Biophysical Chemistry 279 (2021) 106682

Contents lists available at ScienceDirect

Biophysical Chemistry

journal homepage: www.elsevier.com/locate/biophyschem

Multi-start Evolutionary Nonlinear OpTimizeR (MENOTR): A hybrid parameter optimization toolbox

Zachariah M. Ingram [a,1], Nathaniel W. Scull [a,1], David S. Schneider [b,2], Aaron L. Lucius [a,*,2]

[a] Department of Chemistry, University of Alabama at Birmingham, Birmingham, AL, USA
[b] Department of Biochemistry and Molecular Genetics, University of Alabama at Birmingham, Birmingham, AL, USA

ARTICLE INFO

Keywords:
Optimization
Data fitting
Kinetics
Thermodynamics

ABSTRACT

Parameter optimization or "data fitting" is a computational process that identifies a set of parameter values that best describe an experimental data set. Parameter optimization is commonly carried out using a computer program utilizing a non-linear least squares (NLLS) algorithm. These algorithms work by continuously refining a user supplied initial guess resulting in a systematic increase in the goodness of fit. A well-understood problem with this class of algorithms is that in the case of models with correlated parameters the optimized output parameters are initial guess dependent. This dependency can potentially introduce user bias into the resultant analysis. While many optimization programs exist, few address this dilemma. Here we present a data analysis tool, MENOTR, that is capable of overcoming the initial guess dependence in parameter optimization. Several case studies with published experimental data are presented to demonstrate the capabilities of this tool. The results presented here demonstrate how to effectively overcome the initial guess dependence of NLLS leading to greater confidence that the resultant optimized parameters are the best possible set of parameters to describe an experimental data set. While the optimization strategies implemented within MENOTR are not entirely novel, the application of these strategies to optimize parameters in kinetic and thermodynamic biochemical models is uncommon. MENOTR was designed to require minimal modification to accommodate a new model making it immediately accessible to researchers with a limited programming background. We anticipate that this toolbox can be used in a wide variety of data analysis applications. Prototype versions of this toolbox have been used in a number of published investigations already, as well as ongoing work with chemical-quenched flow, stopped-flow, and molecular tweezers data sets.

Statement of significance: Non-linear least squares (NLLS) is a common form of parameter optimization in biochemistry kinetic and thermodynamic investigations These algorithms are used to fit experimental data sets and report corresponding parameter values. The algorithms are fast and able to provide good quality solutions for models involving few parameters. However, initial guess dependence is a well-known drawback of this optimization strategy that can introduce user bias. An alternative method of parameter optimization are genetic algorithms (GA). Genetic algorithms do not have an initial guess dependence but are slow at arriving at the best set of fit parameters. Here, we present MENOTR, a parameter optimization toolbox utilizing a hybrid GA/NLLS algorithm. The toolbox maximizes the strength of each strategy while minimizing the inherent drawbacks.

1. Introduction

Parameter optimization, or more commonly 'data fitting', is a process by which the parameters for a model are optimized to best describe the experimental observations. In the context of biochemical kinetics and thermodynamics, experimental observables are often changes in fluorescence [1–9], absorbance [10–12], heat [3,13], force [14,15], or pixel density [16–19] to name only a few. The first task of the parameter optimization process is to determine a mathematical model that relates the experimental observable to an experimentally adjustable independent variable, i.e. time, concentration, etc. The set of parameters that are sought to be determined with the mathematical model could be kinetic

* Corresponding author.
E-mail address: allucius@uab.edu (A.L. Lucius).
[1] Co-first authors.
[2] Co-corresponding authors.

https://doi.org/10.1016/j.bpc.2021.106682
Received 24 July 2021; Received in revised form 10 September 2021; Accepted 10 September 2021
Available online 29 September 2021

rate constants, equilibrium constants, enthalpy, etc. The objective is to find a set of parameters that when applied to the model best describe the experimental observables. Once a set of parameters are found, the second task is to evaluate how well these parameters are determined. This task includes examining the uncertainty on each of the optimized parameter values as well as the error space around the parameter. The last task, and arguably the primary goal, is to discern the meaning of the parameters in the context of a given system being investigated.

Since the 1980s, nonlinear least squares (NLLS) analysis has been the standard method for performing parameter optimization by biochemists [20–24]. While this analysis strategy is easy to use and robust in reasonably simple models, difficulties are encountered with complex models especially ones involving correlated parameters [1,5,25,26]. As the systems being investigated become increasingly more complex, more advanced analysis tools and strategies are vital. However, to our knowledge, little has been done to apply more sophisticated optimization methods to biochemical kinetic and thermodynamic investigations. This observation is likely a consequence of more advanced optimization methods requiring additional programming expertise. Additionally, NLLS does not lend itself to exploiting available advantages of parallel processing. Consequently, little has been done to capitalize on modern computational power.

Numerous programs use NLLS to optimize parameters in a user defined model. A few examples of common programs used by biochemists are are KaleidaGraph (Synergy Software, Reading PA), Graphical Analysis (Vernier, Beaverton OR), Scientist (Micromath, Saint Louis MO), Origin (OriginLab, Northampton, MA), and KinTek Explorer (KinTek Corporation, Austin TX). A general overview of the mathematical details of NLLS algorithms and a great initial start for researchers from a chemistry or molecular biology background is Chapter 6 from *Data Analysis in Biochemistry and Biophysics* and Michael L. Johnson's methods chapter on using least-squares techniques in biochemistry [22]. Many of the limitations of the technique are described there. However, few solutions to the limitations were offered.

A number of characteristics are conserved across all NLLS techniques. These algorithms require an initial guess of the parameter values. For simple models, less certainty on the guess is required for convergence on the best answer. However, for models that are more complicated the guess must be reasonably close to expect convergence. The algorithm will iteratively improve the initial set of parameters until there is no longer a significant difference between the preceding set of parameters and the resultant improved parameters [27]. An analogy of this process is a ball placed on a curved surface as illustrated in Fig. 1 a. In this analogy, the ball represents the current set of parameters while the surface is the goodness of the fit parameters, e.g. chi-squared. The ball will roll down the surface until the ball arrives at the minimum chi-squared value. The bottom of the surface corresponds to the best estimate of the given parameters, since it yields the lowest chi-squared. NLLS is classified as a deterministic method because if the algorithm is started at the same starting point it will always arrive at the same result. The NLLS algorithm does not contain any randomness.

It is important to point out that simple NLLS routines always go downhill as illustrated in Fig. 1a. That is to say, the routine seeks lower and lower values of the chi-squared starting from the initial guess. This immediately leads to a dependence of the results on the initial user provided guess [28,29]. If the error contour does not have local minima, then NLLS will arrive at the same global minimum irrespective of initial guess as illustrated by Fig. 1a. However, consider the case where a local minimum is present as illustrated in Fig. 1b. If one always chooses initial guesses from the right-hand side of the curve, then a NLLS routine will always find the minimum on the right. Whereas, if initial guesses are chosen on the left-hand side, then the NLLS optimization will always find the minimum on the left, which, in this example is the lowest. This type of emergence of local minima often occurs with correlated parameters. Moderately difficult mathematical strategies to overcome correlated parameters include the use of orthogonal polynomials and

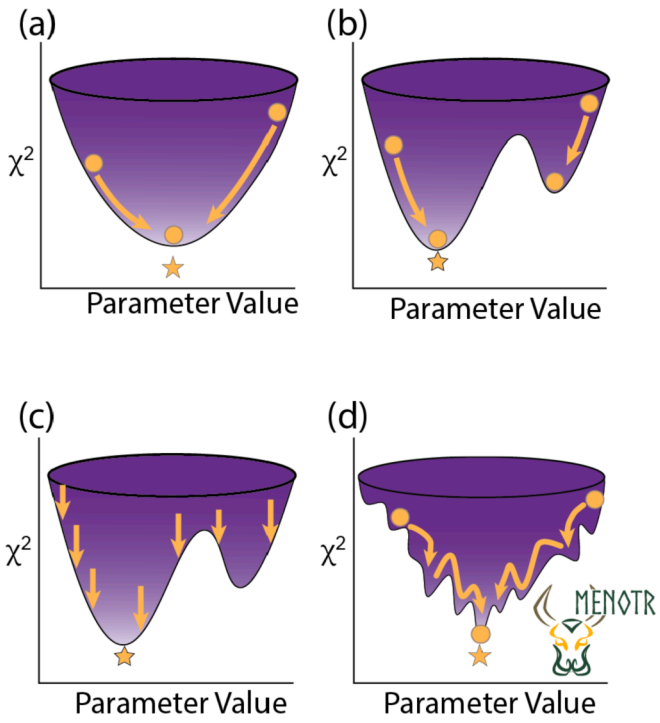


Fig. 1. Illustration outlining deterministic, random, and hybrid algorithms approach to finding minima in error contour. a) NLLS, a deterministic method will quickly converge on solution. If one minimum is present, the algorithm will converge on identical position irrespective of starting point. b) NLLS has well known initial guess bias in cases involving multiple minima in error contour. Different starting points will result in different optimized parameters. c) The genetic algorithm will randomly probe the error space at different parameter values. This algorithm overcomes local minima but has difficulty in finding the absolute minimum. d) MENOTR, a hybrid NLLS-genetic algorithm, takes advantage of the strengths of both approaches while minimizing the weaknesses. The genetic algorithm component of MENOTR escapes local minima and the NLLS quickly converges on a solution.

