Published in final edited form as:

Anal Biochem. 2007 February 15; 361(2): 153–161.

Kinetic Modeling and Fitting Software for Inter-connected Reaction Schemes: VisKin

Xuan Zhang, Jared N. Andrews, and Steen E. Pedersen*

Department of Molecular Physiology and Biophysics Baylor College of Medicine Houston, TX 77030

Abstract

Reaction kinetics for complex, highly-interconnected kinetic schemes are modeled using analytical solutions to a system of ordinary differential equations. The algorithm employs standard linear algebra methods that are implemented using MatLab functions in a Visual Basic interface. A graphical user interface for simple entry of reaction schemes facilitates comparison of a variety of reaction schemes. To ensure microscopic balance, graph theory algorithms are used to determine violations of thermodynamic cycle constraints. Analytical solutions based on linear differential equations result in fast comparisons of first order kinetic rates and amplitudes as a function of changing ligand concentrations. For analysis of higher order kinetics, we also implemented a solution using numerical integration. In order to determine rate constants from experimental data, fitting algorithms using the Levenberg-Marquardt algorithm or using Broyden-Fletcher-Goldfarb-Shanno (BFGS) methods were implemented that adjust rate constants to fit the model to imported data. We have included the ability to carry out global fitting of data sets obtained at varying ligand concentrations. These tools are combined in a single package, which we have dubbed VisKin, to guide and analyze kinetic experiments. The software is available online for use on PCs.

Introduction

Kinetic modeling of reaction mechanisms is important for guiding experiments in complex biochemical and chemical systems. A number of programs have been developed for prediction of kinetic reaction progress, notably KinSim [1,2] and several successors including KinTekSim (KinTek Corp., Austin, TX) and Dynafit [3] that are generally applicable and useful for a variety of kinetic schemes. Full analysis of the transformations of many proteins, especially voltage and ligand-gated ion channels, however, requires consideration of a large number of inactive and active conformational states in addition to several ligand-binding or voltage-activated steps. Inclusion of all possible binding events and conformations leads to a large number of interconvertible states. For one example, a minimum of 12 distinct states of binding and conformation are required for a minimal functional description of the ion channel, the nicotinic acetylcholine receptor (Fig. 1). The model includes 24 possible transitions among these states. Examination of such models by single-channel data [4–9] analysis using the QuB suite or ChannelLab (Synaptosoft Inc.), which includes whole-cell trace analysis, is specifically oriented towards electrophysiological data.

Our goal was to develop an interface for simple entry of kinetic schemes for proteins with multiple ligand binding and conformational changes. Effectively modeling a large number of

* To whom correspondence should be addressed: Department of Molecular Physiology and Biophysics, Baylor College of Medicine BCM335, One Baylor Plaza, Houston, TX 77030. Phone: (713)798-3888. Fax: (713)798-3475. E-mail: pedersen@bcm.tmc.edu.

Publisher's Disclaimer: This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final citable form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

states also requires consideration of the constraints of thermodynamic cycles that form in such systems. For example, the model in Fig. 1 includes 24 equilibrium constants; 13 constants are redundant and can be defined in terms of the other 11. In order to ensure microscopic energy balance, we implemented a check for consistency of thermodynamic cycles that uses graph theory to determine a full set of cyclic reaction paths and then checks each cycle for consistency.

A further concern for large, interconnected models is that they can be difficult to test objectively because of the large number of adjustable parameters (rate constants or equilibrium constants) that may be fitted to kinetic data with limited information content. Without prior knowledge of many of the constants, fitting to single data sets is often insufficient. Global fitting to multiple data sets can be carried out by systematic variation of other parameters such as ligand concentration that may further constrain the model.

In order to simulate large reaction schemes and fit their rate constants to data, we implemented a program with a graphical interface for entry of reaction intermediates and rate constants that rapidly and analytically computes reaction progress. The program solves the kinetic progress assuming a mixing of up to five distinct ligands at the initial time point. In addition to computing kinetic progress by numerical integration, we also implemented analytical computation from first order differential equations. While this approach is limited to reactions that obey first order conditions it is substantially faster than numerical integration and provides direct information on the changes of rates and amplitudes with ligand concentrations. A further advantage of this approach is that the first derivatives, which are required for iterative optimization algorithms, can also be determined faster [10].

Experimental methods and theory

Kinetic reaction mechanisms are described by a series of differential equations that relate reactant concentrations and elementary rate constants to changes in product concentrations with time. Most cases are unimolecular or bimolecular reactions described by a system of first and second order ordinary differential equations. In the absence of covalent bond breaking or formation, the elementary transitions constitute either conformational changes or binding events. The binding reactions can be further treated as pseudo-first order reactions when one reactant is in large concentration excess. In this case, linear algebra can be applied to derive closed-form expressions for reaction progress and for their first derivatives [11], whereas the more general case must be solved by numerical integration. The closed-form solutions to such problems can be determined faster than by numerical integration and provide rapid modeling and data fitting, particularly for models with a large number of reactants or states.

