Estimating Rate Constants in Cell Cycle Models

Jason W. Zwolak*, John J. Tyson, and Layne T. Watson***
Departments of Computer Science* and Biology**
Virginia Polytechnic Institute and State University
Blacksburg, Virginia 24061-0106
e-mail: jzwolak@vt.edu

Keywords: Computational biology, ordinary differential equations, parameter estimation

Abstract

Cell cycle models used in biology can be very complex. These models have parameters with initially unknown values. The values of the parameters vastly affect the accuracy of the models in representing real biological cells. Typically people search for the best parameters to these models using computers only as tools to run the models. In this paper a method and results are described for a computer program that searches for parameters to a specific model. The code for this program uses ODRPACK for parameter estimation and LSODAR to solve the differential equations that make up the model. The resulting parameters fit the experimental data with a total relative error of 2.11×10^{-2}.

1. INTRODUCTION

Computational models of cell growth and division involve digital representation of a complex network of biochemical reactions within cells. These reactions can be described by a system of nonlinear ordinary differential equations, according to the principles of biochemical kinetics. Rate constants and binding constants enter as parameters in the differential equations, and must be estimated by fitting solutions of the equations to experimental data.

This work concerns some classical experiments on activation of MPF (M-phase promoting factor) in frog egg extracts. MPF is a dimer of cyclin and Cdc2 (a protein kinase that drives egg nuclei into mitosis). In the experimental preparation, a fixed amount of cyclin is added to an extract containing an excess of Cdc2 subunits. If the amount of cyclin added is below a threshold, MPF activity never appears. Above the threshold, MPF is activated but only after a characteristic time lag. The time lag decreases abruptly as total-cyclin-added increases above the threshold. The goal is to fit this data with a reasonable model of the underlying biochemistry, which keeps track of cyclin monomers, Cdc2 monomers, and the phosphorylation state of cyclin/Cdc2 dimers.

ODRPACK, based on the orthogonal distance between experimental data and the model, is used for the nonlinear regression to estimate the unknown rate constants (ODE parameters). The ability of this algorithm to arbitrarily weight data values, and to treat both the abscissa and ordinates as uncertain, is crucial, given the sparsity and uncertainty of available biological data. Constructing the model function values requires simulating MPF activity as a function of time after addition of cyclin. These simulations yield the cyclin threshold for MPF activation, and the time lag (the time necessary for MPF activity to reach one-half of its asymptotic value, for supra-threshold stimulation).

The complete calculation is expensive, because the ODEs are stiff, and must be solved numerous times for the nonlinear regression. Also, because of local minima, the nonlinear regression must be done from many starting points to adequately explore the parameter space. Potential sources for parallelism are the ODE solution itself, the estimation of partial derivatives of the ODE solution, and multiple starting points for regression. Numerical results are presented for a relatively simple two-component model, as well as scalability results for shared memory and distributed memory parallel computers.

To study realistic models of cell cycle control, more components must be added to the model, and other measurable phenomena incorporated in the cost function. As the modeling fidelity is increased, the mathematical and computational complexities of the problem grow rapidly. Efficient and accurate tools for parameter estimation will be needed to build computational models of the complex control networks operating within cells, which is one of the main goals of bioinformatics in the postgenomic era.

Section 2 outlines the biological model and provides the experimental data. An overview of the code along with descriptions of the tools (ODRPACK and LSODAR) used by the code can be found in Section 3. Section 4 contains a more detailed pseudocode for the algorithm. The results of the parameter estimation are in Section 5.

2. PROBLEM STATEMENT

The differential equation describing the concentration and rate of change in active MPF in a frog egg with a fixed concentration of total cyclin is

$$\frac{dM}{dt} = -k_{wee}M + (k_{25} + k'_{25}M^2)(C - M),$$

where k_{wee}, k_{25}, and k'_{25} are rate constants, C is concentration of total cyclin, and M is the concentration of active MPF versus time [13].

In cells, MPF is the primary protein that determines when the cell divides. However, MPF does not promote cell division unless MPF is active. Other proteins in the cell inhibit or promote MPF activation. The most dominant proteins in this model are Cdc25 and Wee1. Wee1 deactivates MPF and the rate constant k_{wee} represents Wee1's affect on MPF inactivation. Cdc25 activates MPF and the rate constants k_{25} and k'_{25} represent Cdc25's affect on MPF activation [13].

