A First Course on Kinetics and Reaction Engineering

Example 20.1

Problem Purpose

This example shows how to analyze a multi-step operational protocol for a batch reactor.

Problem Statement

The rate expression for reaction (1) is given in equation (2). The heat of reaction (1) may be taken to be constant and equal to -22,200 cal mol^{-1}. A jacketed, perfectly mixed batch reactor is charged with 4 L of a 2 M solution of A at 23 °C. The heat capacity of the solution is approximately constant and equal to 440 cal L^{-1} K^{-1}, and its density is constant. A controller maintains a constant pressure in the reactor headspace. The jacket volume is 0.5 L with a heat transfer area of 0.6 ft^2 and a heat transfer coefficient of 1.13 x 10^4 cal ft^{-2} h^{-1} K^{-1}. Initially there is no flow to the jacket, but it is filled with water, also at 23 °C; the water may be taken to have a constant density of 1 g cm^{-3} and a constant heat capacity of 1 cal g^{-1} K^{-1}. A heating coil with a heat transfer coefficient of 3.8 x 10^4 cal ft^{-2} h^{-1} K^{-1} and a heat transfer area of 0.23 ft^2 is submerged in the reacting solution.

$$A \rightarrow B \tag{1}$$
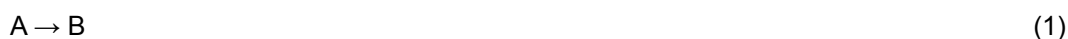
$$r_1 = \left(2.59 \times 10^9 \text{ min}^{-1}\right) \exp\left(\frac{-16500 \text{ cal mol}^{-1}}{RT}\right) C_A \tag{2}$$

In the first step of the operating protocol, saturated steam at 120 °C is admitted to the coil. This step continues until the reaction volume reaches a temperature of 50 °C. At that point, the steam flow to the coil is shut off and cooling water at 20 °C starts to flow into the jacket at a rate of 0.5 kg min^{-1}. This continues until the reaction volume reaches a temperature of 25 °C. At that point, the reactor is drained and preparations for processing the next batch begin. If the turnaround time for the reactor is 25 min, what is the net rate of production of B?

In analyzing the first step of this protocol, you should account for heat transfer between the coil and the reacting fluid and between the reacting fluid and the jacket. In the second step, you may assume that any heat transfer between the reacting fluid and the coil is negligible. You may also assume that the jacket is perfectly mixed throughout both steps of the protocol.

Problem Analysis

The problem describes an operational protocol for a batch reactor that consists of two steps: a heating step and a cooling step. To answer the question posed, mole and energy balances will be written and solved for each of the processing steps. The reactor jacket contains a perfectly mixed heat transfer fluid, and in that situation, the temperature of the heat transfer fluid may change as the reaction progresses. Therefore, energy balances will need to be written for the heat transfer fluid in each step, and solved simultaneously with the other design equations for that step.

Problem Solution

This problem involves the same reaction and reactor as Example 19.3, where only the cooling step was analyzed and the initial conditions were different. Recall from that example that the general mole balance for a batch reactor is given in equation (3), and that writing this equation for A and for B leads to equations (4) and (5).

$$\frac{dn_i}{dt} = V \sum_{\substack{j=all \\ reactions}} \nu_{i,j} r_j \tag{3}$$

$$\frac{dn_A}{dt} = -V r_1 \tag{4}$$

$$\frac{dn_B}{dt} = V r_1 \tag{5}$$

The general energy balance on the reaction volume is given in equation (6). As discussed in Example 19.3, the work term, the derivative of the volume with respect to the time and the derivative of the pressure with respect to time are negligible, and the summation of the molar heat capacities can be replaced by the overall volume-specific heat capacity of the solution, leading, after rearrangement, to equation (7). After rearrangement, a general energy balance on the heat transfer fluid takes the form of equation (8), again, as shown in Example 19.3.

