Published in final edited form as:

Comput Chem Eng. 2014 April 1; 63(17): 152–158. doi:10.1016/j.compchemeng.2014.01.006.

Parameter estimation in stochastic chemical kinetic models using derivative free optimization and bootstrapping

Rishi Srivastava^{a)} and James B. Rawlings^{b)}

Department of Chemical and Biological Engineering, University of Wisconsin-Madison

Abstract

Recent years have seen increasing popularity of stochastic chemical kinetic models due to their ability to explain and model several critical biological phenomena. Several developments in high resolution fluorescence microscopy have enabled researchers to obtain protein and mRNA data on the single cell level. The availability of these data along with the knowledge that the system is governed by a stochastic chemical kinetic model leads to the problem of parameter estimation. This paper develops a new method of parameter estimation for stochastic chemical kinetic models. There are three components of the new method. First, we propose a new expression for likelihood of the experimental data. Second, we use sample path optimization along with UOBYQA-Fit, a variant of Powell's unconstrained optimization by quadratic approximation, for optimization. Third, we use a variant of Efron's percentile bootstrapping method to estimate the confidence regions for the parameter estimates. We apply the parameter estimation method in an RNA dynamics model of *E. coli*. We test the parameter estimates obtained and the confidence regions in this model. The testing of the parameter estimation method demonstrates the efficiency, reliability, and accuracy of the new method.

1. Introduction

Intrinsic noise with its ability to drive processes of significant biological interest and stochastic chemical kinetic models with their ability to explain intrinsic noise have seen significant attention recently [32, 2, 15, 36, 4, 41, 29]. An exhibit of intrinsic noise is when similar cells give rise to markedly different behaviors. Stochastic chemical kinetic model, which is known as continuous time discrete jump Markov chain in Mathematics and Probability theory, models such similar cells as realizations of a Markov chain. One useful tool for understanding these models is the chemical master equation, which describes the evolution of the probability density of the system. The solution of the master equation is computationally tractable only for simple systems. Rather, approximation techniques such as finite state projections [34] or stochastic simulation algorithm (SSA) [18, 19] are employed

© 2014 Elsevier Ltd. All rights reserved.

^{a)}rishisrivastava1984@gmail.com

^{b)}rawlings@engr.wisc.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.

to reconstruct a system's probability distribution and statistics (usually the mean and variance). Applying these techniques to solve models of biological processes leads to significant improvements in our understanding of intrinsic noise and its effect on cellular behavior.

With the advent of fluorescence microscopy [30], gene expression data at the protein level has become available. Also, Golding et al. [20, 21] have quantified the mRNA level data using fluorescence microscopy. The availability of these data, along with stochastic chemical kinetic models for the experimental systems naturally gives rise to the need to develop parameter estimation methods.

Unlike ordinary differential equation based models, for which fairly robust and efficient parameter estimation techniques exist, there are not many such techniques for stochastic chemical kinetic models. Boys et al. [5] propose generating many samples of the full master equation consistent with the given measurement. They then use Markov chain Monte Carlo to obtain the posterior distribution of the parameter. The first step is computationally intractable for the models of interest here. Golightly et al. [22] use the Fokker-Planck approximation of the master equation. This diffusion approximation is not generally applicable in discrete stochastic chemical kinetics. Tian et al. [48] express the likelihood $p(y|\theta)$ as a product of transition densities $p(y|\theta) = \prod_{i=1}^n p(y_{i+1}|y_i, \theta)$. Each $p(y_{i+1}|y_i, \theta)$ is evaluated using 5000 SSA simulations. A genetic algorithm is used to maximize $p(y|\theta)$. This procedure is computationally inefficient because 5000 SSA simulations are used for each transition, and we expect many transitions in the experiments of interest. Reinker et al. [42] calculate the likelihood analytically using an artificial maximum number of reactions that can occur within a given time interval. They use a quasi-newton method to maximize the likelihood. The assumption about the maximum number of reactions is restrictive and unrealistic.

Poovathingal et al. [39] propose to evaluate the likelihood using the solution to the master equation. Their proposed function is not the likelihood, but some other merit function. They estimate the solution of the master equation by SSA simulations. This is computationally intensive and requires a binning strategy. They use directed evolution to optimize. Henderson et al. [26] replace stochastic chemical kinetic model with a statistical model, and use the statistical model for obtaining parameter estimates. The replacement with the statistical model makes their method limited to cases where such a replacement is accurate and possible. Indeed, the general replacement problem is more difficult than the problem of parameter estimation problem without replacement. Wang et al. [50] propose a stochastic gradient descent method that requires generating many samples of the master equation consistent with the measurement. They then use reversible jump Markov chain Monte Carlo to obtain the posterior distribution of the parameter. The first step is computationally intractable for the models of interest here.