This paper describes the methods and results of estimating the rate constants k_{wee}, k_{25}, and k'_{25}. Estimating k_{wee}, k_{25}, and k'_{25} requires experimental data related to the ODE. The available data is in Table 1.

Table 1. Experimental data [8]

Total Cyclin	Time Lag (min)
0.15	55
0.20	45
0.25	40
0.30	30
0.50	20

The time lag appears in the ODE as the point where the active MPF concentration is half its asymptotic value. For low concentrations of total cyclin, MPF never activates (i.e., the concentration of active MPF never rises above half the concentration of total cyclin). For higher concentrations of total cyclin MPF activates after some time lag. The higher the concentration of total cyclin the smaller the time lag. This behavior can be seen in the experimental data in Table 1 and the plots in Figure 1.

3. METHODS

The variables that correspond to the data in Table 1 are not all present in the ODE (total cyclin is in the ODE, time lag is not in the ODE). The first step is to use the ODE to calculate another function f in terms of the variables corresponding to the data in Table 1.

Let $f(x)$ be the time lag for total cyclin x, where time lag is the time for MPF to activate or deactivate (depending on whether MPF was initially active or inactive). More precisely the time lag is the time where

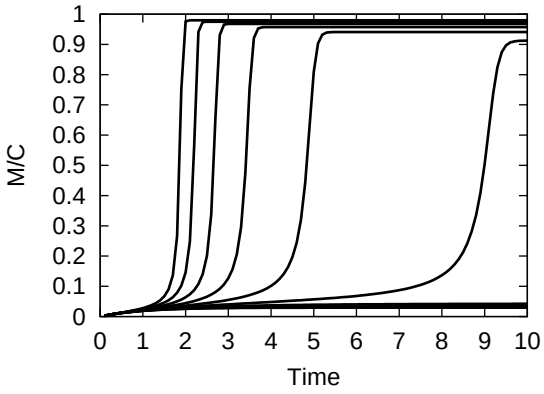

Figure 1. Percent total cyclin in active MPF, M/C, versus time t for multiple concentrations of total cyclin

the active MPF concentration is the average of the initial concentration of active MPF and the asymptotic concentration of active MPF.

LSODAR is used to solve the ODE and to find the time lag from the solution to the ODE. The time lag versus total cyclin function $f(x)$ is also a function of the rate constants in the ODE. This function is used by ODRPACK to find the rate constants giving the curve $f(x)$ that best fits the experimental data in Table 1.

3.1. LSODAR

LSODAR is a variant of LSODE ([10], [5], [6]) that automatically switches between stiff and non-stiff methods and has a root finder. LSODAR starts with a non-stiff method and switches to a stiff method if necessary. LSODAR also has a built in root finder, which is used in this application to find the time lag for MPF activation.

For non-stiff problems LSODAR uses Adams-Moulton (AM) of orders 1 to 12. For stiff problems LSODAR uses backward differentiation formulas (BDF) of orders 1 to 5. With both methods LSODAR varies the step size and order. LSODAR switches from AM to BDF when AM is no longer stable for the problem or cannot meet the accuracy requirements efficiently [9].

The present problem uses LSODAR to solve for $M(t)$ (the concentration of active MPF with respect to time). The tolerances are set to 10^{-12} for both relative and absolute error. A tolerance of 10^{-10} is used when calculating a root for a function of the form

$$M(t) - M_{root},$$

where M_{root} is the value of the function $M(t)$ for which a time, t, is desired.

LSODAR takes, as an argument, a user written function, GEX, that evaluates equations based on the

variables involved in the ODE that LSODAR is solving. For this problem GEX evaluates $M - M_{root}$ as mentioned earlier. GEX returns evaluations of its equations to LSODAR and LSODAR looks for roots for those equations. When a sign change is detected LSODAR has bracketed a root and begins an algorithm based on the ROOT function described below. After each iteration of ROOT, LSODAR must evaluate a point on the solution curve of the ODE as requested by ROOT. Each evaluation involves interpolating the ODE solution $M(t)$. This interpolation formula is defined as part of the AM [12] or BDF [4] method (depending on which is currently being used by LSODAR).

3.2. ODRPACK

ODRPACK is used to estimate the rate constants that fit time lag versus total cyclin to the experimental data in Table 1. ODRPACK finds an estimate for the rate constants by minimizing the weighted orthogonal distance between the experimental data and the calculated curve.

