

Use Spreadsheets to Estimate Modeling Parameters

TOMAS B. CO
MICHIGAN TECHNOLOGICAL UNIV.

Dynamic process modeling based on non-uniformly sampled data and multiple sampling runs, even for nonlinear systems, can be performed using spreadsheet implementations of the weighted-least-squares (WLSQ) and Runge-Kutta WLSQ methods.

Dynamic processes are often modeled by a set of differential equations. Depending on the modeling objectives, the process models can be obtained from either physico-chemical laws or by empirical approaches. Process models based on physical phenomena usually start with the laws of conservation supplemented by constitutive equations. The empirical approach, on the other hand, describes the behavior of the process using standard mathematical models. Both approaches require the parameters for the model's equations to be determined with sufficient confidence that the resulting process models match the behavior of the real process.

In some applications, time-series data are obtained under uniform sampling times, often with the aid of online computerized data-logging. However, when the time-series data are obtained manually, they are not likely to be sampled uniformly over time. Moreover, time-series data are sometimes obtained from different process runs under different initial conditions and over different lengths of time.

This article discusses the least-squares method (LSQ) for parameter estimation (1); and the practical implementation of LSQ techniques using spreadsheets such as Microsoft Excel and their optimization tools (e.g., Solver).

The weighted least-squares approach

To illustrate the weighted LSQ approach for parameter estimation, consider the simplest case, where the time-series data were obtained from a single run and where the set of

differential equations has been solved analytically. The predicted values can be evaluated directly using standard spreadsheet formulas.

Set up a spreadsheet consisting of one column for time data and a pair of columns for each process variable, *i.e.*, one column for the observed data points and the other column for the corresponding predicted values. Since the objective is parameter estimation, the cells for the model parameters should be located near each other. In cases where the signals contain significant noise, the initial conditions may also be estimated, and the cells for those initial conditions should be located near the cells for the other parameters.

To each pair of columns of observed and predicted values, attach an additional column that calculates the difference. The standard approach uses the squared (residual) error, *i.e.*, $\text{error}^2 = (\text{observed} - \text{predicted})^2$. Then, set up a cell for the root mean square (RMS) value to evaluate the goodness-of-fit achieved by the process model. (The RMS is the square root of the average of all the squared errors.) When some portions of data need a better fit than other data, extra weights can be included. The resulting method is then known as the weighted least-squares (WLSQ) method.

After the spreadsheet has been created, use an optimization tool to minimize the RMS value by adjusting the parameters and initial conditions. In some cases, constraints need to be included to set bounds on the estimates as well as the stability of the process model.

Example 1: FOPDT. Empirical first-order-plus-dead-

Modeling

Nomenclature

- k = growth rate (Malthusian parameter) in the logistic process
- K = process gain in the first- and second-order processes
- k_{ab} = specific rate constant for the reaction $A \rightarrow B$
- t, t_{start} = process time and step-change point, respectively
- t_k = k th time instant in $t^{(combined)}$
- t_M, t_N, t_P = maximum time instant in $t^{(data)}$, $t^{(RK)}$ and $t^{(combined)}$, respectively
- $t^{(data)}$ = set of data sampling-time points
- $t^{(RK)}$ = set of Runge-Kutta simulation-time points
- $t^{(combined)}$ = union of $t^{(data)}$ and $t^{(RK)}$, arranged in ascending order
- Δt_k = k th simulation-time interval
- u = input variable in the first- and second-order process models
- x, x_0, x_{SS} = state variable, initial value and steady state value, respectively
- x_a, x_b, x_c = concentration (weight fraction) of component A, B and C, respectively; x_{a0}, x_{b0}, x_{c0} are the concentrations at $t = 0$
- y, y_k = process variable and k th sampled process variable, respectively, in a nonlinear model

Greek Letters

- α, β = real and imaginary part of the eigenvalues of the second-order under-damped process model
- $\delta_k^{(i)}$ = i th correction in the Runge-Kutta method for the k th simulation point
- ϵ = a logistic model parameter, asymptotic initial value
- λ = a logistic model parameter, reciprocal of the carrying capacity
- η = transient component of a second-order under-damped process
- τ, τ_n, τ_d = time constant for first-order process, second-order process and dead time, respectively
- τ_{start} = time instant where the step change in input variable occurs (used in FOPDT and SOUPDT model identifications)
- ζ = damping coefficient

time (FOPDT) models are used extensively in process control for tuning of proportional-integral-derivative (PID) controllers (2-4). Several existing methods can be used to estimate FOPDT models (3, 4). However, most of these methods rely on smoothed data to determine critical values, whereas the WLSQ method can be used with noisy data. Moreover, the WLSQ method can be applied to higher-order models with time-delay, including processes that contain inverse response behavior.