We present here a novel method of parameter estimation in stochastic chemical kinetic models. As noted by Wets [49, 11], the method developed here employs tight integration of statistical sampling and mathematical optimization. The method is based on constructing a new expression for the likelihood of the data. The likelihood expression is in the form of an

expectation of a smooth function of the data, model parameters, and a regularization parameter. We then use the sample path method [27, 28, 38, 44, 46, 16, 23, 24, 37, 45] to approximate the likelihood, which is in the form of an expected value, to a sample average. The basic idea behind the sample path method is that in calculating the objective function, i.e., the likelihood, we use common random numbers. Since the gradient and Hessian of the likelihood are hard to estimate both due to the requirement of computation and the inaccuracy in their estimation, we pursue a derivative free optimization method, UOBYQA [40]. For some other derivative free methods, see references [8, 7, 31, 6, 43]. We adapt UOBYQA to UOBYQA-Fit that suits the optimization problem of interest here. Unlike [9], who assume that the objective function at the points of the fitting set are components of a multivariate normal to obtain the number of sample paths for different iterations, we keep the number of sample paths fixed.

Point estimates of the parameter alone are not useful when one is dealing with a stochastic model. Confidence region around the point estimate tells us the precision of the point estimate and how much information is contained in the experimental data. We discuss in Section 5 that neither finite sample distribution nor asymptotic distribution of parameter estimates are obtainable. Bootstrap estimation of confidence regions becomes useful when one does not have much information about the distribution of the point estimates. We use a variant of Efron's percentile method [12, 14, 13] to estimate the parameter confidence region. Along with estimation of confidence region, we test the quality of this confidence region. The successful testing of the confidence region is a reliable indicator that the UOBYQA-Fit along with the variant of the percentile method solves the parameter estimation problem reliably and accurately.

This paper is arranged as follows. In section 2, we describe the stochastic chemical kinetic modeling framework. In section 3, we provide the expression of the likelihood of the data and describe the optimization problem. Section 4 describes the UOBYQA-Fit algorithm and provides the pseudo-code for it. Section 5 describes the bootstrapping confidence estimation algorithm and provides the pseudo code for both the confidence region estimation algorithm and the verification of the confidence region algorithm. An application of the technique described in this paper is given in section 6. Finally, section 7 discusses the conclusions of this paper and summarizes the contribution.

2. The stochastic chemical kinetic model

The dynamic state of a well stirred mixture of $N_s - 1$ species $\{S_1, S_2, \dots, S_{N_s}\}$ under the influence of $M - 1$ reaction channels $\{R_1, R_2, \dots, R_M\}$ can be specified as

$$\mathbf{X}(t) = [X_1(t), X_2(t), \dots, X_{N_s}(t)]'$$

in which

$$X_i(t) = \text{The population of species } i \text{ at time } t, \quad \forall i \in \{1, 2, \dots, N_s\}$$

A probability density $P(x, t)$ for the random variable $\mathbf{X}$ can be defined as the probability of system being in state x at time t , where we note that even though time t is a continuous variable the random variable $\mathbf{X}$ takes on finitely many or countably infinite values. The underlying governing equation for $P(x, t)$ is a linear first order differential equation known as master equation which can be written as :

$$\frac{dP(x, t)}{dt} = \sum_{j=1}^M k_j a_j(x - \nu_j, t) P(x - \nu_j) - k_j a_j(x) P(x, t) \quad (1)$$

In equation (1), ν_j is stoichiometric vector of j th reaction of the reaction network. If we had a method whereby we could solve explicitly the time evolution of $P(x, t)$ for all x in the state space of the biological network, we would be done. However, solving this system of ODEs in equation (1) either analytically or numerically is challenging even for moderately large state-space systems. There is a method [17] available to generate realization of the system described by master equation, instead of solving for the time evolution of probabilities of different states explicitly. This method is also known as Gillespie algorithm and usually this is the preferred way to simulate complex biological networks.

3. The negative log likelihood minimization problem

To obtain the parameter estimates, we pursue likelihood maximization. The likelihood of the data is poorly conditioned, and to improve the conditioning we formulate the standard equivalent problem of minimizing the negative log likelihood. In Appendix we show that the exact likelihood of the experimental data, composed of m replicates $y = \{y_1, y_2, \dots, y_m\}$, is given by

$$L(y|\theta) = \lim_{R \rightarrow 0} \frac{1}{(2\pi)^{mn_d/2} |R|^{m/2}} \prod_{j=1}^m \mathbb{E}_{\theta} [e^{-(1/2)(y_j - x)' R^{-1} (y_j - x)}] \quad (2)$$

in which x is the random vector of the population of species coming from the SSA simulation of the model, m is the number of replicates, n_d is the number of sample points in each replicate, R is an $n_d \times n_d$ positive definite matrix representing both measurement noise and smoothing, and θ is the vector of parameters to be estimated. It is possible that likelihood defined in equation (2) is non-convex and multimodal in such cases several different starting points can be used in the method described in this paper. The use of several different starting points reduces the possibility of the method terminating at a local optima. For any positive definite value of R , the right hand side of equation (2) after the limit is an approximation to the exact likelihood. For such a value of $R \rightarrow 0$, using sample mean as the estimator for the expectation in equation (2), we obtain an estimator of $L(y|\theta)$

$$\hat{L}(y|\theta, N) = \frac{1}{(2\pi)^{mn_d/2} |R|^{m/2} N^m} \prod_{j=1}^m \sum_{i=1}^N e^{-(1/2)(y_j - x_i(\theta))' R^{-1} (y_j - x_i(\theta))} \quad (3)$$