The present problem explicitly relates time lag to the total concentration of cyclin in the cell. Precisely,

$$y = f(x; \beta),$$

where y is time lag, x is total cyclin, and β is a vector of the rate constants. ODRPACK takes an equation of this form and experimental data for x and y to minimize

$$\min_{\beta,\delta,\epsilon} \left(\sum_{i=1}^{n} w_{\epsilon_i} \epsilon_i^2 + w_{\delta_i} \delta_i^2 \right),$$

where n is the number of experimental data points, ϵ_i is the error in the dependent variable y for point i, δ_i is the error in the explanatory variable x for point i, and w_{ϵ_i} and w_{δ_i} are the weights for ϵ_i and δ_i, respectively. β, δ, and ϵ are subject to the constraints

$$y_i = f(x_i + \delta_i; \beta) - \epsilon_i,$$

where $i = 1, \ldots, n$ indexes the experimental data points.

ODRPACK actually minimizes a more general objective function

$$\min_{\beta,\delta,\epsilon} \left(\sum_{i=1}^{n} \epsilon_i^T w_{\epsilon_i} \epsilon_i + \delta_i^T w_{\delta_i} \delta_i^2 \right),$$

where ϵ_i and δ_i are vectors for the errors in the dependent variable and errors in the explanatory variable, respectively. w_{ϵ_i} and w_{δ_i} are matrices of weights for ϵ_i and δ_i, respectively [2] [1]. Note that x and y, from the previous description of ODRPACK, are vectors and the function f is a vector-valued function in the general case. The present problem can be thought of as using the scalar version of ODRPACK, since the present problem has w_{ϵ_i} and w_{δ_i} as matrices of one element and ϵ_i and δ_i as vectors of one element.

The function $f(x + \delta; \beta)$ is implemented in FORTRAN and used by ODRPACK. Constraints are put on β by setting a flag (when β is invalid) before returning from the user supplied function. This is used to prevent the rate constants from becoming negative, which does not make sense biologically.

ODRPACK uses a trust region Levenberg-Marquardt method with scaling to minimize the objective function [2]. In doing so ODRPACK needs to calculate the Jacobian matrices for β and δ. ODRPACK can calculate the Jacobian matrices by finite differences or by a user supplied routine. Finite differences were used here.

3.3. ROOT

ROOT is based on ZEROIN [11], which is in turn based on code by Dekker [3]. ROOT uses a combination of the secant and bisection methods where the secant method is used by default. ROOT has two working approximations of the root: A and B. The approximations always satisfy the constraint

$$g(A) * g(B) \leq 0,$$

where $g(t) = M - M_{root}$ and t is time in this problem (note that M is dependent on t). Furthermore, A is the better approximation of the root of $g(t)$. A is replaced in each iteration by a better approximation and B remains the same or changes to the old A, whichever satisfies the above equation. ROOT switches to the bisection method under two circumstances: when the secant method is converging too slowly, or when a large error is introduced because of limitations in machine precision. Notice the bisection method will not suffer from large error because it computes

$$\frac{A + B}{2}$$

for each iteration.

The initial approximations, for A and B, come from LSODAR's evaluation of GEX before and after LSODAR noticed a sign change. ROOT then requests values for g at new times until the approximation for the root of g is within the requested relative and absolute error.

4. ALGORITHM

In this section the algorithm used is described in some detail using pseudocode. Many of the function arguments used with ODRPACK's subroutine DODRC and ODEPACK's subroutine LSODAR do not appear in the pseudocode. Most of these arguments were set to default values, and others are not relevant to

understanding the methods used to solve the present problem.

The main program sets up the input for DODRC and is as follows:

begin

$s := 8$; s is the number of significant digits in the response variable of f.

$n := 5$; n is the number of experimental data points.

$x := (0.15, 0.20, 0.25, 0.30, 0.50)$; the vector x contains the total cyclin components of the experimental data.

$y := (55, 45, 40, 30, 20)$; the vector y contains the time lags corresponding to total cyclin concentrations from above.

$w_\delta := (44.44, 25, 16, 11.11, 4)$; w_δ contains the weights for the errors in x.

