17th European Symposium on Computer Aided Process Engineering – ESCAPE17
V. Plesu and P.S. Agachi (Editors)

Modular and Multilayer Modeling – Application to Biological Processes

Michael B. Cutlip[a] and Mordechai Shacham[b]

[a]University of Connecticut, Storrs,CT 06269,USA, E-mail: michael.cutlip@uconn.edu
[b]Ben-Gurion University, Beer-Sheva 84105,Israel, E-mail: shacham@bgu.ac.il

Abstract

Construction of dynamic simulation programs for complex processes is accomplished by a modular and sequential approach. The complex simulation model is broken down into smaller modules. These modules are modeled and tested using a mathematical software package that requires modest technical coding efforts for computer implementation of the model. The verified model is then used as the basis for the program that represents the entire process and for documentation. The proposed approach is demonstrated for the construction of a simulation program for semi-continuous fed-batch and cyclic–fed batch operation of a bioreactor. A simple and straightforward extension of the program enables optimization of the process with respect to various input variables. The effort required for the construction of the simulation program and the associated documentation is significantly reduced.

Keywords

Bioreactor, modeling, modular, multilayer, fed-batch, cyclic

1. Introduction

Dynamic modeling and simulation of biological reactors is being extensively used to design and optimization of the operation of such reactors [1-3] and to evaluate the control systems for the reactor [4]. Many biological processes consist of several basic modes (such as initiation, processing and harvesting) where the various basic modes are represented by different mathematical models. The models of the modes represent the first layer of the modeling. The processes can be operated in various operational modes (such as cycling) that may require repetitive modeling of the individual modes. This type of operation represents the second layer of modeling. Optimization and control of the process represents an additional modeling layer.

The first layer of modeling, in this case a biological reactor, involves the development of modules that describe each basic mode of the reactor operation. This is most easily accomplished with a software package that requires very little technical coding and which provides efficient modeling with flexibility and user friendliness. This work has used POLYMATH software for this purpose (http:www.polymath-software.com). This software also now supports the automated export of programs to MATLAB (http://www.mathworks.com). Thus MATLAB provides more programmable options that are extremely useful for the additional layers of modeling.

This approach to the first layer of modeling with POLYMATH enables considerable efficiency in developing the models that describe the various basic modes of reactor operation. The automated generation of the MATLAB code by POLYMATH is extremely convenient and allows for easy model verification and integration into operational models for the dynamic reactor operation, the second layer of modeling. Additional modeling for optimization and control (the third layer of modeling) is also enabled within MATLAB.

2. Simulation of the Semi-batch and Cyclic Operational Modes of a Bioreactor

The three basic modes in a semi-batch bioreactor are shown in Fig. 1 for the reaction $S + X \rightarrow P + nX$ where S is substrate reactant, X is the cell, and P is the product. The biochemical reaction is inhibited by the substrate. The first basic mode involves the initiation or startup of the reactor from an initial innoculum concentration of cells with substrate feed rate of F_I over time interval from $t = 0$ to t_I. The processing is then followed by controlled addition of the substrate at a rate of F_P over time interval from t_I to t_P during which the reactor volume increases from V_I to V_H. The substrate addition is then stopped and the batch bioreactor is harvested with only a discharge flow rate of F_H over time interval from t_P to t_H during which the volume decreases from V_H to V_I. In fed-batch operation, the initiation, processing, and harvesting basic modes are utilized in a single sequence. Cyclic operation involves the repetitive cycle of processing and harvesting modes where *n* is the cycle number. More details are given by Cutlip and Shacham [3].

The objective of the final simulation is to describe the semi-batch or cyclic operation of the bioreactor. This work is limited to a determination of the optimal substrate feed concentration for a fixed set of operational conditions.

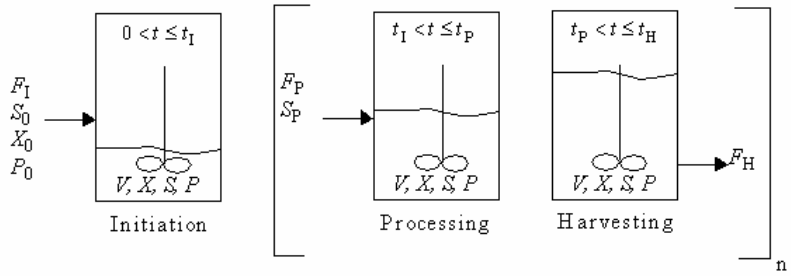

Figure 1. The Three Modes in the Operation of the Semi-batch Bioreactor (For fed batch operational mode n =1, for cyclic mode n>1)