$$\dot{Q} - \dot{W} = \frac{dT}{dt} \sum_{\substack{i=all \\ species}} \left(n_i \hat{C}_{p,i} \right) + V \sum_{\substack{j=all \\ reactions}} \left(r_j \Delta H_j \right) - V \frac{dP}{dt} - P \frac{dV}{dt} \tag{6}$$

$$\frac{dT}{dt} = \frac{\dot{Q} - V \Delta H_1 r_1}{V \hat{C}_{p,\text{soln}}} \tag{7}$$

$$\frac{dT_{e,jacket}}{dt} = \frac{\dot{m} \tilde{C}_{p,e} \left(T_e^0 - T_e \right) - U_{jacket} A_{jacket} \left(T_{e,jacket} - T \right)}{\rho_e V_e \tilde{C}_{p,e}} \tag{8}$$

Analysis of the first processing step. In the first processing step, there are two heat input terms, one corresponding to heat transfer from the coil to the reaction volume, equation (9) and the other corresponding to heat transfer from the jacket to the reaction volume, equation (10). Physical reasoning reveals that the latter heat flow will be negative (i. e. from the reaction volume to thejacket). In addition, there is no flow of coolant into the jacket, so $\dot{m}$ in equation (8) is equal to zero. Making these substitutions results in the set of design equations given in equations (11) through (14) as the model for the first processing step.

$$\dot{Q}_{coil} = U_{coil} A_{coil} \left(T_{e,coil} - T \right) \tag{9}$$

$$\dot{Q}_{jacket} = U_{jacket}A_{jacket}\left(T_{e,jacket} - T\right) \tag{10}$$

$$\frac{dn_A}{dt} = -Vr_1 \tag{11}$$

$$\frac{dn_B}{dt} = Vr_1 \tag{12}$$

$$\frac{dT}{dt} = \frac{U_{coil}A_{coil}\left(T_{e,coil} - T\right) + U_{jacket}A_{jacket}\left(T_{e,jacket} - T\right) - V\Delta H_1 r_1}{V\widehat{C}_{p,\text{soln}}} \tag{13}$$

$$\frac{dT_{e,jacket}}{dt} = \frac{-U_{jacket}A_{jacket}\left(T_{e,jacket} - T\right)}{\rho_e V_e \tilde{C}_{p,e}} \tag{14}$$

The problem specification provides enough information to calculate the initial value of the dependent variables, so the design equations can be solved numerically using software for the solution of initial-value ODEs. Supplemental Unit S5 provides a brief overview of how such software works. There are many software packages you can use in order to do this; you should pick the one you are most comfortable using. No matter what software you elect to use, you will need to provide three things as input to that software:

- the initial values of the independent and dependent variables
- the final value of either t or one of the dependent variables
- code that evaluates each of the derivatives (i. e. the right side of equations (11) through (14)) given a value for t and values for each of the dependent variables along with the additional information provided in the problem specification

At the start of this processing step, the elapsed time for this step is, of course, equal to zero. The reaction volume is initially charged with a solution containing only A at a concentration, C_A^0, of 2M. Therefore the initial moles of A are found using equation (15) and the initial moles of B is equal to zero. The initial temperature of the reaction volume and of the fluid in the jacket are equal to 23 ºC. Thus, the initial values of the independent and dependent variables are known.

$$n_A^0 = C_A^0\ V \tag{15}$$

This processing step continues until the reaction volume reaches a temperature of 50 ºC. Since the temperature of the reaction volume is one of the dependent variables, this can be used directly as the final value.

The third thing that we need to provide in order to solve the design equations (11) through (14) numerically is code to evaluate the right hand side of those equations, given values for t, n_A, n_B, T and $T_{e,jacket}$ along with any other information from the problem statement or reference books. Looking at the right hand sides of equations (11) through (14), the reaction volume, heat transfer coefficients, heat

transfer areas, densities, heat capacities, heat of reaction and coil temperature are all constants; the only additional variable quantity is the rate of reaction. That can be calculated using equation (2), where the ideal gas constant is a known constant and the temperature will be given. That leaves only the concentration of A to be determined, and that is easily calculated using equation (16) since the moles of A will be given and the reaction volume is a known constant.