The Q-matrix approach for solving kinetic progress

The transition matrix, or **Q**-matrix, occurs in the solutions to Markov processes for prediction of ion-channel open and closed states [6]. The solution here is obtained in a similar manner. The basis of the **Q**-matrix solution is demonstrated here for the four-state cyclic mechanism shown in Fig. 2. For a given reaction scheme, entered by the user in the graphic interface, we parse the model and determine the differential equation system:

$$\begin{aligned}
 d[R]/dt &= -k_1^+[R][L] - m_1^+[R] + k_1^-[RL] + m_1^-[D] \\
 d[RL]/dt &= k_1^+[R][L] - k_1^-[RL] - m_2^+[RL] + m_2^-[DL] \\
 d[D]/dt &= m_1^+[R] - k_2^+[D][L] - m_1^-[D] + k_2^-[DL] \\
 d[DL]/dt &= m_2^+[RL] + k_2^+[D][L] - k_2^-[DL] - m_2^-[DL]
 \end{aligned}
 \tag{1}$$

Assuming the ligand concentration, $[L]$, to be constant, the system becomes pseudo-first order. Rewritten in matrix form, we obtain the following equations:

$$\begin{pmatrix} d[R]/dt \\ d[RL]/dt \\ d[D]/dt \\ d[DL]/dt \end{pmatrix} = \begin{pmatrix} -k_1^+[L] - m_1^+ & k_1^- & m_1^- & 0 \\ k_1^+[L] & -k_1^- - m_2^+ & 0 & m_2^- \\ m_1^+ & 0 & -k_2^+[L] - m_1^- & k_2^- \\ 0 & m_2^+ & k_2^+[L] & -k_2^- - m_2^- \end{pmatrix} \begin{pmatrix} [R] \\ [RL] \\ [D] \\ [DL] \end{pmatrix} \quad (2)$$

Or

$$d\mathbf{n}/dt = \mathbf{Q}\mathbf{n}(t) \quad (3)$$

$$\mathbf{n}(t) = e^{\mathbf{Q}t}\mathbf{n}_0 \text{ or } \mathbf{n}(t) = \mathbf{V}e^{\mathbf{\Lambda}t}\mathbf{V}^{-1}\mathbf{n}_0 \quad (4)$$

$$\mathbf{n}(t) = \mathbf{\omega}\mathbf{V}e^{\mathbf{\Lambda}t}\mathbf{V}^{-1}\mathbf{n}_0 \quad (5)$$

where $\mathbf{n}$ is the vector of reactant or state concentrations and $\mathbf{Q}$ is the transition rate matrix; the elements of $\mathbf{Q}$ describe the rates of transition among the states. The solution is given by the eq. 4 where $\mathbf{V}$ is the matrix of eigenvectors of $\mathbf{Q}$, $\mathbf{\Lambda}$ is the diagonal matrix of eigenvalues (λ_i), and $\mathbf{n}_0$ is the vector of the initial state concentrations [12]. In the second form of the solution (eq. 5), the vector $\mathbf{\omega}$ is included as a weighting factor for each state, for example, for fluorescent detection of stopped flow experiments where states may have distinct quantum yields and contribute unequally to the signal.

Analysis of rates and amplitudes

The general solution in equation 4 is a sum of exponentials for each state with rates given by the eigenvalues, λ_j , and a corresponding set of amplitudes given by the following equation for each of the $i = 1$ to n states:

$$\mathbf{n}(t)_i = \sum_{j=1}^n \mathbf{V}_{ij}(\mathbf{V}^{-1}\mathbf{n}_0)_j e^{\lambda_j t} \quad (6)$$

Because the rates and amplitudes are directly available from the analytical solution, they can be examined over a range of ligand concentrations. This serves as a useful tool for comparison of the model with experimentally observed rates and amplitudes, which are often measured across a variety of conditions. The program will plot the amplitudes for each rate and for each state over a selected range of ligand concentrations. However, because the algorithm for computing the eigenvalues and eigenvectors does not sort them similarly at every ligand concentration, such plots would generally yield mixed rates that correspond to different aspects of the mechanism if plotted simply according to eigenvalue number. To properly associate a set of rates and amplitudes across a range of ligand concentrations, the program compares changes in amplitudes and rates assuming smoothly varying values by first sorting the absolute value of the eigenvalues for each ligand concentration. Then the further adjustment is made by sorting the eigenvalues again to minimize changes in the rates with ligand concentration. With a sufficient number of concentrations, the rates and the amplitudes can be sorted to vary consistently with ligand.

Numerical methods

To accommodate reaction mechanisms that are not first or pseudo-first order, we also implemented computation of reaction progress by numerical integration. The kinetics of biological reactions often occur over a broad time scale, which constitutes a stiff numerical problem. The most popular methods for the solution of stiff problems are the backward differentiation formulas. The algorithms described by Shampine and Reichelt [13] for the MatLab ODE suite were implemented in Visual Basic (vs. 5, Microsoft, Redmond, WA). This approach incorporates interpolation to smooth the integration and deals with problems introduced from the changing time-step size.

Thermodynamic cycle constraints

Cyclic reaction schemes must obey the principle of conservation of energy and path independence to give realistic simulations. We implemented a function to check all thermodynamic cycles that is also used to provide constraints during data fitting (see below). An algorithm is used to find a basis set for all cycles in a model using graph theory and the grafpack algorithms maintained by John Burkardt [14] (http://www.scs.fsu.edu/~burkardt/f_src/grafpack/grafpack.html). First, a graph is created according to the reaction model where each reactant, or state, constitutes a vertex and each state-transition constitutes an edge. For illustration, the 8-state model in Fig. 3A constitutes a graph G . Then, a spanning tree T is computed, which will include all vertices of G but no cycles. The spanning tree T and the remaining edges (the set of edges $G - T$) of the 8-state model are illustrated in Figs. 3B and 3C. Next, for each edge in $(G - T)$, a unique cycle is defined after adding the edge back to T . For example, if we add the edge 3–4 in $(G - T)$ to T , the cycle 1–2–3–4–1 is reconstituted (Fig. 3D). The collection of cycles from each addition of an edge from $(G - T)$ to T represents a complete set of redundant equilibrium constants, though not the only possible set.