As $N \rightarrow \infty$ and $R \rightarrow 0$, the estimated likelihood, $\hat{L}(y|\theta, N)$, approaches the exact likelihood $L(y|\theta)$. The estimate of negative log likelihood which we use as objective function for minimization is:

$$\hat{\phi}(\theta, N) = -\log \hat{L}(y|\theta, N) = -\sum_{j=1}^m \log \left[\frac{1}{N(2\pi)^{n_d/2} |R|^{1/2}} \sum_{i=1}^N e^{-(1/2)(y_j - x_i(\theta))' R^{-1} (y_j - x_i(\theta))} \right] \quad (4)$$

The parameter estimates are then given by the minimization of the negative log likelihood

$$\hat{\theta} = \arg \min_{\theta} \hat{\phi}(\theta, N) \quad (5)$$

4. The algorithm for the negative log likelihood minimization problem

We rewrite equation (5) to denote the dependence on the randomness ω , which is parametrized by all the random numbers used to generate N SSA sample paths

$$\hat{\theta}(N, \omega) = \arg \min_{\theta} \hat{\phi}(\theta, N, \omega) \quad (6)$$

To use sample path optimization, we freeze the N streams of random numbers and use these streams repeatedly to calculate the value of $\hat{\phi}(\theta, N, \omega)$ for different value of θ . The freezing of the random numbers means that the problem (6) becomes a deterministic optimization problem. Furthermore, we define θ to be the logarithms of the rate constants to keep the deterministic optimization problem unconstrained. We develop UOBYQA-Fit, which is a variant of Powell's UOBYQA [40], for solving the sample path optimization problem (6). The lack of availability of good derivative estimates for the objective function $\hat{\phi}(\theta, N, \omega)$ makes use of derivative free methods like UOBYQA attractive here. Like UOBYQA, UOBYQA-Fit is a model based approach which constructs a series of quadratic models approximating the objective function. Like UOBYQA, UOBYQA-Fit uses a trust region framework [33, 35]. However, unlike UOBYQA, which does exact interpolation to obtain the quadratic model, UOBYQA-Fit fits the quadratic model to the sample points. As described in section 4.1, the use of fitting instead of exact interpolation enables UOBYQA-Fit to have a fairly mild adequacy criteria for the fitting points.

4.1. The quadratic model

At every iteration of the algorithm, UOBYQA-Fit fits a quadratic model

$$Q(\theta) = c + g^T (\theta - \theta_k) + \frac{1}{2} (\theta - \theta_k)^T G (\theta - \theta_k) \quad (7)$$

to an adequate set of sample points $S_k = \{z_1, z_2, \dots, z_L\}$. The value of the objective function is obtained using (4) along with N SSA simulations. We describe subsequently the notion of adequacy and how we obtain the number N . The fitting procedure solves the following minimization problem.

$$\{c_k, g_k, G_k\} = \arg \min_{c, g, G} \sum_{i=1}^L (Q(z_i) - \hat{\phi}(z_i, N, \omega))^2 \quad (8)$$

The point θ_k is the center of the trust region, c_k is a scalar, g_k is a vector in $\mathbb{R}^n$, and G_k is a symmetric $\mathbb{R}^{n \times n}$ matrix. The quadratic model is expected to approximate the function $\hat{\phi}$ around θ_k . The number of points in the fitting set S_k is chosen to be double the number of points that can determine a unique quadratic, i.e.,

$$L = (n+1)(n+2) \quad (9)$$

The fitted quadratic that is obtained after solving minimization problem (8) is

$$Q_k(\theta) = c_k + g_k^T (\theta - \theta_k) + \frac{1}{2} (\theta - \theta_k)^T G_k (\theta - \theta_k) \quad (10)$$

The notion of adequacy—Unlike UOBYQA, which requires that no nonzero quadratic function should vanish at all the interpolation points, we impose an adequacy criteria that is relatively mild. The points in the fitting set must neither be too close to each other, nor too far away from the current iterate θ_k . We ensure that any two points in the fitting set are no closer than δ_k/L , where closeness is measured by the Euclidean distance. We also ensure that

$$\|z_i - \theta_k\| \leq 2\delta_k, \quad z_i \in S_k.$$

The adequacy criteria ensures that the fitted quadratic model Q_k is a good local approximation to the function $\hat{\phi}$ in the neighborhood of θ_k .