The FOPDT model for a unit step change in input variable u is given by:

$$\tau \frac{dx}{dt} + x = Ku(t - \tau_d) + x_0 \quad (1)$$

	A	B	C	D	E	F
1	τ	1.96869	RMS	0.169		
2	τ_d	3.71908				
3	K	3.02169				=SQRT(AVERAGE(F9:F48))
4	x_0	2.04102				
5	x_{SS}	5.06274				=B\$4+B\$3*1
6	t_{start}	2				
7						
8	t	x	x_{est}	τ	τ_d	RMS
9	0.01	2.01	2.04102	0.001	10	0.0096
10	0.51	2.04	2.04	15.06	10	1E-05
11				0.037	10	0.0372
12				0.08	10	4E-05
13				0.1	10	
14				0.00	10	
15				0	10	
16	3.56	2.08	2.04102	0.001		
47	15.08	5.02	5.03669	0.0003	10	0.0028
48	15.31	4.97	5.03956	0.0048	10	0.0484

Figure 1. A spreadsheet such as this can be used for FOPDT modeling. The RMS value is minimized using the Solver tool by adjusting the values for τ , τ_d , K and x_0 .

where $x(t) = x_0$ for $t \leq t_{start}$, $u(t) = 0$ for $t \leq t_{start}$, and $u(t) = 1$ for $t > t_{start}$.

The analytical solution is given by:

$$x(t) = x_0 \quad \text{for } t \leq t_{start} + \tau_d \quad (2a)$$

$$x(t) = x_{SS} + (x_0 - x_{SS}) \exp\left(-\frac{t - t_{start} - \tau_d}{\tau}\right) \quad \text{for } t > t_{start} + \tau_d \quad (2b)$$

where the steady-state value of x is $x_{SS} = K + x_0$.

Figure 1 is a spreadsheet for estimating the FOPDT model using the data in Table 1, with $t_{start} = 2.0$, and Figure 2 compares the predicted and observed data. The data have been partitioned into three phases: the initial phase ($t < t_{start}$), the transient phase ($t_{start} < t < 10$) and the final phase ($t > 10$), and the data within the initial and final phases were weighted to be ten times the data in the transient phase. This allows for a more-accurate estimate of the process gain, K , which is an important parameter for tuning controllers.

Example 2: SOUPDT. A significant improvement can be achieved by increasing the order of the empirical fit, for instance, using a second-order-underdamped-plus-dead-time (SOUPDT) model:

$$\tau_n^2 \frac{d^2x}{dt^2} + 2\tau_n\zeta \frac{dx}{dt} + x = Ku(t - \tau_d) + x_0 \quad (3)$$

where $x(t) = x_0$ and $dx/dt = 0$ for $t \leq t_{start}$, and $u(t) = 0$ for $t < t_{start}$ and $u(t) = 1$ for $t > t_{start}$.

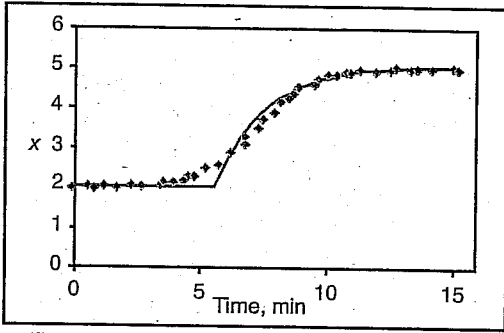


Figure 2. This plot compares the FOPDT model using parameters calculated by the spreadsheet in Figure 1 with the data in Table 1.

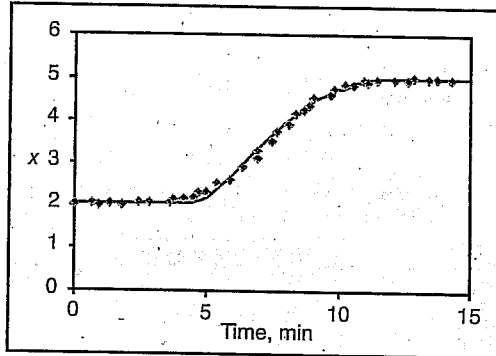


Figure 3. The SOUPDT model provides a better fit for the data in Table 1 than a first-order model.

