2 as mineral 2 (rather than mineral 1)).*

Finally, the user inputs various run parameters, including initial concentrations of each aqueous species and mineral phase (including a conversion factor if units are different, as discussed in the Section 7 Example Calculation), the maximum number of iterations to perform, boundaries on parameter space, and a refresh interval that controls the frequency of program output during optimization. The maximum number of iterations to perform is of no consequence to the final results, since the program allows the user to perform additional iterations in the event that optimization is not achieved within the previous limit.

5 Optimization algorithm

MinKin feeds the user input into a code based on differential evolution (DE), an evolutionary optimization algorithm^{13,14} that fits the constructed kinetic model to the experimental data. Because a generalized kinetic model consists of ODEs rather than

direct equations for mineral/species abundance, many of the common steepest descent techniques for objective function optimization (such as the Levenberg-Marquardt algorithm) cannot be directly applied to kinetic problems. MinKin circumvents this difficulty by making use of a set of nested functions in which output from a numerical ODE solver is passed into an error function, whose output is then passed into the DE algorithm, which evolves populations of points in parameter space towards the minima of the error function. As described in section 2, the coupling of DE to a numerical ODE solver is a unique and novel mathematical approach capable of solving kinetic problems that are intractable by other methods. Figure 2 shows a simplified flow chart of MinKin's operation.

In DE, an initially random population of points within the defined bounds of parameter space is generated. In our example kinetic model, each of these points would consist of a vector of rate constants for each of the four reactions, $\langle k_1, k_2, k_3, k_4 \rangle$. At each iteration, the program uses these values to numerically solve the model's differential equations with Matlab's most versatile ODE solver for potentially stiff equations, ode15s. The output from the ODE solver for each parameter vector is then passed to an error function, which computes the sum of square errors between each model output and the experimental data using the following equation:

$$SSE = \sum_i \left(\sum_j (O_{i,j} - f_j(t_i))^2 \right) \quad (\text{Eq. 5})$$

where the inner sum is over all species included in the data set, and the outer sum is over all time steps. $O_{i,j}$ is the i th observation of species j in the data set, $f_j(t)$ is the concentration of species j predicted by the model at time t , and t_i is the time of the i th point in the data set. For the example mineral system with metal M, Eq. 5 would become:

$$\text{SSE} = \sum_i \left((M_{1i} - f_{M1}(t_i))^2 + (M_{2i} - f_{M2}(t_i))^2 \right) \quad (\text{Eq. 6})$$

where M_{1i} is the observed concentration of Mineral 1 at data point i , M_{2i} is the observed concentration of Mineral 2 at point i , $f_{M1}(t)$ is the model prediction for Mineral 1 at time t , $f_{M2}(t)$ is the model prediction for Mineral 2 at time t , and t_i is the measurement time of the i th point in the data set. The sum is taken over all observations in the data set. The SSE function automatically weights each observation equally, and consequently each data point is assigned equal importance in optimizing the parameter values.

During each iteration, the SSE value corresponding to each of the population's parameter vectors is computed. The population member yielding the lowest SSE value is stored, and it is also output to the screen periodically at a regular iteration interval set by the user (called the “refresh interval”). The DE algorithm then “evolves” the population by taking the differences between random pairs of population vectors, and adding them to a chosen base vector (for details of the DE algorithm, see Ref. 13). The net result is that a new population of points is generated, and these points progressively cluster in regions of parameter space that produce small SSE values. Once the new population has been

produced, the next iteration begins and the program again passes the population's parameter vectors to the ODE solver, the ODE output to the SSE function, and the SSE values to the evolutionary algorithm.

MinKin has been designed to update the user on its progress during optimization, so that convergence of parameter values can be viewed and potential problems can be identified even before the calculation is complete. MinKin produces four output graphs, two during optimization and two after optimization. The graphs produced are illustrated in Figure 3, and described below:

1. *Error minimization.* (Fig. 3A) After each refresh interval, the program updates a graph that displays the sum of square errors value of the best population member on the y-axis versus that population's iteration number on the x-axis. This allows the user to track the progress of the goodness of fit during optimization.

2. *Parameter convergence.* (Fig. 3B) After each refresh interval, the program updates a graph showing the parameter values of the best population member on the y-axis versus that population's iteration number on the x-axis. Each parameter is plotted with its own color-coded curve, and a key within the graph indicates the name of the parameter associated with each curve. This allows the user to graphically view the convergence of parameter values during optimization.

3. *Model fit.* (Fig. 3C) After optimal parameter values have been calculated, MinKin evaluates the model's differential equations at the best parameter values, numerically solves them and plots the predicted mineral abundances of each mineral in the system on the y-axis versus time on the x-axis. The model prediction for each mineral

is plotted as a smooth curve. In the same graph, the experimental abundance versus time data for each mineral is plotted using “+” symbols. For ease of comparison, the color used to display the data points for a given mineral is the same as that used to plot the model prediction curve for that mineral. A key within the graph labels the observations and model predictions for each mineral. This allows users to qualitatively evaluate the goodness of fit achieved at the end of optimization.

