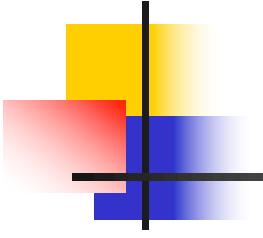
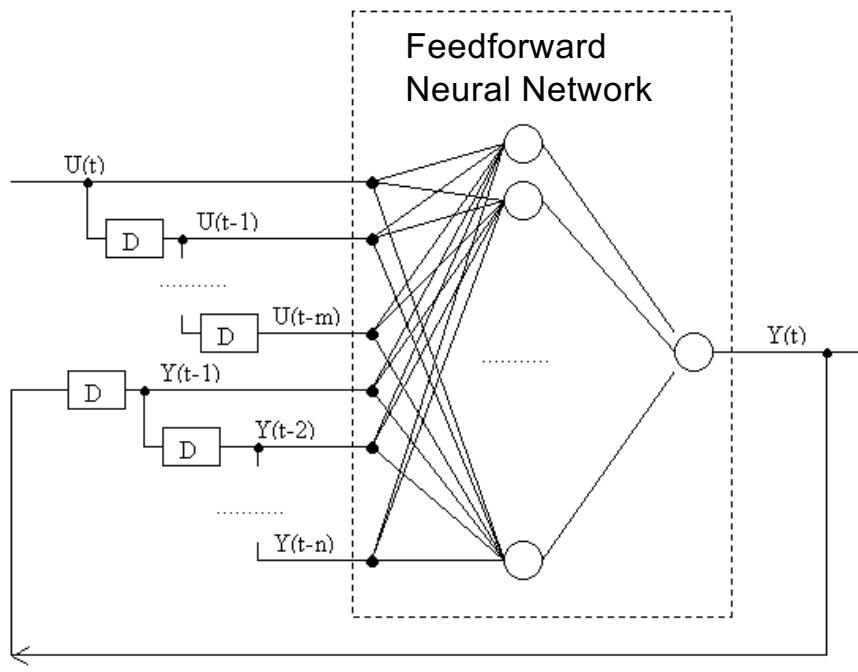


Neural Networks based control of nonlinear systems

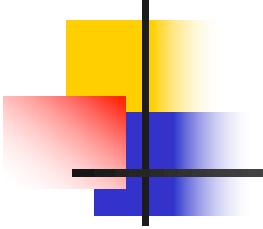


Identification of dynamic systems with Artificial Neural Networks

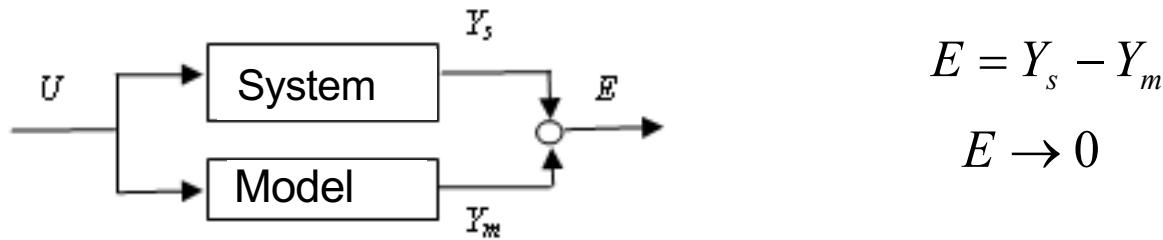
Dynamic/ feedback network:



$$\begin{aligned}
 y(t) &= F_2 \cdot (W_2 \cdot (F_1 \cdot (W_1 \cdot \begin{bmatrix} u(t) \\ \vdots \\ u(t-m) \\ y(t-1) \\ \vdots \\ y(t-n) \end{bmatrix} + \Theta_1) + \Theta_2) = \\
 &= f_{nn}(u(t), \dots, u(t-m), y(t-1), \dots, y(t-n))
 \end{aligned}$$