$w_\epsilon := (3.305 \cdot 10^{-4}, 4.938 \cdot 10^{-4}, 6.25 \cdot 10^{-4}, 1.111 \cdot 10^{-3}, 2.4 \cdot 10^{-3})$; w_ϵ contains the weights for the errors in y. The weights are the squared reciprocals of the corresponding data values, which makes all the errors in the objective function relative instead of absolute.

$\beta := (0.5, 0.06, 80)$; β contains the initial guess for the rate constants. After DODRC has been called β will contain ODRPACK's best estimate for β given the arguments to DODRC.

DODRC(FCN, $n, s, x, y, w_\delta, w_\epsilon, \beta, \ldots$); the ODRPACK subroutine used is DODRC. FCN is defined below.

end

The function procedure FCN takes concentrations of total cyclin and parameters to the ODE and returns time lags for each concentration of total cyclin. ODRPACK does not give FCN the total cyclin concentrations from the experimental data. Instead, ODRPACK gives FCN the total cyclin concentrations plus some error δ. In most cases the time lags returned by FCN will not match the time lags from the experimental data. Errors in measurements in the experimental data contribute to this mismatch. ODRPACK handles this by labeling the output of FCN as $y + \epsilon$. Precisely, let $X = x + \delta$ and $Y = y + \epsilon$. FCN takes arguments β and X and returns Y. The code for FCN follows.

subroutine FCN
for $i := 1$ **step** 1 **until** n **do**
begin
$C := X(i)$; (set the total cyclin value)
$T := 0$; (the initial time)
$R_{tol} := 10^{-12}$; (relative error tolerance)
$A_{tol} := 10^{-12}$; (absolute error tolerance)
$M_{init} := 0$; (initial MPF concentration)
$T_{out} := 1440$; (solve for the MPF concentration at this time)
$N_g := 0$; (no roots are desired from LSODAR)
M_{inf} :=LSODAR(FEX, M_{init}, T, T_{out}, R_{tol}, A_{tol}, N_g, JEX, GEX, ...);
if $M_{inf} < C/2$ **then**
$Y(i) := 1440$; (pseudo-infinite-lag)
cycle;
endif
$M_{root} := M_{inf}/2$; (find a root at $M_{inf}/2$)
$N_g := 1$; (one root is desired from LSODAR)
LSODAR(FEX, M_{init}, T, T_{out}, R_{tol}, A_{tol}, N_g, JEX, GEX, ...);
$Y(i) := T_{out}$; (the root is returned in T_{out})
end

The number 1440, construed as a pseudo-infinite-lag, is used to put a limit on how long to search for MPF activation. Effectively, the (computed) curve in Figure 2 will be flat when it reaches 1440 minutes. The true physical curve continues to increase after 1440 minutes. This modification creates a curve that does not precisely match the actual curve, but this modification does not affect the computation. All the experimental data is well below 1440 minutes (1 day). ODRPACK looks for the point on the calculated curve that is closest to the experimental data when calculating the error. Since the initial guess is not closer to the horizontal line at 1440 minutes than to the real curve, the flat portion will not cause ODRPACK to make wrong estimates for the rate constants.

Subroutine FEX solves for the change in MPF concentration given MPF concentration, time, values for the parameters, and total cyclin concentration. Note that time does not appear directly in the ODE, but M is dependent on time. FEX is used by LSODAR when computing M numerically. FEX takes the MPF concentration M and returns the derivative M_t of MPF concentration with respect to time. JEX computes the partial derivative P of the ODE with respect to the dependent variable M, and takes the same arguments as FEX. LSODAR returns a root for the function G evaluated in GEX. GEX takes the same arguments as FEX. Pseudocode for FEX, JEX, and GEX follows.

subroutine FEX
begin
$M_t := -\beta_1 M + (\beta_2 + \beta_3 M^2)(C - M)$;
end

subroutine JEX
begin
$P_{1,1} := -\beta_1 - \beta_2 + 2\beta_3 CM - 3\beta_3 M^2$;
end

subroutine GEX
begin
$G_1 := M - M_{root}$; M_{root} is set elsewhere to a desired value of the solution M to the ODE defined in FEX.
end

5. RESULTS

Many variations on the parameters to ODRPACK have been tried. The best results so far are in Table 2. These results were obtained using the experimental data in Table 1 and Figure 2.

