

The object of this column is to enhance our readers' collections of interesting and novel problems in chemical engineering. We request problems that can be used to motivate student learning by presenting a particular principle in a new light, can be assigned as novel home problems, are suited for a collaborative learning environment, or demonstrate a cutting-edge application or principle. Manuscripts should not exceed 14 double-spaced pages and should be accompanied by the originals of any figures or photographs. Please submit them to Dr. Daina Briedis (e-mail: briedis@egr.msu.edu), Department of Chemical Engineering and Materials Science, Michigan State University, East Lansing, MI 48824-1226.

BIOKINETIC MODELING OF IMPERFECT MIXING IN A CHEMOSTAT *an Example of Multiscale Modeling*

MICHAEL B. CUTLIP

University of Connecticut • Storrs, CT 06269

NEIMA BRAUNER

Tel-Aviv University • Tel-Aviv 69978, Israel

MORDECHAI SHACHAM

Ben-Gurion University of the Negev • Beer-Sheva 84105, Israel

Mathematical software packages such as Excel[®], MAPLE[™], MATHCAD[®], MATLAB[®], Mathematica[®], and POLYMATH[™] are currently used routinely for numerical problem solving in engineering education.^[1, 2] From the numerical solution perspective, it is convenient to characterize the various problems as Single Model-Single Algorithm (SMSA) problems and complex problems with some combination of Multiple Models and Multiple Algorithms (MMA). A typical example of an SMSA problem is the solution of a system of ordinary differential equations coupled with explicit algebraic equations where one numerical integration algorithm (such as the 4th order Runge-Kutta) can be used to solve the problem (e.g., steady-state operation of a tubular reactor).

The application of mathematical software packages for solving SMSA problems has essentially replaced all other solution techniques, as can be seen in many recent textbooks (see, for example, Fogler^[3]). For complex and/or multi-scale problems, however, the solution process is often more involved.

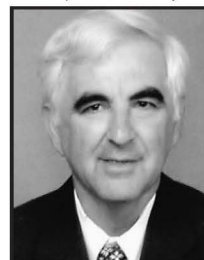
The types of models included in the "complex" category are:

1. *Multiple Model-Single Algorithm (MMSA) Problem.*
A typical example is the cyclic operation of a semi-batch bioreactor.^[4] The three modes of operation of the



Michael B. Cutlip is professor emeritus of the Chemical, Materials, and Biomolecular Engineering Department at the University of Connecticut and has served as department head and director of the university's Honors Program. He has B.Ch.E. and M.S. degrees from Ohio State and a Ph.D. from the University of Colorado. His current interests include the development of general software for numerical problem solving and application to chemical and biochemical engineering.

Neima Brauner is a professor in the School of Mechanical Engineering and Heat Transfer at the Tel-Aviv University, Tel Aviv, Israel. She received her B.Sc. and M.Sc. in chemical engineering from the Technion Institute of Technology, Haifa, Israel, and her Ph.D. in mechanical engineering from the Tel-Aviv University. Her research interests include hydrodynamics and transport phenomena in two-phase flow systems.



Mordechai Shacham is the Benjamin H. Swig professor and head of the Department of Chemical Engineering at the Ben-Gurion University of the Negev in Israel. He received his B.Sc. and D.Sc. degrees from the Technion, Israel Institute of Technology. His research interests include analysis, modeling, and regression of data, applied numerical methods, and prediction and consistency analysis of physical properties.

bioreactor (initialization, processing, and harvesting) are represented by different models comprising ordinary differential equations and explicit algebraic equations. All models can be solved by one numerical integration algorithm (such as the 4th order Runge-Kutta).

2. Single Model-Multiple Algorithm (SMMA) Problem.
Typical examples are the solution of two-point boundary value problems, where the integration of the model is carried out in the inside loop and a nonlinear equation solver algorithm adjusts the boundary values in an outer loop, or the solution of differential-algebraic systems of equations where the same algorithms are used but in an opposite hierarchy.
3. Multiple Model-Multiple Algorithm (MMMA) Problem.
A typical example is the modeling of an exothermic batch reactor, where the two stages of operation (heating and cooling) require different models and different integration algorithms (stiff and non-stiff).