The constraint imposed by microscopic balance is that the energy between two points in the cycle is independent of the path. This is equivalent to requiring that the product of the equilibrium constants equals 1 when they are defined consistently in one direction around the cycle: for the cycle shown in Fig. 3D, $K_{12}K_{23}K_{34}K_{41} = 1$. In terms of rate constants, where $K_{12} = k_{12}/k_{21}$, we constrain the rate constants to obey eq. 7.

$$\frac{k_{12}}{k_{21}} \cdot \frac{k_{23}}{k_{32}} \cdot \frac{k_{34}}{k_{43}} \cdot \frac{k_{41}}{k_{14}} = 1 \quad (7)$$

During the optimization algorithms described below, we fit the log of the rate constants which permits us to formulate this relation as a linear constraint.

The graph theory algorithms are also employed to set up calculation of equilibrium distributions of the states over a range of ligand concentrations by using a spanning tree to define each state through one path from a reference state.

Optimization methods

One goal of kinetic studies is to extract elementary rate constants from experimental data. In principle, this could be accomplished by least squares fitting of experimental data to the kinetic model directly. Fitting algorithms require first derivatives for determining the optimal step size, at each iteration. An advantage of the pseudo-first order kinetic condition is that first derivatives can be determined as closed-form expressions, thereby speeding up the fitting procedure. Two methods for determining the step size at each iteration were implemented: the Levenberg-Marquardt method and the BFGS (Broyden, Fletcher, Goldfarb, and Shanno) method. In addition, the fitted parameters, which are usually the rate constants in the model, can be constrained to fixed values or to other rate constants. This is critical for fitting complex

models because data often do not contain sufficient information to determine all the rate constants. The logs of the rate constants are fitted to facilitate implementing ratiometric constraints using linear matrix methods and to improve the fitting because the step sizes can accommodate the broad scale of rate constant values that are frequently encountered when treating conformational changes and ligand binding simultaneously.

The Levenberg-Marquardt method is a well-established, widely used method to solve non-linear least-squares problems that minimizes the object function $S = \sum [y(i) - f(\theta, t(i))]^2$. In this expression, S is the sum of squares, $y(i)$ are the observed values and $f(\theta, t(i))$ is the fit function of the fitted parameters, θ , and time, $t(i)$, where i indexes the data points. The fitted parameters are estimated iteratively by $\theta_{k+1} = \theta_k + (\mathbf{J}^T \mathbf{J} + \alpha \mathbf{I})^{-1} \mathbf{J}^T \mathbf{r}$, where θ is the vector of the fitted parameters, $\mathbf{J}$ is the Jacobian (derivative) matrix of $f(\theta, t(i))$ with respect to θ , $\mathbf{r}$ is the vector of residuals ($y(i) - f(\theta, t(i))$), α is a small constant, and $\mathbf{I}$ is the identity matrix. Both analytical and numerical calculations of $\mathbf{J}$ were implemented. For the analytical approach, the methods described by Jennrich and Bright [10] were used. For numerical calculation, a strategy for selecting and adjusting the difference increment described by Salane [15] was used. Addition of the matrix $\alpha \mathbf{I}$ is important to the Levenberg-Marquardt approach to prevent ill-conditioned matrices and to control the step size and direction [16,17]. To accommodate a broad range of fit function values, α was initially set to the norm of the diagonal elements of the matrix $\mathbf{J}^T \mathbf{J}$. α is decreased 10 fold if the current iteration least squares value, S , is smaller than the previous iteration, otherwise α is increased 10 fold and the step recalculated. α is also adjusted according to the step size at each iteration. If the norm of the current step is larger than a limit of 10 ln units, α is increased by factor of three and the step recalculated.

The BFGS fitting method approximates the inverse Hessian of the object function at the $(k + 1)^{\text{th}}$ iteration from successive iterations of the fitting with the correction pairs (s_k, z_k) such that $s_k = \theta_{k+1} - \theta_k$ and $z_k = dS_{k+1}/d\theta - dS_k/d\theta$ where S is the object function described above. Given an initial matrix $\mathbf{H}^{-1}_k$ that approximates the inverse Hessian matrix, a new approximation, $\mathbf{H}^{-1}_{k+1}$, is defined by the recursive formula $\mathbf{H}^{-1}_{k+1} = \mathbf{W}^T \mathbf{H}^{-1}_k \mathbf{W} + \rho \mathbf{s} \mathbf{s}^T$ where $\rho = 1/z^T s$ and $\mathbf{W} = \mathbf{I} - \rho \mathbf{z} \mathbf{s}^T$. The step at each iteration is defined as above using $\mathbf{H}^{-1}_{k+1}$ instead of $(\mathbf{J}^T \mathbf{J} + \alpha \mathbf{I})^{-1}$: $\theta_{k+1} = \theta_k + \mathbf{H}^{-1}_{k+1} \mathbf{J}^T \mathbf{r}$ [18].