4. *Parameter contours.* (Fig. 3D) After optimal parameter values have been calculated, the program plots the sum of square error on the y-axis versus the value of each parameter on the x-axis, holding all other parameters at their optimal values. A separate, color-coded curve is used for each parameter, and a key within the graph labels the parameter corresponding to each curve. For example, for the three mineral system plotted in Figure 3D, there are six reactions and therefore six model parameters (equal to the rate constants of each reaction). Suppose the optimal parameter vector calculated by MinKin is $p_{opt} = \langle k_{1opt} \ k_{2opt} \ k_{3opt} \ k_{4opt} \ k_{5opt} \ k_{6opt} \rangle$. The curve associated with the first parameter (k_1 = the Mineral A precipitation constant) in the parameter contours graph would plot the SSE associated with model $\langle k_1 \ k_{2opt} \ k_{3opt} \ k_{4opt} \ k_{5opt} \ k_{6opt} \rangle$ as a function of k_1 (dark blue curve in Fig. 3D). This curve should show a global minimum SSE at a value of $k_1 = k_{1opt}$. Similarly, the curve associated with parameter k_2 plots the SSE of model $\langle k_{1opt} \ k_2 \ k_{3opt} \ k_{4opt} \ k_{5opt} \ k_{6opt} \rangle$ as a function of k_2 , and so on. This feature allows users to graphically examine the minimum they have reached, and identify potential false minima. Examples of each type of graph are shown in Figure 3.

Upon convergence, MinKin explicitly prints the following to the Matlab control screen: 1) optimal values of all model parameters to 6 decimal places, 2) the final sum of

square errors to 6 decimal places, 3) the computation time, 4) the χ^2 (chi-squared) value of the fit, and 5) the left-tailed p-value associated with the χ^2 value for N-P degrees of freedom, where N = the total number of data points in the data set, and P = the number of optimized parameters. All of this information is also saved into an output file with the same name as the experimental data file, but with a ".out" extension.

MinKin continues performing iterations until either 1) a convergence criterion is reached in which all parameter values are within 0.5% of their mean values during the last five refresh intervals AND those values have not remained perfectly static for the most recent two refresh intervals, or 2) the maximum number of specified iterations has been performed. The reason for requiring the parameter values to change by a nonzero amount is that perfectly static parameter values are a symptom of entrapment in a false minimum (since the program is continuing to evolve populations, but no members of those populations have bested the previous best population member, as would be expected if the population were actually converging on the global minimum). As demonstrated by the example calculations, this feature provides powerful protection against converging on a false minimum.

If the maximum number of iterations is performed without reaching convergence, the user is given the option of continuing optimization starting with the most recent population of parameter vectors. This allows the user to avoid wasting computation time in the event that optimization is not achieved in the first run cycle. The user is then prompted for the number of additional iterations to perform, and optimization continues from where it left off. This process can be continued an arbitrary number of times until convergence is achieved. The total number of iterations needed to reach convergence can

be anywhere from 100 to several thousand depending on the number of parameters and complexity of the problem, and computation times range from seconds to hours.

Graphical output from an example optimization using a simulated data set illustrates some characteristic behaviors of MinKin in the course of the refinements (Figures 3). (This optimization was set up in a similar way to the resolution tests described below, except that data points were not perturbed by random error). Note that in Figure 3B, parameter values occasionally remain perfectly static during optimization for periods of up to 100 iterations. This is symptomatic that the population of parameter vectors is temporarily occupying a false minimum, and is a normal feature in most computations. However, because the convergence criteria require parameter values to shift slightly before convergence is declared, MinKin prevents premature convergence into such minima.

In contrast, note that after iteration 400 (Figure 3B), once parameter values are close to their correct values, very minor (but nonzero) adjustments occur before convergence is declared. In Figure 3D, the surface of the SSE function (as demonstrated by its cross sections along each of the parameter axes, plotted in different colors) forms a roughly symmetric distribution about the global minimum, and no signs of local minima are observed (which would appear as additional wells). These observations instill confidence that the correct minimum was reached. In contrast, the observation of a second (local) minimum in one of the parameter contours, a minimum that is at the very edge of parameter space, or an asymmetric SSE surface in which the SSE value is less sensitive to one parameter than to others, would indicate that a false minimum may have been reached. For example, Figure 3E shows the parameter contours for a calculation that

mis-converged. Note that the sensitivities to each parameter are substantially different, and one parameter (the Mineral B dissolution constant, in light blue) has a minimum at the lower edge of parameter space.