Fourier series analysis [25,30].

One way a user can overcome local minima problems is by starting NLLS routines with multiple initial guesses and tabulating the resulting optimized parameters with the corresponding goodness of fit. The tabulated values are then ranked based on their respective goodness of fit, and the minimum value is assumed to be the global minimum. A reasonable question naturally arises: when have enough different initial guesses been investigated to conclude that the lowest goodness of fit score has yielded the best parameter values achievable? The answer is as many as possible. However, this process is both tedious and laborious. Moreover, the answer will not be the same for every model. The researcher may also be tempted to only use initial guesses, which yield successful convergence onto a result. This is because making large changes in the initial guesses can often lead to divergence or "crashing" of the software, both of which are often interpreted as evidence of a bad model. However, failure of code should not be interpreted as failure of the model. Thus, the method of manually testing many different initial guesses is not only a laborious task but also one that can easily introduce user bias.

Here we sought to develop a method that overcomes this initial guess dependence. On the surface, this sounds trivial; code a computer to give many starting points to a NLLS routine. However, the problem with that solution is what was articulated above. If initial guesses that are too far from a local minimum or the best fit are given to a NLLS routine, then the routine is likely to fail and failure of the routine cannot be used to rule out a set of parameters.

Metaheuristics are "upper-level methodologies" (meta) that work "to

discover" (heuristic) solutions to a problem. Metaheuristics have been an avenue of active research for nearly four decades and have yielded good results in solving high-level optimization problems across a variety of fields [31–33]. Examples include simulated annealing, ant colony optimization, particle swarm optimization, bees algorithm, and the genetic algorithm [34–39]. As might be obvious from the names, these methods are often nature-inspired and utilize ideas like mutation, fitness, gene crossover, and natural selection to solve optimization problems [40]. The underlying characteristic of these methods that lends itself to solving such problems is the stochastic nature of these processes.

The genetic algorithm (GA) is a well-established optimization method. This method mimics the principles of biological evolution to arrive at the best solution or to identify the parameter that best describes the experimental data. A number of books are available to explain the details concerning GAs, but here we will discuss some general features [40–42].

Consider the simplest case involving a one parameter model, y = m*x, where the slope, m, is the parameter to be optimized. An initial value for m is provided by the user and is used to randomly generate many unique initial guesses. This set of initial guesses is referred to as a population matrix. For each of the experimental x and y pairs, a simulated set of x and y pairs are generated from each of the initial guesses in the population matrix, and a corresponding chi-squared is calculated. The example in Fig. 1 uses chi-squared to quantify goodness of fit but other metrics, like root mean squared deviation (RMSD) or variance could be used. Each arrow in Fig. 1 c illustrates how different parameter values result in different chi-squared values. The sampling of the parameter values to generate the population is a stochastic process meaning the sampling is discreet and random. The resultant chi-squared values are ranked and the parameters with the lowest chi-squared are considered the best. Unlike NLLS methods, which are trying to find the best parameters by systematically minimizing the chi-squared value of the fit, the GA is significantly less likely to be trapped in local minima. Equally important, the calculation of chi-squared is a simple mathematical operation and the code is unlikely to fail at this stage. This is in stark contrast to NLLS where the hunt for the minimum is often a failing point of the code because the code is trying to find a minimum by executing the first derivative of the fitting function, setting it equal to zero, and then finding solutions. In the last step of the GA, a new population matrix is formed with values centered around the 'best fit' parameter value from the previous population. This process is performed iteratively until the user defined stopping criteria are met. In effect, this strategy results in thousands of initial guesses. While genetic algorithms do not get stuck in local minima all GAs have difficulty in resolving the absolute minimum. This is a direct consequence of the stochastic nature of GAs. The only way to reach the absolute minimum in a GA is for the optimal parameters to be randomly selected in the population, which is inherently unlikely.

Here we report the development of MENOTR, a hybrid algorithm that balances the strengths of NLLS and GAs to offset their corresponding limitations. MENOTR, **M**ulti-start **E**volutionary **N**onlinear **O**p**T**imize**R** was developed from the MATLAB (MathWorks, Inc., Natick MA) scripts used in our previous kinetic data analysis and was designed to address NLLS's dependence on initial guesses through an easy to use MATLAB toolbox [17,19]. MENOTR was designed to give researchers with a limited coding background access to a more advanced optimization tool for the analysis of complex kinetic and thermodynamic models. MENOTR is a hybrid NLLS-genetic algorithm in which the GA portion of the code ranks thousands of initial guesses before performing NLLS optimization. This process is repeated multiple times, further refining the parameters to achieve a lower chi-squared until the global minimum is reached. This approach eliminates the optimization routine's dependence on the user provided initial guesses and overall minimizes user bias. In addition, MENOTR provides a greater search of the possible parameter values compared to what one could achieve by manually varying the initial guesses, thus giving greater confidence that the resultant optimized parameter values are "the best". Fig. 2 outlines a sampling of the features present in MENOTR. MENOTR is capable of accommodating a number of different types of models describing experimental data. Such examples are systems of ordinary differential equations describing chemical kinetics and closed form expressions involving both simple and complex mathematical expressions (addition, subtraction, Laplace transform, inverse Laplace transform, Fourier transform, etc.). In addition, MENOTR can globally optimize parameters present in equations describing different experimental observables. An example is shared parameters used to simultaneously describe changes in both fluorescence and anisotropy [43].

MENOTR has additional functionalities that are of immediate usefulness. MENOTR contains two methods for calculating parameter uncertainty. The first utilizes Monte Carlo simulations and reflects how reproducible fit parameters would be if a large number of replicates were performed [44]. The second method, grid search analysis, identifies confidence regions for each parameter. These regions correspond to parameter values that describe the experimental data. These methods allow a user to identify parameters that are correlated and aids in identifying unconstrained parameters. Additionally, MENOTR's ability to run independently of user input makes it ideal for being run on high-performance computing clusters (HPC). Many parameter optimization programs do not have the ability to run independently of user input and doing so increases user productivity by allowing multiple models to be optimized simultaneously.

Here we present three case studies to demonstrate MENOTR's parameter optimization capabilities. The first case study contains published data describing DNA unwinding by the helicase RecBCD, the

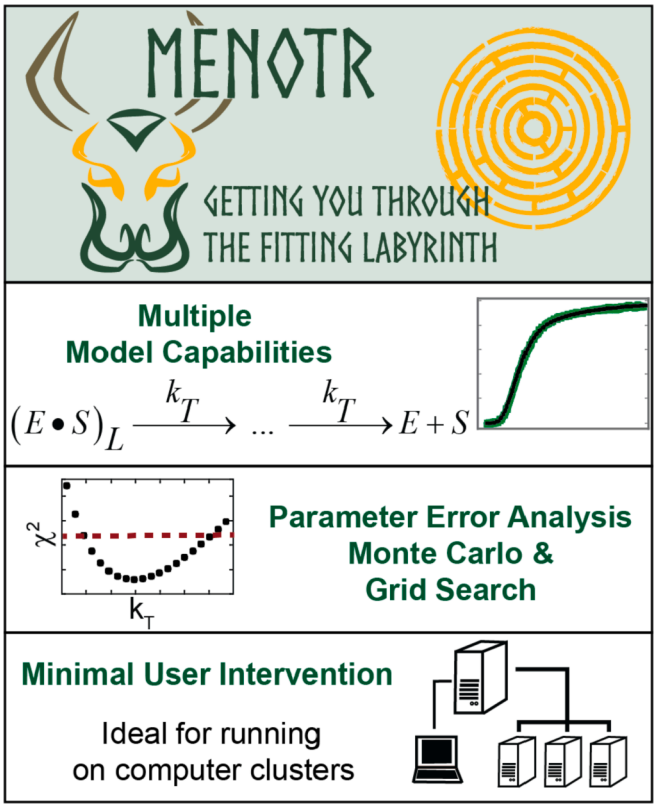


Fig. 2. Functionalities of the MENOTR toolbox. MENOTR can optimize parameters describing floating or static models in addition to closed form expressions. Errors on resultant parameters may be ascertained using Monte Carlo or grid search analysis. The toolbox can be run with minimal user intervention, making it ideal for being launched on a node of a high-performance computer cluster.

second presents published data describing polypeptide unfolding/translocation catalyzed by the AAA+ chaperone ClpA, and the last study uses simulated data describing three classic thermodynamic ligand binding models. The first two examples were chosen because the data were previously published, and the models used in the analysis are mathematically challenging. The equations used are first derived in the Laplace domain. The numerical inverse Laplace transform is then necessary to simulate time courses in the time domain. Additionally, several parameters are correlated making the optimization extremely challenging. The third example was chosen because it requires the use of implicit fitting, which again results in a non-trivial optimization problem. Implicit equations emerge in even simple thermodynamic models describing ligand binding. Finally, two simple examples with simulated data described by a line and kinetic model are present in the supplemental as a resource for training new users.