The analytical solution of the SOUPDT model is:

$$x(t) = x_0 \quad \text{for } t \leq t_{start} + \tau_d \quad (4a)$$

$$x(t) = x_{SS} + (x_0 - x_{SS})\eta(t - t_{start} - \tau_d) \quad (4b)$$

$$\text{for } t > t_{start} + \tau_d$$

where the steady-state value of x is x_{SS} and

$$\eta(t) = e^{-\alpha t} \left(\cos(\beta t) + \frac{\alpha}{\beta} \sin(\beta t) \right); \quad \alpha = \frac{\xi}{\tau_n}; \quad \beta = \frac{\sqrt{1 - \xi^2}}{\tau_n}$$

Using the same weights as in Example 1, the weighted least-squares estimates for the SOUPDT model are: $K = 2.852$, $\tau_n = 1.979$, $\xi = 0.706$, $\tau_d = 2.263$, $x_0 = 2.028$. Figure 3 plots the results for this model.

Runge-Kutta least-squares method

Analytical solutions may be difficult to obtain when the process models are nonlinear. A numerical solution can be used to generate model predictions in these situations. In particular, the fourth-order Runge-Kutta method is useful for integrating the process models and supplying the predictions for the weighted least-squares method. This approach is referred to as the RKLSQ method.

For the nonlinear differential equation

$$dy/dt = f(t, y); \quad y(0) = y_0 \quad (5)$$

the fourth-order Runge-Kutta method (5) applies the following update formulas:

$$y_{k+1} = y_k + (1/6)[\delta_k^{(1)} + 2\delta_k^{(2)} + 2\delta_k^{(3)} + \delta_k^{(4)}] \quad (6)$$

where:

$$\begin{aligned} \delta_k^{(1)} &= \Delta t_k f(t_k, y_k), \\ \delta_k^{(2)} &= \Delta t_k f(t_k + 0.5\Delta t_k, y_k + 0.5\delta_k^{(1)}), \\ \delta_k^{(3)} &= \Delta t_k f(t_k + 0.5\Delta t_k, y_k + 0.5\delta_k^{(2)}), \\ \delta_k^{(4)} &= \Delta t_k f(t_k + \Delta t_k, y_k + \delta_k^{(3)}). \end{aligned}$$

The k th time interval, Δt_k , can be chosen to be uniform or can be adjusted to control the error with each update. In either case, a Runge-Kutta time sequence is obtained from:

$$t^{(RK)} = \{t_0^{(RK)}, t_1^{(RK)}, \dots, t_N^{(RK)}\} \quad (7)$$

The time sequence can be obtained from the raw data by:

$$t^{(data)} = \{t_0^{(data)}, t_1^{(data)}, \dots, t_M^{(data)}\} \quad (8)$$

where $t_0^{(data)} \geq t_0^{(RK)}$ and $t_M^{(data)} \leq t_N^{(RK)}$.

The two time sequences are then combined by inserting the sequence from $t^{(data)}$ into $t^{(RK)}$ such that:

$$t^{(combined)} = \left(t^{(RK)} \right) \cup \left(t^{(data)} \right) = \{t_0, t_1, \dots, t_p\} \quad (9)$$

where $t_k \subseteq t^{(RK)}$ or $t_k \subseteq t^{(data)}$, and $t_{k+1} > t_k$.

The Runge-Kutta method is then applied to the combined time sequence to generate the model predictions. The new sampling intervals, Δt_k , will likely be non-uniform when the real data are not sampled uniformly. Since the time-sequence insertion procedure yields smaller time intervals in the neighborhood of the insertion points, the convergence behavior of the the Runge-Kutta method guarantees that the accuracy of the corresponding predicted values will not be reduced (5).

The Runge-Kutta predictions of the process variables at the times corresponding to the data time sequence are extracted and aligned with the values of the process vari-

Table 1. Time series data set 1:

Time	X
0.01	2.01
0.61	2.04
0.88	1.98
1.25	2.04
1.76	2.00
2.36	2.06
2.77	2.04
3.56	2.08
3.65	2.13
4.06	2.14
4.48	2.19
4.66	2.29
4.94	2.29
5.35	2.50
5.86	2.59
6.32	2.89
6.88	3.10
6.88	3.28
7.43	3.50
7.61	3.73
8.03	3.89
8.30	4.15
8.63	4.25
8.81	4.36
9.00	4.53
9.64	4.60
9.78	4.72
10.19	4.83
10.52	4.83
10.88	4.92
11.07	4.89
11.44	4.95
12.08	4.94
12.64	4.95
12.82	5.02
13.42	4.97
13.70	5.00
14.25	4.97
15.08	5.02
15.31	4.97