6 Accuracy tests using synthesized data sets

To evaluate MinKin's effectiveness with different types of systems, a variety of test runs were performed in which simulated mineral abundance data sets were generated by choosing a predetermined kinetic model with predetermined parameter values. Each simulated data set involved a system of 1, 2, or 3 mineral phases that precipitate from an initially homogenous aqueous solution of 0.5 mol/L, so that $[M_{aq}]_0 = 0.5$, and $[A]_0 = [B]_0 = [C]_0 = 0$. Reactions that convert one mineral into another were suppressed in these simulations, and parameter values ranged from 0.1-1.0 hr^{-1} . In all tests, each reaction was given a 1:1 stoichiometry, and a reaction order of 1 was used for each rate law (as in Eq 3 and 4). Although the maximum rate constant used to construct the datasets was 1.0 hr^{-1} , parameter space boundaries of $0 \leq k \leq 10 \text{ hr}^{-1}$ were used for all k during optimization. (In other words, a somewhat larger parameter space was searched than was actually necessary, as would be the case in a typical optimization since the correct values are not normally known in advance.) The tests used the following procedure:

- 1) A system involving the homogeneous precipitation of 1, 2, or 3 minerals was simulated by choosing rate constants for each reaction (i.e., precipitation and dissolution

of each mineral), and forward-calculating the resulting mineral concentrations with time using the numerical ODE solver.

2) Individual data points in these synthesized data sets were perturbed by a normally distributed random error, with a standard deviation of 1%, 5%, or 10% of the original value. These perturbations of the data points were performed to simulate increasing degrees of random experimental error.

3) The resulting perturbed data sets were fed into MinKin to observe its effectiveness in extracting the known rate constants.

For each combination of mineral system (1, 2, and 3 minerals) and relative error of the datasets (1%, 5%, 10%), these three steps were repeated in triplicate, for a total of 27 resolution tests. No correlation was observed between the number of minerals in the system and the accuracy of refined rate constants (i.e., MinKin's accuracy is not affected by the complexity of the problem). However, as expected, the amount of experimental error had a marked impact on MinKin's accuracy.

When the *average accuracy* of the refined rate constants was tabulated for all the tests using the same experimental error, the average relative error of the rate constants was less than 1% for each experimental error, indicating that after many optimizations MinKin's results are centered on the true values. However, the *precision* of the calculated rate constants (i.e., the width of the distribution of relative errors) became progressively worse as the experimental error increased from 1% to 10%. Three histograms of the relative errors of optimized rate constants, one for each experimental error, are shown in

Figure 4A, while Figure 4B shows the 1σ uncertainties for optimized rate constants calculated from these histograms. Table 2 illustrates typical results (for one repetition) of the resolution tests. The results demonstrate that users can expect errors in optimized rate constants that are on par with the relative errors present in the input data sets. Output graphs for the two-mineral system at each experimental error demonstrate that MinKin produces similar model predictions even when the noise of experimental data is increased (Figure 5). In summary, the uncertainties of calculated rate constants in an individual optimization depend primarily on the uncertainties of the experimental data. However, even when experimental data points have high uncertainties, calculated rate constants will still cluster around the true values after many independent optimizations.

Because experimental data with very low error can potentially produce rate constants with very low errors, MinKin provides the optimized rate constants to six decimal places so that users who do highly precise determinations do not lose information. However, users should always keep in mind that the accuracy of the experimental data limits the accuracy of the refined rate constants in any optimization. In order to estimate the error bars on optimized rate constants, we therefore recommend either 1) replicating experiments and performing identical MinKin optimizations using data from each experimental replication (in cases for which replications can be easily done), or 2) using knowledge of the experimental error in conjunction with the results on synthesized data sets plotted in Figure 4 as a guide in estimating errors.

.

7 Example Calculation: Obtaining a time series of mineral abundances from real X-ray diffraction data

Before MinKin can be utilized to calculate rate constants, suitable data sets of mineral abundance versus time must be obtained for as many minerals as possible in the experimental system. Diffraction data may originate from a single, time-resolved experiment (as in synchrotron TR-XRD), or from a series of batch experiments run under identical conditions. To extract mineral abundances from these data, a variety of crystallographic software packages can be used to calculate integrated intensities of appropriate diffraction peaks, or Rietveld refinement software can be used to refine mineral abundance values from whole patterns. However it is achieved, the data set must have values of mineral abundance that are comparable to each other, both for different minerals in the same time step, and for the same mineral across different time steps.

Note that these abundance values must reflect the *mass* (or volume) of each phase for a given time step rather than simply the relative abundance, as is reflected in the weight percentage of each phase in the total mineral assemblage. For example, if the crystallization of Mineral A precedes that of Mineral B, then Mineral A will constitute an unchanging 100 wt% of the solid product even as its quantity increases during precipitation. Fortunately, integrated intensities of individual diffraction peaks are directly correlated with mineral mass and can be used as a proxy for mineral production. In the following example calculation, values of mineral abundance for the TiO₂ polymorphs anatase and rutile are obtained by the Rietveld analysis of a time-resolved series of synchrotron XRD patterns.¹¹

Figure 6 shows stacked XRD patterns for an experiment in which anatase and rutile are crystallized at 100 °C from an aqueous solution of 0.5 M TiCl₄ at beamline X7B of the National Synchrotron Light Source, Brookhaven National Labs.¹¹ These patterns show an initial increase in the abundance of both anatase and rutile during the first two hours of the experiment, and then a subsequent increase in rutile abundance with concurrent loss of anatase. To obtain values of mineral abundance, the General Structure Analysis System (GSAS) of Larson and von Dreele²⁴ was used to model selected diffraction patterns and refine scale factors that vary linearly with the mass of mineral in solution. These values are shown in columns 2 and 3 of Table 3.