Implementation of fitting constraints

For a system of linear constraints, the constraints are described by $\Gamma \theta = \xi$, where θ is the full parameter vector, Γ is a coefficient matrix and ξ is a constant vector. To set up the constraints matrix Γ and the constants matrix ξ , each entry in a row of the constraints matrix multiplies each corresponding parameter of θ . There is one row per individual constraint. The linear combination of parameters equals the corresponding entry in ξ . The constraints should be linearly independent - i.e. the Γ matrix should have full rank, which is tested in the algorithm. The parameter vector is divided into two parts containing constrained (θ_1) and unconstrained (θ_2) parameters:

$$\theta = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} \quad (8)$$

Only the parameter subset θ_2 is optimized; the remaining parameters are determined using $\theta = \mathbf{P}(\mathbf{A}\theta_2 + \mathbf{b})$; the permutation matrix $\mathbf{P}$, $\mathbf{A}$, $\mathbf{b}$, and θ_2 are defined by the QR decomposition of Γ and ξ as detailed by Qin *et al.* [7–9,19].

Boundary constraints

The gradient projection method was used for optimization with constraints that provide boundary conditions on the parameter space [20]. At each iteration, the normal step ($\Delta \theta_2$) was

first computed and then evaluated to determine whether boundary conditions were violated. If the step would violate a boundary condition and if there are no active constraints, the step is truncated to meet the boundary condition(s) and the boundary constraints marked as active. In the case of active constraints where the step would violate boundary conditions, the following projection method is used to determine the step direction and size.

The boundary constraints take the form $\mathbf{L}^T \boldsymbol{\theta} > \mathbf{m}$, where $\mathbf{L}$ is a constraints matrix where each column defines a constraint vector, $\mathbf{I}_j$, $\mathbf{m}$ is a vector of upper or lower bounds and $\boldsymbol{\theta}$ is the vector of the full parameter set. For the reduced parameter set, $\boldsymbol{\theta}_2$, determined according to equality constraints, the constraint condition is tested at each iteration after calculation of the new step, $\Delta \boldsymbol{\theta}_2$:

$$\mathbf{L}^T \mathbf{P}(\mathbf{A}(\boldsymbol{\theta}_2 + \Delta \boldsymbol{\theta}_2) + \mathbf{b}) - \mathbf{m} > \mathbf{0} \quad (9)$$

For each index, j , of the vector defined by the left hand side that violates the condition, the following expression determines a factor $v = \min_{(all\ j)} [(m_j - \mathbf{I}_j^T (\mathbf{P}(\mathbf{A}(\boldsymbol{\theta}_2 + \Delta \boldsymbol{\theta}_2) + \mathbf{b}))) / (\mathbf{I}_j^T (\mathbf{P}(\mathbf{A} \Delta \boldsymbol{\theta}_2 + \mathbf{b})))]$ that reduces the length of the step size such that it hits the closest boundary and does not exceed any boundary conditions. The correct step size will then be determined by $\Delta \boldsymbol{\theta}_{2i} = v \Delta \boldsymbol{\theta}_2$ for the i^{th} iteration.

Fitted parameter error estimates

The standard errors for the parameter estimates are obtained from the covariance matrix where $\mathbf{V}(\boldsymbol{\theta}) \approx \mathbf{H}(\boldsymbol{\theta})^{-1}$. For the Levenberg-Marquardt method, $\mathbf{V} = (\mathbf{J}^T \mathbf{J})^{-1} \mathbf{G}(\mathbf{J}^T \mathbf{J})^{-1}$ and

$$g(\boldsymbol{\theta}_2)_{jk} = \sum_i r_i^2 J_{ij} J_{ik} \quad (10)$$

where i indexes the time points [20]. For the BFGS method, $\mathbf{H}^{-1}$ is given by the approximation at the last iteration; the estimates for the variances of the full parameter set can be derived from the variances of the reduced set as follows [8]:

$$v(\boldsymbol{\theta}_2) \approx \sum_k \left(\frac{\partial \boldsymbol{\theta}_2}{\partial \boldsymbol{\theta}_k} \right)^2 v(\boldsymbol{\theta}_{2k}) \quad (11)$$

where $\mathbf{V}(\boldsymbol{\theta})$ denotes variance of the complete parameter sets, which are the diagonal elements of the covariance matrix. Since the parameters are fitted as the logarithm of the rate constants, the error reported in terms of the standard deviation is $\sigma_i = q_i v_{ii}^{1/2}$, where each q_i is a rate constant in the $\mathbf{Q}$ matrix. The covariance matrix for the rate constants is then recomputed as $v_{ij, new} = q_i q_j v_{ij, old}$.

To further evaluate the goodness of fit, the software computes several other statistics. The cross correlation matrix, $\mathbf{CC}$ [20], is computed, where $cc_{ij} = v_{ij} (v_{ii} v_{jj})^{-1/2}$. A maximum deviation, σ_{max} , is computed to estimate the maximum standard deviation of a parameter upon covariance from other parameters as follows [20]: From the eigenvalue decomposition of the covariance matrix, $\mathbf{V} = \mathbf{U} \boldsymbol{\lambda} \mathbf{U}'$, where $\mathbf{U}$ is the eigenvector matrix and $\boldsymbol{\lambda}$ is the diagonal matrix of eigenvalues, assign $\sigma_{max} = \mathbf{w}_{max}^{1/2}$ where $w_{max}(i)$ is the maximum value of row i in the matrix $\mathbf{W} = |\mathbf{U} \boldsymbol{\lambda}|$. The dependency for each parameter is given as $d_i = 1 - v_{ii}/v_{max}(i)$, where v_{max} is the largest value in row i of $\mathbf{V}$ [21].

