# Jacketed Continuous Stirred-tank Reactor – Models, Control Systems and Simulators

Dr. James H. Taylor
Department of Electrical & Computer Engineering
University of New Brunswick
Fredericton, NB CANADA E3B 5A3
e-mail: jtaylor@unb.ca

Date: 23 November 2009

*Abstract:* This document serves as a model description and user's guide for a Jacketed Continuous Stirred-tank Reactor (JCSTR) simulator to be used in the PAWS Wireless Networked Control System Coordination Agent (WNCSCA) development effort. We start with the nonlinear JCSTR model[1] in differential equation form, and then we depict an appropriate linear control configuration for this nonlinear plant. Next, the static relations corresponding to an operating point are given and a linearized version of the model is derived. Based on this, we present a linear control system configuration used for analysis and design of a controller for the nonlinear process. Finally, some MATLAB simulation results are provided from an example run and a listing of the JCSTR control system simulation model is provided.

## 1 Nonlinear JCSTR Model

In this JCSTR, shown in Fig. 1[2], the tank inlet stream (described by $F_i$ or inlet flow rate and $T_i$ or inlet temperature) is received from another process unit, and there is a heat transfer liquid circulating through the jacket to heat the liquid in the tank (described by $F_{jin}$ or jacket flow rate and $T_{jin}$ or jacket inlet temperature). The objective is to control the temperature and the volume inside the tank by varying the jacket inlet valve flow rate (the temperature control or TC loop) and tank outlet valve flow rate (the level control or LC loop) respectively.

The following assumptions were made in order to derive the dynamic modeling equations of the tank and jacket temperatures:

- Liquids have constant density and heat capacity.
- Mixing in both the tank and the jacket are perfect.
- The amount of liquid in the jacket is constant, i.e., $F_{jin} = F_{jout}$.
- The tank inlet flow rate, tank inlet temperature and jacket inlet temperature may change; these are uncontrolled inputs.

[1]The following JCSTR model presentation is based on Dr. Maira Omana's MScEng thesis.
[2]This schematic is taken from Mr. Hazem Saad Ibrahim's MScEng thesis.

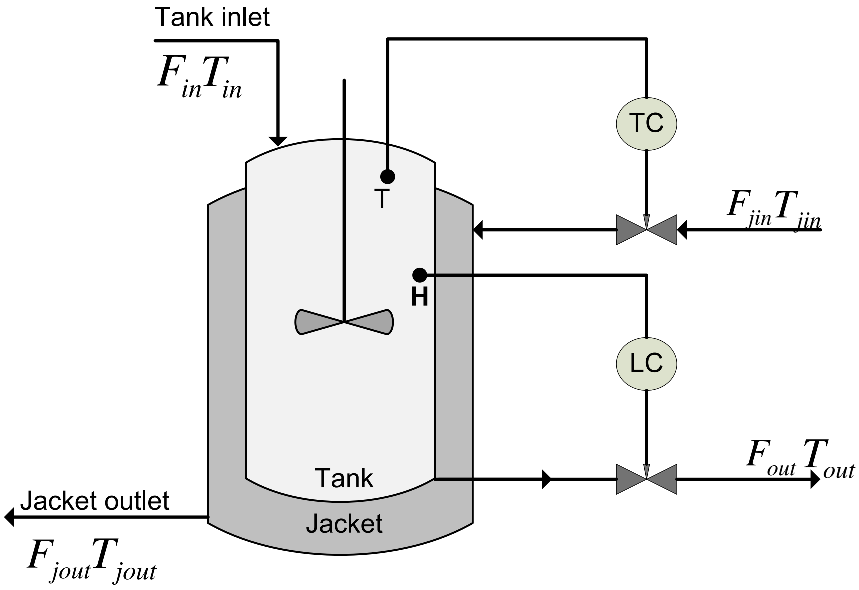


Figure 1: Jacketed Continuous Stirred-tank Reactor Schematic

- The tank outlet flow rate and jacket flow rate are control inputs.
- The rate of heat transfer from the jacket to the tank is governed by the equation $Q = U\,A_H\,(T_j - T)$, where $U$ is the overall heat transfer coefficient and $A_H$ is the area for heat transfer.