However, because the total X-ray intensity at beamline X7B degrades over time, peak intensities (and hence refined scale factors) from different XRD patterns are *not* comparable across all time steps until they are normalized by the incoming beam flux. The variation in peak intensity due to diminishing beam flux can be removed from the data by normalizing to a background intensity point at $2\theta = 20^\circ$, which contributes no Bragg scattering from either of the two mineral phases (shown in column 4 of Table 3). The resulting normalized scale factors (columns 5 and 6 of Table 3, graphed in Figure 7) are now uniform across time as well as phase, and are therefore suitable for input into MinKin. Note that for data sets in which beam intensity is constant throughout an experiment, such a normalization would be unnecessary.

Next, columns 1, 5, and 6 of Table 3 would be used to create a delimited text file that can be input into MinKin (in the same format as Table 1). The other inputs for this optimization (i.e., run parameters) are shown in Table 4, as are the refined rate constants at the end of optimization for each of the four reactions occurring in the system:

474

479

480 The sum of square errors asymptotically declined during optimization; all
481 parameters simultaneously converged to a single value within the bounds of parameter
482 space; the refined model is a good fit to the data; and parameter contours form a steep, V-
483 shaped well (Figure 8). These are all indications of a successful optimization. Using the
484 refined rate constants calculated by MinKin, the best fit kinetic model to this data would
485 be described by the following equations:

486

$$\frac{d[\text{Anatase}]}{dt} = 2.073 \text{ hr}^{-1}[\text{TiCl}_4] - 0.243 \text{ hr}^{-1}[\text{Anatase}]$$

$$\frac{d[\text{Rutile}]}{dt} = 1.626 \text{ hr}^{-1}[\text{TiCl}_4] - 0.102 \text{ hr}^{-1}[\text{Rutile}]$$

487

488 The first term in each equation represents the rate of precipitation of the mineral in
489 question, and the second term represents the dissolution rate. It should be remembered

that since rate constants are temperature dependent, the refined rate constants in this example are only valid at 100 °C, the temperature at which these data were collected. Further analysis of the kinetics of this particular system will be detailed in a subsequent publication. The individual experiment and optimization discussed here are provided only to illustrate how to apply MinKin to a real geochemical system.

MinKin does not require that mineral abundances are expressed in any particular units, as long as the units are consistent for all minerals and time steps. Likewise, any unit of time may be used as long as it is consistent throughout the data set. The concentrations of aqueous species can also be provided in any units, but if these are different than the units of mineral concentration, then a conversion factor relating these two units must also be provided. This is because MinKin mathematically converts between aqueous material and mineral material during the course of a refinement using the reactions of the kinetic model. Thus, if a relationship between aqueous quantities and mineral quantities is not specified, MinKin cannot model reactions properly and the results of a refinement are not meaningful.

It is worth noting that because MinKin uses rate laws of the form $d[R]/dt = k[R]^A$, the optimized rate constant k for a given reaction will be in units of $conc^{1-A} \cdot time^{-1}$, where R is a chemical species, $conc$ is the unit of concentration for that species, $time$ is the unit of time, and A is the reaction order. Note that in the relatively common case of first-order reactions ($A=1$), calculated k values will be in units of reciprocal time, and completely independent of the units used to express mineral abundance.

8 System Requirements and Future work

MinKin can be operated on a platform of Matlab R2007a or higher by placing the original Matlab files within Matlab's workspace, or as a standalone executable in combination with any version of Matlab Compiler Runtime (MCR), which is available free from the MathWorks® website.²⁵ Experimental data must be placed in a space, comma or tab delimited text file containing no column headers, with time data in the first column and species abundance data in subsequent columns. Output files are placed in the same folder as the input data files. Interested parties may write to the author to obtain a free copy of the software. Researchers are asked to cite this publication in any paper presenting data, kinetic models, or other results obtained using MinKin.

Future developments for MinKin are planned that will expand its capacity for solving kinetic problems. First, we will include additional choices for rate laws that are potentially useful in complex systems, such as the Avrami (or JMAK) equation, the Michaelis-Menten equation, etc. Second, we plan to update the programming of the kinetic model so that more than two fluid components and more than three mineral components can be part of the kinetic system. Since the complexity of the differential equations describing the system grows rapidly with each added component, this upgrade will take significant time but will make MinKin even more powerful in analyzing the complex, multi-component systems found in nature.

