University Badji Mokhtar - Annaba, Engineering Faculty
Department of Electronics

Section : M1 Auto 2023/2024

Course : Nonlinear systems

Lab 1 : Using S-function blocks in Simulink

I Motivation

With the complexity of medium-size to large-size nonlinear models, it may be more efficient to use a set of differential equations written in an m-file. These m-files will be accessed by Simulink through the S-function block. Thus, this method mixes the advantages of an m-file which can be run directly by solvers such as *ode*45, with the graphical links to other Simulink blocks.

II Example System

$$\frac{dC_a}{dt} = \left(\frac{F}{V}\right) \cdot (C_{af} - C_a) - k_0 \cdot \exp\left[-\frac{E_a}{R(T+460)}\right] \cdot C_a$$
$$\frac{dT}{dt} = \left(\frac{F}{V}\right) \cdot (T_f - T) - \frac{\Delta H}{\rho C_p} \cdot \left(k_0 \cdot \exp\left[-\frac{E_a}{R(T+460)}\right] \cdot C_a\right) - \left(\frac{U \cdot A}{\rho \cdot C_p \cdot V}\right) \cdot (T - T_j)$$

We want to model this system in which we will treat the jacket temperature, T_j, as the input (i.e. manipulated variable). We will also want to monitor concentration and temperature of the liquid in the CSTR as our outputs.

III Write the m-file

Recall that we could model the process by writing an m-file to be used by Matlab solvers such as ode45. One such file, which we will name as **reactor.m**, is shown in Figure 1.
Test the model to make sure it works. For instance, with $Tj = 55$:

```
» [t,x]=ode45(@reactor, [0 10], [0.1 ; 40], [],55) ;
```

Note/recall :
The command-line specifies : a simulation-time span of **[0 10]**, an initial-value column vector : **[0.1 ;40]**, a null placeholder, **[]** , for default options, and setting Tj with a value equal to **55**.

IV Write an S-function file

This file will also be saved as an m-file such that Simulink can access information from Matlab. Figure 2 is the S-function file reactor_sfcn.m

```
function    dx = reactor(t,x,Tj)
%
%   model for reactor
%

    Ca   = x(1)           ;   % lbmol/ft^3
    T    = x(2)           ;   % oF

    Ea   = 32400          ;   % BTU/lbmol
    k0   = 15e12          ;   % hr^-1
    dH   = -45000         ;   % BTU/lbmol

    U    = 75             ;   % BTU/hr-ft^2-oF
    rhocp = 53.25         ;   % BTU/ft^3
    R    = 1.987          ;   % BTU/lbmol-oF
    V    = 750            ;   % ft^3
    F    = 3000           ;   % ft^3/hr
    Caf  = 0.132          ;   % lbmol/ft^3
    Tf   = 60             ;   % oF

    A = 1221              ;   % ft^2

    ra   = k0*exp(-Ea/(R*(T+460)))*Ca;
    dCa  = (F/V)*(Caf-Ca)-ra;
    dT   = (F/V)*(Tf-T)-(dH)/(rhocp)*ra...
             -(U*A)/(rhocp*V)*(T-Tj);

    dx =[dCa;dT];
```