The solution of such complex problems can be rather cumbersome and time consuming even if mathematical software packages are used, as manual transfer of data from one model/problem to another and consecutive manual reruns are often required. Combining the use of several software packages of various levels of complexity, flexibility and user friendliness, however, can considerably reduce the time and effort required for solving complex models.

Following this premise, the models representing the various stages of the problems can be coded and tested using a software package (for example, POLYMATH^[5]) that requires very little technical coding effort. After testing each of the modules separately, they are combined into one program using a programming language, or a mathematical software package that supports programming (say, MATLAB^[6]). To minimize the probability of introducing errors into the model equations, the POLYMATH input for the various modules can be automatically converted within POLYMATH to MATLAB code. This allows MATLAB functions to be created that enable the consecutive and repetitive calls to the various models, apply the appropriate solution algorithms, and assign the hierarchy of the computations during the solution.

A homework assignment that demonstrates this suggested approach is the following problem of biokinetic modeling of a chemostat with imperfect mixing. This problem is a modified version of a problem presented by Cutlip and Shacham.^[7] The solution algorithm presented for this problem includes the use of various computing tools in the different stages of the problem solution (the solution of an SMSA problem, parametric runs of an SMSA problem, and the solution of an SMMA problem).

PROBLEM BACKGROUND

Biokinetic Modeling of Imperfect Mixing in a Chemostat

A chemostat is usually considered to be a completely mixed reactor; however, this is not always the case. Consider the situation where the chemostat may be considered to be modeled as a reactor with a completely-mixed volume V_1 (dm^3) that interacts with another completely-mixed volume V_2 (dm^3) as shown in Figure 1. Volume V_2 with an exchange flow rate F_2 (dm^3/hr) may be considered to model the poorly mixed regions within a production fermenter. The microbial

TABLE 1
POLYMATH Model for the Chemostat with Imperfect Mixing

No.	Equation # Comment
1	$f(S1) = F1*S0 + F2*S2 - (1/Y_{xs}) * (\text{mum} * S1 / (Ks + S1)) * X1 * V1 - F1 * S1 - F2 * S1$ # Substrate balance on volume V1
2	$f(S2) = F2*S1 - (1/Y_{xs}) * (\text{mum} * S2 / (Ks + S2)) * X2 * V2 - F2 * S2$ # Substrate balance on volume V2
3	$f(X1) = F2*X2 + (\text{mum} * S1 / (Ks + S1) - kd) * X1 * V1 - F1 * X1 - F2 * X1$ # Cell balance on volume V1
4	$f(X2) = F2*X1 + (\text{mum} * S2 / (Ks + S2) - kd) * X2 * V2 - F2 * X2$ # Cell balance on volume V2
5	$F1 = 0.17$ # Feed flow rate to volume V1 (dm^3/hr)
6	$F2 = 0.2 * F1$ # Feed flow rate to volume V2 (dm^3/hr)
7	$P1 = Y_{ps} * (S0 - S1)$ # Production (g/dm^3)
8	$D = F1 / (V1 + V2)$ # Dilution rate (1/hr)
9	$S0 = 0.6$ # Feed substrate concentration (g/dm^3)
10	$kd = 0.002$
11	$Y_{xs} = 0.4$ # Yield coefficient (g cells/g substrate)
12	$Y_{ps} = 0.2$ # Yield coefficient (g product/g substrate)
13	$Ks = 0.2$ # Saturation constant (g substrate/ dm^3)
14	$\text{mum} = 0.2$ # Maximal specific growth rate (1/hr)
15	$V1 = 1.7$ # Volume V1 (dm^3)
16	$V2 = 0.3$ # Volume V2 (dm^3)
17	$PR_DX1 = D * X1$ # Cell production rate (g/hr)
18	$PR_DP1 = D * P1$ # Product production rate (g/hr)
19	$S1(0) = 0$ # Substrate concentration in volume V1 (g/dm^3)
20	$S2(0) = 0$ # Substrate concentration in volume V2 (g/dm^3)
21	$X1(0) = 0.025$ # Cell concentration in volume V1 (g/dm^3)
22	$X2(0) = 0.025$ # Cell concentration in volume V2 (g/dm^3)

system to be modeled involves substrate S (g/dm^3) going to product P (g/dm^3) only under the action of cells X (g/dm^3). The following separate balances on the substrate, cells, and product in each reactor volume use Monod kinetics and a cell death rate constant given by k_d (hr^{-1}).