Identification with Artificial Neural Networks



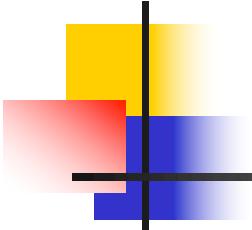
$$E = Y_s - Y_m$$

$$E \rightarrow 0$$

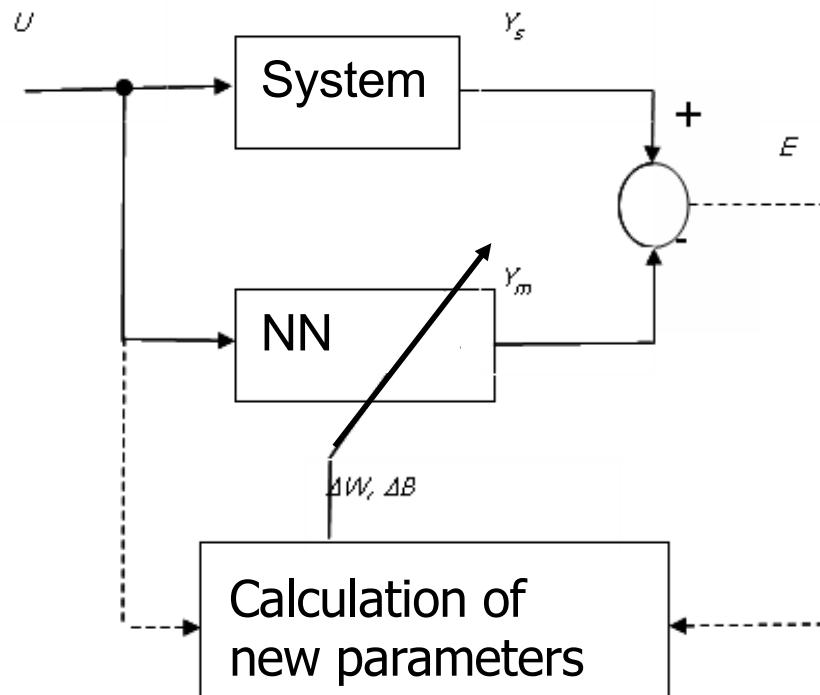
U is the input of the system and the model