9 Acknowledgments

This work was made possible by National Science Foundation Grant No. EAR07-45374 and EAR11-47728, and by the Center for Environmental Kinetics Analysis (CEKA), an NSF- and DOE-sponsored Environmental Molecular Science Institute (NSF CHE-0431328). We would also like to thank the following individuals for valuable assistance and advice on kinetic modeling and the development of the MinKin code: Joshua Dorin, Klaus Keller, Joel Bandstra, Rudy Slingerland, Frank Hummer, Paul Winberry, Nick Chernyy, and Eliza Richardson.

10 References

- [1] Millange, Franck; Walton, Richard I.; O'Hare, Dermot. Time-resolved in situ X-ray diffraction study of the liquid-phase reconstruction of Mg-Al-carbonate hydrotalcite-like compounds. *J. Mater. Chem.* **2000**, *10*, 1713-1720.
- [2] Meller, Nicola; Hall, Christopher; Jupe, Andrew C.; Colston, Sally L.; Jacques, Simon D. M.; Barnes, Paul; Phipps, Jonathan. The paste hydration of brownmillerite with and without gypsum: a time-resolved synchrotron diffraction study at 30, 70, 100, and 150 °C. *J. Mater. Chem.* **2004**, *14*, 428-435.
- [3] Coker, Victoria S.; Bell, Anthony M. T.; Pearce, Carolyn L.; Pattrick, Richard A. D.; van der Laan, Gerrit; Lloyd, Jonathan R. Time-resolved synchrotron powder X-ray diffraction study of magnetite formation by the Fe(III)-reducing bacterium *Geobacter sulfareducens*. *Am. Min.* **2008**, *93*, 540-547.

- 553 [4] Heaney, Peter J.; Post, Jeffrey E.; Fischer, Timothy B; Hummer, Daniel R.; Lopano,
554 Christina L.; Wall, Andrew J. Applications of time-resolved synchrotron X-ray
555 diffraction to cation exchange, crystal growth and biomineralization reactions.
556 *Mineralogical Magazine* **2008**, 72, 179-184.
- 557 [5] Perrillat, J. P. Kinetics of high-pressure mineral phase transformations using in situ
558 time-resolved X-ray diffraction in the Paris-Edinburgh cell: a practical guide for
559 data acquisition and treatment. *Mineralogical Magazine* **2008**, 72, 683-695.
- 560 [6] Fischer, Timothy B.; Heaney, Peter J.; Jang, Je-Hun; Ross, Daniel E.; Brantley, Susan
561 L.; Post, Jeffrey E.; Tien, Ming. Continuous time-resolved X-ray diffraction of the
562 biocatalyzed reduction of Mn oxide. *Am. Min.* **2008**, 93, 1929-1932.
- 563 [7] Lopano, Christina L.; Heaney, Peter J.; Post, Jeffrey E. Cs-exchange in birnessite:
564 Reaction mechanisms inferred from time-resolved X-ray diffraction and
565 transmission electron microscopy. *Am. Min.* **2009**, 94, 816-826.
- 566 [8] Lopano, Christina L.; Heaney, Peter J.; Bandstra, Joel Z.; Post, Jeffrey E.; Brantley,
567 Susan L. Kinetic analysis of cation exchange in birnessite using time-resolved
568 synchrotron X-ray diffraction. *Geochim. Cosmochim. Acta* **2011**, 75, 3973-3981.
- 569 [9] Wall, Andrew J.; Heaney, Peter J.; Mathur, Ryan; Post, Jeffrey E.; Hanson, Jonathan
570 C.; Eng, Peter J. A flow-through reaction cell that couples time-resolved X-ray
571 diffraction with stable isotope analysis. *J. Appl. Crystall.* **2011**, 44, 429-432.

- 572 [10] Wall, Andrew J.; Mathur, Ryan; Post, Jeffrey E.; Heaney, Peter J. Cu isotope
573 fractionation during bornite dissolution: An in situ X-ray diffraction analysis. *Ore*
574 *Geology Reviews* **2011**, *42*, 62-70.
- 575 [11] Hummer, Daniel R.; Kubicki, James D.; Kent, Paul R. C.; Post, Jeffrey E.; Heaney,
576 Peter J. Origin of nanoscale phase stability reversals in titanium oxide
577 polymorphs. *J. Phys. Chem. C* **2009**, *113*, 4240-4245.
- 578 [12] Hummer, Daniel R.; Heaney, Peter J.; Post, Jeffrey E. In situ observations of particle
579 size evolution during the hydrothermal crystallization of TiO₂: A time-resolved
580 synchrotron SAXS and WAXS study. *J. Cryst. Growth* **2012**, *344*, 51-58.
- 581 [13] Storn, Rainer; Price, Kenneth. Differential evolution – A simple and efficient
582 heuristic for global optimization over continuous spaces. *J. Global Optimization*
583 **1997**, *11*, 341-359.
- 584 [14] Price, Kenneth; Storn, Rainer. Differential Evolution. *Dr. Dobbs Journal* **1997**,
585 *22*(4).
- 586 [15] Skinner, Gordon B. **1974** *Introduction to Chemical Kinetics*. Academic Press, New
587 York, NY, Ch. 2.
- 588 [16] Schott, J.; Brantley, Susan L.; Crerar, D.; Guy, C.; Borcsik, M.; Williame, C. **1989**
589 Dissolution kinetics of strained calcite. *Geochim. Cosmochim. Acta* *53*, 373-382.
- 590 [17] White, Art F.; Brantley, Susan L. **1995** Chemical weathering rates of silicate
591 minerals: an overview. *Chem. Weath. Rates Silicate Miner.* *31*, 1-22.