Steady-State Substrate Balance on Volume V_1

$$F_1 S_0 + F_2 S_2 + \frac{1}{Y_{X/S}} \left(\frac{\mu_m S_1}{K_S + S_1} \right) X_1 V_1 = F_1 S_1 + F_2 S_1 \quad (1)$$

where F is flow rate (dm^3/hr), $Y_{X/S}$ is yield coefficient ($\text{g cells}/\text{g substrate}$), μ_m is the maximal specific growth rate (hr^{-1}), and K_S is the saturation constant ($\text{g substrate}/\text{dm}^3$). The indexes 0, 1, and 2 are used as shown in Figure 1.

Steady-State Substrate Balance on Volume V_2

$$F_2 S_1 + \frac{1}{Y_{X/S}} \left(\frac{\mu_m S_2}{K_S + S_2} \right) X_2 V_2 = F_2 S_2 \quad (2)$$

Steady-State Cell Balance on Volume V_1

$$F_2 X_2 + \left(\frac{\mu_m S_1}{K_S + S_1} - k_d \right) X_1 V_1 = F_1 X_1 + F_2 X_1 \quad (3)$$

Steady-State Cell Balance on Volume V_2

$$F_2 X_1 + \left(\frac{\mu_m S_2}{K_S + S_2} - k_d \right) X_2 V_2 = F_2 X_2 \quad (4)$$

Overall Steady-State Material Balance for the Product

$$P_1 = Y_{P/S} (S_0 - S_1) \quad (5)$$

where $Y_{P/S}$ is the yield coefficient ($\text{g product}/\text{g substrate}$).

PROBLEM STATEMENT

Microbial growth has been studied in a continuous culture, and the following parameters were obtained: $\mu_m = 0.2 \text{ h}^{-1}$, $K_S = 0.2 \text{ g}/\text{dm}^3$, $k_d = 0.002 \text{ hr}^{-1}$, $Y_{X/S} = 0.4 \text{ g cells}/\text{g substrate}$, and $Y_{P/S} = 0.2 \text{ g product}/\text{g substrate}$. Tracer studies have indicated that the incomplete mixing can be described by a well-mixed

volume $V_1 = 1.7 \text{ dm}^3$ and a volume of $V_2 = 0.3 \text{ dm}^3$ with an exchange flow rate F_2 . The flow rate relationship with the overall flow rate to chemostat, F_1 , is given by $F_2 = 0.2 F_1$ in dm^3/hr . Chemostat operation is such that $F_1 = 0.17 \text{ dm}^3/\text{hr}$, $X_0 = 0$ and $S_0 = 0.6 \text{ g}/\text{dm}^3$, and the endogenous metabolism can be neglected.

- Create a single graph of S_p , X_p , and P_p vs. the dilution rate defined by $D = F_1/V_1$.
- Plot the cell production rate, the product DX_p , and the product production rate, the product of DP_p , as functions of the dilution rate between 0.05 and 0.130 hr^{-1} .
- Estimate the dilution rate that will maximize the production rate, DX_p , for the cells and the dilution rate that will maximize the production rate, DP_p , for the product.

PROBLEM SOLUTION

Modeling the Chemostat and Solving the Single Model-Single Algorithm (SMSA) Problem

The mathematical model of the chemostat can be formulated as a system of nonlinear algebraic equations (NLEs) that can be solved by a single algorithm. This simple, uncomplicated model can be easily solved with POLYMATH version 6.1 to obtain the solution of this SMSA problem.