$$C_A = \frac{n_A}{V} \tag{16}$$

We now have everything we need in order to solve the design equations numerically. Doing so we Find that the reaction volume reaches 50 ºC in approximately 3.9 minutes at which time it contains 7.77 moles of A and 0.23 moles of B. The temperature of the water in the jacket has risen to 35.3 ºC at the end of this step. Figure 20.1 shows the concentration of A in the reaction volume as a function of elapsed time during the first step and Figure 20.2 shows the temperatures of the reaction volume and coolant.
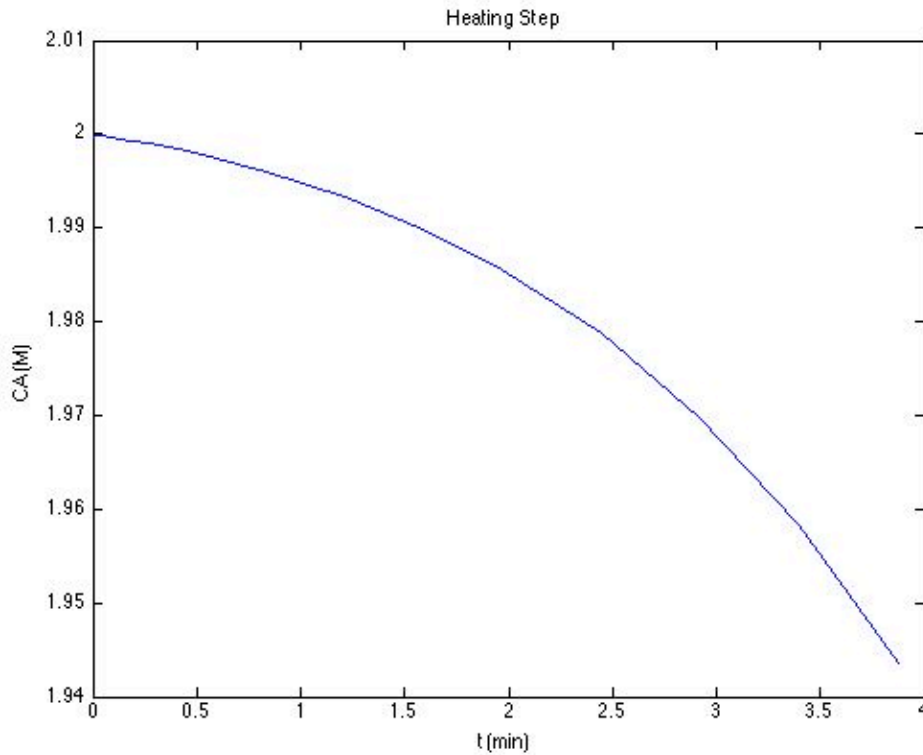

Figure 20.1. Variation in the concentration of A during the first processing step.
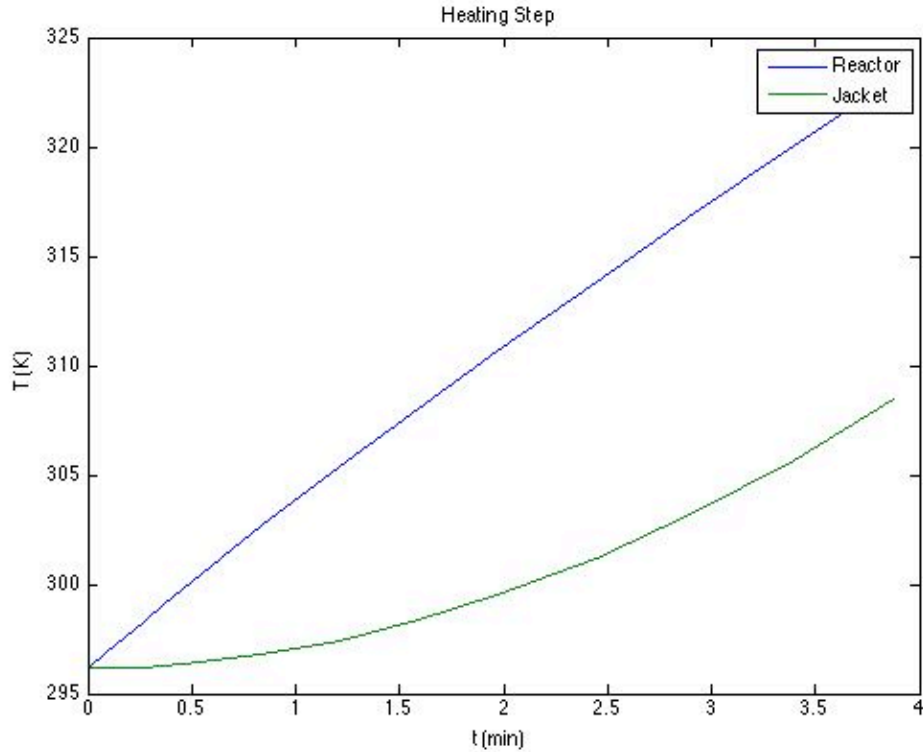
Figure 20.2. Variation in the reaction volume temperature and the coolant temperature during the first processing step.