Obtaining number of SSA simulations N and smoothing/noise parameter R—

As evident from the equation (4), the estimation of negative log likelihood requires the values of N and R . In the absence of any external information about measurement noise, we use

$$R = \alpha I_{n_d \times n_d} \quad (11)$$

in which the value of α depends on N via

$$\alpha = \frac{c}{\sqrt{N}}. \quad (12)$$

The value of c is a problem dependent parameter. Intuitively equations (11) and (12) imply that a large number of SSA simulations (N) drive the requirement of smoothing (α) to a small value. To obtain N , we implement the UOBYQA-Fit algorithm described in the next

section, with $N = N_i \in \{1, 2, 3 \dots\}$ values to obtain several optimal parameter estimates θ_i . We choose the smallest $N = N_i$ such that

$$\|\theta_i - \theta_{i-1}\| \leq \varepsilon$$

in which ε is a small number. With c and the obtained N values, we use equation (12) to obtain α . The value of α , which determines R by equation (11), and N , the number of SSA simulations, are the two required parameter for the determination of $\hat{\varphi}$ in equation (4).

4.2. The UOBYQA-Fit algorithm

In this section we present the core algorithm, which inherits several basic features of the UOBYQA algorithm [40]. Starting the algorithm requires a starting point θ_0 , and an initial trust region radius δ_0 . At iteration k , we fit the quadratic model (7) as described in section 4.1. As in a classic trust region method, a new promising point θ_{k+1} is obtained by solving the subproblem

$$\min_{s \in \mathbb{R}^n} Q_k(\theta_k + s), \quad \text{subject to } \|s\| \leq \delta_k \quad (13)$$

The new promising point s^* is accepted if the ‘degree of agreement’

$$\rho_k = \frac{\hat{\varphi}(\theta_k) - \hat{\varphi}(\theta_k + s^*)}{Q_k(\theta_k) - Q_k(\theta_k + s^*)} \quad (14)$$

is large enough. Otherwise the next iterate $\theta_{k+1} = \theta_k$. If ρ_k is large enough, which indicates good match between the quadratic model Q_k and the function $\hat{\varphi}$, the point $\theta_k + s^*$ is set as the iterate $k + 1$ and it is put inside the interpolation set S_{k+1} .

UOBYQA-Fit algorithm

Input

- Objective function $\hat{\varphi}$ which depends on the model, experimental data, N, R
- Trust region parameters

$$0 \leq \eta_a \leq \eta_0 \leq \eta_1, \quad 0 \leq \gamma_0 \leq \gamma_1 \leq 1 \leq \gamma_2, \delta_0, \delta_{end}$$

- Starting point θ_0

Output

- Parameter estimate θ^* and objective function value, φ^* , at the parameter estimate θ^*
1. Generate the initial fitting set S_1 . Using equation (4), evaluate $\hat{\varphi}$ for each point in the fitting set S_1 . The first iterate is the point $\theta_1 \in S_1$ that minimizes $\hat{\varphi}$ over the points in the set S_1

2. For iteration $k = 1, 2, \dots$

- a. Construct a quadratic model of the form (10) which fits points in S_k by solving the optimization problem (8)
- b. Solve the trust region problem (13). Evaluate $\hat{\varphi}$ at the new point $\theta_k + s^*$, and compute the agreement ratio, ρ_k , defined in equation (14)
- c. If $\rho_k \geq \eta_1$, increase the trust radius δ_k by using

$$\delta_k \leftarrow \delta_k(1 + \gamma_2)/2$$

otherwise, decrease the trust radius by using

$$\delta_k \leftarrow \delta_k(\gamma_0 + \gamma_1)/2$$

- d. If $\rho_k \geq \eta_0$, accept the point $\theta_k + s^*$ as the next iterate

$$\theta_k \leftarrow \theta_k + s^*$$

otherwise, set the next iterate as the current iterate

$$\theta_k \leftarrow \theta_k$$

- e. If $\rho_k \geq \eta_a$, improve the quality of fitting points in S_k as described in section 4.1
- f. Check whether any of the termination criteria is satisfied, otherwise repeat the loop. The termination criteria include hitting the limit on the number of function $\hat{\varphi}$ evaluations and $\delta_k \leq \delta_{end}$

3. Evaluate and return the final solution point θ^* and value of the objective function $\hat{\varphi}$ at θ^* .

5. Estimation of confidence region

As we explained in previous sections, the distribution of parameter estimates is not known apriori. Without any information about this distribution, we choose ellipsoidal confidence region as it is a typical choice of confidence region in statistics. Next we present the estimation of confidence region algorithm: Bootstrapping is the method of choice to estimate confidence regions when we do not have any information about the distribution of the sample statistics [3, 12, 25, 10]. The parameter estimates obtained for a given experimental data y , consisting of m replicates, depend upon the experimental data y . We do not have any specific information about the distribution of y , implying that the distribution of sample statistics, i.e., parameter estimate $\hat{\theta}$, is also unavailable to us.

To generate a large population of replicates from the experimental data y , we make c_y copies of y and generate a bootstrapped population of replicates Y . From the bootstrapped population of replicates, we pick m replicates by doing random sampling with replacement.

This random sampling with replacement gives us one bootstrapped experimental data set. We repeat this random sampling with replacement N_B times to obtain N_B sets of bootstrapped experimental data, each consisting of m replicates. Next we present the algorithm to estimate the parameter confidence region, which is an ellipsoid, given the experimental data y and confidence level α .