2. Results & discussion

Genetic algorithms are unlikely to get stuck in local minima, but they do have difficulty converging on the absolute minimum. In contrast, NLLS will easily find the absolute minimum if an initial guess is provided that is close to that minimum. Thus, we hypothesized that by combining the two we could overcome the limitations of both. After many iterations of the GA, a set of parameters could be provided to the NLLS routine that could be confidently assumed to be close to the absolute minimum. In addition, a third algorithm is used that we refer to as a multi-start routine. The initial user provided guess is first coarsely refined by the GA before being further refined by NLLS. This process is setup in a cyclic form where the output of the NLLS algorithm is passed back to the GA and the process is repeated until an optimal value is achieved. The outcome is a survey of the error surface illustrated in Fig. 1a–d. This idea is not novel, several research fields have implemented variations of this optimization strategy including mathematics [45,46], engineering [47,48], computer science [49], and systems biology. However, to our knowledge no readily accessible analysis tool is available to optimize parameters in the complex kinetic and thermodynamic models that we will be presenting here. Moreover, we did not find any available tools that were easily adaptable to the unique challenges presented by transient state kinetic data and the statistical thermodynamic models required to describe the thermodynamics of ligand binding.

2.1. Overview of how MENOTR optimizes parameters

MENOTR, Multi-start Evolutionary NLLS OpTimizeR, is a custom-built MATLAB toolbox used to optimize parameters that requires minimal user intervention. A general explanation of how MENOTR optimizes parameters is shown below while a more nuanced description can be found in the supporting material and throughout the source code.

Like all optimization algorithms, MENOTR requires a set of user supplied initial guesses as a first step, depicted at the top of Fig. 3. While the exact values are unknown, typically a user will have reasonable guesses for each parameter value. Preferably within one or two orders of magnitude of the optimal value. Initial guesses closer to that of the true answer result in faster optimizations. In MENOTR, these initial guesses are used to generate a search area of different parameter values. It is often advantageous to establish a parameter search area encompassing values spanning several orders of magnitude. MENOTR uses the $\log_{10}$ of each parameter value to establish the order of magnitude of the initial guesses. A population of different parameter values is then generated for each initial guess. The generated population values are a Gaussian distribution centered on the $\log_{10}$ of the initial guess while the standard deviation is held constant at 1.The standard deviation value of 1 was chosen to create a population encompassing parameter values one order of magnitude above and below the initial guess. For example, consider an initial guess for a parameter, k_1, is 100 s^{-1}. MENOTR first takes the log of this value, $\log_{10}(100) = 2$. A gaussian distribution of parameter values is generated centered on 2 with a standard deviation of 1. The antilog of each parameter value is calculated resulting in parameter values from ~10 s^{-1} to ~1000 s^{-1}.

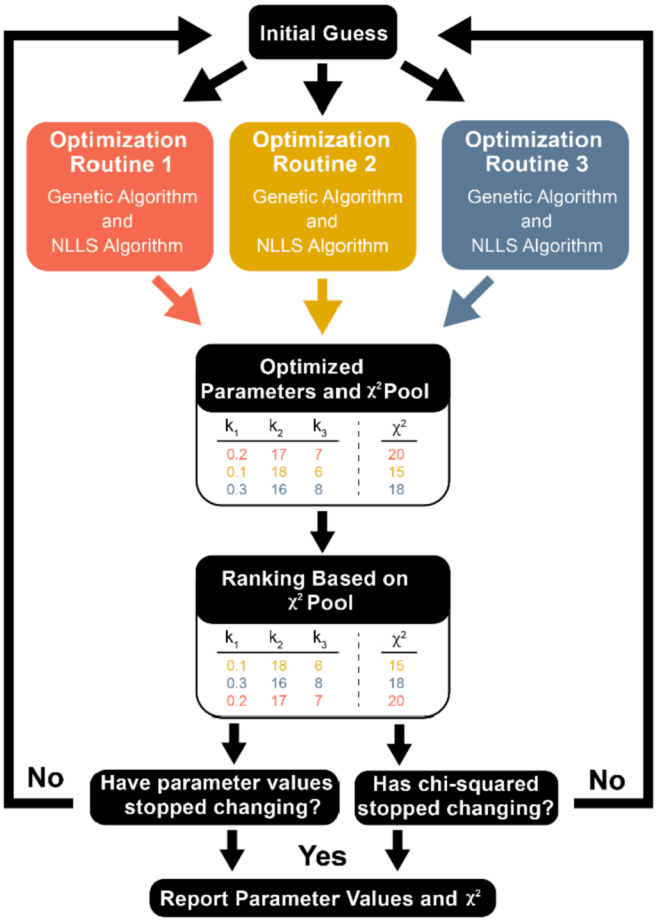


Fig. 3. Flow diagram overview illustrating how MENOTR optimizes parameters. In the first step, an initial guess is passed to three different optimization routines. The parameters are optimized individually and then pooled together. The pooled parameters are then compared to see if they are different. If they are different, then the best set of parameters is used as the new initial guess. If they are identical then the parameter values are reported.

Once the initial guess values are chosen for each parameter, the parameter values are passed to a user-defined number of optimization routines, this is the multi-start component of the algorithm. The default number of optimization routines in MENOTR is three, this is depicted in Fig. 3 with the red, yellow, and blue boxes. Each of the optimization routines begin by generating a separate parameter population (parameter search area) as described in the previous paragraph. While the mean of each parameter population within an optimization routine will be identical, the individual parameter values within each population will be different and lead to diversification of the surveyed parameter values. Within each optimization routine, a genetic algorithm will be used to identify parameter values with small chi-squared values. Some of the best parameter values are then passed to a NLLS algorithm that further optimizes the parameter values. The algorithms have been structured to take advantage of the inherent strengths of each optimization methodology while minimizing the drawbacks. A more detailed explanation of how the genetic algorithm and NLLS algorithm work to achieve optimized parameter values can be found in the supporting material. Each of the optimization routines are performed independently with no cross-over of information between optimization routines. The separation of the optimization routines is a multi-start process because each optimization routine is starting from a different population of parameter

values.

Upon completion, the set of parameters with the lowest chi-squared value from each optimization routine are pooled together with their corresponding chi-squared value. Shown in Fig. 3 is an example where three kinetic rate constants, k_1–k_3, are being optimized to describe a hypothetical experimental data set. Each optimization routine results in a different set of parameter values and a corresponding chi-squared value. After being pooled, the parameter sets are then ranked based on the chi-squared values.

The next task is to determine if the optimized parameters are the best possible parameters or if further refinement is necessary. We designed MENOTR to address this question by requiring the pooled parameter values to satisfy two conditions. The first condition is that the reported best parameter values between different optimization routines should agree. If disagreement is present between the parameter values, then further refinement is necessary, and the parameter set with the lowest chi-squared is used as the initial guess for another round of optimization routines. The second condition that must be met is that the chi-squared must stop changing between optimization routines. If a set of parameters has been fully optimized, then the chi-squared should be minimized and a smaller chi-squared is not achievable. Thus, if multiple optimization routines are all reporting the same chi-squared value a smaller chi-squared value is unlikely. If different chi-squared values are reported from the optimization routines, then the parameter set with the smallest chi-squared is used as the initial guess for a second round of optimization. However, neither the parameters nor the chi-squared is ever going to be mathematically exactly equal. Rather for practical purposes it is necessary to determine equality based on some tolerancer. MENOTR has this built-in capability and when both stopping conditions are met, MENOTR reports the optimized set of parameters with the corresponding chi-squared value as the final solution.

2.2. Parameter uncertainty analysis in MENOTR

Once the parameters are optimized for a given model, the next step is to measure the uncertainty on each of the optimized parameters. Such error measurements are beneficial for determining trends in parameter values, or even more simply, when two parameter values are statistically different or identical. MENOTR presents two different strategies to assess parameter uncertainty: Monte Carlo analysis and Grid-search analysis.

Monte Carlo simulations for the purpose of uncertainty estimates on parameters assumes that: 1) the model accurately describes the experimental observable and 2) the deviation between the fit and the experimental observable is similar to the deviation one would expect for many experimental replicates. During a Monte Carlo simulation, a large number of simulated data sets are generated by applying random error to each data point of the same magnitude as the deviation between the experimental observable and the best fit. Thus, these simulated data sets represent simulated replicates with error comparable to that of the experimental observable and the best fit. Each simulated time course is subjected to NLLS analysis using the set of best-fit parameters of the data set for initial guesses. The resultant best-fit parameters for each simulated data set are tabulated and a standard deviation for each parameter is calculated. This standard deviation represents an estimate of the error associated with that parameter if the experiment had been repeated as many times as data sets were simulated. This allows for estimates of error that would require unrealistic numbers of experimental replicates. It is important to note that Monte Carlo lends itself to parallel computing, and as such can be performed quickly within the MENTOR toolbox.

In addition to the determination of the standard deviation on each parameter the Monte Carlo simulation reveals information about parameter correlation. After the Monte Carlo simulation is done the experimentalist is left with thousands of estimates of the parameters that represent a simulation of thousands of experimental replicates. Construction of plots of one parameter vs. another reveal how the parameter pairs are correlated or not correlated. An analysis of this parameter correlation can aid the experimentalist in interpreting the level of confidence one has in a given parameter and protect against over interpreting the determined values.