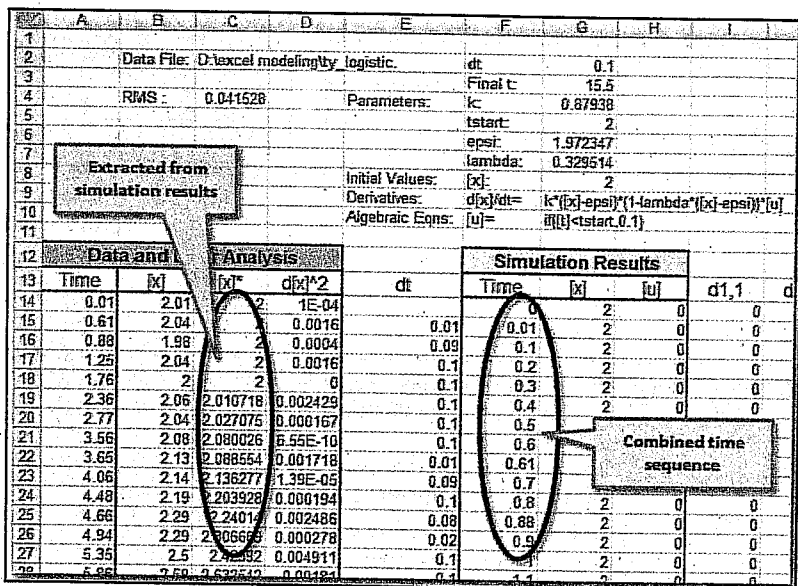


Figure 4. This spreadsheet implements the RKLSQ method using the add-in tool RK4.

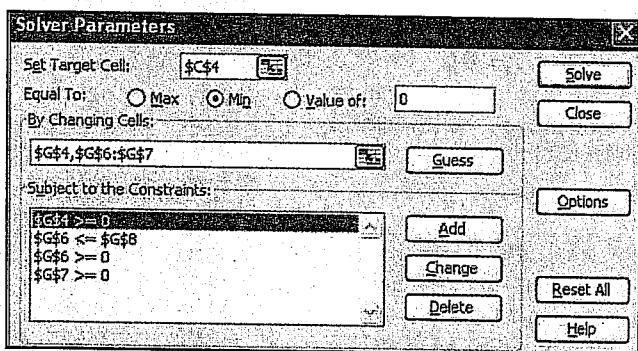


Figure 5. Solver is used to minimize the RMS cell (the cell addresses correspond to the spreadsheet shown in Figure 4).

ables from the time-series data. Then the residual errors are evaluated to determine the RMS error. Finally, the process parameters are estimated by minimizing the RMS error.

Instead of an empirical approach (e.g., the FOPDT and the SOUPDT models described previously), a logistic process model can be used to implement the RKLSQ method. This is given by:

$$\frac{dx}{dt} = k(x - \epsilon)[1 - \lambda(x - \epsilon)]u \quad (10)$$

where $x(t) = x_0$ for $t \leq t_{start}$, $u(t) = 0$ for $t \leq t_{start}$, and $u(t) = 1$ for $t > t_{start}$.

An add-in tool for Microsoft Excel, RK4, implements the fixed-interval Runge-Kutta method (see box on p. 50). It is available for free download, along with the related manuals, at www.chem.mtu.edu/~tbco/cm416/RK4_v3_0.html. Figure 4 is a snapshot of the spreadsheet using RK4 applied to the

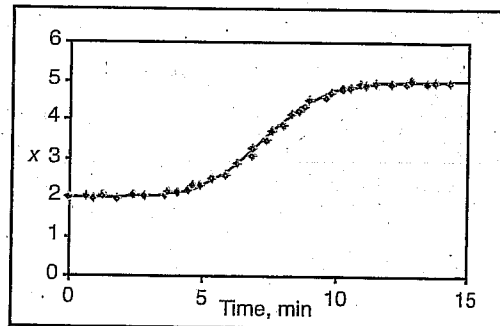


Figure 6. The logistic model provides a better fit for the Table 1 data.

data in Table 1 to obtain initial estimates of the modeling parameters. The parameters are then estimated using Solver, with some additional constraints applied to guarantee bounds on the simulation results, as shown in Figure 5. In this case, the initial condition was fixed at $x_0 = 2.00$.

The parameters for the logistic model were then estimated to be: $k = 0.879$, $\epsilon = 1.972$ and $\lambda = 0.330$. A plot of the logistic model with estimates obtained from the RKLSQ approach is shown in Figure 6.