5.1. The confidence region estimation algorithm

Input

- Experimental data y
- Number of copies of experimental data c_y
- Number of bootstrapped experimental data sets N_B
- The level of confidence for parameter estimates α

Output

- Center of the confidence region, θ_c
- G and b_α that characterize the confidence region

$$(\theta - \theta_c)' G (\theta - \theta_c) \leq b_\alpha$$

- Bounding box half edge lengths l
1. Make c_y copies of experimental data y to generate bootstrapped population of replicates Y
 2. Generate N_B bootstrapped experimental data sets from Y using random sampling with replacement
 3. For each bootstrapped experimental data run UOBYQA-Fit of section 4.2 to obtain a bootstrapped parameter estimate
 4. Make a grid in the parameter space centered around the median of the bootstrapped parameter estimates obtained from step 3
 5. Evaluate objective function $\hat{\varphi}$ at each point of the grid of step 4
 6. Fit a quadratic

$$(\theta - \theta_c)' G (\theta - \theta_c) = d$$

through the grid points and the value of the objective functions obtained in step 5 at the grid points. θ_c is the median of the bootstrapped parameter estimates of step 3

7. For each bootstrapped parameter estimate obtained from step 3, evaluate $f(\theta) = (\theta - \theta_c)' G (\theta - \theta_c)$
8. Sort $f(\theta)$ values obtained from step 7 in ascending order. Assign b_α to 100 α percentile point of this sorted $f(\theta)$ values

9. The vector of bounding box half lengths is given by

$$l = \sqrt{(b_\alpha \text{diag}(G^{-1}))}$$

10. Return θ_c, G, b_α, l

Next we check the accuracy of the obtained confidence region. We generate several sets of experimental data, and obtain parameter estimates for each experimental data set. We obtain the confidence region, using just the first experimental data set. We verify the accuracy of this confidence region by checking how many of the parameter estimates fall inside this confidence region. Ideally, the fraction of points inside the confidence region should be equal to the confidence level α . We are now ready to present, the verification procedure for the confidence region algorithm.

5.2. The verification of confidence region algorithm

Input

- N_e sets of experimental data. Each experimental data set consists of m replicates
- C , a vector of confidence levels

Output

- Vector I with elements as I_i such that for each confidence level $\alpha_i \in C$, I_i is the number of points inside α_i confidence hyper-ellipsoid
1. For each experimental data, obtain parameter estimate using UOBYQA-Fit algorithm
 2. For $i = 1, 2, \dots$ to number of confidence levels
 - a. Call the confidence region estimation algorithm of section 5.1 with confidence level α_i and first experimental data set
 - b. For each parameter estimate $\hat{\theta}$ obtained in step 1 check

$$(\theta_i - \theta_c)' G (\theta_i - \theta_c) \leq b_{\alpha_i} \quad (15)$$

- c. Assign I_i to the number of parameter estimates from step 5.2 that satisfy (15)
3. Return vector I

6. Application: RNA dynamics in Escherichia coli

In this section we apply the new method to estimate 3 parameters in an important biological process. We obtain both parameter estimates and confidence region of the estimates.

Golding et al. [21] developed a fluorescence microscopy method to quantify molecular level of mRNAs in individual E. coli cells. The method is based on amplification of a fluorescence protein having the capability to bind to a reporter RNA. To obtain the number

of mRNA molecules, the fluorescence flux produced in the cells is compared with the fluorescence produced by a single mRNA molecule. The mRNA signal was shown to rise until 80 minutes and then plateau. They fit the experimental data with a mass action kinetic model given by reactions

Since our aim here is to test the algorithm developed, we test it with simulated experimental data assuming model (16)–(18) as the true system. The amount and quality of experimental data is important to obtain good parameter estimates and tight Figure 1 shows 10 replicates of the experiment. The large replicate to replicate variability is an indicator of lack of enough experimental data. We use $m = 100$ replicates as our experimental data set. For each replicate we use a sampling time of 0.5 minutes, which is the same as the successive image time of the experimental protocol.

We apply the UOBYQA-Fit algorithm described in section 4.2 to obtain the point estimates of the parameters. The parameters used in the implementation of the algorithm are listed in Table 1.

Unlike [39], we provide a confidence region using the method described in section 5. In 2-D the confidence region for the parameters is an ellipse and the bounding box is the rectangle aligned with the coordinate axes and tangent to the ellipse. Figure 2 shows an ellipse with its rectangular bounding box.

In 3-D the confidence region for the parameters is an ellipsoid and the bounding box is defined to be the box aligned with the coordinate axes and tangent to the ellipsoid. The parameter estimates obtained and the corresponding confidence region, both the bounding box and the extreme points of the ellipsoid, are listed in Table 2. On a normal desktop PC, it takes 9 minutes to obtain the parameter estimation and the confidence region listed in Table 2

The ratio of volumes of the 95% confidence region ellipsoid and the bounding box corresponding to this ellipsoid is 0.28. This ratio indicates the significant stretch in the confidence region when we put the bounding box around the 95% confidence region ellipsoid.