Grid-search analysis is another method for calculating the uncertainty on fit parameters [17–19,50,51]. However, grid-searching yields additional information on the level of constraint of a given parameter and the symmetry or asymmetry in the error space. In this method, values larger and smaller than an optimized parameter value are selected. The parameter is fixed at each of its selected values and a NLLS minimization is performed to optimize all other parameters. Initial guesses for the NLLS routine are the previously optimized parameters. The tabulated chi-squared values are then used to generate a plot of chi-squared as a function of the parameter being searched. The curve is concave up with a minimum at the optimized parameter value. In principle, the curve rises to the left and right of the best-fit value, because deviating from this optimized value causes an increase in the chi-squared. The chi-squared values on the y-axis are normalized to the minimum chi-squared value and this generates a new set of numbers called the F-statistic values. An example contour from a parameter grid-search is shown in Fig. 4. The minimum of the contour is the optimized parameter value (~12.5) and the F-statistic values increase to the left and right of this minimum indicating that varying this parameter causes an increase in the chi-squared values. The minimum of the curve is 1, because the error values are normalized to the minimum chi-squared value. In MENOTR, a 68% confidence interval is the default selection and the F-critical is automatically calculated for the user-supplied data set. In Fig. 4, the F-critical line is displayed as the horizontal black dashed line. The intersection of the F-calculated value with the F-critical value generates a 68% confidence lower and upper bound for the parameter value. In MENOTR, uncertainty on fit parameters are calculated using a built-in script that has been adapted to work with the outputs from the parameter optimization routine, making the process easily executed and user friendly.

In principle a plot such as that shown in Fig. 4 should be a symmetric parabola. However, the parabola is often asymmetric. The asymmetry reveals that the error on the left of the parameter is different from the

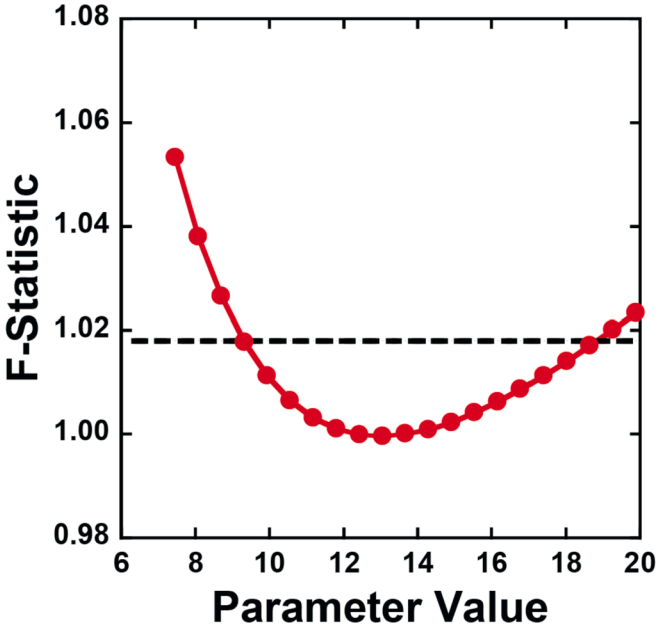


Fig. 4. Example of resultant contour generated from grid-search analysis for a given parameter. (Circles) Individual F-statistic values, (solid lines) interpolation between data points, and (Broken line) F-critical value.

error on the right. That is to say, there is asymmetric error on the parameter. Simple calculation of the standard deviation, standard error, etc. available in most fitting routines will never reveal information on asymmetric error. Knowledge of this asymmetry allows the experimentalist to better understand how well a given parameter is constrained and protects the experimentalist from overinterpreting parameters.

2.3. Case studies

2.3.1. Case Study I: Duplex DNA unwinding catalyzed by RecBCD helicase

Case Study 1 is an analysis of a set of DNA unwinding time courses that were previously published in 2004 by Lucius et al. These time courses, shown as solid traces in Fig. 5, were collected in the investigation of DNA unwinding catalyzed by *E. coli* RecBCD using a FRET based stopped-flow assay. This assay monitors the FRET signal between a Cy3 and Cy5 pair attached on either side of a nick in a duplex DNA. At time zero, the signal from Cy3 is low and the signal for Cy5 is high. Upon DNA unwinding, the two dyes are separated, resulting in an increase in Cy3 signal and a decrease in Cy5 signal. The time courses shown in Fig. 5 come from the Cy3 signal in the experiment.

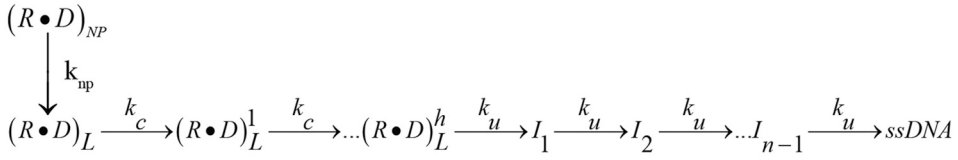
Weighted global nonlinear least squares analysis of unwinding by RecBCD was performed using Scheme 1 and Eq. (1). The resulting published fit parameters are shown in Table 1. The fit was performed for eight different lengths of duplex substrates using a fitting strategy where the parameters k_U, k_C, k_{NP}, m, and h are constrained as global fitting parameters with the same value for all duplex lengths. While A and x are local parameters with unique values for each duplex length. This fit was also subjected to Monte Carlo analysis to generate uncertainties at 68% confidence; these correspond to the error on the fit parameters in Table 1 for the previously published fits.

Table 1
Optimized parameter comparison for kinetic benchmarks I and II.

RecBCD-catalyzed DNA unwinding parameters					
Parameter	k_U (s^{-1})	k_C (s^{-1})	k_{NP} (s^{-1})	m (bp $step^{-1}$)	h (steps)
Published [2]	200 ± 40	51 ± 5	6.0 ± 0.3	3.4 ± 0.6	3.2 ± 0.3
MENOTR	185.5 ± 0.1	54.9 ± 0.6	6.49 ± 0.08	3.68 ± 0.02	3.31 ± 0.04

ClpA-catalyzed polypeptide translocation parameters					
Parameter	k_T (s^{-1})	k_d (s^{-1})	k_C (s^{-1})	k_{NP} (s^{-1})	m (aa $step^{-1}$)
Published [1]	1.39 ± 0.06	ND	0.22 ± 0.01	0.047 ± 0.001	14.0 ± 1.5
MENOTR	1.5 ± 0.2	ND	0.165 ± 0.008	0.040 ± 0.003	14 ± 1

k_U, unwinding rate constant; k_C, slow conformational change; k_{NP}, rate constant for change to productive complex; m, kinetic step size; h, number of steps with rate constant k_C; k_T, translocation rate constant; k_d, dissociation rate constant. Errors reported in this plot come from Monte Carlo analysis of 1000 simulated time courses.



Scheme 1

$$f_{RecBCD}(t) = A\mathscr{L}^{-1}\left(\frac{k_c^h k_t^{L/m}(k_{np}+s\cdot x)}{s(k_c+s)^h(k_{np}+s)(k_t+s)^{L/m}}\right) \quad (1)$$

Parameter optimization using MENOTR was executed on the same eight published RecBCD time courses using Scheme 1 and Eq. (1). Three of the eight RecBCD time courses along with the best-fit lines generated from the MENOTR optimized parameters are shown in Fig. 5. Inspection of the time courses and best-fit line indicate good agreement between the model and the data. The chi-squared value from the fit using MENOTR was found to be 367, which is ~71% smaller than the previously published fit chi-squared of 515. The MENOTR fit was found to be statistically better at a 68% confidence interval by F-statistics.

The fit parameters determined using MENOTR and the previously reported values are compared in Table 1. Interestingly, the kinetic parameters did not vary dramatically compared to published results despite finding a statistically better fit. To determine if the parameters were within error of the previously published values, a Monte Carlo analysis was performed to generate uncertainties. The uncertainties on the parameters shown in Table 1 are from the Monte Carlo simulation and indicate the 68% confidence interval. All but one of the kinetic parameters were found to be within error of the previous results. The parameter k_{NP} was found to have a value of (6.49 ± 0.08) s^{-1} from the analysis using MENOTR while its published value is (6.0 ± 0.3) s^{-1}.

A key difference in the execution of these fits is that MENOTR ran unsupervised until an optimized set of parameters was reported. In order to be confident that the lowest chi-square was determined, the previous published NLLS fit required the user to manually start the NLLS routine at different starting points, record the resultant outputs, and try new starting points. In contrast, the MENOTR analysis represents the best fit after starting from thousands of different initial guesses. This would be an intractable number of restarts when doing manually initiated NLLS. However, it is the number of restarts that one needs to have confidence that the lowest chi-squared has been found and the analysis is not simply "stuck" in a local minimum. Finally, the MENOTR analysis was performed completely unsupervised and finished in approximately 24 h on a quad-core computer. Table 1. Optimized parameter comparison for kinetic benchmarks I and II.