Parameter estimation based on multiple data sets

Sometimes the data used for parameter estimations come from different runs of the same process. The various runs may have different initial conditions or operating conditions but share the same subset of parameters. In this case, one worksheet is allotted for each data set, while another worksheet contains all the parameters and initial conditions. Figure 7 shows one possible arrangement of worksheets.

Example 3: Reversible reactions. Modeling parameters need to be estimated for the isothermal reversible reactions of chemicals A, B and C shown in Figure 8. When all are first-order reactions, an analytical approach that applies linear algebra to find the kinetic constants by matching straight-line reactions with eigenvectors of the linear system (6) is applicable.

The theory, however, breaks down when the reactions are not linear, such as the reactions in Figure 8, which are all first-order except the forward reaction from A to B, which is a second-order reaction. In this case, the RKLSQ approach can handle the parameter estimation as follows:

$$\begin{aligned} \frac{dx_a}{dt} &= -k_{ab}x_a^2 - k_{ac}x_a + k_{ba}x_b + k_{ca}x_c \\ \frac{dx_b}{dt} &= k_{ab}x_a^2 - k_{ba}x_b - k_{bc}x_b + k_{cb}x_c \\ x_c &= 1 - x_a - x_b \end{aligned} \quad (11)$$

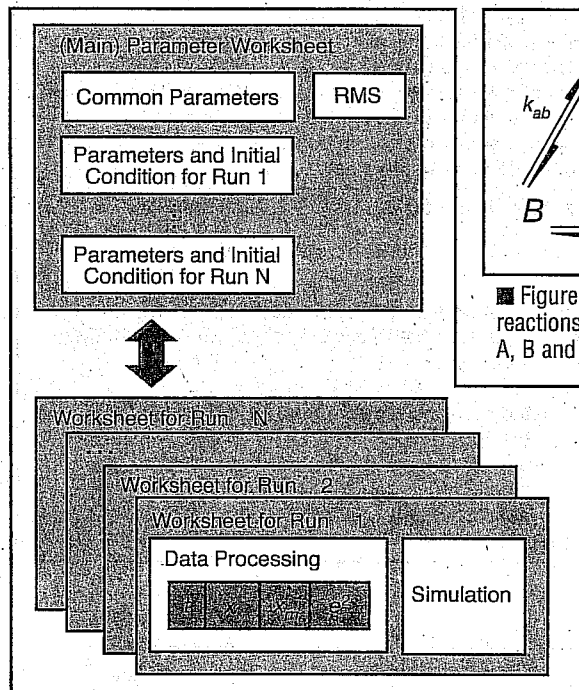


Figure 7: Multiple worksheets are set up to handle data from several different runs.

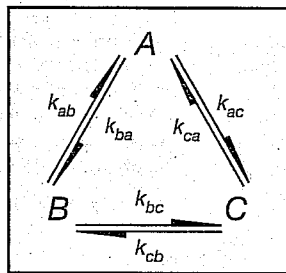


Figure 8: Reversible reactions among chemicals A, B and C.

Table 3: Parameter estimates using the RKLSQ method for the reaction system of Example 3.

	Original Input to the Simulation	Estimate	Relative Error
k_{ab}	4	4.082	2.05%
k_{ac}	2	1.933	-3.35%
k_{ba}	1.2	1.162	-3.17%
k_{bc}	0.2	0.231	15.50%
k_{ca}	1.5	1.523	1.53%
k_{cb}	0.8	0.771	-3.63%

with initial conditions $x_a(0) = x_{a0}$, $x_b(0) = x_{b0}$ and $x_c(0) = x_{c0}$.

Now consider the data in Table 2 for multiple runs of this reaction system. The data were obtained by simulating Eq. 11 using the data in Table 3, and including measurement noise.

Using the add-in tool RK4, a spreadsheet can be generated that implements the scheme shown in Figure 7 to handle multiple data sets (see box on p. 50). Figure 9 is a snapshot of the spreadsheet using RK4 to analyze the data in Table 2 to obtain initial parameter estimates. In this example, minimizing the value of the RMS adjusts the initial conditions automatically. Table 3 compares the estimated parameters, i.e., the kinetic rate constants for the reactions, to the original values used to generate the data in Table 2. The response of the system using the parameters obtained from the RKLSQ approach and the data are shown in Figure 10.

The relative error for k_{bc} is about 16%, which is due mainly to the fact that k_{bc} is so much smaller than the other constants (about an order of magnitude). The other parameters are within $\pm 4\%$ relative error.

Closing thoughts

The weighted-least-squares method for parameter estimation can be efficiently implemented in spreadsheets for problems involving non-uniformly sampled data as well as multiple sampling runs. For nonlinear systems,

Table 2: Data for reaction system of Example 3 (Figure 8).