The following equations describe the ordinary differential equation (ODE) model for the JCSTR reactor; the notation used is listed below.

$$\dot{V} = F_{in} - F_{out} \tag{1}$$

$$\dot{T} = \frac{F_{in}\,(T_i - T)}{V} + \frac{U A_H\,(T_j - T)}{\rho\,C_p\,V} \tag{2}$$

$$\dot{T}_j = \frac{F_{jin}\,(T_{jin} - T_j)}{V_j} - \frac{U A_H\,(T_j - T)}{\rho\,C_p\,V_j} \tag{3}$$

Note that an old but **important error** has been corrected – the heat flow **into** the reactor must come **from** the jacket, hence the opposite signs on that term in Eqns. (2) and (3) above.

The TC and LC loops are both closed by PI controllers, i.e.,

$$F_{out} = K_{PH}\,e_H + K_{IH}\int e_H \tag{4}$$

$$F_{jin} = K_{PT}\,e_T + K_{IT}\int e_T \tag{5}$$

where the error signals are given in terms of the level (height) and temperature set-points $H_{SP}$, $T_{SP}$ by

$$e_H = H_{SP} - H \tag{6}$$

$$e_T = T_{SP} - T \tag{7}$$

**Parameters:**

| | | |
|---|---|---|
| $D_r$ | Diameter of the reactor tank | 5 $m$ |
| $A_B$ | Area of the tank base $= \pi D_r^2/4$ | 19.635 $m^2$ |
| $H$ | Height of liquid in the reactor | 10 $m$ (maximum) |
| $A_H$ | Area for heat transfer $= A_B + \pi D_r H$ | *varying* |
| $\rho$ | Density (mass/vol) | 997.95 $kg/m^3$ |
| $C_p$ | Heat capacity (energy/mass·temp) | 4.1868*1000 $J/kg \cdot K$ |
| $F_i$ | Volumetric flow rate (volume/time) | 0.10 $m^3/s$ |
| $F_J in$ | Volumetric flow rate (volume/time) | 0.15 $m^3/s$ |
| $T_i$ | Tank inlet temperature | 283 $^oK$ (nominal) |
| $T_{jin}$ | Jacket inlet temperature | 419 $^oK$ (nominal) |
| $U$ | Heat transfer coefficient (energy/time·area·temp) | 2,130 $W/m^2 \cdot K$ |
| $V$ | Volume of liquid in tank $= A_B \cdot H$ | *varying* |

**Control gains:**

| | |
|---|---|
| $K_{PH}$ | 0.05 |
| $K_{IH}$ | 3.0e-005 |
| $K_{PT}$ | – 0.4 |
| $K_{IT}$ | – 9.2e-004 |

The state variables of the system are level (liquid height), $H$, tank temperature, $T$, and jacket liquid temperature, $T_j$, i.e., $x = [\,H \;\; T \;\; T_j\,]^T$, and the control inputs are $u = [\,F_{out} \;\; F_{jin}\,]^T$. Finally, the uncontrolled inputs are $w = [\,F_i \;\; T_i \;\; T_{jin}\,]^T$. Thus we may use $H = V/A_B$ to reformulate the model in Eqns. (1 - 3) as:

$$\dot{x}_1 = \frac{w_1 - u_1}{A_B} \tag{8}$$

$$\dot{x}_2 = \frac{w_1\,(w_2 - x_2)}{A_B x_1} + \frac{U\,(x_3 - x_2)}{\rho\, C_p\, x_1} + \frac{\pi\, D_r\, U\,(x_3 - x_2)}{\rho\, C_p\, A_B} \tag{9}$$

$$\dot{x}_3 = \frac{u_2\,(w_3 - x_3)}{V_j} - \frac{U\, A_B\,(x_3 - x_2)}{\rho\, C_p\, V_j} - \frac{\pi\, D_r\, U\, x_1\,(x_3 - x_2)}{\rho\, C_p\, V_j} \tag{10}$$

# 2 JCSTR Control System Configuration

A control system for a highly nonlinear plant cannot be configured as shown in most control text books. This is due to the fact that the linear controller must operate on *small perturbation signals,* i.e., on variations about a set point (operating point). This is done by using feed forward of the plant input corresponding to the operating point, denoted $\overline{u}$, and having the controller $C(s)$ operate on the error signal $e = \delta y$ to produce the perturbation $\delta u$, as shown in Fig. 2. One must derive the equations for $U(\,\cdot\,)$ based on the desired operating point in order to construct this system. Note that the plant output is simply $y = [\,H \;\; T\,]^T = [\,x_1 \;\; x_2\,]^T$ for the system under consideration.