Y_s is the output of the system

Y_m is the output of the model



Identification with Artificial Neural Networks

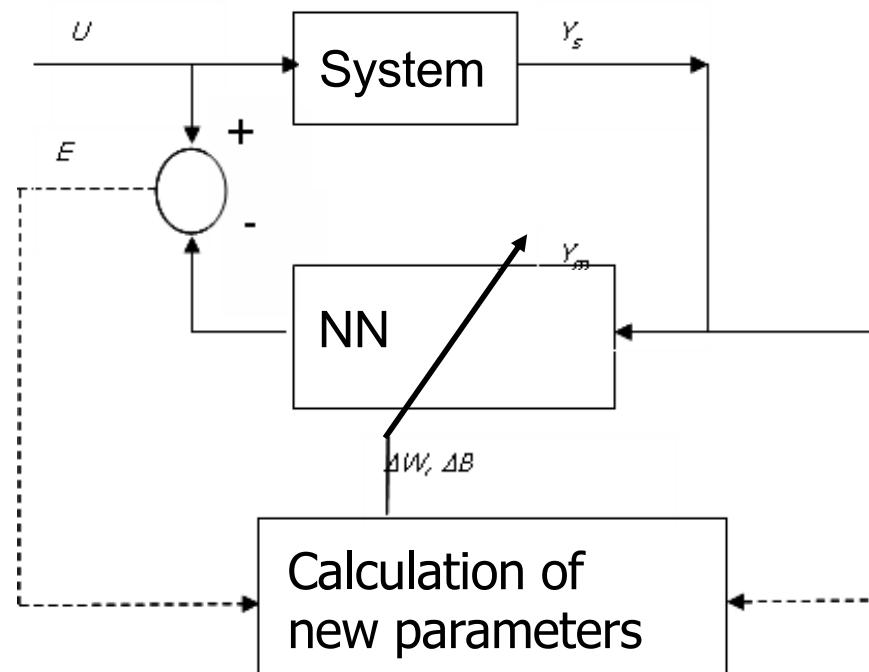


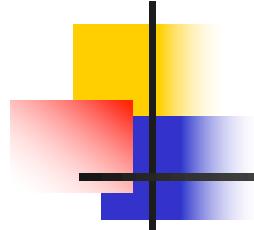


1918

TALLINNA
TEHNIKAÜLIKOOOL

Inverse model





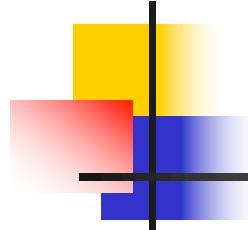
Inverse model

$$f^{-1} : Y(t) \rightarrow U(t-1)$$

$$U(t-1) = f^{-1}(Y(t), \dots, Y(t-q), U(t-2), \dots, U(t-r))$$

Consider a system: $y(t) = f(y(t-1), y(t-2), u(t-1), u(t-2))$

$$u(t-1) = f^{-1}(y(t), y(t-1), y(t-2), u(t-2))$$



Inverse model based control