In this study, parameter optimizations performed with MENOTR were able to reproduce results comparable to methods that implemented only NLLS strategies. The simulated best-fit lines and the values of the kinetic parameters determined from the analysis using MENOTR agreed with both the experimental time courses and the previously published results. However, unlike previous analysis strategies, MENOTR was able to perform this fit with minimal user intervention. By automating this process three goals are achieved: 1) minimization of user bias, 2) ease of use is improved, because no user intervention is necessary throughout the optimization process, and 3) the user gains the ability to run multiple model optimizations simultaneously on computer clusters which are increasingly more accessible to researchers. The previously published analysis using NLLS was carried out using a program called *CONLIN* [52]. While robust, this program has a significant learning curve and requires a user to manually probe different initial guesses and manually

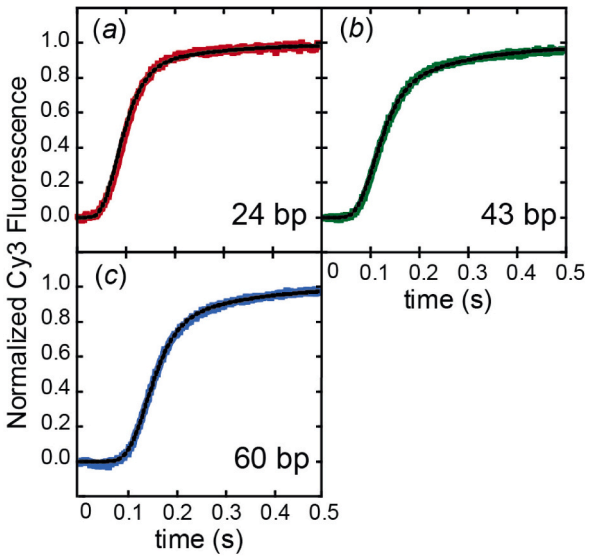


Fig. 5. MENTOR analysis of previously published single turnover RecBCD catalyzed DNA unwinding. Fitting was performed globally across eight DNA duplex lengths using Eq. (1). Parameters k_U, k_C, k_{NP}, m, and h were assigned as global parameters while A and x were local parameters for each length. Here we show three representative data sets (colored traces) of duplex lengths (a) 24 bp, (b) 43 bp, and (c) 60 bp with the corresponding best-fit simulations (black traces) based on Eq. (1) and the optimized parameters (Table 1). A corresponding figure for the original fits and analysis of this data can be found in Lucius 2004, fig. 8 [1].

record which sets of initial guesses give rise to better chi-squared values. Other programs utilizing NLLS are available, but few are able to optimize models using Laplace transform/inverse Laplace transform functions or sets of differential equations.

While the time courses analyzed in Case Study I are the time courses published in the original manuscript, the resultant parameter uncertainty values calculated from the Monte-Carlo analysis in MENOTR were smaller for all parameter values compared to the published values. A few analysis details may explain this observation. First, the previous analysis resulted in a slightly higher variance of the fit compared to the lower value obtained with MENOTR. Because the MENOTR analysis yielded a lower variance, a smaller simulated noise value was applied in the Monte Carlo simulations. Since there is less simulated error there is a resultant lower error on the parameters. Second, it should be noted that the parameter uncertainty analysis in MENOTR used 1000 Monte Carlo simulations compared to 50 in the previous published analysis. In the previously published Monte-Carlo simulation it was not possible to take advantage of parallel processing. Each simulation was done by generating time courses by numerically solving the inverse Laplace transform and then fitting those time courses by numerically solving the inverse Laplace transform. Further, each one of those cycles were done sequentially since they could not be done in parallel. Thus, doing more than fifty cycles was inordinately time consuming. This problem is solved by MENOTR since each cycle can be done in parallel. This is one explanation for the differences observed in the parameter uncertainty values. In general, it is recommended to perform hundreds of Monte Carlo simulations.

By doing Monte Carlo simulations many things can be learned about the data and the model being used to describe the data. For example, plots of one parameter vs. another can yield insight into the degree of parameter correlation. Also, the Grid search routine is another method we have built into MENOTR for error analysis. But, in addition to determining the uncertainty on a parameter, the grid search reveals information on the level of constraint on a given parameter and asymmetries that may exist in the error space. Although both Monte-Carlo and Grid Search will estimate parameter uncertainty and both techniques yield additional insights, we recommend collecting the experimental data at least three times, fitting each set of data independently, and reporting the standard error on the resultant parameters from three replicates. In our experience the parameter uncertainty estimated by replicates is a better representation of the overall reproducibility of the experimental observable. However, to yield the most insight into the model and protect oneself from overfitting or over interpretation we recommend all three, grid search, Monte Carlo, and experimental replicates.

2.3.2. *Case Study II: polypeptide translocation catalyzed by ClpA*

The second case study covers the use of MENOTR in fitting time courses describing polypeptide translocation catalyzed by *E. coli* ClpA. In 2010, Rajendar et al. developed a fluorescence stopped-flow method for studying ClpA catalyzed translocation of polypeptide substrate [1]. The assay monitors the change in fluorescence signal of fluorecein-5-maleimide as translocation occurs. When ClpA is bound to the polypeptide substrate the fluorescence is quenched. During translocation, ClpA resides on the polypeptide substrate and the fluorescence remains quenched. Upon completion of translocation, ClpA dissociates, and fluorescence is restored. This assay allows for quantitative measurements of the ClpA translocation kinetics. Translocation time courses were collected for three polypeptide lengths shown in Fig. 6 as solid colored traces. For this stopped-flow method signal can occur at every dissociation step. Thus, Scheme 2, in contrast to Scheme 1, incorporates a substrate, S, release step at each intermediate translocation step. Weighted global nonlinear least squares analysis of polypeptide translocation catalyzed by ClpA was performed using a function S*(t)* derived

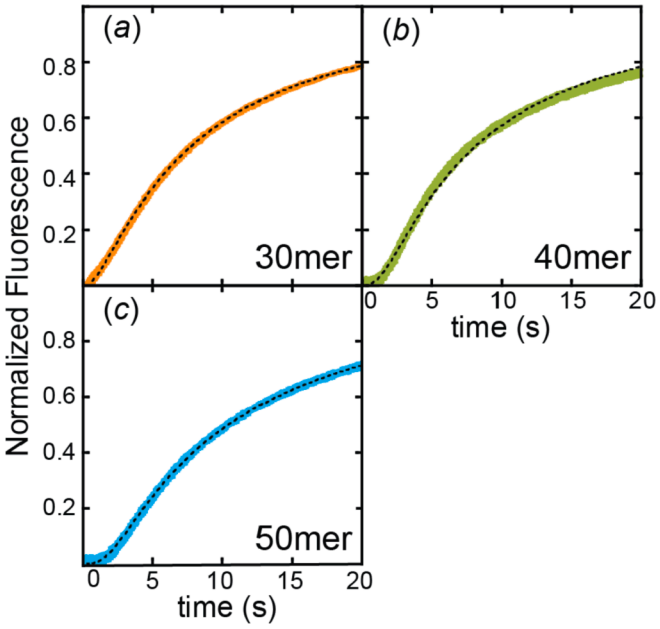
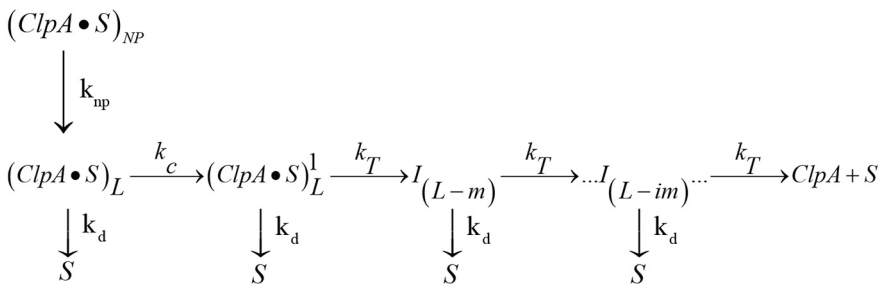


Fig. 6. MENOTR analysis of previously published fluorescence time courses for ClpA catalyzed polypeptide translocation. Translocation time courses were collected on fluorescein-SsrA 30mer, 40mer and 50mer out to 200 s. Fitting was performed globally across all three peptide lengths using Eq. (2). Parameters k_T, k_d, k_C, k_{NP}, h, m, and b were optimized globally, while A and x were optimized locally for each length. The first 20 s of the time courses are plotted here as solid traces with the dashes representing a best-fit simulation using Eq. (2) and the optimized parameters (Table 1). A corresponding figure for the original fits and analysis of this data can be found in Lucius et al. 2010, Fig. 3 [2].

from Scheme 2 to find the set of best-fit parameters shown in Table 1. The full expression of Eq. (2) can be found in Rajendar et al., 2010 [1]. The parameters were optimized using a strategy where k_T, k_{NP}, k_C, m and h were global fitting parameters, while A and x were local parameters for each time course.



Scheme 2

$$S(t) = A\mathscr{L}^{-1}S\left(s, k_T, k_d, k_c, k_{np}, m, b, x, h\right) \tag{2}$$

MENOTR was initialized using the same published model shown in Scheme 2. Best-fit simulations of the data generated using Eq. (2) are shown in Fig. 6 as black dashed traces for the three substrate lengths. The optimized parameters are tabulated in Table 1. Inspection of the time courses, much like in Case Study I, showed good agreement between the best-fit simulation and the experimental data. MENOTR was able to achieve a 50% reduction in the fit chi-squared (from 5934 to 3002) which resulted in a statistically better fit at a 68% confidence interval.