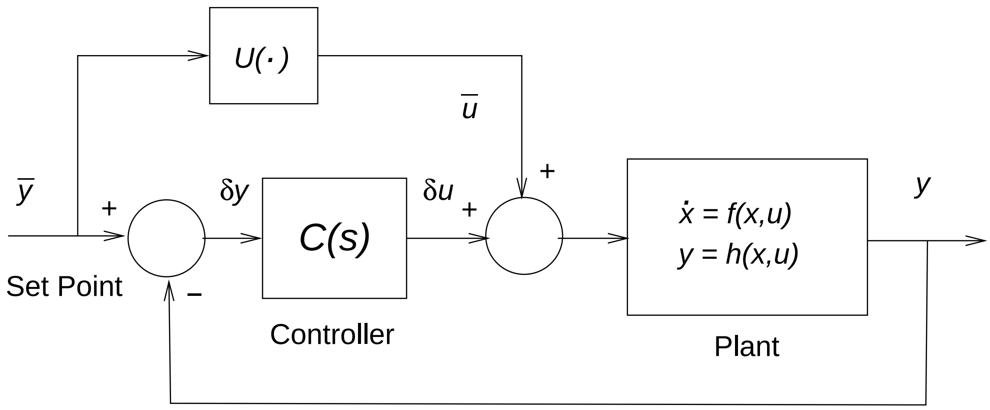


Figure 2: Linear Control for a Nonlinear System

# 3 JCSTR Operating Point

An operating point $(\overline{x}, \overline{u})$ is, by definition, a point where $\dot{x} = 0$. For the JCSTR we start by defining the desired set point of the system, $\overline{x}_1$, $\overline{x}_2$ and derive $\overline{x}_3\,, \overline{u}_1$ and $\overline{u}_2$ corresponding to the set point.

From $\dot{x}_1$ above we see that the level will not change as long as $\overline{u}_1 = w_1$ or $F_{out} = F_{in}$, so that part is simple; by physical reasoning this condition causes $x_1 = H$ to remain at any arbitrary value $\overline{x}_1$.

Setting $\dot{x}_2$ and $\dot{x}_3$ to zero yields the following simultaneous equations:

$$\dot{x}_2 = 0 \quad \rightarrow \quad w_1\,(w_2 - x_2) = -A_B\,(\alpha + \beta\,x_1)\,(x_3 - x_2) \tag{11}$$

$$\dot{x}_3 = 0 \quad \rightarrow \quad u_2\,(w_3 - x_3) = V_j\,(\gamma + \eta\,x_1)\,(x_3 - x_2) \tag{12}$$

where we define $\alpha$, $\beta$, $\gamma$ and $\eta$ for simplification as follows:

$$\alpha = \frac{U}{\rho\,C_p} \tag{13}$$

$$\beta = \frac{\pi D_r U}{\rho\,C_p\,A_B} = \alpha\,\frac{\pi\,D_r}{A_B} \tag{14}$$

$$\gamma = \frac{U A_B}{\rho\,C_p\,V_j} = \alpha\,\frac{A_B}{V_j} \tag{15}$$

$$\eta = \frac{\pi D_r U}{\rho\,C_p\,V_j} = \gamma\,\frac{\pi\,D_r}{A_B} = \alpha\,\frac{\pi\,D_r}{V_j} \tag{16}$$

Solving Eqn. (11) for $\overline{x}_3$ then substituting $\overline{x}_3$ into Eqn. (12) we can obtain $\overline{u}_2$, as follows:

$$\overline{x}_3 = \overline{x}_2 + \frac{w_1\,(\overline{x}_2 - w_2)}{\alpha\,(A_B + \pi D_r \overline{x}_1)} \tag{17}$$

$$\overline{u}_2 = \frac{w_1\,(\overline{x}_2 - w_2)}{w_3 - \overline{x}_3} \tag{18}$$

# 4 Linearized JCSTR Model

Given a dynamical system and output equation $\dot{x} = f(x, u, w)$, $y = h(x, u, w)$, some desired operating-point values for $u_0$ and $w_0$ and the corresponding equilibrium $x_0$, then we can evaluate the corresponding output value, $y_0 = h(x_0, u_0, w_0)$. To proceed, define the perturbation variables $\delta x = x - x_0$, $\delta u = u - u_0$, $\delta w = w - w_0$, $\delta y = y - y_0$.