- 592 [18] Brantley, Susan I.; White, A.F.; Hodson, M.E. **1999** Surface Area of Primary
593 Silicate Minerals. In: Jamtveit, B; Meakin, P. (Eds.) Growth, dissolution and
594 pattern formation on geosystems. Dordrecht, Kluwer. p. 291-326.
- 595 [19] Luttge, Andreas; Arvidson, R.L. **2008** The mineral-water interface. In: Brantley,
596 Susan L.; Kubicki, James D.; White, Art F. (Eds.) *Kinetics of Water-Rock*
597 *Interaction*. Springer, p. 73-107.
- 598 [20] Putnis, C.V.; Ruiz-Agudo, E. **2013** The mineral-water interface: where minerals
599 react with the environment. *Elements* 9, 177-182.
- 600 [21] Schott, J.; Pokrovsky, O.S.; Oelkers, Eric H. **2009** The link between mineral
601 dissolution/precipitation kinetics and solution chemistry. *Rev. Mineral. Geochem.*
602 70, 207-258.
- 603 [22] Fischer, Cornelius; Kurganskaya, Inna; Schafer, Thorsten; Luttge, Andreas. **2014**
604 Variability of crystal surface reactivity: what do we know? *Appl. Geochem.* 43,
605 132-157.
- 606 [23] Fischer, Cornelius; Arvidson, R.S.; Luttge, Andreas. **2012** How predictable are
607 dissolution rates of crystalline material? *Geochim. Cosmochim. Acta* 98, 177-185.
- 608 [24] Larson, Alan C.; Von Dreele, Robert B. General Structure Analysis System (GSAS)
609 **2000**, (Report LAUR 86-748). Los Alamos, New Mexico: Los Alamos National
610 Laboratory.
- 611 [25] *MATLAB Runtime*. [Computer Software]. *MATLAB Runtime - MATLAB Compiler*.
612 MathWorks, Inc., n.d. Web. <http://www.mathworks.com/products/compiler/mcr/>.

613 **Table 1:** Sample input file of mineral abundance data.

Time (hrs)*	Mineral A	Mineral B	Mineral C
0.0000	0.0000	0.0000	0.0000
0.5000	0.1247	0.0836	0.0949
1.0000	0.1454	0.1095	0.1205
1.5000	0.1440	0.1205	0.1289
2.0000	0.1391	0.1268	0.1325
2.5000	0.1347	0.1311	0.1343
3.0000	0.1312	0.1344	0.1354
3.5000	0.1287	0.1369	0.1360
4.0000	0.1268	0.1388	0.1363
4.5000	0.1254	0.1402	0.1365
5.0000	0.1244	0.1414	0.1366
5.5000	0.1236	0.1423	0.1366
6.0000	0.1231	0.1430	0.1365
6.5000	0.1226	0.1435	0.1365
7.0000	0.1223	0.1440	0.1364
7.5000	0.1221	0.1443	0.1364
8.0000	0.1219	0.1446	0.1363
8.5000	0.1218	0.1448	0.1363
9.0000	0.1217	0.1450	0.1362
9.5000	0.1216	0.1451	0.1362
10.0000	0.1215	0.1452	0.1361

614

615 * In the actual text file column headers would not be used, but are provided here for clarity.

616

617 **Table 2:** Sample table of MinKin accuracy tests. All the tests represented in this table
618 were repeated in triplicate, and all reactions were modeled using first order rate laws.