2.1. Development of the Models of the Individual Operational Modes

Separate models for the three basic modes of the bio-reactor were derived and tested using Polymath. The Polymath model for the "Initiation" mode is shown in Table 1. (Note that the programs used in this work can be found at **http://www.engr.uconn.edu/~cutlipm/escape17**.) The Polymath models (including the "comments" which start with the # sign) provide complete documentation of the models, the values of the constants and the initial and final conditions. Statements 1 through 4 of Table 1 present the differential equations for obtaining the total quantities of the cells N_X, substrate N_S, the product N_P, and the culture volume V. The additional variables included in the differential equations are the specific growth rate μ_{net}, cell concentration X, substrate concentration S and the product concentration P (see statements 5 through 8). Additional parameters and constants are described starting in statement 9. Also the initial conditions for the independent variable and dependent variables are specified, and the final value of the independent variable (*t)* is set. Note that the duration of the initiation mode is one hour.

Table 1. POLYMATH Model for Initiation Mode of the Semi-batch Bioreactor

No.	Equation # Comment
1	d(Nx) / d(t) = (FI*X0+munet*V*X) # Quantity of cells (g)
2	d(Ns)/d(t)= (FI*S0+munet*V*X/Yxs) # Quantity of glucose (g)
3	d(Np)/d(t)=(munet*V*X/Yxp) # Quantity of fermentation products (g)
4	d(V)/d(t)=FI # Culture volume (L)
5	munet=mum*S/(Ks+S+S^2/KI) # Specific growth rate (g product/L-hr)
6	X=Nx/V # Cell concentration (g/L)
7	S=Ns/V # Substrate concentration (g/L)
8	P=Np/V # Product concentration (g/L)
9	FI=0.2 # Feed flow rate initiation stage (L/hr)
10	X0=0 # Feed cell concentration (g/L)

11	S0=200 # Feed substrate concentration, initiation stage (g glucose/L)
12	Yxs=0.4 # Yield coefficient (g cells/g glucose)
13	Yxp=0.15 # Yield coefficient (g cells/g product)
14	mum=0.3 # Maximal specific growth rate (1/hr)
15	Ks=1 # Monod constant (g glucose/L)
16	KI=300 # Monod constant (g glucose/L)^2
17	Nx(0) = 24 # Initial quantity of cells (g)
18	Ns(0)=0 # Initial quantity of substrate (g)
19	Np(0)=0 # Initial quantity of products (g)
20	V(0)=.8 # Initial culture volume (L)
21	t(0) = 0
22	t(f) = 1

The model for the basic processing mode is essentially the same as for the basic initiation mode except that the flow-rate F_I= 0.2 L/hr is replaced by F_P = 0.5 L/hr. The duration of the basic processing mode is 5 hours.

In the harvesting basic mode model of Table 2, there is no inflow but outlet flow of cells, substrate and product. A differential equation is added to calculate the total amount of the fermentation products (P_R). The duration of the basic harvesting mode is 1 hour.

Table 2. Partial POLYMATH Model for the Harvesting Step of the Semi-batch Bioreactor

No.	Equation # Comment
1	d(Nx) / d(t) =(-FH*X+munet*X*V) # Quantity of cells (g)
2	d(Ns)/d(t)= (-FH*S-munet*X*V/Yxs) # Quantity of glucose (g)
3	d(Np)/d(t)= (-FH*P-munet*X*V/Yxp) # Quantity of fermentation products (g)
4	d(V)/d(t)=(-FH) #Culture volume (L)
5	d(PR)/d(t)=FH*P # Production rate (g/hr)
6	FH=2.5 # Outlet flow rate, harvesting stage (L/hr)

The results of the initiation mode and the first cycle of processing and harvesting modes are summarized in Table 3 and Fig. 2. The culture volume increases during the initiation mode from 0.8 L to 1 L, there is a further increase up to 3.5 L during the processing mode. After the harvesting mode the volume returns to 1 L (the same value as after the initiation mode). The product concentration reaches a maximum of 133.14 g/L after the processing mode and it is reduced to 89.14 g/L after the harvesting mode.