Then, *if* the perturbations are small *and if* continuous partial derivatives exist at $(x_0, u_0, w_0)$ the behaviour of the original system near $x_0$ is similar to that of $\dot{\delta x} = A\,\delta x + B\,\delta u + E\,\delta w$ and $\delta y = C\,\delta x + D\,\delta u + F\,\delta w$ where

$$A = \left[\frac{\partial f}{\partial x}\right]_{x_0,u_0,w_0} \quad , \quad B = \left[\frac{\partial f}{\partial u}\right]_{x_0,u_0,w_0} \quad , \quad E = \left[\frac{\partial f}{\partial w}\right]_{x_0,u_0,w_0} \tag{19}$$

$$C = \left[\frac{\partial h}{\partial x}\right]_{x_0,u_0,w_0} \quad , \quad D = \left[\frac{\partial h}{\partial u}\right]_{x_0,u_0,w_0} \quad , \quad F = \left[\frac{\partial h}{\partial w}\right]_{x_0,u_0,w_0} \tag{20}$$

The procedure defined in Eqns (19,20) is called **small signal linearization** (SSL), since it provides a good model for the system's dynamic behaviour only if the perturbation variables $\delta x$, $\delta u$, $\delta w$ are small.

Taking the partial derivatives of Eqns. (8–10) we obtain:

$$A = \begin{bmatrix} 0 & 0 & 0 \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} , \; B = \begin{bmatrix} -1/A_B & 0 \\ 0 & 0 \\ 0 & b_{32} \end{bmatrix} , \; E = \begin{bmatrix} 1/A_B & 0 & 0 \\ e_{21} & e_{22} & 0 \\ 0 & 0 & u_{20}\,/\,V_j \end{bmatrix} \tag{21}$$

$$C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} , \; D = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} , \; F = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{22}$$

where

$$a_{21} = -\frac{w_{10}(w_{20} - x_{20})}{A_B\,x_{10}^2} - \frac{\alpha(x_{30} - x_{20})}{x_{10}^2} \tag{23}$$

$$a_{22} = -\frac{w_{10}}{A_B\,x_{10}} - \frac{\alpha}{x_{10}} - \beta \tag{24}$$

$$a_{23} = \frac{\alpha}{x_{10}} + \beta \tag{25}$$

$$a_{31} = -\,\eta\,(x_{30} - x_{20}) \tag{26}$$

$$a_{32} = \gamma + \eta\, x_{10} \quad (27)$$

$$a_{33} = -\frac{u_{20}}{V_j} - \gamma - \eta\, x_{10} \quad (28)$$

$$(29)$$

$$b_{32} = \frac{w_{30} - x_{30}}{V_j} \quad (30)$$

$$(31)$$

$$e_{21} = \frac{w_{20} - x_{20}}{A_B x_{10}} \quad (32)$$

$$e_{22} = \frac{w_{10}}{A_B\, x_{10}} \quad (33)$$

# 5 Linearized Control System

In our studies, we may wish to use a linearized model in place of the nonlinear plant, so we implement a linear control system as in Fig. 3. We use this model both for designing the linear controller (for use with either the linearized or nonlinear system), as well as for stability analysis. Note that $\overline{y}$ is, in effect, irrelevant in the linear control context – we can set it to zero and then study, for example, how the control system will respond to unit step inputs. The result will be exactly the same if we add a unit step to $\overline{y}$ except for the offset.

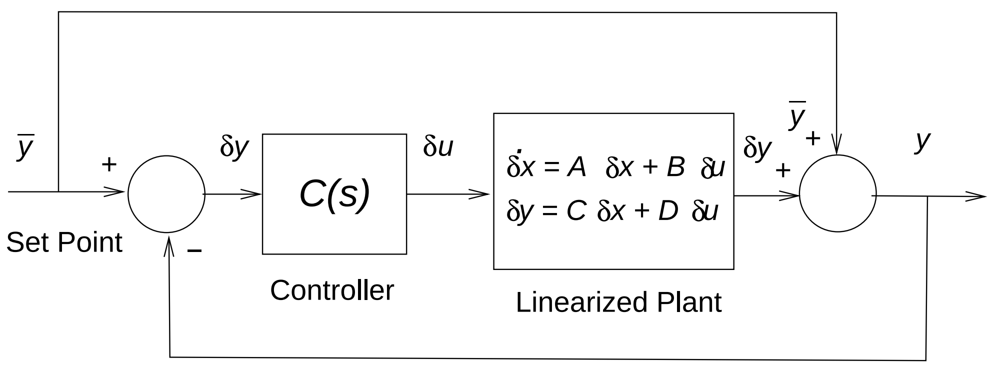


Figure 3: Linear Control with a Linearized Plant

Note that we have ignored the secondary plant input $w$; since we are only considering $w$ equal to a constant it too can be eliminated from consideration *except* in the evaluation of $A$, $B$, $C$, $D$.