Relative error in mineral datasets	No. of minerals in system	Reaction	Actual rate constant (hr ⁻¹)	Calculated rate constant (hr ⁻¹)	Relative error (%)
1%	1	Mineral A precipitation	0.700	0.706	0.911
		Mineral A dissolution	0.100	0.104	4.061
	2	Mineral A precipitation	0.500	0.497	-0.534
		Mineral A dissolution	0.300	0.302	0.538
		Mineral B precipitation	0.600	0.596	-0.670
		Mineral B dissolution	0.400	0.398	-0.408
	3	Mineral A precipitation	1.000	1.041	4.091
		Mineral A dissolution	0.800	0.834	4.288
		Mineral B precipitation	0.600	0.604	0.721
		Mineral B dissolution	0.400	0.405	1.137
		Mineral C precipitation	0.700	0.700	-0.055
		Mineral C dissolution	0.500	0.502	0.423
5%	1	Mineral A precipitation	0.700	0.637	-9.009
		Mineral A dissolution	0.100	0.083	-16.630
	2	Mineral A precipitation	0.500	0.502	0.448
		Mineral A dissolution	0.300	0.300	-0.150
		Mineral B precipitation	0.600	0.610	1.673
		Mineral B dissolution	0.400	0.399	-0.361
	3	Mineral A precipitation	1.000	0.971	-2.913
		Mineral A dissolution	0.800	0.758	-5.268
		Mineral B precipitation	0.600	0.591	-1.532
		Mineral B dissolution	0.400	0.382	-4.606
		Mineral C precipitation	0.700	0.716	2.344
		Mineral C dissolution	0.500	0.504	0.777
10%	1	Mineral A precipitation	0.700	0.702	0.282
		Mineral A dissolution	0.100	0.112	11.880
	2	Mineral A precipitation	0.500	0.528	5.679
		Mineral A dissolution	0.300	0.288	-3.868
		Mineral B precipitation	0.600	0.606	0.964
		Mineral B dissolution	0.400	0.327	-18.141
	3	Mineral A precipitation	1.000	0.783	-21.695
		Mineral A dissolution	0.800	0.600	-25.021
		Mineral B precipitation	0.600	0.502	-16.282

Mineral B dissolution	0.400	0.306	-23.497
Mineral C precipitation	0.700	0.638	-8.801
Mineral C dissolution	0.500	0.443	-11.301

619

620

Table 3: Rietveld refined scale factors representing the abundances of anatase and rutile during a time-resolved synchrotron XRD experiment. In this experiment, anatase and rutile precipitate from a Ti^{4+} -bearing aqueous fluid.

Time (hrs)	GSAS refined scale factors			Normalized Scale factors	
	Anatase	Rutile	Intensity at $2\theta = 20^\circ$	Anatase (au)	Rutile (au)
0.326	138.31	101.96	21234	0.006514	0.004802
0.392	137.36	105.21	21077	0.006517	0.004992
0.457	134.33	101.81	20876	0.006435	0.004877
0.522	134.46	111.44	20903	0.006433	0.005331
0.588	135.44	114.65	20632	0.006565	0.005557
0.718	135.18	120.60	20313	0.006655	0.005937
1.044	135.62	119.98	19962	0.006794	0.006010
1.371	134.74	116.80	19694	0.006842	0.005931
1.697	136.54	121.83	19658	0.006946	0.006198
1.762	119.51	129.94	19604	0.006096	0.006628
1.828	137.60	123.88	19536	0.007043	0.006341
1.893	136.35	123.12	19549	0.006975	0.006298
1.958	113.85	147.40	19462	0.005850	0.007574
2.024	128.58	128.12	19203	0.006696	0.006672
2.154	120.95	137.67	18968	0.006377	0.007258
2.350	128.34	131.14	18797	0.006828	0.006977
2.481	126.06	122.31	18577	0.006786	0.006584
2.676	120.51	133.54	18416	0.006544	0.007251
2.872	118.69	129.62	18030	0.006583	0.007189
3.133	111.24	136.35	17828	0.006240	0.007648
3.329	116.26	125.92	17521	0.006635	0.007187
3.525	113.77	123.07	17260	0.006592	0.007130
3.786	110.45	123.09	17172	0.006432	0.007168
3.982	104.40	125.37	16809	0.006211	0.007458
4.308	104.96	121.29	16356	0.006417	0.007416
4.635	107.65	119.63	15950	0.006749	0.007500
4.961	94.69	118.85	15177	0.006239	0.007831
5.614	86.47	116.97	14402	0.006004	0.008122
6.593	79.99	110.80	13434	0.005954	0.008247
7.572	135.78	187.08	23024	0.005897	0.008126
8.225	126.24	199.73	22541	0.005600	0.008861
9.857	130.18	189.31	22458	0.005796	0.008429
10.314	138.31	101.96	21234	0.006514	0.008402

Table 4: Refinement conditions and results for an example optimization: The crystallization of anatase and rutile from 0.5 M TiCl₄ at 100 °C. Graphical output from this optimization is shown in Figure 7.