Next we verify whether the large confidence region depicted in Table 2 is due to the lack of information in the experimental data or it is an artifact of our confidence region estimation algorithm. We generate 500 sets of experimental data set where each experimental data set consists of 100 replicates. We use the verification of the confidence region algorithm of

section 5.2 to obtain the number of parameter estimates inside different confidence level ellipsoids.

Figure 3 shows, the number of these points that are inside α level confidence region as a function of α . For small values of α the number of points inside α level confidence ellipsoid are close to the expected number of points inside the α level confidence ellipse. Figure 3 illustrates the quality of the confidence region generated by the confidence region estimation algorithm. Therefore the confidence region algorithm generates reliable confidence regions, and the large confidence region depicted in the table 2 is due to the lack of information in the experimental data.

7. Conclusions

In this paper we presented a new method of parameter estimation in stochastic chemical kinetic models. The method is based upon a negative log likelihood minimization approach in which the likelihood expression had several nice properties. The likelihood expression was in the form of the expectation of a function of data, parameters, and a smoothing parameter. The estimation of this likelihood is possible even with just 1 SSA simulation. We describe a procedure to obtain the number of SSA simulations, N , which can give us reliable parameter estimates. We provide a heuristic expression for the connection between the smoothing parameter, R , and the number of SSA simulations, N . Equipped with the likelihood expression, N , and R , we used the sample path method to estimate the negative log likelihood. To minimize this negative log likelihood, we developed a derivative free optimization method UOBYQA-Fit, a variant of Powell's UOBYQA algorithm. To estimate the confidence region, we developed a variant of Efron's percentile method.

We tested the obtained parameter estimates and the confidence regions by generating several experimental data sets. The tests indicate that both the optimization and the confidence estimation algorithms are producing reliable parameter estimates and confidence regions, respectively.

Parameter estimation in stochastic chemical kinetic models is an important problem. We believe that the new proposed parameter estimation method has significant advantages over previous methods. An active area of our ongoing research is use of derivative based optimization for the negative log likelihood function. We are evaluating the effectiveness of derivatives estimates obtained by a good sensitivity estimator, e.g. coupled-finite difference [1, 47], in performing the parameter estimation in stochastic chemical kinetic models.

References

1. Anderson DF. An efficient finite difference method for parameter sensitivities of continuous time Markov chains. *SIAM J Num Anal.* 2012; 50:2237–2258.
2. Arkin A, Ross J, McAdams H. Stochastic kinetic analysis of developmental pathway bifurcation in phage lambda-infected Escherichia coli cells. *Genetics.* Aug; 1998 149(4):1633–1648. [PubMed: 9691025]
3. Babu, GJ.; Singh, K. *The Annals of Statistics.* 1983. Inference on means using the bootstrap; p. 999-1003.

4. Blake WJ, Kaern MA, Cantor CR, Collins JJ. Noise in eukaryotic gene expression. *Nature*. 2003; 422(6932):633–637. [PubMed: 12687005]
5. Boys RJ, Wilkinson DJ, Kirkwood TBL. Bayesian inference for a discretely observed stochastic kinetic model. *Stat and Comput*. 2008; 18(2):125–135.
6. Conn, AAR.; Scheinberg, K.; Vicente, LN. Introduction to derivative-free optimization. Vol. 8. SIAM; 2009.
7. Conn, AR.; Scheinberg, K.; Toint, PL. Recent progress in unconstrained nonlinear optimization without derivatives. *MATHEMATICAL PROGRAMMING*; 16th International Symposium on Mathematical Programming (ismp97); LAUSANNE, SWITZERLAND. 1997; Oct 1. 1997 p. 397-414.
8. Conn, AR.; Toint, PL. An algorithm using quadratic interpolation for unconstrained derivative free optimization. IBM TJ Watson Research Center; 1995.
9. Deng G, Ferris M. Variable-number sample-path optimization. *Math Prog*. 2009; 117:81–109.
10. Diccio, TJ.; Romano, JP. *Journal of the Royal Statistical Society Series B (Methodological)*. 1988. A review of bootstrap confidence intervals; p. 338-354.
11. Dong MX, Wets RJB. Estimating density functions: A constrained maximum likelihood approach. *JOURNAL OF NONPARAMETRIC STATISTICS*. 2000; 12(4):549–595.
12. Efron B. Bootstrap confidence intervals for a class of parametric problems. *Biometrika*. 1985; 72(1):45–58.
13. Efron B, Halloran E, Holmes S. Bootstrap confidence levels for phylogenetic trees. *PROCEEDINGS OF THE NATIONAL ACADEMY OF SCIENCES OF THE UNITED STATES OF AMERICA*. Jul 9; 1996 93(14):7085–7090. [PubMed: 8692949]
14. Efron B, Tibshirani R. Bootstrap methods for standard errors, confidence intervals, and other measures of statistical accuracy. *Statistical science*. 1986; 1(1):54–75.
15. Elowitz MB, Levine AJ, Siggia ED, Swain PS. Stochastic gene expression in a single cell. *Science*. 2002; 297(5584):1183–1186. [PubMed: 12183631]
16. Ferris, MC.; Munson, TS.; Sinapiromsaran, K. A practical approach to sample-path simulation optimization. *Proceedings of the 32nd conference on Winter simulation*; Society for Computer Simulation International; 2000. p. 795-804.
17. Gillespie DT. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *J Comput Phys*. 1976; 22:403–434.
18. Gillespie DT. Exact stochastic simulation of coupled chemical reactions. *J Phys Chem*. 1977; 81:2340–2361.
19. Gillespie DT. A rigorous derivation of the chemical master equation. *Physica A*. 1992; 188:404–425.
20. Golding I, Cox EC. Rna dynamics in live escherichia coli cells. *Proceedings of the National Academy of Sciences of the United States of America*. 2004; 101(31):11310. [PubMed: 15277674]
21. Golding I, Paulsson J, Zawilski SM, Cox EC. Real-time kinetics of gene activity in individual bacteria. *Cell*. 2005; 123(6):1025–1036. [PubMed: 16360033]
22. Golightly A, Wilkinson DJ. Bayesian inference for nonlinear multivariate diffusion models observed with error. *Comput Stat Data Anal*. 2008; 52(3):1674–1693.
23. Gürkan, G.; Özge, AY.; Robinson, SM. Sample-path solution of stochastic variational inequalities, with applications to option pricing. *Proceedings of the 28th conference on Winter simulation*; IEEE Computer Society; 1996. p. 337-344.
24. Gürkan, G.; Özge, AY.; Robinson, SM. Solving stochastic optimization problems with stochastic constraints: An application in network design. *Proceedings of the 31st conference on Winter simulation: Simulation—a bridge to the future*; ACM; 1999. p. 471-478.
25. Hall P. On the number of bootstrap simulations required to construct a confidence interval. *The Annals of Statistics*. 1986; 14(4):1453–1462.
26. Henderson DA, Boys RJ, Krishnan KJ, Lawless C, Wilkinson DJ. Bayesian emulation and calibration of a stochastic computer model of mitochondrial dna deletions in substantia nigra neurons. *Journal of the American Statistical Association*. 2009; 104(485):76–87.