Analysis of the second processing step. In the second processing step there is only heat transfer between the reaction volume and the coolant in the jacket, so equation (10) alone is used for the heat input. In addition, there is coolant flowing into the jacket during the second step, so $\dot{m}$ is not equal to zero. With these changes, the design equations take the form shown in equations (17) through (20).

$$\frac{dn_A}{dt} = -Vr_1 \tag{17}$$

$$\frac{dn_B}{dt} = Vr_1 \tag{18}$$

$$\frac{dT}{dt} = \frac{U_{jacket}A_{jacket}\left(T_{e,jacket} - T\right) - V\Delta H_1 r_1}{V\widehat{C}_{p,\text{soln}}} \tag{19}$$

$$\frac{dT_{e,jacket}}{dt} = \frac{\dot{m}\tilde{C}_{p,e}\left(T_e^0 - T_e\right) - U_{jacket}A_{jacket}\left(T_{e,jacket} - T\right)}{\rho_e V_e \tilde{C}_{p,e}} \tag{20}$$

Once again, these equations will be solved numerically. At the start of this processing step, the elapsed time for the step is zero. The initial values of the dependent variables are equal to their values at

the end of the first step ($n_A{}^0$ = 7.77 moles, $n_B{}^0$ = 0.23 moles of B, T = 50 °C and $T_{e,jacket}$ = 35.3 °C). The final condition for the second processing step is that the temperature of the reaction volume is equal to 25 °C. We've already discussed how to calculate everything on the right hand side of the design equations except $\dot{m}$, which is a known constant. Therefore, at this point we can solve the equations numerically. Doing so, we find that the second processing step takes 76.9 min and at the end, the reaction volume contains 3.6 moles of A and 4.4 moles of B. Figure 20.3 shows how the concentration of A varies during the second step, and Figure 20.4 shows how the reaction temperature and the coolant temperature vary.
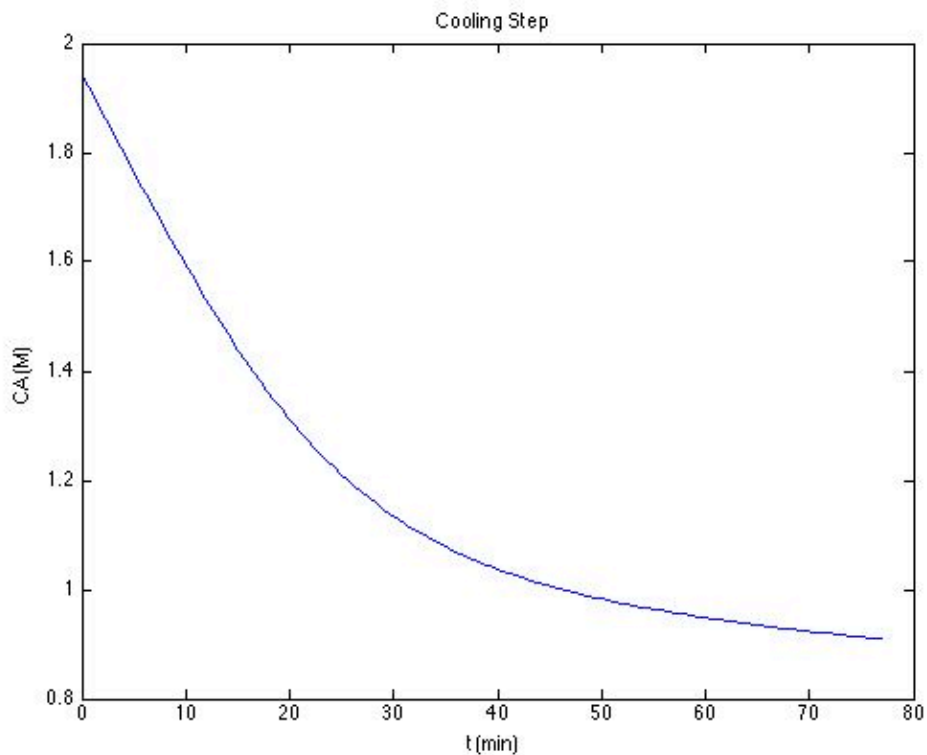