$$u(t-1) = f^{-1}(y(t), y(t-1), y(t-2), u(t-2))$$

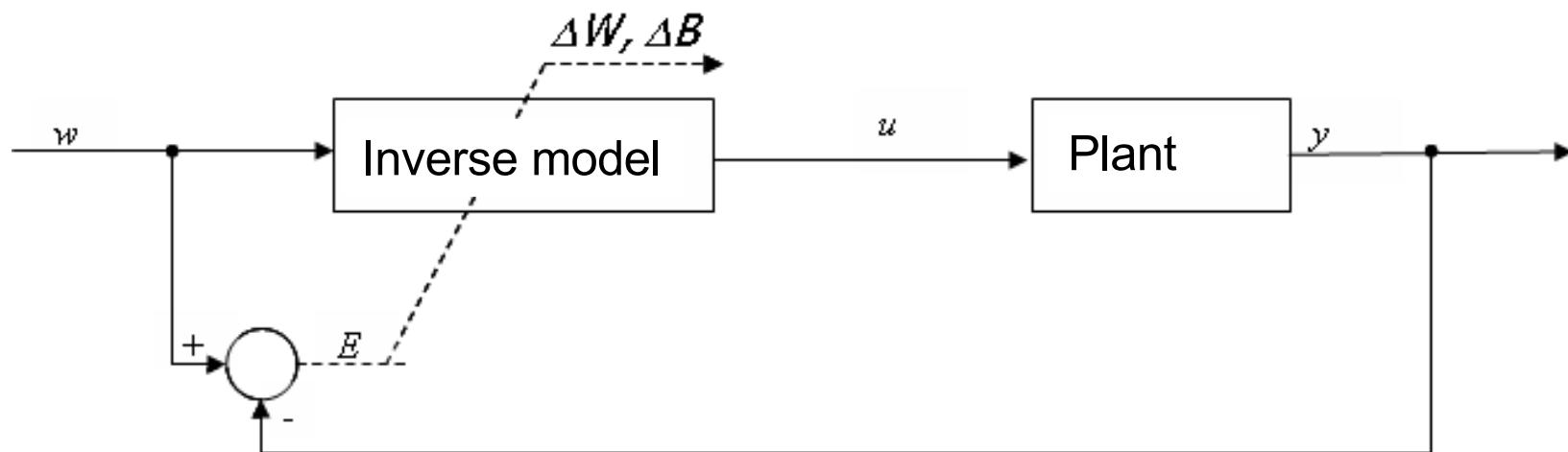
Here $y(t) = w(t)$



1918

TALLINNA
TEHNIKAÜLIKOO

Inverse model based adaptive control

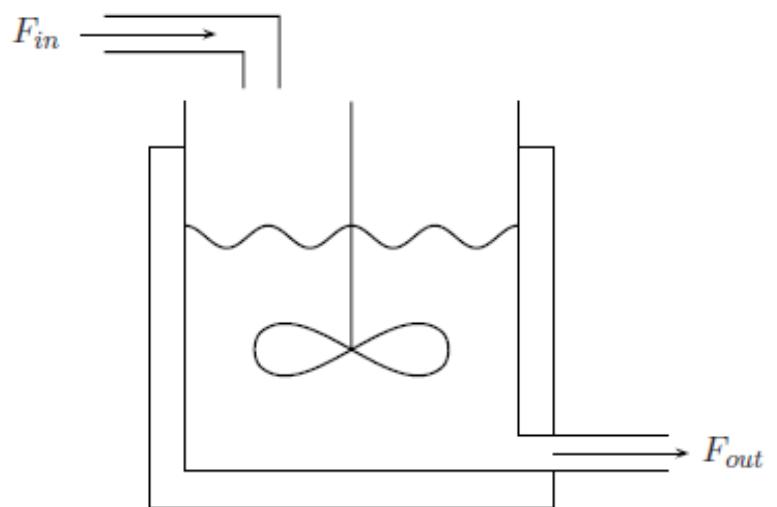


$$E \rightarrow 0$$

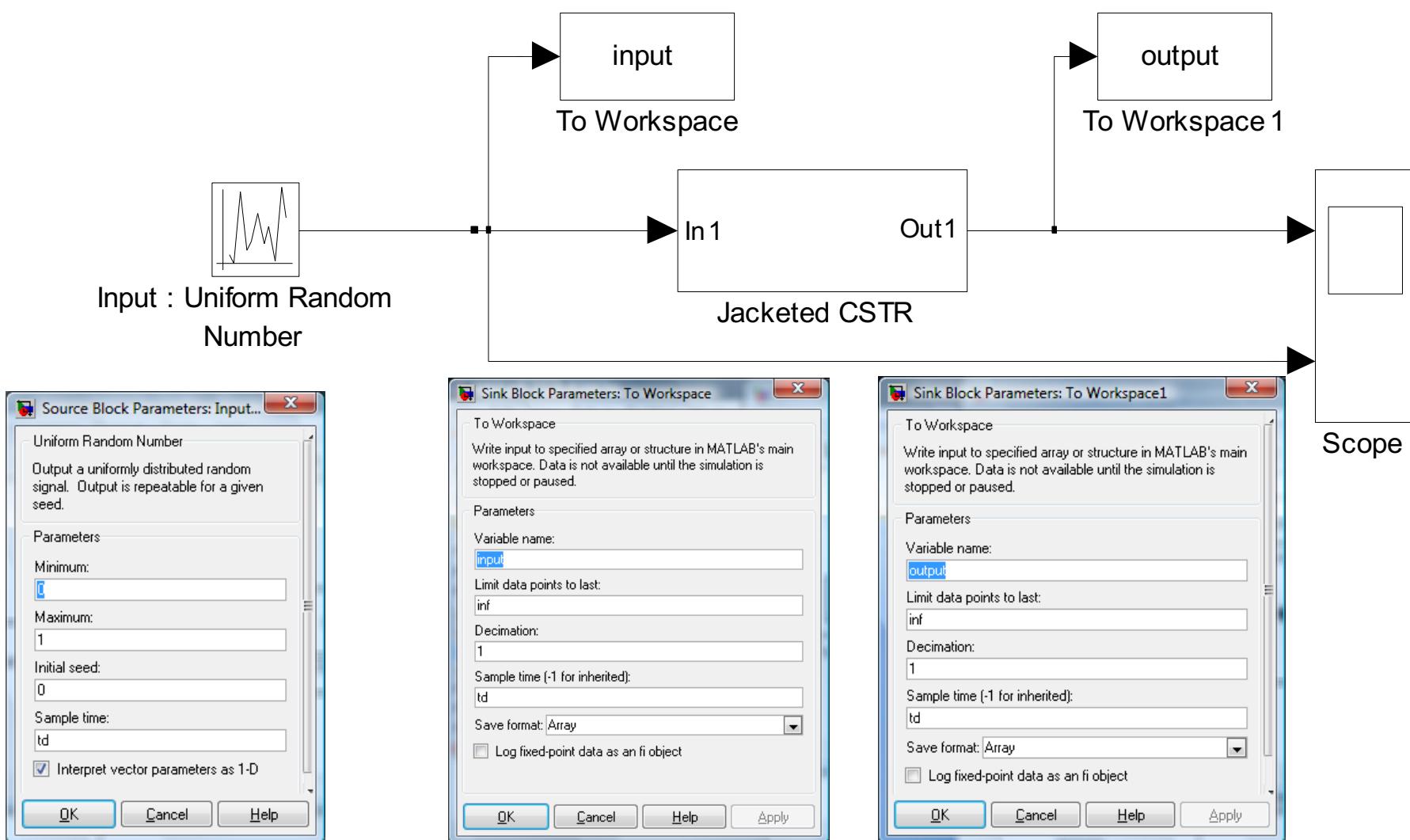
Example: Jacketed CSTR (Continuous Stirred Tank Reactor)

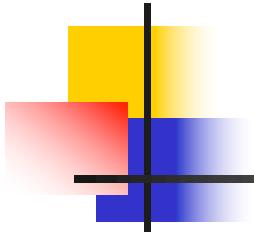
Input-Output equation:

$$\begin{aligned}y(t+2) = & 0.7653y(t+1) - 0.231y(t) + \\& + 0.4801u(t+1) - 0.6407y^2(t+1) + \\& + 1.014y(t)y(t+1) - 0.3921y^2(t+1) + \\& + 0.592y(t+1)u(t+1) - 0.5611y(t)u(t+1)\end{aligned}$$