27. Homem-De-Mello T. Variable-sample methods for stochastic optimization. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*. 2003; 13(2):108–133.
28. Homem-de Mello T. On rates of convergence for stochastic optimization problems under non-independent and identically distributed sampling. *SIAM Journal on Optimization*. 2008; 19(2): 524–551.
29. Hooshangi S, Thiberge S, Weiss R. Ultrasensitivity and noise propagation in a synthetic transcriptional cascade. *Proc Natl Acad Sci USA*. 2005; 102(10):3581–3586. [PubMed: 15738412]
30. Lakowicz JR, Masters BR. Principles of fluorescence spectroscopy. *Journal of Biomedical Optics*. 2008; 13:029901.
31. Marazzi M, Nocedal J. Wedge trust region methods for derivative free optimization. *Math Prog*. 2002; 91:289–305.
32. McAdams HH, Arkin A. Stochastic mechanisms in gene expression. *Proc Natl Acad Sci USA*. Feb 4; 1997 94(3):814–819. [PubMed: 9023339]
33. Moré JJ, Sorensen DC. Computing a trust region step. *SIAM Journal on Scientific and Statistical Computing*. 1983; 4:553.
34. Munsy B, Khammash M. The finite state projection algorithm for the solution of the chemical master equation. *J Chem Phys*. 2006; 124:044104. [PubMed: 16460146]
35. Nocedal, J.; Wright, SJ. *Numerical Optimization*. 2. Springer; New York: 2006.
36. Ozbudak EM, Thattai M, Kurtser I, Grossman AD, van Oudenaarden A. Regulation of noise in the expression of a single gene. *Nat Genet*. 2002; 31(1):69–73. [PubMed: 11967532]
37. Plambeck, EL.; Fu, BR.; Robinson, SM.; Suri, R. Throughput optimization in tandem production lines via nonsmooth programming. *Proceedings of the 1993 Summer Computer Simulation Conference*; 1993. p. 70-75.
38. Plambeck EL, Fu BR, Robinson SM, Suri R. Sample-path optimization of convex stochastic performance functions. *Mathematical Programming*. 1996; 75(2):137–176.
39. Poovathingal SK, Gunawan R. Global parameter estimation methods for stochastic biochemical systems. *BMC Bioinf*. 2010; 11(1):414.
40. Powell MJD. UOBYQA: unconstrained optimization by quadratic approximation. *Math Prog*. 2002; 92:555–582.
41. Raser JM, O'Shea EK. Control of stochasticity in eukaryotic gene expression. *Science*. 2004; 304(5678):1811–1814. [PubMed: 15166317]
42. Reinker S, Altman R, Timmer J. Parameter estimation in stochastic biochemical reactions. *IEE Proc-Syst Biol*. 2006; 153(4):168.
43. Rios LM, Sahinidis NV. Derivative-free optimization: A review of algorithms and comparison of software implementations. *Journal of Global Optimization*. 2012:1–47.
44. Shapiro A. Statistical inference of stochastic optimization problems. *NONCONVEX OPTIMIZATION AND ITS APPLICATIONS*. 2000; 49:282–307.
45. Shapiro, A. Monte carlo sampling approach to stochastic programming. *ESAIM: Proceedings*; 2003. p. 65-73.
46. Shapiro A, Homem-de Mello T. A simulation-based approach to two-stage stochastic programming with recourse. *Mathematical Programming*. 1998; 81(3):301–325.
47. Srivastava R, Anderson DF, Rawlings JB. New methods to obtain sensitivities of stochastic chemical kinetic models. 2012 To be submitted to *Journal of Chemical Physics*.
48. Tian T, Xu S, Gao J, Burrage K. Simulated maximum likelihood method for estimating kinetic rates in gene expression. *Bioinformatics*. 2006; 23(1):84. [PubMed: 17068087]
49. Wets, RJB. Statistical estimation from an optimization viewpoint. *ANNALS OF OPERATIONS RESEARCH*; 8th International Conference on Stochastic Programming; UNIV BRITISH COLUMBIA, VANCOUVER, CANADA. AUG 08–16, 1998; 1999. p. 79-101.
50. Yuanfeng W, Scott C, Xiaohui X. Parameter inference for discretely observed stochastic kinetic models using stochastic gradient descent. *BMC Syst Biol*. 2010; 4