Table 2. Optimal rate constants from ODRPACK

Rate Constant	Optimal Value
k_{wee}	$1.117 \cdot 10^{-10}$
k_{25}	$3.277 \cdot 10^{-3}$
k'_{25}	$8.719 \cdot 10^{0}$

In Figure 2 the fitted curve was generated from the rate constants in Table 2. The fit appears good and the weighted sum of squares of the δ's and ϵ's are $3.039 \cdot 10^{-03}$ and $1.810 \cdot 10^{-02}$, respectively. So indeed the curve fits the given data well. However, currently the code used does not take into account the thresholds for MPF activation and inactivation. This curve has a threshold for MPF activation around or below 0.03. Experiments estimate the threshold to be about 0.1. One of the next steps for this problem is to integrate the empirical thresholds for MPF activation and inactivation into the code.

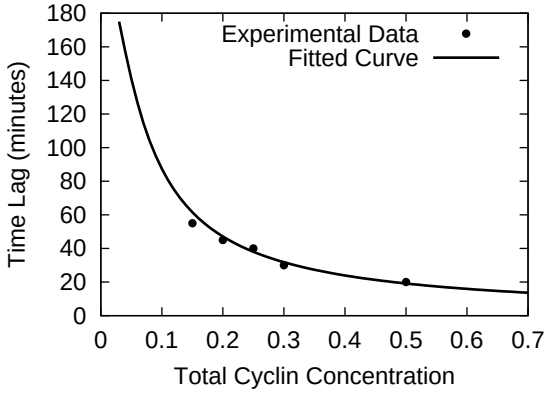

Figure 2. Time lag for MPF activation versus total cyclin

REFERENCES

[1] Boggs, P.T.; R.H. Byrd; J.R. Donaldson; R.B. Schnabel. 1989. "Algorithm 676 — ODRPACK: Software for Weighted Orthogonal Distance Regression." *ACM Trans. Math. Software 15*, No. 4:348–364.

[2] Boggs, P.T.; R.H. Byrd; J.E. Rogers; R.B. Schnabel. 1992. *User's Reference Guide for ODRPACK Version 2.01: Software for Weighted Orthogonal Distance Regression*. Center for Computing and Applied Mathematics, U.S. Department of Commerce, Gaithersburg, MD.

[3] Dekker, T.J. 1969. "Finding a zero by means of successive linear interpolation." In *Constructive Aspects of the Fundamental Theorem of Algebra* (B. Dejon and P. Henrici, eds.). Wiley-Interscience, London.

[4] Gear, C.W. 1971. *Numerical Initial Value Problems in Ordinary Differential Equations*. Prentice-Hall, Englewood Cliffs, NJ.

[5] Hindmarsh, A.C. 1980. "LSODE and LSODI, Two New Initial Value Ordinary Differential Equation Solvers." *ACM SIGNUM Newsletter 15*, No. 4:10–11.

[6] Hindmarsh, A.C. 1983. "ODEPACK: A Systematized Collection of ODE Solvers." In *Scientific Computing* (R.S. Stepleman, et al., eds.). North Holland Publishing Co., New York, 55–64.

[7] Kahaner, D.; C. Moler; S. Nash. 1989. *Numerical Methods and Software*. Prentice-Hall, Inc., Englewood Cliffs, NJ.

[8] Moore, J. 1997. Private Communication, Aug.

[9] Petzold, L. 1983. "Automatic Selection of Methods for Solving Stiff and Nonstiff Systems of Ordinary Differential Equations." *SIAM Journal on Scientific and Statistical Computing 4*, 136–148.

[10] Radhakrishnan, K.; A.C. Hindmarsh. 1993. *Description and Use of LSODE, the Livermore Solver for Ordinary Differential Equations*. NASA Reference Publication 1327. Lawrence Livermore National Laboratory, Livermore, CA, Dec.

[11] Shampine, L.F.; R.C. Allen. 1973. *Numerical Computing: An Introduction*. W. B. Saunders Company, Philadelphia, PA.

[12] Shampine, L.F.; M.K. Gordon. 1975. *Computer Solution of Ordinary Differential Equations, The Initial Value Problem*. W. H. Freeman, San Francisco, CA.

[13] Tyson, J.J.; B. Novak; K. Chen; J. Val. 1995. "Checkpoints in the cell cycle from a modeler's perspective." *Progress in Cell Cycle Research 1*, 1–8.