The software also computes the mean square fitting error, $S/(n_d - p)$ [22], the sum of squared residuals where n_d is the number of data points and p the number of fitted parameters, the Durbin-Watson statistic [23], the F-statistic [24], and outputs these values to a text file along with tables of the fitted value, $\boldsymbol{\sigma}_i$, σ_{max} , dependencies, $\mathbf{V}$, and $\mathbf{CC}$. Confidence limits to visualize

the goodness of fit, are computed as $f(\theta, t) \pm c_1^{1/2} * F_{crit}$, where $c_1 = \max ((\mathbf{J}\mathbf{V}) \cdot \mathbf{J})_i$, the maximum row value in row i , and F_{crit} is the critical value for the F -statistic for p degrees of freedom, n_d data points and a user-selectable confidence limit [20]. The limits can be plotted together with the best fit and the data.

Coding

The software was written in Visual Basic (vs. 5) and used linear algebra functions and graphics from Matrix VB, a MatLab (Newton, MA) add-in for Visual Basic. The algorithms were often first written and tested in MatLab before coding for Visual Basic. The program executable is available for Windows operating systems at the following web site: http://www.bcm.edu/physio/lab_pages/pedersen/index.html.

Results and Discussion

The goal of creating this software was to provide a tool for rapid evaluation of kinetic models by incorporating a simple interface for model entry into software with sufficient flexibility and power to evaluate a broad range of complex models through direct comparisons to data and back-fitting of data. Fig. 4 shows a screen capture of a simple four-state model that includes two conformational changes and two ligand binding steps, as shown schematically in Fig. 2. This is a minimal model for illustrating interactions between ligand binding and conformational changes that are common to modeling the kinetics of allosteric proteins, especially ligand-gated ion channels. Each state can be color coded and kinetic traces will be colored accordingly.

As more conformations and ligand binding states are taken into consideration, the number of possible combinations of states grows substantially. For example, as shown in Fig. 1 for the nicotinic acetylcholine receptor, a model that minimally includes two distinct binding site and three allosteric states, resting, open, and desensitized, requires 12 states with 24 state transitions described by 48 rate constants. For this reason, the input form can accommodate up to 100 states in a 10x10 array, where state transitions can be visualized between any adjacent states (including diagonals). In the cases where state-transitions are defined for non-adjacent states the transition is indicated by a line, rather than a display of the constants, and the rate constants are modified through a separate form.

Upon initiation of a kinetic simulation, the model is parsed and the rate constants entered into the Q-matrix format. The initial starting concentrations for each state and for the ligands are entered through a separate form. The kinetic progress for the model in Fig. 4 is shown in Fig. 5A on a log time scale, as the kinetic changes occur over 5 orders of magnitude. Since this was a cyclic model, the thermodynamic cycle constraints were checked using the graph theory algorithm described in the Methods and Theory section. We implemented this check because violations of thermodynamic cycle constraints can lead to unrealistic and inconsistent simulations. The check indicates to the user which cycles contain discrepancies so that rate constants can be changed to be consistent. For the four-state model in Fig. 4, thermodynamic cycle violations are easy to detect, but as models become more intertwined, such as in Fig. 1 that has 13 of 24 redundant state transitions, the ability to determine where violations occur is critical for repairing models efficiently.

The four states each change their concentration over time with three characteristic rate constants ($n - 1$ characteristic rates where n is the number of states); the last rate is zero and sets a constant value. The three rates are the same for all states but the amplitudes vary for each rate and state. The program will also calculate kinetics for a pre-set range of ligand concentrations, as shown in Fig. 5B for 10 nM to 100 μ M. In this case, the sum of the two states, RL and DL, is plotted to indicate the total amount of ligand bound. The program can plot any linear combination of states. This allows the program to account for multiple ligand

occupancy and is also convenient for fluorescent experiments where signals with distinct quantum yields may arise from several states.

The eigenvalue approach to computing reaction kinetics only solves first order differential equations and this imposes a pseudo-first order condition on all ligand binding steps whereby the ligand concentration is assumed to not change. In cases where ligand is not in excess and it is necessary to account for ligand depletion during the reaction, the user can opt to solve the model by numerical integration. An example comparing two ligand concentrations, one higher and one lower than the receptor concentration, is shown in Fig. 5C. In the case of the higher, excess ligand concentration, the kinetics modeled by either method gives similar results. In the case of the 100 nM ligand concentration, which is lower than the receptor concentration, using numerical integration to account for ligand depletion yields substantially different kinetics and net binding. In this case the eigenvalue method may be inappropriate for a typical test-tube experiment of fixed volume. In other cases, for example, a patch-clamp current measurement of channel activity in response to ligand-application, the eigenvalue method may still be accurate if the ligand solution is constantly replenished from a reservoir through a streaming pipette, as is often done in such experiments. In our implementation, the eigenvalue method computes kinetics about ten times faster than numerical integration for models of this size. This is a substantial advantage for examining a variety of models and conditions quickly, while the numerical integration is still available for models that do not satisfy the pseudo-first order criterion.

A further advantage of the eigenvalue computation is that the rates and amplitudes can be extracted directly from the computation. VisKin will plot rates and amplitudes according to ligand concentration for any state or linear combination of states. The rates and amplitudes corresponding to the plots in Fig. 5B are shown in Figs. 6A and 6B, respectively. This permits direct correlations with experimental data where varying ligand concentration is often a diagnostic tool for correlating characteristic rates with binding steps in the model. However, the eigenvalue determination does not always yield rates (eigenvalues) in the same order at different ligand concentrations. To match rates and amplitudes corresponding to the same process, the program sorts the values according to absolute value and then according to the derivatives of the values with ligand concentration. This sorting works better when more ligand concentrations (*i.e.* ten or more) are used to give smoothly varying changes in rates and amplitudes. For analysis outside the program, the program includes ASCII export functions for data and rate constants. These can be set to include noise to simulate random experimental variables.