The complete POLYMATH code for the chemostat model is given in Table 1. The model includes four implicit nonlinear algebraic equations that are obtained from the material balances. The POLYMATH model (including the "comments," which start with the # sign) provides complete documentation of the equations, the values of the constants, and the initial estimates used for the four unknowns: S_1 , S_2 , X_1 , and X_2 . Statements 1 through 4 present the implicit equations for obtaining the substrate concentration in the well-mixed volumes (S_1 , S_2 , respectively), and the cell concentration in the well-mixed volumes (X_1 , X_2 , respectively). Explicit variables and constants are described in statements 5-18. Initial estimates for the unknowns in the nonlinear equations are provided in lines 19 to 22.

The results for the case where $F_1 = 0.17 \text{ dm}^3/\text{hr}$ and the initial estimates $S_{1,0} = S_{2,0} = 0$, $X_{1,0} = 0.025$, and $X_{2,0} = 0.025$ are given in Table 2. For this case with the dilution rate $D =$

Variable	Value	f(x)	Initial Guess
S_1 (g/dm^3)	0.1821	4.20E-11	0
S_2 (g/dm^3)	0.03589	3.91E-11	0
X_1 (g/dm^3)	0.1631	-1.68E-11	0.025
X_2 (g/dm^3)	0.2178	-1.56E-11	0.025
D (1/hr)	0.085		
F_1 (dm^3/hr)	0.17		
F_2 (dm^3/hr)	0.034		
PR_DPI (g/hr)	0.00711		
PR_DX1 (g/hr)	0.01387		

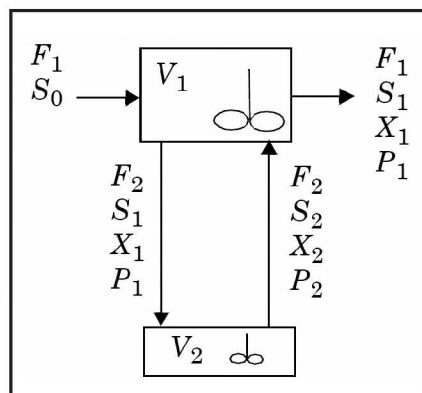


Figure 1.
Chemostat model.

TABLE 3 MATLAB Function (Model) for the Chemostat with Imperfect Mixing	
No.	Equation % Comment
1	function fx = MNLEfun(x, F1);
2	S1 = x(1); %Substrate concentration in volume V1 (g/dm ³)
3	S2 = x(2); %Substrate concentration in volume V2 (g/dm ³)
4	X1 = x(3); %Cell concentration in volume V1 (g/dm ³)
5	X2 = x(4); %Cell concentration in volume V2 (g/dm ³)
6	F1 = 0.17; %Feed flow rate to volume V1 (dm ³ /hr)
7	F2 = 0.2 * F1; %Feed flow rate to volume V2 (dm ³ /hr)
8	Yps = 0.2; %Yield coefficient (g product/g substrate)
9	V2 = 0.3; %Volume V2 (dm ³)
10	S0 = 0.6; %Feed substrate concentration (g/dm ³)
11	kd = 0.002; %Cell death rate (1/hr)
12	Yxs = 0.4; %Yield coefficient (g cells/g substrate)
13	P1 = Yps * (S0 - S1); %Production (g/dm ³)
14	Ks = 0.2; %Saturation constant (g substrate/dm ³)
15	mum = 0.2; %Maximal specific growth rate (1/hr)
16	V1 = 1.7; %Volume V1 (dm ³)
17	D = F1 / (V1 + V2); %Dilution rate (1/hr)
18	PR_DX1 = D * X1; %Cell production rate (g/hr)
19	PR_DPI = D * P1; %Product production rate (g/hr)
20	fx(1,1) = F1 * S0 + F2 * S2 - (1 / Yxs * mum * S1 / (Ks + S1) * X1 * V1) - (F1 * S1) - (F2 * S1); %Substrate balance on volume V1
21	fx(2,1) = F2 * S1 - (1 / Yxs * mum * S2 / (Ks + S2) * X2 * V2) - (F2 * S2); %Substrate balance on volume V2
22	fx(3,1) = F2 * X2 + (mum * S1 / (Ks + S1) - kd) * X1 * V1 - (F1 * X1) - (F2 * X1); %Cell balance on volume V1
23	fx(4,1) = F2 * X1 + (mum * S2 / (Ks + S2) - kd) * X2 * V2 - (F2 * X2); %Cell balance on volume V2