Figure 20.3. Variation in the concentration of A during the second processing step.
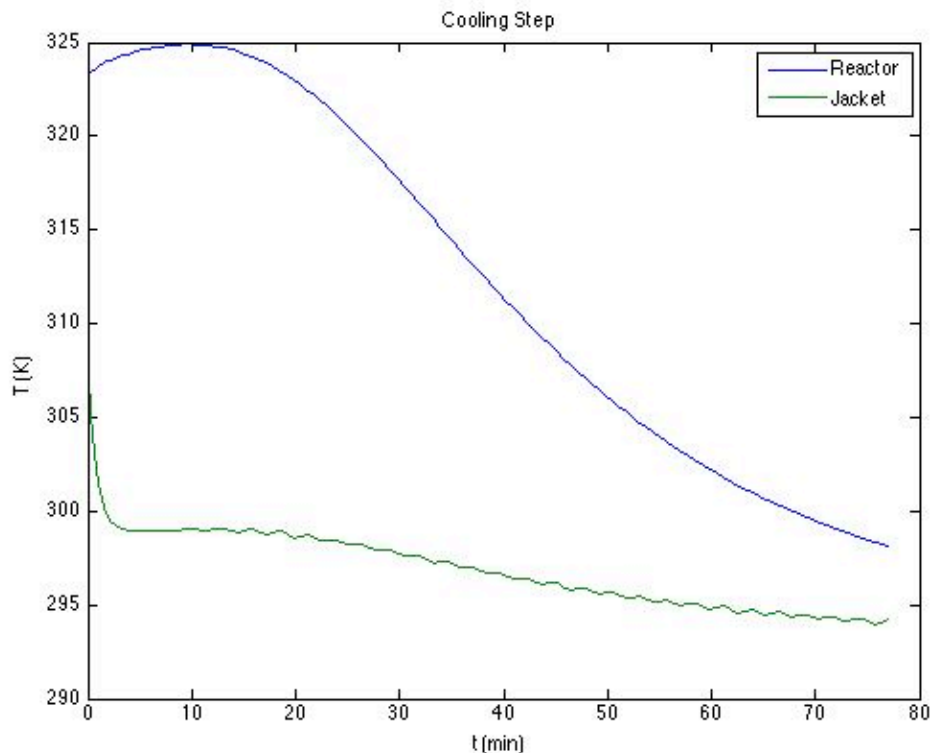
Figure 20.4. Variation in the reaction volume temperature and the coolant temperature during the second processing step.

The net rate of production of B, accounting for the processing time for each step as well as the turnaround time, can be calculated using equation (21). The operating protocol described in the problem statements leads to a net rate of production of B of 0.01 mol B per min.

$$r_{B,net} = \frac{n_B^f - n_B^0}{t_{step\ 1} + t_{step\ 2} + t_{turnaround}} \tag{21}$$

Calculation Details Using MATLAB

The design equations for each of the processing steps are initial value ordinary differential equations, and in both steps the final value of a dependent variable is known. Supplemental Unit S5 describes how to solve this type of equation numerically and provides template files for doing so. For this problem, the appropriate template file is SolvIVDifD.m. I elected to make two copies of this file and modify one of them, named Example_20_1_a.m, to solve the design equations for the first processing step and the second, named Example_20_1_b.m, to solve the design equations for the second processing step. I then wrote a third function that calls each of the first two and uses the results to calculate the net rate of production of B. The third file is named Example_20_1.m; all three files accompany this solution.