In addition to Kinetics, the software will also calculate equilibrium concentrations over a range of concentrations to generate binding or activity plots. These can be directly replotted as Hill plots to extract the Hill coefficient for analysis of cooperativity.

Data fitting

While simulation can often determine whether a model can account for experimental data, it is desirable to directly determine rate constants from data. Backfitting a model to experimental data can be carried out using the eigenvalue approach as the derivatives required for non-linear least squares fitting can also be determined analytically. For complex models, however, there are likely more rate constants than can be defined by the data. For example, the model in Fig. 1 has 48 rate constants. It is unlikely any kinetic trace contains sufficient information to define all 48 constants. Therefore we included the ability to fit data sets at multiple ligand concentrations and the ability to constrain variables using prior information and using thermodynamic cycle constraints. Such constraints can reduce the number of fitted constants to a set that can be defined by the information content of the data.

A number of features for evaluating the goodness of fit were implemented. These include plotting the data and fit together and plotting the residuals directly beneath the data for visual inspection (not shown). We find this the most reliable method of determining the goodness of fit. We also compute a covariance matrix, from which we extract the standard error, maximum standard error, and dependency for each fitted parameter and compute a cross-correlation matrix. Examination of these parameters indicates the extent of error of the fitted parameters and indicates possible correlation with other parameters and may point to places to refine fitting for problematic models. We also report the sum of squared residuals, the F-statistic, the Durbin-Watson statistic, and include an option for plotting the confidence limits of the fit.

The four-state model shown in Figs. 2 and 4 was simulated at five ligand concentrations at ten-fold intervals; simulated data sets were then generated with varying amounts of added Gaussian noise. Each data set was then fit to the model using unconstrained global fitting. The fitted constants and the corresponding fractional error estimates are given in Table 1. An example of a fit with data noise at 2% of the maximum signal is shown in Fig. 7. The tabulated data shows the larger fit error estimates with higher data noise. The association rate constants are well determined when considering the consistency of the estimate with change in noise level and both the standard error and the maximum standard error. While the standard errors are tolerable in most other cases, the maximum standard errors, obtained by considering possible covariance (see methods), were quite large in several cases. These values would suggest possible highly correlated variance among some parameters, which may be reasonable since some aspects of the model will depend more strongly on the equilibrium constant (the ratio of two rate constants) than on the rate constants per se. Despite this potentially large covariance, the fitted parameters are close to the expected value, suggesting that the covariance is not a problem for determining a good fit for this particular model or that the parameters began near this particular local minimum.

To demonstrate that the statistics warn of poor fits, we also fit the data to a model where we eliminated the reaction between D and DL, k_2^- and k_2^+ , or constrained k_1^+ to a value of 5×10^7 , or both (see Fig. 2). The model has sufficient redundancy that either change alone produces a plausible, but not perfect, fit, yet the differences in statistics are readily apparent: the Durbin-Watson statistic and F-statistic are both reduced and the mean squared error is increased (Table 2). The poorest fit, with both the missing reaction and constrained k_1^+ value is shown as the dashed lines in Fig. 7 and the statistics are given in Table 2.

Fitting can be carried out using either the Levenberg-Marquard algorithm or the BFGS methods, as described in the Methods and Theory section. Alternatively, derivatives for the fitting can be calculated numerically, but this increases computation time substantially. For data analysis, the program will also fit imported data or simulations to multi-exponential functions. This permits extraction of the apparent rates from simulations, especially for numerical integrations where the rate constants and amplitudes cannot be extracted directly from the computation.

Acknowledgements

We thank Alberto Roca for useful discussions and for the program name. We also thank Robert Meltzer, Fred Mandel, and Iraida Andreeva for useful critique of the software and suggestions for improvement. We especially thank Dr. Stephen Locke for his suggestion of how to apply graph theory to the problem of microscopic balance. John Dani is thanked for his input into the software, suggestions for improvement and his support of this project. This work was supported by NIH grant NS35212 and Robert A. Welch Foundation grant Q-1406 to SEP and NIH grant NS048505 to John Dani.