Case 1			Case 2			Case 3		
Time	x_a	x_b	Time	x_a	x_b	Time	x_a	x_b
0.00	0.00	0.21	0.00	0.10	0.90	0.00	0.50	0.50
0.05	0.08	0.22	0.10	0.17	0.78	0.03	0.46	0.51
0.13	0.13	0.23	0.23	0.22	0.68	0.05	0.44	0.51
0.19	0.20	0.25	0.38	0.26	0.61	0.11	0.39	0.51
0.30	0.24	0.28	0.63	0.28	0.52	0.17	0.36	0.51
0.43	0.28	0.30	1.21	0.29	0.43	0.25	0.33	0.50
0.59	0.29	0.34	1.97	0.29	0.42	0.38	0.31	0.48
0.94	0.30	0.38				0.47	0.30	0.47
1.42	0.29	0.41				0.68	0.30	0.45
1.78	0.30	0.41				0.91	0.29	0.44
						1.38	0.30	0.42
						1.98	0.29	0.42

Literature Cited

- Marquardt, D. W., "An Algorithm for Least-Squares Estimation of Nonlinear Parameters," *J. Soc. Ind. Appl. Math.*, **11** (2), pp. 431-441 (1963).
- Cohen, G. H., and G. A. Coon, "Theoretical Consideration of Retarded Control," *Trans. ASME*, **75**, pp. 827-834 (1953).
- Bequette, B. W., "Process Control: Modeling, Design and Simulation," Prentice Hall, Upper Saddle River, NJ (2003).
- Riggs, J. B., and M. N. Karim, "Chemical and Bio-Process Control," Ferret Publishing, Lubbock, TX (2006).
- Hairer, E., et al., "Solving Ordinary Differential Equations I: Nonstiff Problems," Springer Verlag, New York, NY (2002).
- Wei, J., and C. D. Prater, "The Structure and Analysis of Complex Reaction Systems," *Advances in Catalysis*, Vol. 13. Academic Press, New York, NY (1962).
- Hairer, E., et al., "Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems," Springer-Verlag, New York, NY (2004).

TOMAS B. CO is an associate professor in the department of chemical engineering at Michigan Technological Univ. (Houghton, MI 49931; Phone: (906) 487-2144; E-mail: tbco@mtu.edu). His research interests include plantwide control and nonlinear process modeling using modulating functions and granular computation methods, such as fuzzy sets and neural networks. Another area of interest is building a theory of process-system integrity monitoring and control for the sustained operation of chemical processes in which process control and optimization are balanced with the health of the system away from failure conditions. He holds a BSc from the Univ. of the Philippines and a PhD from the Univ. of Massachusetts.

Implementing the RKLSQ Method Using the RK4 Add-in Tool

Case 1: Logistic Model

First, create a text file that includes the model statements shown in the box below (left), and save it as an *.rk4 file. Next, open an Excel workbook and activate RK4. When the front panel appears, open the *.rk4 file, and go to the "Initialization" tab. Select the "Parameter Estimation" option, and enter the name of the data file. Follow the procedures required to match data columns to variable labels, and a spreadsheet similar to Figure 4 should appear.

RK4 Model File for the Logistic Model

dt: 0.1
 tfinal: 15.5
 [x]; Initial Value: 2, Deriv: $k \cdot ([x] - \text{epsi}) \cdot (1 - \text{lambda} \cdot ([x] - \text{epsi})) \cdot [u]$
 [u]; Eqn: $\text{if}([t] < \text{tstart}, 0, 1)$
 k, Value: 2
 tstart, Value: 2
 epsi, Value: 1
 lambda, Value: 2

Case 2: System of Reversible Reactions

Create a text file that includes the model statements shown in the box below (right), and save it as an *.rk4 file. Follow the same steps as for Case 1, except three separate data files will need to be downloaded. This will alert RK4 to allot one worksheet for the parameters and three more worksheets corresponding to each of the data sets.

RK4 Model File for the System of Reversible Reaction

dt: 0.01
 tfinal: 2
 [xa], Initial Value: 0.3, Deriv: $-k_{ab} \cdot [xa]^2 - k_{ac} \cdot [xa] + k_{ba} \cdot [xb] + k_{ca} \cdot [xc]$
 [xb], Initial Value: 0.3, Deriv: $k_{ab} \cdot [xa]^2 + k_{cb} \cdot [xc] - k_{ba} \cdot [xb] - k_{bc} \cdot [xb]$
 [xc], Eqn: $1 - [xa] - [xb]$
 kab, Value: 2
 kac, Value: 2
 kba, Value: 2
 kbc, Value: 2
 kca, Value: 2
 kcb, Value: 2

the Runge-Kutta method can be combined with the WLSQ method, which can also be implemented in an Excel spreadsheet with the add-in tool RK4.