TABLE 4 Part of the MATLAB "Main Program" for Parametric Studies with the Chemostat	
No.	Equation % Comment
1	options = optimset('Diagnostics','off','TolFun',[1e-9],'TolX',[1e-9]);
2	Yps = 0.2; S0 = 0.6; kd = 0.002; Yxs = 0.4; Ks = 0.2;
3	mum = 0.2; V1 = 1.7; V2 = 0.3;
4	F1=0.1; %Initial feed flow rate to volume V1 (dm ³ /hr)
5	xguess = [0 0 0.025 0.25]; % initial guess vector
6	for k=1:16
7	xsolv=fsolve(@MNLEfun,xguess,options,F1);
8	S1(k)=xsolv(1); S2(k)=xsolv(2); X1(k)=xsolv(3); X2(k)=xsolv(4);
9	F1list(k)=F1; D(k) = F1 / (V1 + V2); P1(k)= Yps * (S0 - S1(k));
10	PR_DX1(k) = D(k) * X1(k); PR_DPI(k) = D(k) * P1(k);
11	F1=F1+0.01; %Incrementing feed flow rate to volume V1 (dm ³ /hr)
12	end

0.085 hr⁻¹, the cell production rate $DX_1 = 0.0139$ g/hr and the product production rate $DP_1 = 0.00711$ g/hr. Lower initial values of $X_{1,0} = X_{2,0}$ that are less than 0.0247 g/dm³ result in negligible steady-state reaction corresponding to cell washout operation. Thus the simulated chemostat has a critical value of initial cell concentration that leads to a sustained steady-state biochemical reaction. The production rates associated with the operation where washout of the cells is avoided will be studied in more detail.

Parametric Studies on the Chemostat

Parametric runs, requested in the second part of the assignment, can be carried out with POLYMATH by manually changing the parameter values. This approach, however, is inefficient and cumbersome—particularly for problems where there are many parameters and a wide range of parameter values to be considered. In such cases, programming is desirable for repetitive solution of the problem with the various parameter values. One option is to carry out the parametric runs efficiently using MATLAB. The MATLAB function representing the operation of the chemostat can be automatically and efficiently generated by POLYMATH (Table 3). Note that MATLAB requires input of the variable values into the function in a single array (**x**, in this case), and return of the function values in a single array (**fx**, lines 20-23 in Table 3). The variable values are put back into variables with the same names as used in the POLYMATH model (lines 2-5) to make the MATLAB code more meaningful. POLYMATH orders the basic model equations sequentially as required by MATLAB and converts any needed intrinsic functions and logical expressions.

Convenient parametric runs can be made for various values of the feed flow rate (F_1), and this variable can be added as an input parameter to the MNLEfun function (Table 3). A main program can be prepared that changes the value of F_1 , solves the system of nonlinear equations, collects the pertinent data, and plots the results of the parametric runs. Part of this main program is shown in Table 4. The value of F_1 is changed starting at $F_1 = 0.1$ up to $F_1 = 0.25$ with steps of 0.01. The MATLAB library function *fsolve* is used to solve the system of algebraic equations as shown in line 7 of Table 4. The variable values needed for preparing the various plots are calculated and stored in lines 8 through 10.

Excel^[8] can also be used for carrying out the parametric runs efficiently. The model can be automatically exported from POLYMATH to Excel with a single key press. Part of the Excel worksheet as generated by POLYMATH is shown in Table 5, where the variable cell calculations are indicated. The variable names are translated to cell addresses, a new equation that calculates the sum of squares of the function values is added, and the equations are rearranged in a form that is appropriate for solving the equation using the *solver* add-in available within Excel. The complete worksheet with the solution obtained using solver is shown in Table 6 (next page). The numerical results are identical to those obtained by POLYMATH. The variable names in column B, the POLYMATH equations in column D, and the variable descriptions in column E provide complete documentation for the Excel formulas in column C.