References

1. Dang Q, Frieden C. New PC versions of the kinetic-simulation and fitting programs, KINSIM and FITSIM. *Trends Biochem Sci* 1997;22:317. [PubMed: 9270307]
2. Barshop BA, Wrenn RF, Frieden C. Analysis of numerical methods for computer simulation of kinetic processes: development of KINSIM--a flexible, portable system. *Anal Biochem* 1983;130:134–145. [PubMed: 6688159]
3. Kuzmic P. Program DYNAFIT for the analysis of enzyme kinetic data: application to HIV proteinase. *Anal Biochem* 1996;237:260–273. [PubMed: 8660575]
4. Sigworth FJ, Sine SM. Data transformations for improved display and fitting of single-channel dwell time histograms. *Biophys J* 1987;52:1047–1054. [PubMed: 2447968]
5. Colquhoun D, Hatton CJ, Hawkes AG. The quality of maximum likelihood estimates of ion channel rate constants. *J Physiol* 2003;547:699–728. [PubMed: 12562901]
6. Colquhoun D, Hawkes AG. On the stochastic properties of single ion channels. *Proc R Soc Lond B Biol Sci* 1981;211:205–235. [PubMed: 6111797]
7. Qin F, Auerbach A, Sachs F. Maximum likelihood estimation of aggregated Markov processes. *Proc R Soc London, Ser B: Biol Sci* 1997;264:375–383.
8. Qin F, Auerbach A, Sachs F. A direct optimization approach to hidden Markov modeling for single channel kinetics. *Biophys J* 2000;79:1915–1927. [PubMed: 11023897]
9. Qin F, Auerbach A, Sachs F. Hidden Markov modeling for single channel kinetics with filtering and correlated noise. *Biophys J* 2000;79:1928–1944. [PubMed: 11023898]
10. Jennrich RI, Bright PB. Fitting Systems of Linear Differential Equations Using Computer Generated Exact Derivatives. *Technometrics* 1976;18:385–392.
11. Bujalowski W, Jezewska MJ. Kinetic mechanism of the single-stranded DNA recognition by Escherichia coli replicative helicase DnaB protein. Application of the matrix projection operator technique to analyze stopped-flow kinetics. *J Mol Biol* 2000;295:831–852. [PubMed: 10656794]
12. Apostol, T. *Calculus*. John Wiley and Sons; New York: 1969.
13. Shampine LF, Reichelt MW. The MatLab ODE Suite. *SIAM Journal on Scientific Computing* 1997;18:1–35.
14. Burkardt, J. School of Computational Sciences. Florida State University; Tallahassee, FL:
15. Salane, DE. Tech. Report SAND86–1319, Sandia National Laboratory; Albuquerque, NM. 1986.
16. Jennrich RI, Ralston ML. Fitting nonlinear models to data. *Ann Rev Biophys Bioeng* 1979;8:195–238. [PubMed: 383001]
17. Smith SB, Shanno DF. *Technometrics* 1971;13:63–74.
18. Benson, SJ.; Moré, JJ. Tech. Report ANL/MCS-P909-0901, Mathematics and Computer Science Division, Argonne National Laboratory; Argonne, IL. 2001.
19. Qin F, Auerbach A, Sachs F. Estimating single-channel kinetic parameters from idealized patch-clamp data containing missed events. *Biophys J* 1996;70:264–280. [PubMed: 8770203]
20. Bard, Y. Academic Press; New York and London: 1974.
21. Sigmaplot 9.0 Manual, Systat Software 2004.
22. Smith LH, McCarty PL, Kitandis PK. Spreadsheet Method for Evaluation of Biochemical Reaction Rate Coefficients and Their Uncertainties by Weighted Nonlinear Least-Squares Analysis of the Integrated Monod Equation. *Applied and Environmental Microbiology* 1998;64:2044–2050. [PubMed: 9603812]
23. Durbin J, Watson GS. Testing for serial correlation in least squares regression. II. *Biometrika* 1951;38:159–178. [PubMed: 14848121]
24. Fisher RA. The Goodness of Fit of Regression Formulæ, and the Distribution of Regression Coefficients. *Journal of the Royal Statistical Society* 1922;85:597–612.

Fig 1.

A model for allosteric activation of a protein with two ligand binding sites and three inter-changeable conformations, for example, for the nicotinic acetylcholine receptor ion channel. This model includes 12 states and 24 equilibrium constants; for a kinetic description, formally, 48 rate constants would be required. However, the 24 equilibrium constants, or the corresponding 48 rate constants, are constrained by noting that the 15 independent cycles constitute a set of constraints reducing the model to 9 independent equilibrium constants, or 33 non-redundant rate constants. This particular model can be further constrained by noting that the ligand binding events for each site will have the same affinity and rate constants, regardless of the occupancy of the other site.

Fig 2.

A four-state model for allosteric activation of a protein. The model includes two binding events and two allosteric transitions, described by second order and first order rate constants.

Fig 3.

A graph-theory approach to determining thermodynamic cycles. (A) The thermodynamic cycles in a highly inter-connected reaction scheme are considered as a set of edges and vertices in a graph, G . (B) A set of independent cycles can be determined by first obtaining a spanning tree of the scheme (graph T), which includes all states but contains no cycles. (C) Subtraction of the spanning tree from the whole scheme yields a set of equilibrium constants that are redundant (the set of connections $G - T$). (D) To determine the relationship between the redundant constants and the remainder of the scheme, each equilibrium constant is added back to reconstitute a single cycle within the reaction scheme. Each cycle includes eight rate constants, of which one is redundant. This relationship provides one additional constraint according to Eq. 7.

The screenshot shows a 'Model Input' window with a grid of input fields for various parameters. The parameters are organized into two main columns, each with a header row (E11, E12) and a data row (E21, E22). To the right of these are two more columns with headers E13 and E23.

E11		E12		E13	
R	k11-12: 1e8 k12-11: 1000 Note: [dropdown]	RL	k12-22: 1 k22-21: 0.01 Note: [dropdown]		
E21	k21-22: 1e8 k22-21: 1 Note: [dropdown]	E22	DL	E23	

Fig 4.
Screen capture of the VisKin model input form.

Fig 5.