The RKLSQ implementation discussed here can be further enhanced. First, the optimizers that are included with spreadsheets, such as Solver, are highly dependent on initial guesses and yield local minima. Other optimizers, such as genet-

ic algorithms, can be applied to increase the probability of obtaining global minima of the RMS values. Second, the Runge-Kutta method used in RK4 is the explicit form. An implicit form is needed to handle systems containing stiff differential systems, e.g., processes containing a very wide range of fast and slow modes (7). And third, although models including delays are handled well in processes that have analytical solutions, e.g., FOPDT and SOUPDT models, the RKLSQ can be coupled with

data interpolation to obtain delayed values for non-linear processes.

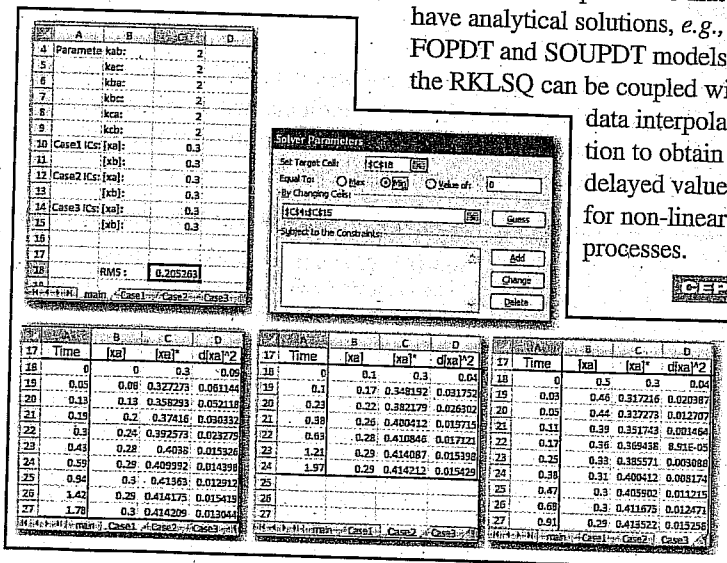


Figure 9. In this spreadsheet, the model parameters are in the main worksheet, while the three data sets are handled in the Case1, Case2 and Case3 worksheets.

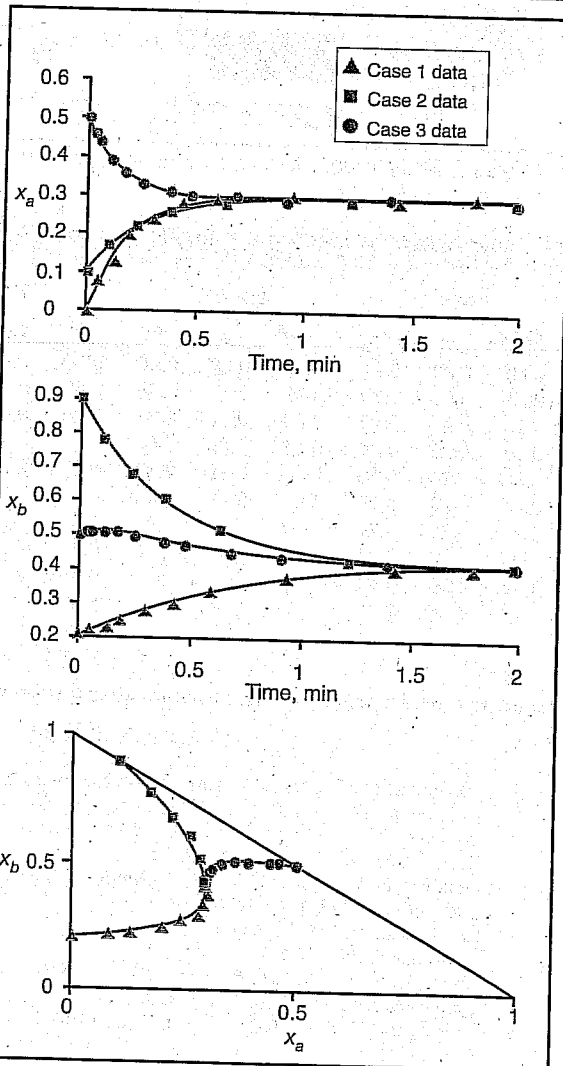


Figure 10. The modeled responses of the reaction system agree well with the data in Table 3.