As in the first case study, the fit was found to be statically better using MENOTR, but minimal differences in the optimized kinetic parameter values were observed. This fit was also subjected to Monte Carlo analysis to generate uncertainties within 68% confidence, found in Table 1 with their corresponding parameter values. The values determined for k_T and m were both within error of their published values. The other two rate constants, k_C and k_{NP}, were just outside of error of their published values, $k_{C,NLLS} = (0.20 \pm 0.003)\ s^{-1}$ compared to $k_{C,MENOTR} = (0.165 \pm 0.008)\ s^{-1}$ and $k_{NP,NLLS} = (0.045 \pm 0.0005)\ s^{-1}$ compared to $k_{NP,MENOTR} = (0.040 \pm 0.003)\ s^{-1}$.

In this investigation, MENOTR reproduced optimized parameters comparable to previous published investigations using only nonlinear least squares. In both cases, MENOTR was able to obtain resultant parameters with minimal user intervention and a lower chi-squared was found.

2.3.3. Case Study III: thermodynamics of ligand binding macromolecule

Unique fitting challenges emerge when using the models that describe ligand binding to macromolecules. Indeed, a simple one-to-one binding interaction is not terribly difficult to describe since the model contains a single equilibrium constant. However, upon departure from the realm of one-to-one binding two major problems arise. If the goal is to elucidate two or more non-identical equilibrium constants, then parameter correlation becomes the first problem. The second problem that emerges is that the equations that are written down for extent of binding (ligand bound per total macromolecule) are a function of the free ligand concentration. However, the experimentally known quantity is the total amount of ligand added to the system. Only in rare experimental cases does the experimentalist directly measure the free ligand concentration, e.g. equilibrium dialysis. In continuous titration experiments like ITC or fluorescence titrations, the experimentalist knows how much total ligand is added to the vessel but does not know how it parses into bound versus free ligand. The parameter that defines how the system parses into bound ligand and free ligand is the parameter that is being sought, the binding equilibrium constant(s).

For a simple one-to-one binding reaction, the latter problem is straightforward to overcome. The equilibrium scheme shown in Eq. (3) is described by the binding equation given by Eq. (4).

$$M + x_f \overset{K_a}{\rightleftharpoons} Mx \tag{3}$$

$$\frac{[Mx]}{[M]_T} = \overline{X} = \frac{K_a x_f}{1 + K_a x_f} \tag{4}$$

In Eqs. (3) and (4), M is a single site macromolecule, x_f is the free ligand concentration, Mx is the bound state, $[M]_T$ is the total macromolecule concentration and K_a is the binding equilibrium constant. To derive an equation in terms of an independent variable that the experimentalist can control one needs the conservation of mass equation given by Eq. (5).

$$x_t = x_f + \sum Mx_i = x_f + \frac{[Mx]}{[M]_T}[M]_T \tag{5}$$

Where the summation of Mx_i represents the summation of all the bound states. For Eq. (4) the only bound state is Mx, which can be determined by multiplying the extend of binding by the total macromolecule concentration as shown in the right hand side of Eq. (5). The extent of binding, $\overline{X}$, multiplied by the total macromolecule concentration represents the summation of all the bound states. Consequently, the conservation of mass equation given by Eq. (6) is general for any number of binding sites on a macromolecule that does not change its assembly state during the titration. This would include a monomer that does not oligomerize during the course of the titration or a larger order oligomer that does not either dissociate or further oligomerize during the course of the titration.

$$x_t = x_f + \overline{X}[M]_T \tag{6}$$

By combining Eqs. (4) and (6) one will arrive at

$$\overline{X} = \frac{K_a\left(x_T - \overline{X}[M]_T\right)}{1 + K_a\left(x_T - \overline{X}[M]_T\right)} \tag{7}$$

Notice that, as written, Eq. (7) is implicit in $\overline{X}$. However Eq. (7) can be rearranged to arrive at the explicit equation given in Eq. (8), which is the well-known Langmuir isotherm expressed in terms of the dissociation equilibrium constant, $K_d = K_a^{-1}$.

$$\overline{X} = \frac{(K_d + x_T + M_T) - \sqrt{(K_d + x_T + M_T)^2 - 4M_T x_T}}{2M_T} \tag{8}$$

Eq. (8) is now a function of both the total ligand concentration and the total macromolecule concentration, two variables that the experimentalist is in control of changing. Importantly, Eq. (8) has the form of

the quadratic equation because the derivation of Eq. (8) from Eq. (7) requires finding the roots of a second order polynomial. Thus, the two non-identical binding sites model given by Eq. (9) will require finding the roots of a third order polynomial when one attempts to derive a function of total ligand and total macromolecule using Eq. (6).

$$\overline{X} = \frac{K_1 x_f + 2K_1K_2x_f^2}{1 + K_1 \cdot x_f + K_1K_2x_f^2} \tag{9}$$

Indeed, there are solutions to third order polynomials. However, they are cumbersome to use in a fitting routing. Moreover, there are no explicit solutions for the fourth order polynomial that emerges once a third binding site is added. An alternate strategy is to perform implicit fitting. In the two site binding example Eqs. (6) and (9) are coded into the fitting routine and the data are fit using the total ligand concentration and the total macromolecule concentration as the independent variable. Behind the scenes, the code is solving the implicit equation given by Eq. (10).

$$\overline{X} = \frac{K_1\left(x_T - \overline{X}[M]_T\right) + 2K_1K_2\left(x_T - \overline{X}[M]_T\right)^2}{1 + K_1\left(x_T - \overline{X}[M]_T\right) + K_1K_2\left(x_T - \overline{X}[M]_T\right)^2} \tag{10}$$

To accomplish this the code invokes a numerical root finder to find the correct root of the third order polynomial that emerges from Eq. (10).

To test the implicit fitting routine in MENOTR we simulated binding isotherms from a simple one-to-one binding model using Eq. (8) at three different total macromolecule concentrations. Random gaussian white noise was added to each of the data points to simulate experimental error. The solid points in Fig. 7a represents the simulated data points with simulated uncertainty. The simulated isotherms were analyzed using Eq. (7) and the implicit fitting routine in MENOTR. The solid lines in Fig. 7a represent the best-fit lines from the optimization. Table 2 shows that we used $K_a = 1 \times 10^5\ M^{-1}$ to generate the simulated isotherms and the same value was returned from the analysis. This observation indicates that the implicit fitting routine in MENOTR is working as expected at least for a simple model.

As discussed, implicit fitting is not required for a simple one-to-one interaction. Thus, to further test the implicit fitting routine in MENOTR we simulated binding isotherms for a two-site binding model given by Eq. (9). Data points were again simulated for three total macromolecule concentrations and simulated uncertainty was added. The simulated data points are shown in Fig. 7b. In this case, data points were simulated using Eq. (6) and Eq. (9) in Micromath Scientist (Micromath, St. Louis MO). Micromath Scientist was used for two reasons. First, it is the only commercially available software that we are aware of that can perform implicit fitting or simulations from implicit equations. Secondly, Scientist was used to reduce bias in the analysis. Scientist was used to generate the simulated data and MENOTR in MATLAB was used in the analysis.

For this test three isotherms were simulated with $K_1 = 1 \times 10^5$ and $K_2 = 1 \times 10^6$, see Table 2. The isotherms were subjected to MENOTR analysis using Eq. (6) and Eq. (9) and the resultant parameters are shown in Table 2. As can be seen in Fig. 7b the data points are well described by the model and the resultant parameters are within error of the simulated parameters. Thus, we conclude that MENOTR and the implicit fitting routine is well equipped to solve implicit models.

Here we have shown that MENOTR is able to extract known thermodynamic parameters out of binding data for thermodynamic binding models that require implicit fitting strategies. Fit parameters were