Table 3. First Cycle Results

Variable		Initiation	Processing	Harvesting
Time (hr)	0	1	6	7
Nx (g)	24	30.96	93.90	33.43
Ns (g)	0	57.40	400.05	97.80
Np (g)	0	46.39	465.97	89.14
V (L)	0.8	1.00	3.50	1.00
X (g/L)	30	30.96	26.83	33.43
S (g/L)	0	57.40	114.30	97.80
P (g/L)	0	46.39	133.14	89.14
PR (g)	0	0.00	0.00	280.21

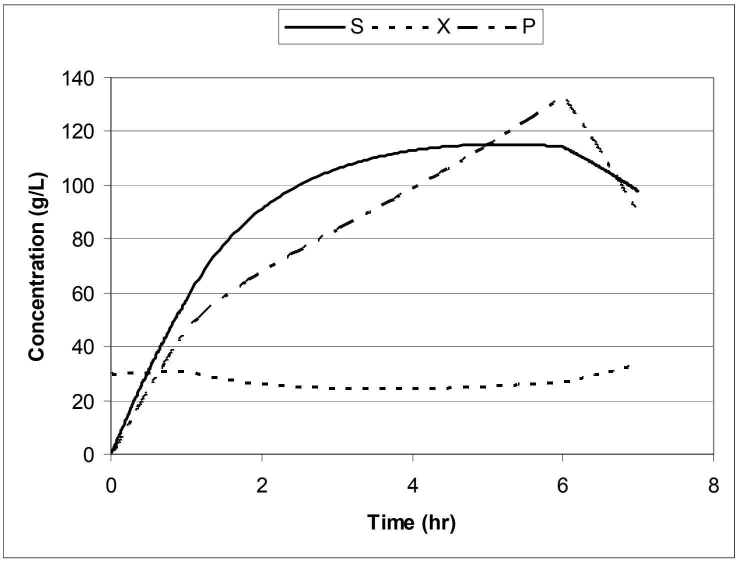

Figure 2. Fed-batch Bioreactor Operaction - Initiation and First Cycle of Processing and Harvesting

2.2. *Integration of the Individual Modules into One Simulation Program*

The POLYMATH models for each of the three basic modes of the bioreactor are exported to MATLAB. The MATLAB function representing a particular basic mode of the process is generated automatically by Polymath. The function representing the initiation mode, for example, is shown in Table 4. Note that MATLAB requires input of the variable values into the function in a single array (*Yfuncvec* , in this case) and return of the derivative values in a single array (*dYfuncvecdt*). The variable values are put back into variables with the same names as used in the Polymath model (details are in Table 4 on the web site) to make the MATLAB code more meaningful. POLYMATH orders the equations sequentially as required by MATLAB and converts any needed intrinsic functions and logical expressions.

Table 4. MATLAB Function (Module) for the Initiation Step of the Semi-batch Bioreactor

No.	Command % Comment
1	function dYfuncvecdt = ODEfun(t,Yfuncvec);
2	Nx = Yfuncvec(1);
3	Ns = Yfuncvec(2);
4	Np = Yfuncvec(3);
5	V = Yfuncvec(4);
6	PR = Yfuncvec(5);
7	mum = .3; % Maximal specific growth rate (1/hr)
8	Ks = 1; % Monod constant (g glucose/L)
9	KI = 300; % Monod constant (g glucose/L)^2
10	Yxs = .4; % Yield coefficient(g cells/g glucose)
11	Yxp = .15; %Yield coefficient (g cells/g product)
12	S0 = 200; % Feed sustrate concentration, initialization stage (g glucose/L)
13	FI = .2; % Feed flow rate, initialization stage (L/hr)
14	S = Ns / V; % Substrate concentration (g/L)
15	X = Nx / V; % Cell concentration (g/L)
16	munet = mum * S / (Ks + S + S ^ 2 / KI); % Specific growth rate (g product/L-hr)
17	P = Np / V; % Product concentration (g/L)
18	X0 = 0; % Feed cell concentration (g/L)
19	dNxdt = FI * X0 + munet * V * X; % Quantity of cells (g)
20	dNsdt = FI * S0 + munet * V * X / Yxs; % Quantity of glucose (g)
21	dNpdt = munet * V * X / Yxp; % Quantity of fermentation products (g)
22	dVdt = FI; % Culture volume (L)
23	dPRdt = 0; % Production rate (g/hr)
24	dYfuncvecdt = [dNxdt; dNsdt; dNpdt; dVdt; dPRdt];

The models of the processing and the harvesting basic modes are exported similarly to MATLAB, and the three functions are renamed to *ODEfunIni, ODEfunProc* and *ODEfunHarv,* respectively.

The next step is to prepare a main program that calls the modules (functions) in the appropriate order to carry out the simulation of the desired operational processing for a predetermined number of cycles. The program for simulating the process which includes the initiation mode and ten processing and harvesting cycles is given in Table 5.