# 6 Nonlinear Control System Performance

A nonlinear control system model was constructed in MATLAB, based on the above developments. The `m-file` for this model will be delivered with this manual, however, a listing is provided here for convenience. Step response plots for (1) a change in level set point at t = 90 min. and then (2) a change in tank temperature set point at t =

120 min. is shown in Fig. 4. The PI control gains could probably be tuned to improve performance – however, given the strongly nonlinear and coupled behaviour of the temperature loop an overshoot of about 30 % is not bad.

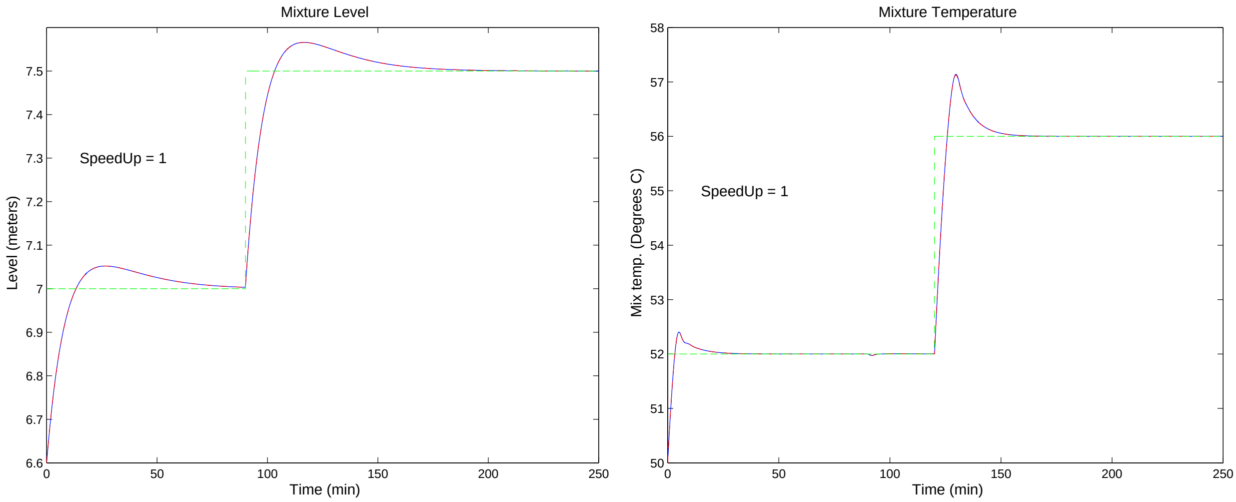


Figure 4: Linear Control for a Nonlinear System

Note that the JCSTR level dynamics are very slow – that is not surprising, given that the tank volume is more than 196 $m^3$. We have included a scale factor `SpeedUp` in the MATLAB model; setting SpeedUp = 60, for example, converts time in minutes to time in seconds.