Each copy of SolvIVDifD.m needed to be modified in six places in order to use it to solve the equations for this problem. In addition, I made a few additional modifications which I'll describe here, as well. In order to model the first processing step, the function name was changed to Example_20_1_a and the initial comment was changed to describe the purpose of the modified file. The first required modification was to enter all of the constant quantities from the problem statement. While entering these quantities, they were changed so that the units were consistent (cal, mol, K, min, ft², kg and L), and I also calculated the initial moles of A and the final temperature, since they, too, are constant. The results of all these modifications are shown in Listing 1.

```
% Modified version of the MATLAB template file SolvIVDifD.m used in the
% solution of Example 20.1 of "A First Course on Kinetics and Reaction
% Engineering." This function models the first step of the operational
% protocol described in that example.
%
function [t_f,z] = Example_20_1_a
    % Known quantities and constants (cal, mol, K, min, ft2, kg, L)
    k0 = 2.59e9; % min-1
    E = 16500; % cal/mol
    R = 1.987; % cal/mol/K
    dH = -22200; % cal/mol
    V = 4; % L
    CA0 = 2; % mol/l
    nA0 = CA0*V; % mol
    Te_coil = 120 + 273.15; % K
    Te0_jacket = 23 + 273.15; % K
    cp = 440; % cal/L/K
    Ve = 0.5; %L
    A_jacket = 0.6; % ft2
    U_jacket = 1.13e4/60; % cal/ft2/min/K
    A_coil = 0.23; % ft2
    U_coil = 3.8e4/60; % cal/ft2/min/K
    rho = 1; % kg/L
    cpe = 1000; % cal/kg/K
    T0 = 23 + 273.15; % K
    Tf = 50 + 273.15; % K
    m = 0; % kg/min (no cooling water flowing)
```

Listing 1. Resulting code after the first set of modifications for the first processing step..

The second and third required modifications involved entering code to evaluate the the right hand side of the ODEs, equations (11) through (14), and provide the initial values. The equations are provided as a vector quantity, and so, it is necessary to map the dependent variables used in the problem statement (n_A, n_B, T and T_e) to a vector z, and the corresponding derivatives are mapped to a vector dzdt. I find it useful at the start of the internal function that will evaluate the derivatives, to define local variables with the names used in the problem statement. This modification is not required, but in my opinion, it makes the code more readable and easier to debug. In addition, the list of variables here serves as a reminder of the mapping of the problem statement variables to the vector z. The rate was calculated next, as described in the problem solution, and the results were saved in the vector dzdt. The

initial values are similarly stored in a vector named z0. The results of these modifications are shown in Listing 2.

```
    % Function that evaluates the ODEs
function dzdt = odeqns(t,z)
    nA = z(1);
    nB = z(2);
    T = z(3);
    Te_jacket = z(4);
    r1 = k0*exp(-E/R/T)*nA/V;
    Q = U_jacket*A_jacket*(Te_jacket-T)+U_coil*A_coil*(Te_coil-T);
    dzdt = [
        -r1*V
        r1*V
        (Q-V*dH*r1)/V/cp
        (m*cpe*(Te0_jacket-Te_jacket)-U_jacket*A_jacket*(Te0_jacket-
    T))/rho/Ve/cpe
        ];
end % of internal function odeqns

% Initial values
t0 = 0.0;
z0 = [
    nA0
    0
    T0
    Te0_jacket
];
```

Listing 2. Resulting code after the second set of modifications for the first processing step..