Table 5. Part of the MATLAB "Main Program" for Simulation of Ten Process Cycle

No.	Command % Comment
1	tspan = [0 1.]; % Range for the independent variable
2	y0 = [24.; 0; 0; 0.8; 0]; % Initial values for the dependent variables
3	tg=[]; Nx=[]; Ns=[];Np=[]; V=[]; PR=[];
4	[t,y]=ode45(@ODEfunIni,tspan,y0);
5	tg=[tg; t]; Nx=[Nx; y(:,1)]; Ns=[Ns; y(:,2)]; Np=[Np; y(:,3)];V= [V; y(:,4)]; PR=[PR; y(:,5)];
6	tspan=[t(end) t(end)+5];
7	y0=y(end,:);
8	ncycle=10;
9	for j=1:ncycle
10	[t,y]=ode45(@ODEfunProc,tspan,y0);
11	tg=[tg; t]; Nx=[Nx; y(:,1)]; Ns=[Ns; y(:,2)]; Np=[Np; y(:,3)];V= [V; y(:,4)]; PR=[PR; y(:,5)];
12	tspan=[t(end) t(end)+1];
13	y0=y(end,:);
14	[t,y]=ode45(@ODEfunHarv,tspan,y0);
15	tg=[tg; t]; Nx=[Nx; y(:,1)]; Ns=[Ns; y(:,2)]; Np=[Np; y(:,3)];V= [V; y(:,4)]; PR=[PR; y(:,5)];
16	tspan=[t(end) t(end)+5];
17	y0=y(end,:);
18	end

The integration of the initiation mode model (*ODEfunIni*) is carried out by the MATLAB library function: *ode45*. The number of cycles is set at 10 for this particular case, and the calculations associated with the processing and harvesting modes are repeated ten times in the defined loop. The cell, substrate and product concentrations during the ten cycles are shown in Fig. 3. As the cell and substrate concentrations approach constant amplitude cycling, there is a slight initial downward slope in the maximal product concentration values. Every cycle adds about the same amount of product. Fig. 4 shows the calculated total amounts of fermentation product. The total amount obtained after 10 cycles (61 hr) is 3.084 kg of product.