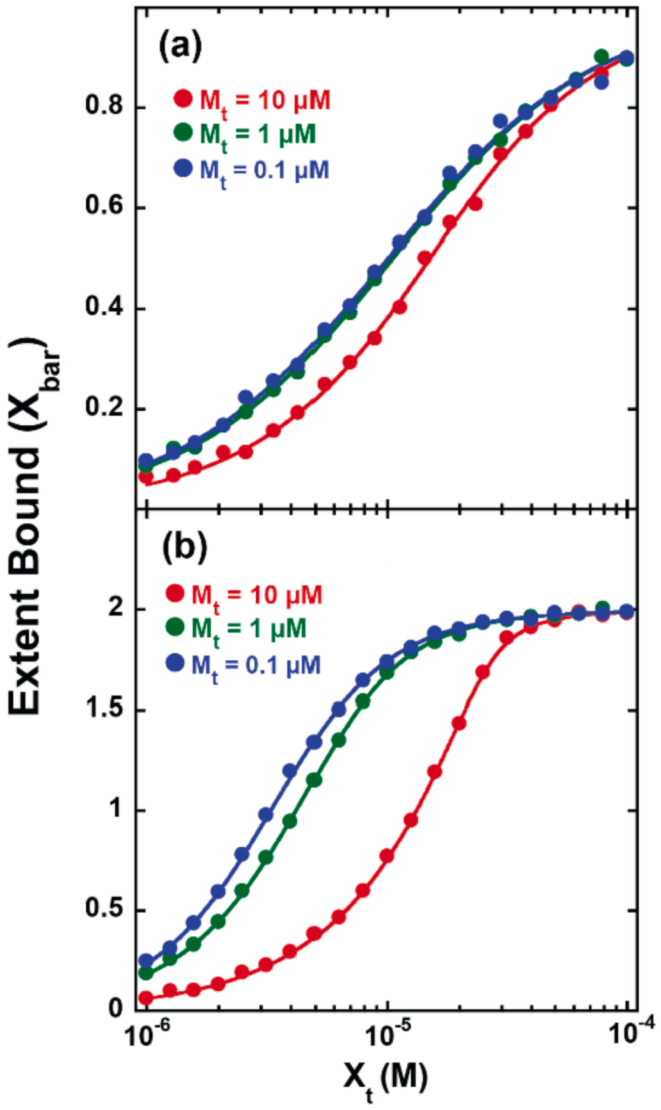


Fig. 7. MENOTR analysis of simulated thermodynamic data for two classic cases of ligand binding to macromolecule using implicit fitting strategies. (a) n-independent and identical: Simulated data were generated using Eq. (8) and Gaussian white noise was added to simulate experimental error. A signal to noise ratio of 30 was used for the experimental noise. The data were fit implicitly using Eq. (4) and Eq. (6) in MENOTR. (b) n-independent non-identical: Data were simulated using a 2-site model with Eq. (6) and Eq. (9) in Micromath Scientist. Gaussian white noise was added to each data set to simulate experimental error using the function AWGN in MATLAB. A signal to noise ratio of 40 was used for the experimental noise. The data were fit implicitly in MENOTR using Eq. (6) and Eq. (9).

Table 2
Optimized parameter comparison for kinetic benchmark III: Thermodynamic macromolecule and ligand binding parameters.

(a) One-to-one binding	
Parameter	K
Simulation values	$1.00 \times 10^5\ M^{-1}$
MENOTR fit results	$(1.00 \pm 0.01) \times 10^5\ M^{-1}$

(b) n-independent non-identical binding		
Parameter	K_1	K_2
Simulation values	$1.00 \times 10^5\ M^{-1}$	$1.00 \times 10^6\ M^{-1}$
MENOTR fit results	$(9.9 \pm 0.2) \times 10^4\ M^{-1}$	$(1.0 \pm 0.2) \times 10^6\ M^{-1}$

(a) Simulations were generated using the Langmuir isotherm, Eq. (8). White Gaussian error was added to each data set with s/n = 40. Simulated data sets were implicitly fit globally across three total macromolecule concentrations in MENOTR using Eq. (4) and Eq. (6). (b) Simulations were generated in Micromath Scientist using Eq. (6) and Eq. (9). White Gaussian error was added to each data set with s/n = 40. Simulated data sets were implicitly fit in MENOTR using Eq. (6) and Eq. (9).

reproduced with less than 1.2% error of the known simulation values.

2.4. Limitations of MENOTR

MENOTR is a useful model optimization tool but has some limitations. One initial drawback is that MENOTR is currently only available to users who have MATLAB. We acknowledge that some users may not have access to MATLAB, but we are currently unaware of open-source scripts for several critical features present in MENOTR. An additional limitation of optimizations using a GA/NLLS (consequently MENOTR also) is that convergence on the most optimized set of parameters is not guaranteed. Theoretically, if the population size in the GA is infinitely large then it is possible to guarantee that the optimized parameters are the best possible set of parameters. However, the calculation time would be, consequently, infinitely long. Thus, it is reasonable to say that the GA/NLLS compared to NLLS alone is more likely to result in the best set of optimized parameters, but it cannot be guaranteed. Figuring out the ideal population size for a given model optimization is difficult and requires a user to test different population size values. We have provided general recommendations, but we encourage users to modify the code to best execute their model optimization.

2.5. When to use vs. when not to use MENOTR

Like all tools, MENOTR is designed to be used to tackle specific types of optimization problems. Consider the simplest case where you have a linear data set, and you want to fit using the linear equation $y = m\,x + b$. In this model the parameters m and b will be optimized to describe the experimental data. In this case the initial guess value is not of great concern, because the values are not correlated and are reasonably easy to optimize. In this situation MENOTR would be much slower compared to standard NLLS programs found in most data analysis packages. Thus, in general we do not encourage using MENOTR to fit simple models with uncorrelated parameters as the optimization attributes of MENOTR are superfluous and a waste of time for simple optimizations.

In contrast, the models presented in this manuscript contain correlated parameters. The initial guess dependence makes model optimizations using standard NLLS approaches more difficult. To overcome these obstacles a user must selectively choose different initial guess values and monitor how the chi-squared is impacted using the chosen initial guess values. This method of manually tabulating different initial guesses and different optimized parameters is certainly possible but is labor intensive and requires intense focus. MENOTR performs a similar process except the computer automatically tabulates the impact of different initial guesses and the resultant optimized parameters. From our experience MENOTR does not necessarily arrive at the optimized parameters faster compared to a researcher experienced with NLLS algorithms. However, unlike human NLLS users, MENOTR is able to execute multiple model optimizations simultaneously. This allows a researcher to probe multiple models simultaneously and additionally allows a researcher to have multiple models tested while they are otherwise preoccupied. Thus, in summary while MENOTR isn't necessarily faster when comparing the optimization of a single model, it is certainly faster when comparing the time taken to optimize multiple models.

3. Conclusion

We have developed a novel MATLAB optimization toolbox, MENOTR, that utilizes a hybrid genetic and nonlinear least squares algorithm. This toolbox optimizes sets of parameters describing various types of chemical data and is designed for users with limited programming experience. The toolbox was used in three benchmark investigations. Two of the benchmarks involved reanalyzing previously published kinetic data demonstrating MENOTR's capability to determine parameters from complex models with highly correlated parameters. The third benchmark demonstrated MENOTR's capability to perform implicit fitting. Here we have shown MENOTR to be a useful tool in the analysis of complex kinetic and thermodynamic data. With the use of MENOTR, rigorous estimates of parameter values and corresponding errors are accessible for models with high degrees of correlation in parameters and numerous local minima. Most importantly, MENOTR solves the problem of knowing when one has chosen an adequate number of initial guesses to have confidence that the final best fit represents the absolute minima in the error space. Moreover, this can be achieved with minimal user intervention and takes advantage of modern parallel processing capabilities. This toolbox is stored on github.com and is freely available (https://github.com/ZachIngram/2021-MENOTR). A new user tutorial can be found in the supplemental information.

Author contributions

N.W.S. and Z.M.I. designed, wrote, and troubleshot the MENOTR toolbox and wrote the manuscript equally. A.L.L. revised the toolbox and wrote the manuscript. D.A.S. wrote the manuscript. All authors read and approved the final manuscript.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The MENOTR scripts can be found at https://github.com/ZachIngram/2021-MENOTR

Acknowledgements

We would like to thank the Lucius and Schneider labs, for their critical discussions and testing of the toolbox. We would especially like to thank the initial contributions of Frank Appling. His MATLAB codes were the initial starting point for this project.

This work was supported by the National Science Foundation (grant MCB-1412624 to A.L.L and MCB-1817749 to A.L.L.) and the National Institutes of Health (grant R01 GM084946 to D.A.S. and R35 GM140710 to D.A.S.)

Computational work done in data analysis was performed using the UAB HPC Cheaha, which is supported in part by the National Science Foundation under Grants No. OAC-1541310 to the University of Alabama at Birmingham, and the Alabama Innovation Fund. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation or the University of Alabama at Birmingham.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.bpc.2021.106682.

References

[1] B. Rajendar, A.L. Lucius, Molecular mechanism of polypeptide translocation catalyzed by the *Escherichia coli* ClpA protein translocase, J. Mol. Biol. 399 (5) (2010) 665–679.

[2] A.L. Lucius, C. Jason Wong, T.M. Lohman, Fluorescence stopped-flow studies of single turnover kinetics of *E. coli* RecBCD helicase-catalyzed DNA unwinding, J. Mol. Biol. 339 (4) (2004) 731–750.

[3] P. McConnell, M. Mekel, A.G. Kozlov, O.L. Mooren, T.M. Lohman, J.A. Cooper, Comparative analysis of CPI-motif regulation of biochemical functions of actin capping protein, Biochemistry 59 (11) (2020) 1202–1215.