Collecting input-output data





Training of inverse model

```
td=1
```

```
N=size(output,1)
```

```
P=[output(3:N)';output(2:N-1)';output(1:N-2)']
```

```
T=input(2:N-1)'
```

```
global net_c
```

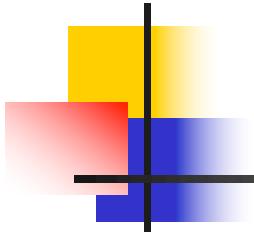
```
net_c=newff([0 1; 0 1; 0 1],[5 1],{'tansig','purelin'})
```

```
net_c.trainParam.show=1;
```

```
net_c.trainFcn='traingd';
```

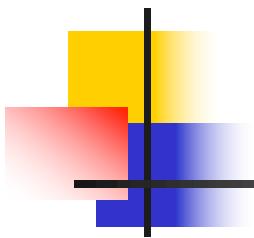
```
net_c.trainParam.epochs=3000;
```

```
net_c=train(net_c,P,T)
```

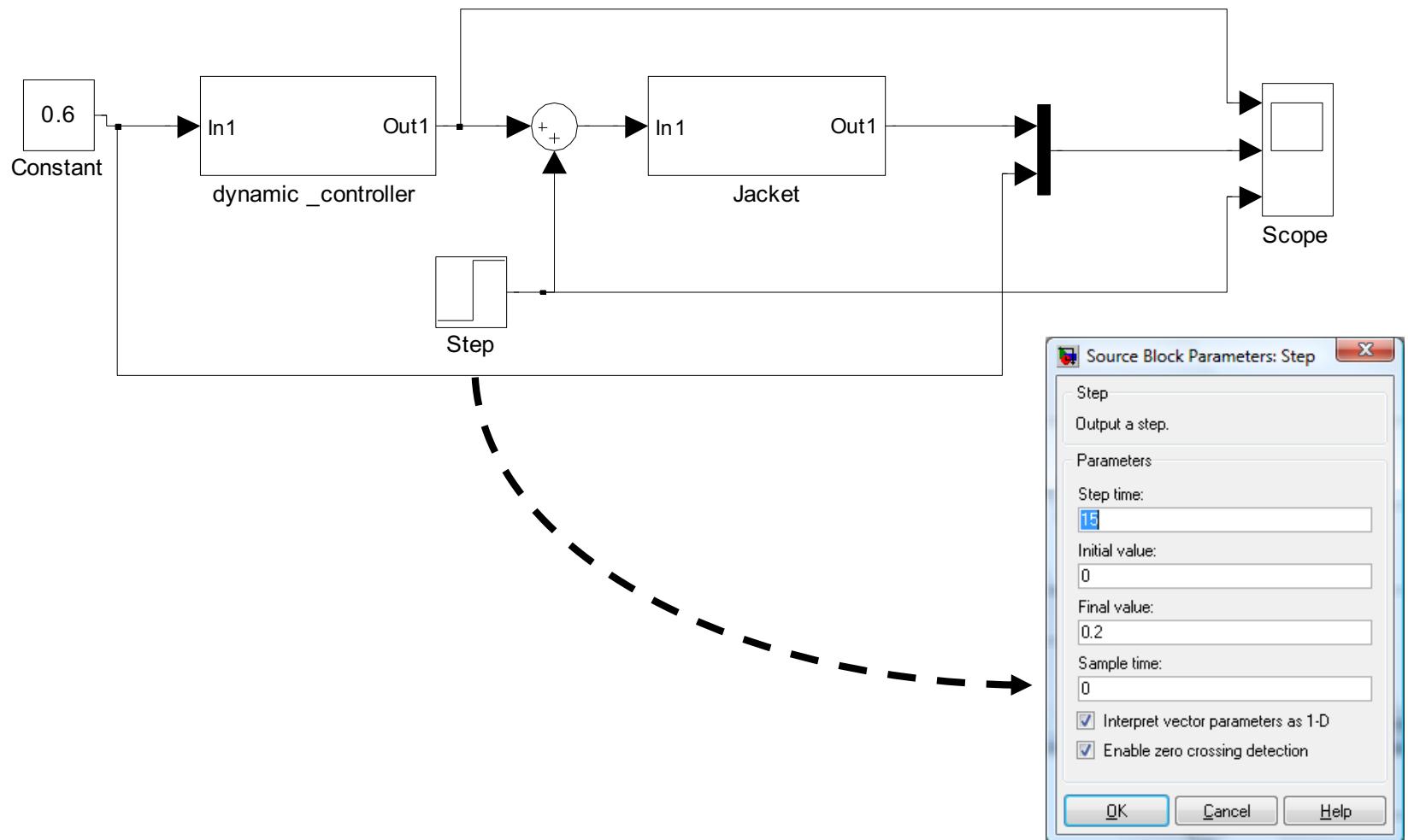


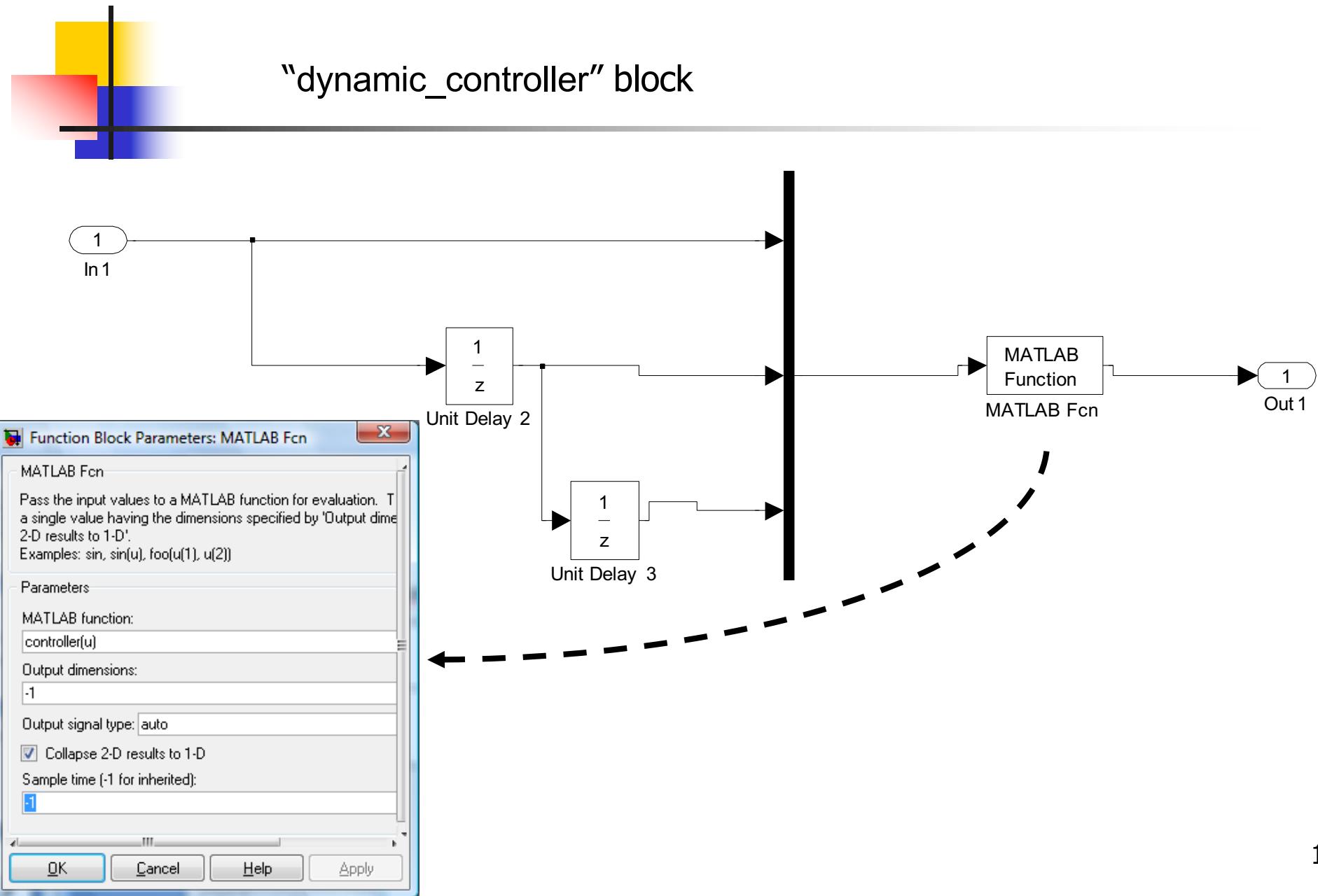
Implementation of the controller in MATLAB (m-file controller.m)