Solution of the system of equations using *solver* for various values of F_1 requires the creation of a macro or a VBA (Visual Basic for Applications^[8]) program. A plot of S_1 , X_1 , and P_1 as functions of the dilution rate is shown in Figure 2, and the cell and product production rates are plotted in Figure 3. Maximum points for the two production rates in the vicinity of $D = 0.1 \text{ hr}^{-1}$ can be observed in this figure. A more precise determination of the maximum is discussed in the next section.

Maximization of the Production Rates by Solving an SMMA Problem

The two optimization problems can be posed as the following minimization problems:

$$\min_{F_1} -DX_1 \text{ and } \min_{F_1} -DP_1 \text{ where } D = F_1 / V_1 \quad (6)$$

(The minus signs in front of DX_1 and DP_1 are used to convert the maximization problems into minimization problems).

The calculation of D , X_1 , and P_1 associated with a particular value of F_1 involves the solution of a system of NLEs, while a minimization algorithm is required in order to find the values of F_1 that satisfy Eq. (1). This is a single model (the chemostat) and multiple algorithms (one for solution of NLEs and one for minimization) problem.

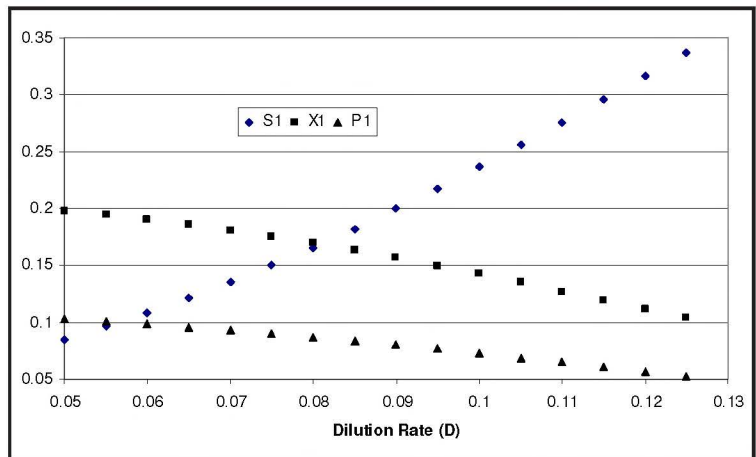


Figure 2. Plot of S_1 , X_1 , and P_1 as functions of dilution rate.

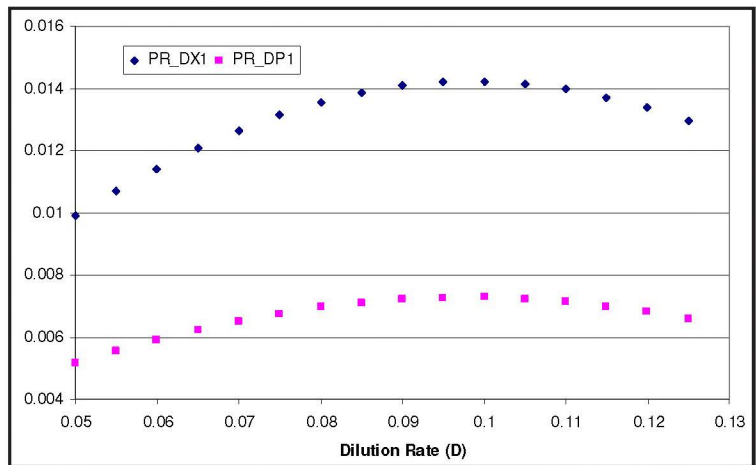


Figure 3. Cell production rate (PR_{DX1}) and product production rate (PR_{DP1}) as functions of dilution rate.