February 2008

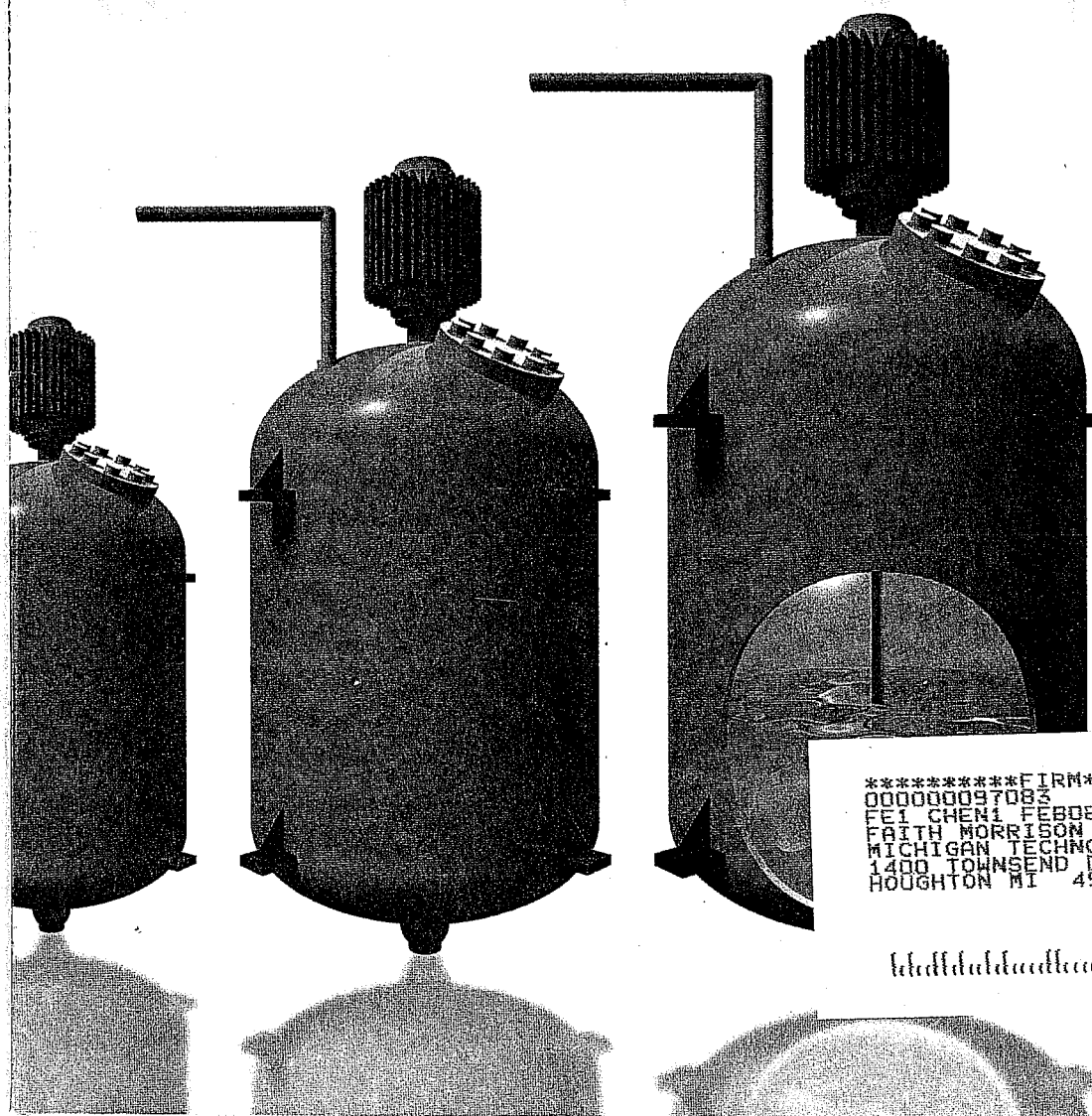
**Chemical
Engineering
Progress**

An AIChE Publication

CEP

www.aiche.org/cep

Scaling Up Bioreactor Agitation



**RFID — Rx for
Drug Security? p.6**

**Chemical Safety in
Warehouses p.35**

**Dealing with Climate
Change Locally p.40**

*****FIRM**CAR-RT LOT**C-002
00000097083
FE1 CHEN1 FEB08 0003 #244 #57798
FAITH MORRISON
MICHIGAN TECHNOLOGICAL UNIVERSITY
1400 TOWNSEND DR
HOUGHTON MI 49931-1200

|||||