Appendix: Proof of exact likelihood of the experimental data

We want to obtain the likelihood of the experimental data $y_j \in \mathbb{R}^n_d$. Using the law of total probability

$$p(y_j; \theta) = \int p(y_j | x) p_s(x) dx \quad (19)$$

in which $p_s(x)$ is the sampled density obtained using N SSA simulations. The difference between experimental data y_j and samples x is explained by the smoothing/noise random variable v , i.e.,

$$p(y | x) = p_v(y - x) \quad (20)$$

in which $v \sim N(0, R)$. The sampled density $p_s(x)$ consists of series of delta function corresponding to the N SSA simulations, and is given by

$$p_s(x) = \frac{1}{N} \sum_{i=1}^N \delta(x - x_i) \quad (21)$$

Substituting (20) and (21) into equation (19) gives us

$$\begin{aligned} p(y_j; \theta) &= \int p_v(y_j - x) \frac{1}{N} \sum_{i=1}^N \delta(x - x_i) dx \\ &= \frac{1}{N} \sum_{i=1}^N p_v(y_j - x_i) \\ p(y_j; \theta) &= \frac{1}{N(2\pi)^{n_d/2} |R|^{1/2}} \sum_{i=1}^N e^{-1/2(y_j - x_i)' R^{-1} (y_j - x_i)} \end{aligned} \quad (22)$$

In the limit of $N \rightarrow \infty$ and $R \rightarrow 0$, the above expression of likelihood turns into the exact likelihood $L(y_j | \theta)$

$$L(y_j | \theta) = \lim_{R \rightarrow 0} \frac{1}{(2\pi)^{n_d/2} |R|^{1/2}} \mathbb{E}_\theta [e^{-1/2(y_j - x)' R^{-1} (y_j - x)}] \quad (23)$$

If we have total m replicates $j = 1, 2, \dots, m$ of the experiment y_j , the likelihood of the experimental data y is given by the product of likelihood of each replicate y_j , which is exactly the equation (2).

HIGHLIGHTS

This paper presents a novel method of parameter estimation for stochastic chemical kinetic models.

These models arise naturally in the modeling of small scale chemical reactors or biological processes.

The method uses tools from statistics, probability theory and numerical optimization.

The combination leads to an accurate and reliable method.

FIG. 1.
10 replicates of mRNA vs time experimental data

FIG. 2.
An ellipse and its bounding box

FIG. 3.
Verification of confidence region using bootstrapping

TABLE I

System and optimization parameters

Parameter	m	N	c	θ	θ_1	δ_1	δ_{and}
Value	100	100	10	$\begin{bmatrix} -1.56 \\ -0.78 \\ -0.40 \end{bmatrix}$	$\begin{bmatrix} -2.00000 \\ -0.39794 \\ -0.45593 \end{bmatrix}$	1	0.004

TABLE II

Parameter estimates and confidence regions. The true parameter values are $[-1.56 \ -0.78 \ -0.40]'$

Parameter estimates	95% Bounding box	6 Extreme points of the ellipsoid
$\begin{bmatrix} -1.40 \\ -1.16 \\ -0.60 \end{bmatrix}$	$\begin{bmatrix} -1.34 \pm 0.97 \\ -1.06 \pm 0.56 \\ -0.52 \pm 1.02 \end{bmatrix}$	$\begin{bmatrix} -1.06 & -1.63 & -0.58 & -2.11 & -1.80 & -0.88 \\ -0.67 & -1.46 & -1.01 & -1.12 & -0.46 & -1.67 \\ -0.24 & -0.81 & -1.36 & -0.32 & -0.90 & -0.15 \end{bmatrix}$