FIGURE 1 – File saved as reactor.m

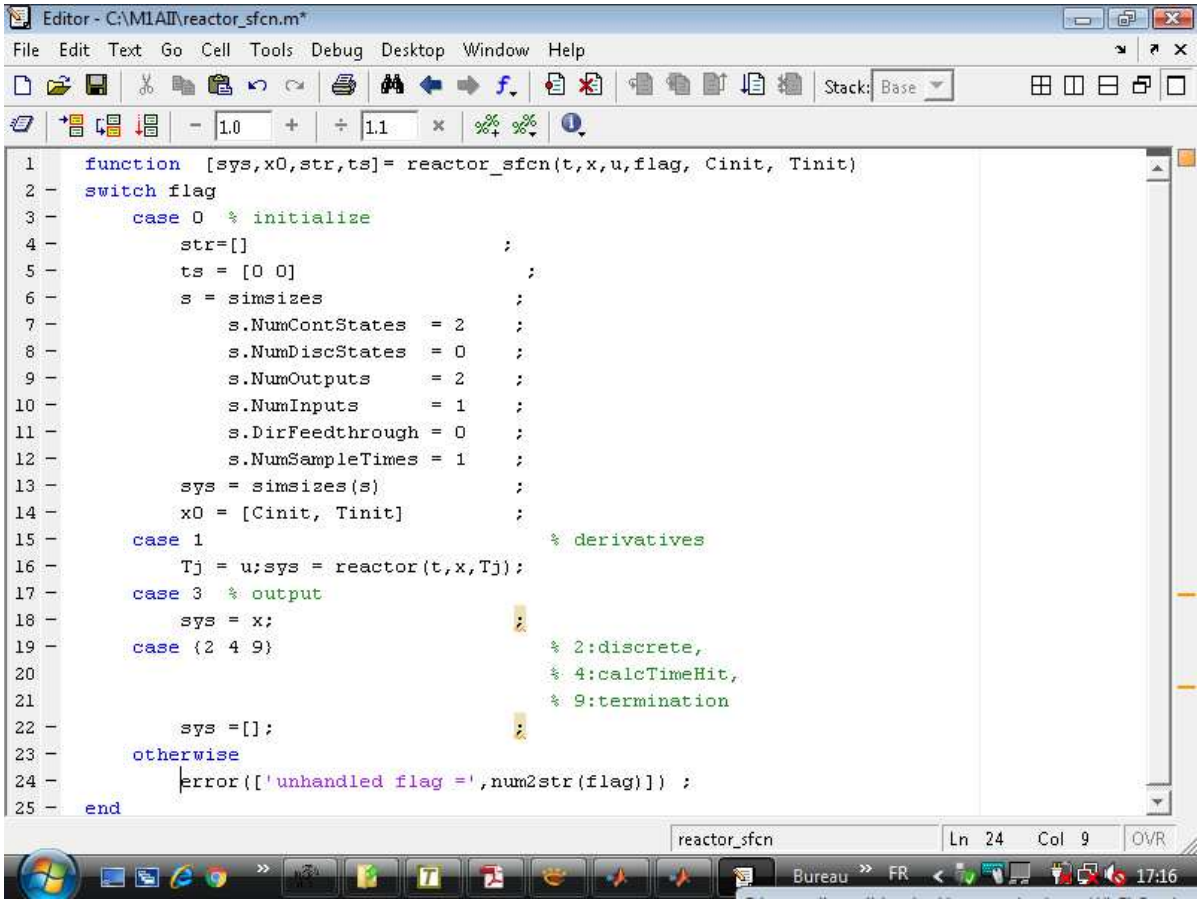

FIGURE 2 – File saved as reactor_sfcn.m

V Insert the S-Function block into the Simulink.

In the Simulink Library browser, go to the [User-Define Functions] subdirectory. Then drag-drop the S-Function block (see Figure 3). Double-click on the S-function block and fill in the parameters. Change the Sfunction name to reactor_sfcn. Also, fill in the parameters. In our case, we input 0.1,40 (which is the value for Cinit and Tinit) as shown in Figure 4.