Example of VisKin output for the model shown in Fig. 2. Rate constants were assigned as follows: $k_{1+} = 1 \times 10^8$, $k_{1-} = 1 \times 10^3$, $k_{2+} = 1 \times 10^8$, $k_{2-} = 1$, $m_{1+} = 1 \times 10^{-3}$, $m_{1-} = 1 \times 10^{-2}$, $m_{2+} = 1$, $m_{2-} = 1 \times 10^{-2}$. The forward binding rate constants have units of $\text{M}^{-1}\text{s}^{-1}$; all others have units of s^{-1} . The initial concentrations were $[R] = 4.5 \times 10^{-7} \text{ M}$, $[D] = 0.5 \times 10^{-7} \text{ M}$, and the ligand concentration, $[L] = 1 \times 10^{-7} \text{ M}$. (A) Kinetic progress of each of the four states ($\mathbf{n}(t)$, eq. 6) as determined by the eigenvalue method. (B) Progress of ligand binding (the sum of the $[RL]$ and $[DL]$) for 15 distinct ligand concentrations between 10 nM and 100 μM . (C) comparison of kinetic progress for binding computed by the eigenvalue method (solid lines) and by numerical

integration (dashed lines) for two ligand concentrations: $[L] = 1 \times 10^{-5} \text{ M}$ and $[L] = 1 \times 10^{-7} \text{ M}$. All the data were replotted for presentation using Sigmaplot (Systat, Richmond, CA).

Fig 6.

The eigenvalue approach determines rates and amplitudes. The data simulated for binding for the four-state model shown in Fig. 4B includes the sum of up to three exponentials with three rates constants and corresponding amplitudes. (A) The rates are plotted according to the change in ligand concentration. (B) The corresponding amplitudes are displayed for the sum of the two bound states ([RL] + [DL]); increases in signal appear as negative amplitudes. Note that the eigenvalue approach yields the same rates for all four states because these depend only on the Q-matrix (see eq. 6) whereas the corresponding amplitudes will differ for each state. VisKin permits plotting the amplitudes together or independently for each state.

Fig 7.

Global fitting to simulated data. Data was simulated by VisKin for the four state model (Fig. 2) at five ligand concentrations ($[L] = 10^{-8}, 10^{-7}, 10^{-6}, 10^{-5},$ and 10^{-4} M) with the rates given in Fig. 4 and with Gaussian noise added (std. dev. of 10 nM or 2 % of the signal). The data was back fit to the model using the Levenberg-Marquardt algorithm and the global fitting feature. The best fits are plotted as the solid curves. The dashed lines represent a poor fit to the same model except that the k_2^- and k_2^+ reactions were omitted and k_1^+ was constrained to 5×10^7 .

Table 1

Global-fitting results and errors upon increasing noise in the data.

	k_1^{+a}	k_1^-	k_2^+	k_2^-	m_1^+	m_1^-	m_2^+	m_2^-
Model value:	1×10^8	1000	1×10^8	1	0.001	0.01	1.00	0.01
Fit start value:	1×10^9	100	1×10^9	0.1	1×10^{-4}	0.001	10	0.1
% Noise ^b	θ_{fit}^c							
0.2	9.99×10^7	999	1.00×10^8	0.994	9.79×10^{-4}	0.0098	0.999	0.01
0.6	9.99×10^7	1000	1.01×10^8	1.02	0.00108	0.0101	1.00	0.01
2.0	9.98×10^7	989	1.02×10^8	1.03	0.00115	0.0108	0.999	0.01
6.0	1.04×10^8	1064	8.44×10^7	0.709	0.00105	0.0202	1.02	0.0104
20.0	1.07×10^8	1020	9.28×10^7	1.18	0.00252	0.0145	0.910	0.0078
	σ_θ/θ							
0.2	7.38×10^{-4}	8.44×10^{-4}	0.004	0.0081	0.028	0.046	9.36×10^{-4}	0.0028
0.6	0.0021	0.0024	0.011	0.0234	0.074	0.127	0.0028	0.0083
2.0	0.0074	0.0085	0.035	0.0770	0.226	0.355	0.0091	0.0276
6.0	0.0219	0.0251	0.101	0.257	0.800	0.884	0.0286	0.0808
20.0	0.0735	0.0812	0.379	0.780	1.09	1.87	0.0902	0.302
	$\sigma_{\theta_{\text{max}}}/\theta$							
0.2	0.0013	.356	0.0037	41.4	712	194	8.15	42.9
0.6	0.0040	1.08	0.0108	123	2020	596	24.0	135
2.0	0.0133	3.62	0.0353	398	6330	1340	77.7	419
6.0	0.0335	8.64	0.102	1290	1.63×10^4	4140	219	1110
20.0	0.129	35.2	0.381	3770	3.23×10^4	1.41×10^4	856	4890

^aParameters are those given in Fig. 2 with initial values as shown in Fig. 4. Units are $\text{M}^{-1}\text{s}^{-1}$ for the bimolecular rate constants ($k^\pm$) and s^{-1} for the first order constants.

^bThe noise level indicates the deviation of the Gaussian-distributed noise added to the simulated data as a percentage of the maximum signal (500×10^{-7}).

^cThe parameter values, θ , are the results of the nonlinear regression; the standard error, σ_θ , and maximum standard error, $\sigma_{\theta_{\text{max}}}$, are reported after being normalized to the fitted value.

Table 2

Fitting statistics for good and poor fits.

Fitting	DW ^a	F-statistic	MSE ^b
Full model	2.00	358	1.0×10^{-16}
k_2^- and k_2^+ omitted	0.71	135	2.8×10^{-16}
k_1^+ constrained ^c	0.68	119	2.9×10^{-16}
both ^d	0.24	44	8.5×10^{-16}

^aThe Durbin-Watson statistic; good fits will have values between 1.7 and 2.3.

^bMean squared error was computed as described in Experimental Methods and Theory.

^c k_1^+ was constrained to equal 5×10^7 .

^dThis fit is illustrated in Fig. 7.