POLYMATH Model of the Chemostat Exported to Excel with Display Formulas Option.			
	A	B	C
1			POLYMATH NLE Migration Document
2		Variable	Value
3	Explicit Eqs	F1	=0.17
4		F2	=(0.2 * C3)
5		P1	=(C10 * (C7 - C17))
6		D	=(C3 / (C13 + C14))
7		S0	=0.6
8		kd	=0.002
9		Yxs	=0.4
10		Yps	=0.2
11		Ks	=0.2
12		mum	=0.2
13		V1	=1.7
14		V2	=0.3
15		PR_DX1	=(C6 * C19)
16		PR_DP1	=(C6 * C5)
17	Implicit Vars	S1	0
18		S2	0
19		X1	0.025
20		X2	0.025
21	Implicit Eqs	f(S1)	=((((C3 * C7) + (C4 * C18)) - (((1 / C9) * ((C12 * C17) / (C11 + C17))) * C19) * C13)) - (C3 * C17)) - (C4 * C17))
22		f(S2)	=((((C4 * C17) - (((1 / C9) * ((C12 * C18) / (C11 + C18))) * C20) * C14)) - (C4 * C18))
23		f(X1)	=((((C4 * C20) + (((C12 * C17) / (C11 + C17)) - C8) * C19) * C13)) - (C3 * C19)) - (C4 * C19))
24		f(X2)	=((((C4 * C19) + (((C12 * C18) / (C11 + C18)) - C8) * C20) * C14)) - (C4 * C20))
25	Sum of Squares:		=((((C21 ^ 2) + (C22 ^ 2)) + (C23 ^ 2)) + (C24 ^ 2))

The MATLAB library function *fminbnd* for single-value minimization can be used for finding the minimum of the functions in Eq. (1). In order to carry out the minimization, two new functions should be prepared. The first one (shown in Table 7) obtains F_1 as input, uses the *fsolve* library function to solve the chemostat model, and returns $-DX_1$ to the calling function. The second function does the same except that it returns the value of $-DP_1$. Two calls to the library function *fminbnd* identify the highest production rate for cells $DX_1 = 0.0142 \text{ g/hr}$ at a dilution rate of $D = 0.0986 \text{ hr}^{-1}$ and the highest production rate for product $DP_1 = 0.00727 \text{ g/hr}$ at a dilution rate of $D = 0.0979 \text{ hr}^{-1}$.

CONCLUSIONS

The example presented here provides an opportunity to practice several aspects of modeling and computation:

- Modeling of a bio-reactor and imperfect mixing.
- Categorizing problems according to the number of models and number of algorithms involved.
- Solving SMSA problems with a software package.
- Using Excel (VBA) or MATLAB programming for parametric runs of SMSA problems.
- Using MATLAB programming for solving SMMA problems.

We suggest that a combination of three popular packages—POLYMATH, Excel, and MATLAB—enables the solution of problems of increasing complexity in the educational setting. The example presented is suitable for courses in chemical reaction engineering, biochemical engineering, numerical methods, and optimization.

The POLYMATH and MATLAB programs used in this study are available at the site <[ftp://ftp.bgu.ac.il/shacham/chemostat/](http://ftp.bgu.ac.il/shacham/chemostat/)>.

REFERENCES

1. Cutlip, M., J.J. Hwalek, H.E. Nuttall, M. Shacham, J. Brule, J. Widman, T. Han, B. Finlayson, E.M. Rosen, and R. Taylor, "A Collection of 10 Numerical Problems in Chemical Engineering Solved by Various Mathematical Software Packages," *Computer Applications in Engineering Education*, **6**, 169 (1998)
2. Shacham, M., and M.B. Cutlip, "Selecting the Appropriate Numerical Software for a Chemical Engineering Course," *Computers and Chemi-*

TABLE 6
Excel Worksheet with Numerical Results and Documentation for the Chemostat Problem.