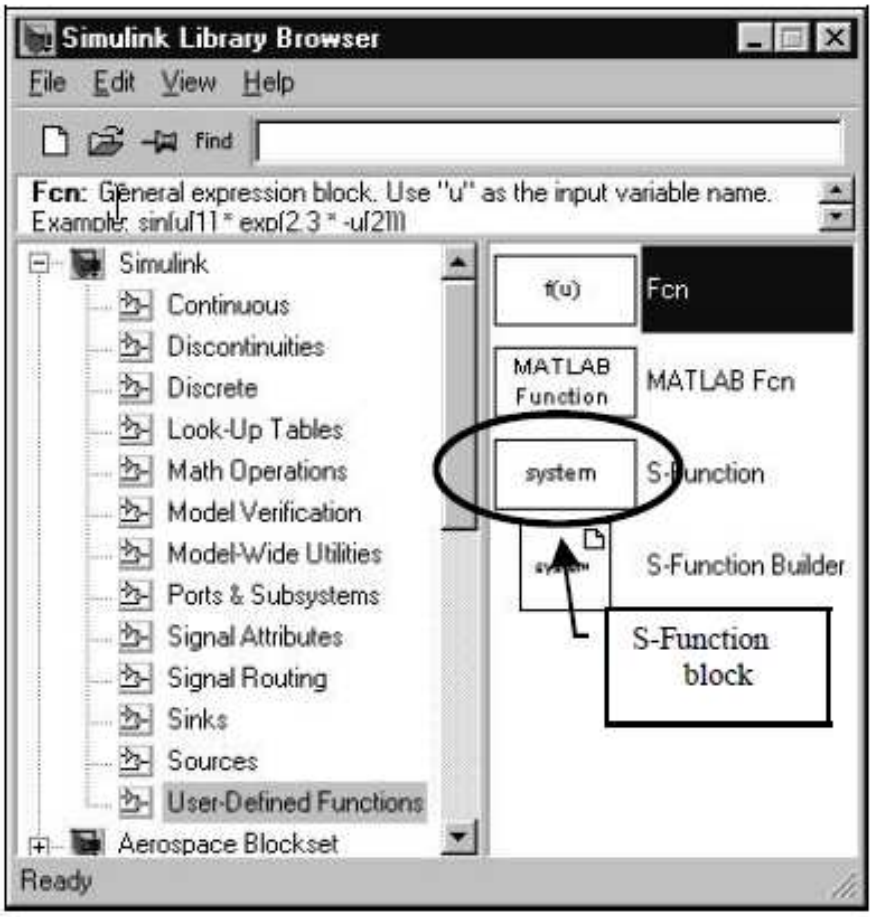

FIGURE 3 –

VI Add other Simulink blocks and simulate

Remark : In figure 5, a demux block (demultiplexer) is included to split the output vector to the 2 elements. In other applications where the input vectors has more than one element, we need a mux block (multiplexer). Both mux and demux blocks reside in the Signal Routing subdirectory of the Simulink Library browser.

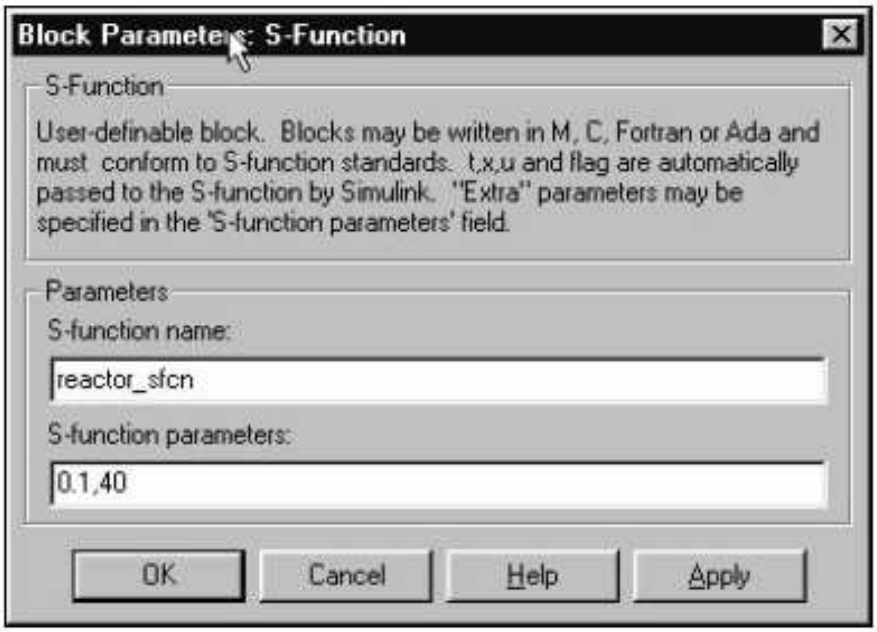

FIGURE 4 –

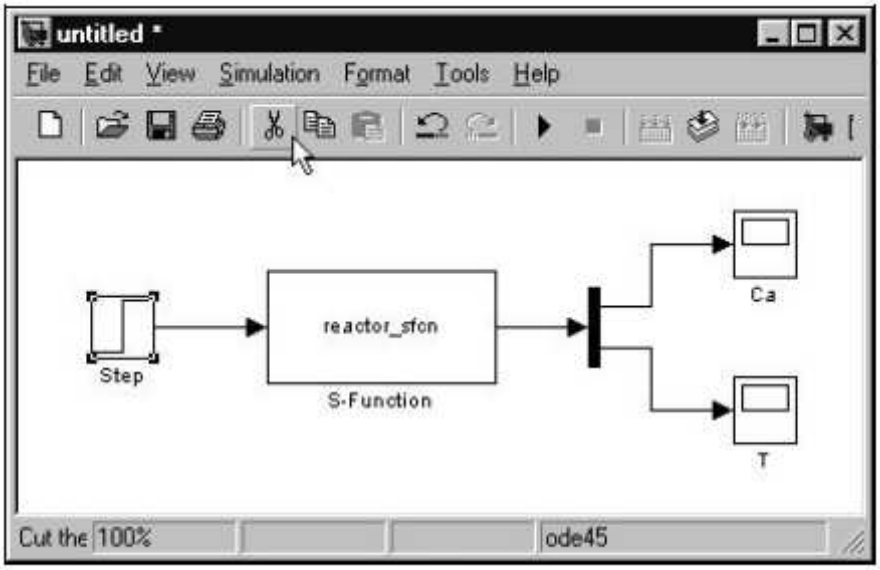

FIGURE 5 –