# 7 Nonlinear Control Simulation Model

```
function [xdot,y] = jcstr(t,x)

% Modified by JH Taylor - 26 August, 7 Sept. & 16-18 Nov. 2009
% You MUST use eufix1y (available from JH Taylor) to integrete
% this model; y = auxiliary vector of outputs of interest.
% Note that this model exhibits very slow dynamics... the
% time constants are in the order of 10 min.

% Augmented with ports for wireless links given
% per discussion on 8 September 2009 and Hazem's
% diagram, Fig. 2 in his memo version 14
% Sensor output path: DIL -> 1 -> 2 -> 3 -> BS -> Gw -> DIL -> controller;
% Controller output path: DIL -> Gw -> BS -> 5 -> 6 -> 7 -> DIL -> Plant

% Added a provision to make the dynamics faster -- setting SpeedUp = 60,
% for example, converts time in minutes to time in seconds.

% x(1) = level of the liquid (m)
```

```
% x(2) = temperature in the reactor (K)
% x(3) = temperature in the jacket (K)
% x(4) = integrator for loop 1 (level) PI controller
% x(5) = integrator for loop 2 (temperature) PI controller

global SpeedUp       %% set in the script for running a simulation
D = 5;               % Diameter of the reactor (m)
A_B = pi*(D^2)/4;    % Area for JCSTR base (m^2)
A_H = A_B+pi*D*x(1); % Area for heat transfer (m^2, dynamic)
Cp = 4186.8;         % Heat capacity (j/kg.K)
rho = 997.95;        % Density (kg/m^3)
U = 2130;            % Heat Transfer coefficient (W/m^2.K)
Vj = 9;              % Heating water Volume (m^3)

%%%%%%%%%%%%%%%%%%%%%%%    Tank      %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Fin = 0.1;             % Mixture inflow 1.766 ft^3/s = 0.1 m^3/s
Tin = 10 + 273;        % Temperature of the mixture feed (283 K)

%%%%%%%%%%%%%%%%%%%%%%%    Jacket    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Fjout = 0.15;          % Heating water Outflow 2.65 ft^3/s = 0.15 m^3/s
Tjin = 146 + 273;      % Temperature of the heating water feed (419 K)

%%%%%%%%%%%%%%%%%%    Controller gains    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Level controller (approx. A_B x larger)
KpH = 0.05;
KiH = 3.0e-005;
% Temperature controller
KpT = - 0.40;  % Atalla's temp. gains WRONG SIGN and size
KiT = - 9.2e-004;
%%%%%%%%%%%%%%%%%%%%%%%    Define set-points    %%%%%%%%%%%%%%%%%%%%%%%%
x_bar(1) = 7;          % nominal set-point for level in the tank, m
T_level = 90.0001*60/SpeedUp; % Time the level set-point will change
%% (make sure T_level is not exactly a data point)
if t > T_level, x_bar(1) = 7.5; end;
%%
x_bar(2) = 273 + 52;   % nominal set-point for temp. in the tank, K
T_temp = 120.0001*60/SpeedUp; % Time the temp. set-point will change
%% (make sure T_temp is not exactly a data point, to avoid a glitch)
if t > T_temp, x_bar(2) = 273 + 56; end;
%%%%%%%%%%%%%%%%%%%%%% Define operating point  %%%%%%%%%%%%%%%%%%%%%%%%
%% x1_bar and x2_bar are "givens"; x_bar(3) and U_bar are derived:
x_bar(3) = x_bar(2)+rho*Cp*Fin*(x_bar(2)-Tin)/(U*(A_B + pi*D*x_bar(1)));
%%
u_bar(1) = Fin;
u_bar(2) = Fin*(x_bar(2) - Tin)/(Tjin - x_bar(3));
```

```
%%%%%%%%%%%%%%%%%%%%%% Implement the control system %%%%%%%%%%%%%%%%%
Sensor_out(1) = x(1);  %% Level
Sensor_out(2) = x(2);  %% Temp.

% Sensor_out enters the controller and Controller_out is produced:
% error signals (negative, due to nature of actuation)
e_level = Sensor_out(1) - x_bar(1);
e_temp = Sensor_out(2) - x_bar(2);

% incremental controller outputs (the plant's "delta u"s)
 du_level = KiH*x(4) + KpH*e_level;
 du_temp  = KiT*x(5) + KpT*e_temp;

% Full controller output (delta u + u_bar)
 Controller_out(1) = du_level + u_bar(1);
 Controller_out(2) = du_temp + u_bar(2);

% Controller_out enters the plant after realistic limits are imposed:
Fout = Controller_out(1);
   if Fout < 0, Fout = 0; end; %% flows can't be negative
Fjin = Controller_out(2);
   if Fjin < 0, Fjin = 0; end;

% state differential equations
xdot(1) = SpeedUp*( Fin - Fout )/A_B;
xdot(2) = SpeedUp*( Fin*(Tin - x(2))/(x(1)*A_B) + ...
          U*A_H*(x(3) - x(2))/(x(1)*rho*Cp*A_B) );
xdot(3) = SpeedUp*( Fjin*(Tjin - x(3))/Vj - U*A_H*(x(3) - ...
          x(2))/(Vj*rho*Cp) ); % sign changed in U*A_H term
xdot(4) = SpeedUp*e_level;
xdot(5) = SpeedUp*e_temp;
xdot = xdot(:);
% aux. output vector (set points)
y(1) = x_bar(1);
y(2) = x_bar(2);
y(3) = x_bar(3);
y(4) = Fout;
y(5) = Fjin;
y = y(:);
```