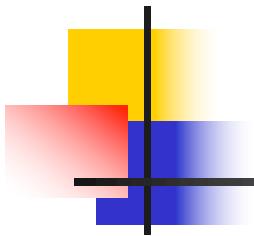
```
function control=controller(u)
global net_c
control=sim(net_c,u);
```



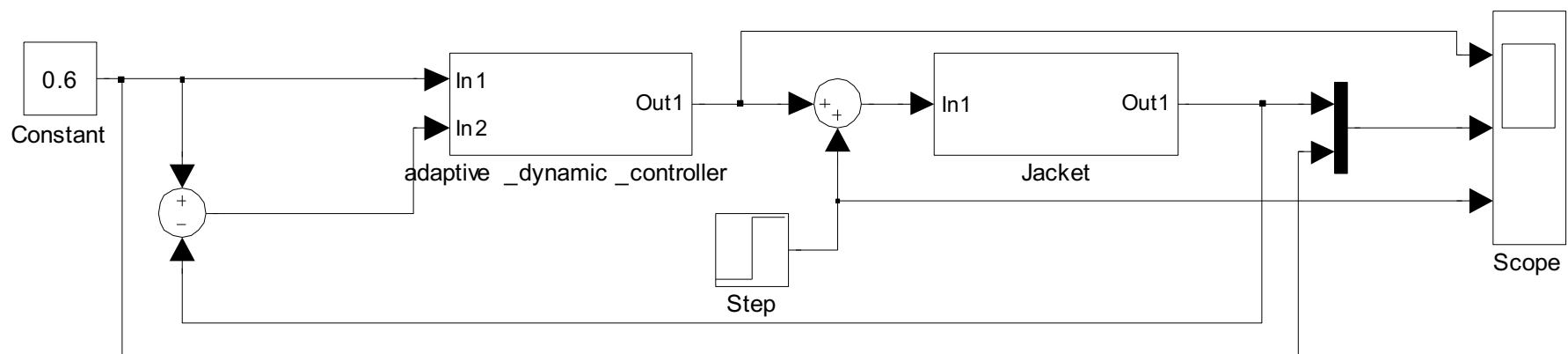
Control scheme

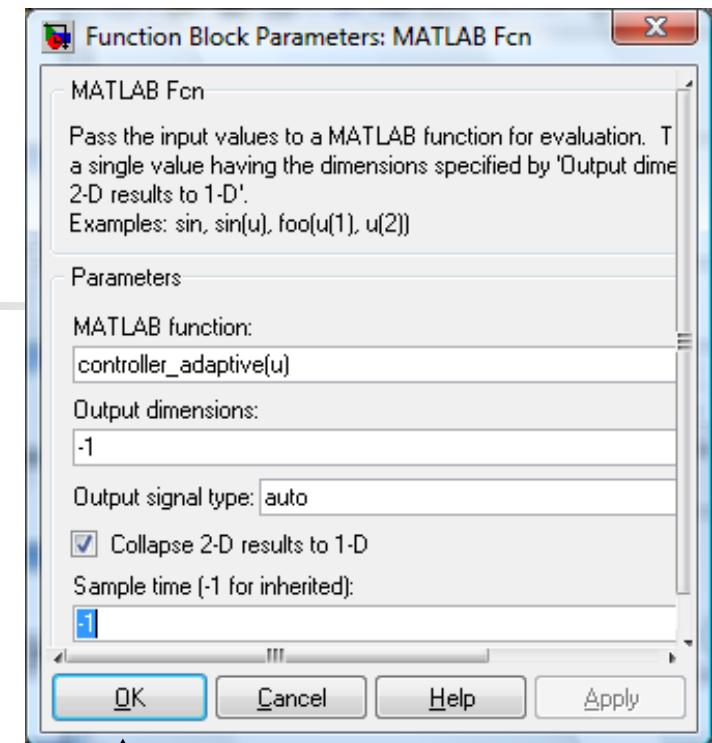
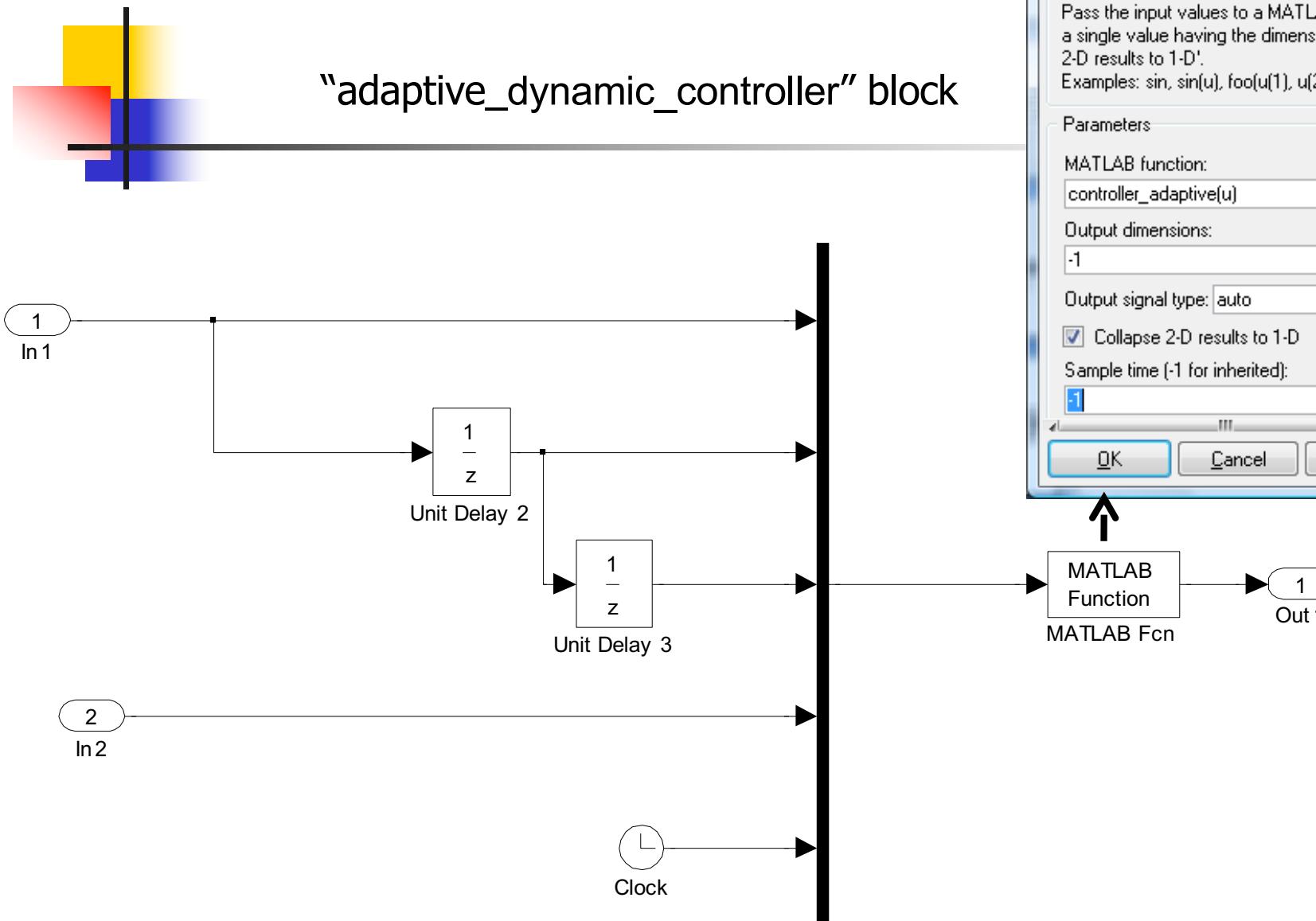


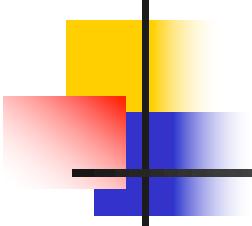




Adaptive control scheme







```
Adaptive controller = function  
"controller_adaptive" (m-file controller_adaptive.m)
```

```
function control=controller_adaptive(u)  
global net_c  
inp=u(1:3);  
error=u(4);  
time=u(5);  
control=sim(net_c,inp);  
if time>10  
net_c=adapt(net_c,inp,control+error);  
end
```