Parameter	Value
Initial conditions	
[TiCl ₄] (mol/L)	0.5
[Anatase] (au)	0
[Rutile] (au)	0
Conversion factor (au•L/mol)*	0.028
Run parameters	
Maximum iterations	1000
Refresh interval	10
Rate constant maximum	10 hr ⁻¹
Solid state reactions	None
Reaction orders	
Anatase precipitation	1
Anatase dissolution	1
Rutile precipitation	1
Rutile dissolution	1
Reaction stoichiometries	
Anatase precipitation	1:1
Anatase dissolution	1:1
Rutile precipitation	1:1
Rutile dissolution	1:1
Refined rate constants	
Anatase precipitation	2.073 hr ⁻¹
Anatase dissolution	0.243 hr ⁻¹
Rutile precipitation	1.626 hr ⁻¹
Rutile dissolution	0.102 hr ⁻¹

*MinKin mathematically converts initial aqueous reactant into mineral products during the course of a refinement. Because reactants and products are in different units, the user *must* supply a conversion factor by which amounts of reactants can be related to amounts of products, otherwise there is no way to simulate reactions in the system. In this example, since the solubility of TiO₂ is very low and all aqueous Ti converts to either anatase or rutile by the end of the experiment, the total normalized scale factors for anatase and rutile (sum of columns 5 and 6, Table 2) at the end of the experiment (~0.014 au) must equate to the initial amount of aqueous Ti (0.5 mol/L). This gives a conversion factor of 0.014 au/0.5 mol/L = 0.028 au•L/mol.

Figure captions

Figure 1: A diagram of the kinetic model used to describe an example geochemical system in which two minerals bearing metal M precipitate from an aqueous M^{n+} -bearing fluid. Arrows represent the four reactions of the model, which shuttle M among three different phases (or reservoirs): Aqueous Solution, Mineral 1, and Mineral 2. For simplicity, direct conversions between Mineral 1 and Mineral 2 are not included in this model.

Figure 2: A simplified schematic flow chart of MinKin. The program accepts as input a user-defined geochemical system and experimental mineral abundance data, defines a standard chemical kinetic model for the system, and uses an iterative optimization algorithm to fit the model parameters to the data. Red boxes show program input, green boxes show program output, and yellow boxes show subroutines that are executed during operation. Blue arrows show data that are passed from one subroutine to the next during each iteration. “DE” stands for the differential evolution algorithm of Price and Storn.^{13,14}

Figure 3: MinKin output graphs for the kinetic analysis of a 3-mineral system with 6 model parameters (data from Table 1). A) The sum of square errors of the best population member vs. iteration number. B) The best parameter values vs. iteration number. Note that at some points during optimization, parameter values remain static, indicating that the population of parameter vectors lies temporarily in a false minimum. C) Comparison of the actual mineral abundance data with the mineral abundances predicted by the best fit model. D) The sum of square errors vs. each parameter value, holding all other parameters at their optimal values. Note that the SSE surface forms a roughly symmetric

well about the global minimum with no signs of local minima. (In each graph, mineral concentration is in units of mol/L, and parameter values are in units of hr^{-1}).

Figure 4: The results of MinKin accuracy tests using synthetic data sets. A) Histograms show how often different relative errors in calculated rate constants occurred for each error in the mineral concentration data. B) Bar chart of the relative error in calculated rate constant values (1σ error for $N=36$ calculations) vs. the error in the mineral concentration data.

Figure 5: Graphical output from three accuracy tests using synthetic data sets perturbed by A) 1% error, B) 5% error, and C) 10% error, in which two minerals precipitated from an initially homogeneous solution. Crosses show the data points given to MinKin, and lines show the predicted mineral concentrations calculated by MinKin using the best fit model.

Figure 6: Stacked diffraction patterns of TiO_2 crystallization from a 0.5 M TiCl_4 solution at 100 °C. Pink peaks (labeled “A”) represent anatase, blue peaks (labeled “R”) represent rutile. Rietveld refinements of selected diffraction patterns yield time-resolved mineral abundance data for anatase and rutile. (After Hummer et al.¹¹)

Figure 7: Mineral scale factors vs. time during TiO_2 crystallization from a 0.5 M TiCl_4 solution at 100 °C (from columns 5 and 6 of Table 2). Scale factors are refined using Rietveld analysis with the General Structure Analysis System (GSAS) (columns 2 and 3 of Table 2), and are then normalized to the intensity at $2\theta = 20^\circ\text{C}$ (column 4 of Table 2) in order to remove the signal caused by drifting beam intensity. These are the data that

683 would be input into MinKin for kinetic analysis. Pink = anatase, Blue = rutile, Brown =
684 total. (After Hummer et al.¹¹)

685 **Figure 8:** MinKin output graphs for the refinement of rate constants involved in the
686 crystallization of anatase and rutile. A) The sum of squares errors during optimization, B)
687 Convergence of rate constants during optimization, C) Comparison between the best fit
688 model and experimental data, D) Sum of squares errors as a function of each parameter
689 value.

690 **Figure 1**

691

692

693 **Figure 2**

694

695

696 **Figure 3**

698

Model fit (with the best parameter values)

699

Parameter contours

700

701

702

703 **Figure 4**

704

705

Figure 5

712

713

714
715 **Figure 6**

716

717

718 **Figure 7**

719

720

721 **Figure 8**

724

725