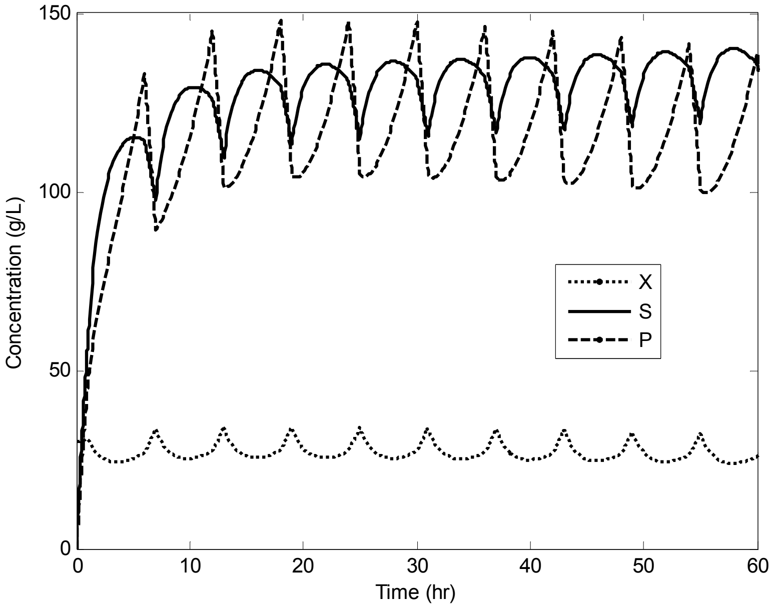

Figure 3. Results for Ten Cycles of Processing and Harvesting

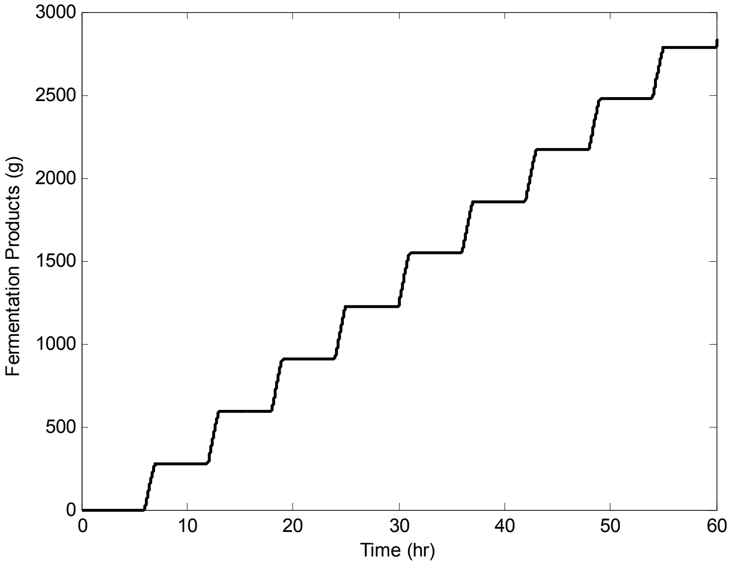

Figure 4. Amount of Harvested Fermentation Products in Cyclic Operation of the Bioreactor

2.3. Adding a Layer for Specialized Calculations

The addition of one or more layers on the top of the simulation program in order to carry out process design and optimization calculations or parameter estimation can be easily accomplished. If the objective is, for example, to determine the optimal substrate concentration value S_P for which the amount of harvested fermentation products P_R is maximal, then the "main program" presented in Section 2.2 must be converted into a function with the input parameter S_P and the returned parameter P_R. A plot of P_R versus S_P, is shown in Fig. 5. There is a maximum in the region of 150 < S_P < 160 g/L. The MATLAB library function *fminbnd* can be used to find the optimal value of $S_P =$ 156.7 g/L where P_R = 8.144 kg.

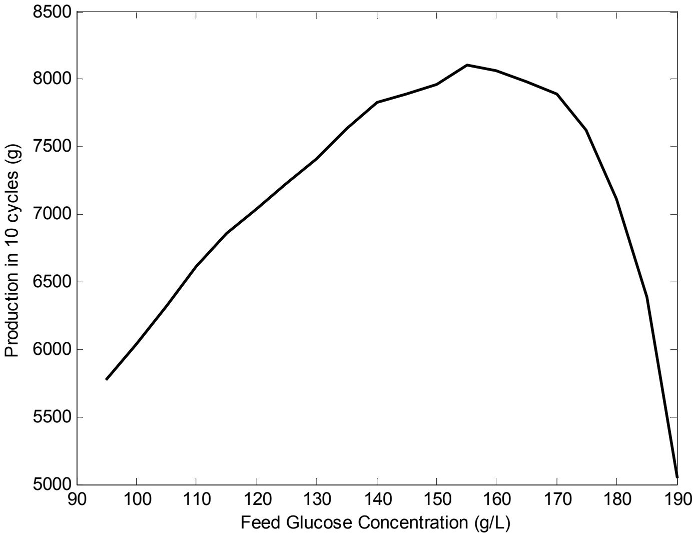

Figure 5. Amount of Harvested Fermentation Products versus Feed Glucose Concentration during Cyclic Operation of the Bioreactor

3. Conclusions

This work demonstrates that the effort involved in construction of dynamic simulation programs for complex processes can be considerably reduced by using the proposed "modular and sequential" approach. This is achieved by utilization of the most effective software tools for the various tasks. Use of a "user-friendly" mathematical software package such as POLYMATH for the preparation of the individual models reduces the technical effort involved in coding and minimizes the potential errors that may enter the final completed mathematical model. Conversion of the verified separated modules into a simulation program within a package such as MATLAB requires essentially the addition of a main executive program that controls the interaction with the user, the sequencing and integration of the results of the various modules, and the reporting of the results.

In addition to reducing the time and effort required for building the process models, the proposed approach yields modular programs that are well documented and easily modified. It also ensures that the models presented in the documentation are almost identical to the models used in the program.

References

1. M. Mussati, P. Aguirre, and N. J. Scenna, Modeling of Real Biological Reactors for the Treatment of Complex Substrates. Dynamic Simulation, Computers Chem. Engng., 22(Suppl) (1998) S723.
2. M. I. Nelson and H.S. Sidhu, Evaluating the Performance of a Cascade of Two Bioreactors, Chemical Engineering Science 61 (2006) 3159.
3. M. B. Cutlip and M. Shacham, Problem Solving In Chemical and Biochemical Engineering with Polymath, Excel and MATLAB, Prentice-Hall, Upper Saddle River, New Jersey, 2007.
4. J. A. Moreno, M. J. Betancur, G. Buitron and I. Moreno-Andrade, Event-Driven Time-Optimal Control for a Class of Discontinuous Bioreactors, Biotechnology and Bioengineering, 94 (2006) 803.