[4] T. Li, A.L. Lucius, Examination of the polypeptide substrate specificity for *Escherichia coli* ClpA, Biochemistry 52 (29) (2013) 4941–4954.
[5] J.M. Miller, J. Lin, T. Li, A.L. Lucius, *E. coli* ClpA catalyzed polypeptide translocation is allosterically controlled by the protease ClpP, J. Mol. Biol. 425 (15) (2013) 2795–2812.
[6] T. Li, C.L. Weaver, J. Lin, E.C. Duran, J.M. Miller, A.L. Lucius, *Escherichia coli* ClpB is a non-processive polypeptide translocase, Biochem. J. 470 (1) (2015) 39–52.
[7] C.K. Hayne, H. Yumerefendi, L. Cao, J.W. Gauer, M.J. Lafferty, B. Kuhlman, D. A. Erie, S.B. Neher, We FRET so you don't have to: new models of the lipoprotein lipase dimer, Biochemistry 57 (2) (2018) 241–254.
[8] J.D. Marsee, A. Ridings, T. Yu, J.M. Miller, *Mycobacterium tuberculosis* ClpC1 N-terminal domain is dispensable for adaptor protein-dependent allosteric regulation, Int. J. Mol. Sci. 19 (11) (2018).
[9] S.P. Singh, A. Soranno, M.A. Sparks, R. Galletto, Branched unwinding mechanism of the Pif1 family of DNA helicases, Proc. Natl. Acad. Sci. U. S. A. 116 (49) (2019) 24533–24541.
[10] W. Kress, H. Mutschler, E. Weber-Ban, Both ATPase domains of ClpA are critical for processing of stable protein structures, J. Biol. Chem. 284 (45) (2009) 31441–31452.
[11] X. Ye, J. Lin, L. Mayne, J. Shorter, S.W. Englander, Structural and kinetic basis for the regulation and potentiation of Hsp104 function, Proc. Natl. Acad. Sci. U. S. A. 117 (17) (2020) 9384–9392.
[12] W. Cao, M.M. Coman, S. Ding, A. Henn, E.R. Middleton, M.J. Bradley, E. Rhoades, D.D. Hackney, A.M. Pyle, E.M. De La Cruz, Mechanism of Mss116 ATPase reveals functional diversity of DEAD-box proteins, J. Mol. Biol. 409 (3) (2011) 399–414.
[13] K.L. Henderson, D.K. Boyles, V.H. Le, E.A. Lewis, J.P. Emerson, Chapter eleven - ITC methods for assessing buffer/protein interactions from the perturbation of steady-state kinetics: a reactivity study of homoprotocatechuate 2,3-dioxygenase, in: A.L. Feig (Ed.), Methods in Enzymology, Academic Press, 2016, pp. 257–278.
[14] Y. Hao, J.P. England, L. Bellucci, E. Paci, H.C. Hodges, S.S. Taylor, R.A. Maillard, Activation of PKA via asymmetric allosteric coupling of structurally conserved cyclic nucleotide binding domains, Nat. Commun. 10 (1) (2019) 3984.
[15] J. Mitra, T. Ha, Streamlining effects of extra telomeric repeat on telomeric DNA folding revealed by fluorescence-force spectroscopy, Nucleic Acids Res. 47 (21) (2019) 11044–11056.
[16] C.E. Scull, A.M. Clarke, A.L. Lucius, D.A. Schneider, Downstream sequence-dependent RNA cleavage and pausing by RNA polymerase I, J. Biol. Chem. 295 (5) (2020) 1288–1299.
[17] C.E. Scull, Z.M. Ingram, A.L. Lucius, D.A. Schneider, A novel assay for RNA polymerase I transcription elongation sheds light on the evolutionary divergence of eukaryotic RNA polymerases, Biochemistry. 58 (2019) 2116–2124.
[18] F.D. Appling, D.A. Schneider, A.L. Lucius, Multisubunit RNA polymerase cleavage factors modulate the kinetics and energetics of nucleotide incorporation: an RNA polymerase I case study, Biochemistry 56 (42) (2017) 5654–5662.
[19] F.D. Appling, A.L. Lucius, D.A. Schneider, Transient-state kinetic analysis of the RNA polymerase I nucleotide incorporation mechanism, Biophys. J. 109 (11) (2015) 2382–2393.
[20] M.K. Shinn, A.G. Kozlov, T.M. Lohman, Allosteric effects of SSB C-terminal tail on assembly of *E. coli* RecOR proteins, Nucleic Acids Res. 49 (4) (2021) 1987–2004.
[21] Y.A. Ordabayev, B. Nguyen, A.G. Kozlov, H. Jia, T.M. Lohman, UvrD helicase activation by MutL involves rotation of its 2B subdomain, Proc. Natl. Acad. Sci. U. S. A. 116 (33) (2019) 16320–16325.
[22] M.L. Johnson, S.G. Frasier, [16] Nonlinear least-squares analysis, in: Methods in Enzymology, Academic Press, 1985, pp. 301–342.
[23] C.A. Brambley, J.D. Marsee, N. Halper, J.M. Miller, Characterization of mitochondrial YME1L protease oxidative stress-induced conformational state, J. Mol. Biol. 431 (6) (2019) 1250–1266.
[24] M.L. Johnson, Why, when, and how biochemists should use least squares, Anal. Biochem. 206 (2) (1992) 215–225.
[25] M. Johnson, Parameter correlations while curve fitting 321, 2000, pp. 424–446.
[26] A.L. Lucius, A. Vindigni, R. Gregorian, J.A. Ali, A.F. Taylor, G.R. Smith, T. M. Lohman, DNA unwinding step-size of *E. coli* RecBCD helicase determined from single turnover chemical quenched-flow kinetic studies, J. Mol. Biol. 324 (3) (2002) 409–428.
[27] M.L. Johnson, L.M. Faunt, [1] Parameter estimation by least-squares methods, in: Methods in Enzymology, Academic Press, 1992, pp. 1–37.
[28] L.V. Stephen Boyd, Convex Optimization, 2004.
[29] C. Loehle, Robust parameter estimation for nonlinear models, Ecol. Model. 41 (1) (1988) 41–54.
[30] K.A. Johnson, 1 Transient-state kinetic analysis of enzyme reaction pathways, in: D.S. Sigman (Ed.), The Enzymes, Academic Press, 1992, pp. 1–61.
[31] F. Glover, Future paths for integer programming and links to artificial intelligence, Comput. Oper. Res. 13 (5) (1986) 533–549.
[32] N.R. Draper, H. Smith, Applied Regression Analysis, Wiley, New York, 1966.
[33] I.H. Osman, G. Laporte, Metaheuristics: a bibliography, Ann. Oper. Res. 63 (5) (1996) 511–623.
[34] S. Forrest, Genetic algorithms: principles of natural selection applied to computation, Science 261 (5123) (1993) 872–878.
[35] J. Maddox, Genetics helping molecular dynamics, Nature 376 (6537) (1995) 209.
[36] M.J. Blommers, C.B. Lucasius, G. Kateman, R. Kaptein, Conformational analysis of a dinucleotide photodimer with the aid of the genetic algorithm, Biopolymers 32 (1) (1992) 45–52.
[37] D.B. McGarrah, R.S. Judson, Analysis of the genetic algorithm method of molecular conformation determination, J. Comput. Chem. 14 (11) (1993) 1385–1395.
[38] H. Mühlenbein, M. Schomisch, J. Born, The parallel genetic algorithm as function optimizer, Parallel Comput. 17 (6) (1991) 619–632.
[39] S. MirjaliliJin, J. Dong, A. Lewis, Nature-Inspired Optimizers, Springer, Berlin Heidelberg, New York, NY, 2019.
[40] M. Mitchell, An Introduction to Genetic Algorithms, MIT Press, Cambridge, Mass, 1996.
[41] S.N. Sivanandam, S.N. Deepa, Introduction to genetic algorithms, Springer, Berlin; New York, 2007.
[42] D.A. Coley, An introduction to genetic algorithms for scientists and engineers, World Scientific, Singapore; River Edge, NJ, 2010.
[43] N.W. Scull, A.L. Lucius, Kinetic analysis of AAA+ translocases by combined fluorescence and anisotropy methods, Biophys. J. 119 (2020) 1335–1350.
[44] M. Straume, M.L. Johnson, Monte Carlo method for determining complete confidence probability distributions of estimated model parameters, Methods Enzymol. 210 (1992) 117–129.
[45] C.W. Lee, Y.C. Shin, Construction of fuzzy systems using least-squares method and genetic algorithm, Fuzzy Sets Syst. 137 (3) (2003) 297–323.
[46] A.D. Olinsky, J.T. Quinn, P.M. Mangiameli, S.K. Chen, A genetic algorithm approach to nonlinear least squares estimation, Int. J. Math. Educ. Sci. Technol. 35 (2) (2004) 207–217.
[47] G.R. Liu, X. Han, K.Y. Lam, A combined genetic algorithm and nonlinear least squares method for material characterization using elastic waves, Comput. Methods Appl. Mech. Eng. 191 (17) (2002) 1909–1921.
[48] A.I. Ferreiro, M. Rabaçal, M. Costa, A combined genetic algorithm and least squares fitting procedure for the estimation of the kinetic parameters of the pyrolysis of agricultural residues, Energy Convers. Manag. 125 (2016) 290–300.
[49] Z.-J. Yang, T. Hachino, T. Tsuji, On-line identification of continuous time-delay systems combining least-squares techniques with a genetic algorithm, Int. J. Control. 66 (1) (1997) 23–42.
[50] F.D. Appling, C.E. Scull, A.L. Lucius, D.A. Schneider, The A12.2 subunit is an intrinsic destabilizer of the RNA polymerase I elongation complex, Biophys. J. 114 (11) (2018) 2507–2515.
[51] F.D. Appling, A.L. Lucius, D.A. Schneider, Quantifying the influence of 5'-RNA modifications on RNA polymerase I activity, Biophys. Chem. 230 (2017) 84–88.
[52] C. Fleury, CONLIN: an efficient dual optimizer based on convex approximation concepts, Struct. Optim. 1 (2) (1989) 81–89.