The fourth and fifth *required* modifications involve providing the final value of one dependent variable, in this problem T. The first step (fourth required modification) is to select a value for t_f that is much greater than the actual final time. Of course, you don't know what the final time is at this point, so all you can do is set t_f to a large number. It is important to check the answer to make sure that the final time that is reported is smaller than this value; otherwise, you need to make this value larger and repeat the calculations. For this problem, I arbitrarily set t_f to 5000 min, so I'll need to check the final time that is calculated and make sure it is less than 5000 min.

The second step (fifth required modification) is to provide the final condition. Basically, this involves adding the code to calculate the value of a variable named stop_when. The variable, stop_when, should become equal to zero when the final condition is reached. Here the final condition we want to reach is that the temperature of the reaction volume should equal 50 ºC. When I entered the data from the problem statement, I used Tf to represent this concentration. Hence, noting that z(3) is the temperature of the reaction volume, the final condition is given by equation (22). I can rearrange that equation so that there is a zero on the left-hand side as shown in equation (23). Clearly, the final condition will be reached when the right hand side of equation (23) becomes equal to zero. Therefore, for the fifth required modification, I set the variable stop_when equal to the right hand side of equation (23). The results of the fourth and fifth modifications are shown in Listing 5.

$$z(3) = T_f \tag{22}$$

$$0 = T_f - z(3) \tag{23}$$

```
tf = 5000.0;
options = odeset('Events',@stop);
    [t, zz, te, ze, ie] = ode45(@odeqns,[t0, tf],z0,options);

% Function that provides the integration stopping criterion
function [stop_when, isterminal, direction] = stop(t,z)
    isterminal = 1;
    direction = 0;

    stop_when = Tf - z(3);
end % of internal function stop
```

Listing 3. Results of the fourth and fifth required modifications.

The sixth and final _required_ modification is to use the results from solving the ODEs to calculate whatever the problem requested. The problem did not ask for any information beyond the solution of the ODEs for this step, but I decided to generate a set of plots like those in Example 19.3. The results of these modifications are shown in Listing 4.

```
% Plot CA vs. t
CA = zz(:,1)/V;
figure
plot(t,CA)
title(['Coolant flow = ',num2str(m),' kg/min'])
xlabel('t (min)')
ylabel('CA (M)')
% Plot Te and T vs. t
figure
plot(t,zz(:,3),t,zz(:,4))
title(['Coolant flow = ',num2str(m),' kg/min'])
xlabel('t (min)')
ylabel('T (K)')
legend('Reactor','Jacket')
```

Listing 4. Results of the sixth required modification.

The modifications to the second file are similar and won't be described here in detail. One significant change was that the function was modified so that the initial values of the dependent variables could be passed in as an argument. The variable z0 was used to receive the argument, and therefore the section of the code where one normally made a required modification to set the values of z0 was simply deleted. The remaining modifications are similar to those made in the first file; they can be examined by opening the file Example_20_1_b.m that accompanies this solution.

Finally, a short function was written to integrate the two processing steps and perform the final calculation of the net rate of production of B. This function first calls Example_20_1_a. It then takes the resulting values of the dependent variables and calls Example_20_1_b using them as an argument. Finally, it calculates the total processing time, not including turnaround time, and the net rate of production

of B and displays them. The code is shown in Listing 5. When it is run, it generates the plots shown in this solution and produces the output shown in Listing 6.

```
% Function to solve the design equations for Example 20.1 from "A First
% Course on Kinetics and Reaction Engineering" for both steps of the
% operating protocol
function Example_20_1
    % Solve the design equations for the first step
    [t_step1,z_step1] = Example_20_1_a;

    % Solve the design equations for the second step
    [t_step2,z_step2] = Example_20_1_b(z_step1);

    % Report the results
    display(['Total processing time (min): ',num2str(t_step1+t_step2)])
    net_rate = z_step2(2)/4/(t_step1 + t_step2 + 25);
    display(['Net rate of B production (mol/min): ',num2str(net_rate)])

end % of Example_20_1
```

Listing 5. Contents of the function Example_20_1.

```
>> Example_20_1
Total processing time (min): 80.7676
Net rate of B production (mol/min): 0.010311
```

Listing 6. Output from the execution of Example_20_1.