	A	B	C	D	E	F
1	POLYMATH NLE Migration Document					
2		Variable	Value		Polymath Equation	Comments
3	Explicit Eqs	F1	0.17		$F1=0.17$	Feed flow rate to volume V1 (dm ³ /hr)
4		F2	0.034		$F2=0.2*F1$	Feed flow rate to volume V2 (dm ³ /hr)
5		P1	0.083588165		$P1=Yps*(S0-S1)$	Production (g/dm ³)
6		D	0.085		$D=F1/(V1+V2)$	Dilution rate (1/hr)
7		S0	0.6		$S0=0.6$	Feed substrate concentration (g/dm ³)
8		kd	0.002		$kd=0.002$	Cell death rate (1/hr)
9		Yxs	0.4		$Yxs=0.4$	Yield coefficient (g cells/g substrate)
10		Yps	0.2		$Yps=0.2$	Yield coefficient (g product/g substrate)
11		Ks	0.2		$Ks=0.2$	Monod constant (g substrate/dm ³)
12		mum	0.2		$mum=0.2$	Maximal specific growth rate (1/hr)
13		V1	1.7		$V1=1.7$	Volume V1 (dm ³)
14		V2	0.3		$V2=0.3$	Volume V2 (dm ³)
15		PR_DX1	0.013867371		$PR_DX1=D*X1$	Dilution rate (1/hr)
16		PR_DP1	0.007104994		$PR_DP1=D*P1$	Cell production rate (g/hr)
17	Implicit Vars	S1	0.182059175		$S1(0)=0$	Substrate concentration in volume V1 (g/dm ³)
18		S2	0.035888744		$S2(0)=0$	Substrate concentration in volume V2 (g/dm ³)
19		X1	0.163145542		$X1(0)=0.025$	Cell concentration in volume V1 (g/dm ³)
20		X2	0.217791093		$X2(0)=0.025$	Cell concentration in volume V2 (g/dm ³)
21	Implicit Eqs	f(S1)	-7.7311E-07		$f(S1)=F1*S0+F2*S2-(1/Yxs)*(mum*S1/(Ks+S1))*X1*V1-F1*S1-F2*S1$	
22		f(S2)	-5.0319E-07		$f(S2)=F2*S1-(1/Yxs)*(mum*S2/(Ks+S2))*X2*V2-F2*S2$	
23		f(X1)	8.79323E-07		$f(X1)=F2*X2+(mum*S1/(Ks+S1)-kd)*X1*V1-F1*X1-F2*X1$	
24		f(X2)	-5.0428E-07		$f(X2)=F2*X1+(mum*S2/(Ks+S2)-kd)*X2*V2-F2*X2$	
25	Sum of Squares:		1.8784E-12		$F = f(S1)^2 + f(S2)^2 + f(X1)^2 + f(X2)^2$	

TABLE 7
A Function for Calculating the Cell Production Rate for a Single Value of F_1

No.	Equation % Comment
1	function PR_DX=ProdRateCell(F1) %Cell production rate (g/hr)
2	V1 = 1.7; %Volume V1 (dm ³)
3	V2 = 0.3; %Volume V2 (dm ³)
4	xguess = [0 0 0.025 0.025]; %initial guess vector
5	options = optimset('Diagnostics','off','TolFun',[1e-9],'TolX',[1e-9]);
6	xsolv=fsolve(@MNLEfun,xguess,options,F1);
7	X1=xsolv(3); %Cell concentration in volume V1 (g/dm ³)
8	D = F1 / (V1 + V2); %Dilution rate (1/hr)
9	PR_DX = -D* X1; %Cell production rate (g/hr)

cal Engineering, **23**(suppl.), S645 (1999)

3. Fogler, H.S., *Elements of Chemical Reaction Engineering*, 4th Ed, Prentice-Hall, Upper Saddle River, New Jersey (2005)
4. Cutlip, M.B., and M. Shacham, "Modular and Multilayer Modeling – Application to Biological Processes" pp. 1019-1024 in V. Plesu and P. S. Agaci (Editors), *Proceedings of the 17th European Symposium on Computer Aided Process Engineering*, Elsevier (2007)
5. POLYMATH is a product of Polymath Software <<http://www.polymath-software.com>>.
6. MATLAB is a trademark of The Math Works, Inc. <<http://www.mathworks.com>>
7. Cutlip, M.B., and M. Shacham, *Problem Solving in Chemical and Biochemical Engineering with POLYMATH, Excel and MATLAB*, 2nd Ed., Prentice-Hall, Upper Saddle River, New Jersey (2008)
8. Visual Basic for Applications and Excel are trademarks of Microsoft Corporation <<http://www.microsoft.com>> □