

scambiatore in controcorrente a tubi coassiali

Fluido di processo da raffreddare: olio lubrificante, fluisce nella porzione anulare (mantello) Fluido refrigerante: acqua, fluisce nel tubo interno

```
%a=acqua (fluido freddo)
%b=olio lubrificante (fluido caldo)
```

```
clear all
clc
```

```
global wa wb Tain Tbin ca cb Rba
```

```
wa=0.2 %kg/s portata acqua
```

```
wa = 0.2000
```

```
wb=0.1 %kg/s, portata olio
```

```
wb = 0.1000
```

```
Tain=30 °C, Tin_acqua
```

```
Tain = 30
```

```
Tbin=100 °C, Tin_olio
```

```
Tbin = 100
```

```
% DIAMETRI TUBAZIONI
```

```
Di=0.025;
De=0.02667;
Dan=0.045;
```

```
ca=4178; % J/(kg °K), calore specifico acqua
cb=2131; % J/(kg °K), calore specifico olio
```

```
rhoa=997; %kg/m^3 densita acqua
rhob=890; %kg/m^3 densita olio
```

```
L=78.5; % Lunghezza totale [m]
```

```
ma=rhoa*pi*Di^2/4*L; % [kg] massa acqua nella tubazione
mb=rhob*pi*(Dan^2-De^2)/4*L; % [kg] massa olio nella tubazione
```

```
K_acc=50; %[W/m^2K] conducibilita termica dell'acciaio laminato
Ki=2287; %[W/m^2K] coeff. scambio termico conduttivo lato interno (interfaccia acqua-acciaio)
Ke=30; %[W/m^2K] coeff. scambio termico conduttivo lato esterno (interfaccia olio-acciaio)
```

```
Rparete=log(De/Di)/(2*pi*L*K_acc);
```

```
Ri=1/(Ki*pi*Di*L);
Re=1/(Ke*pi*De*L);
```

```
Rba=Ri+Re+Rparete
```

```
Rba = 0.0051
```

calcolo del punto di equilibrio

```
x0 = [90,30];  
xSOL = fsolve(@sistemaNL,x0);
```

```
Equation solved.
```

```
fsolve completed because the vector of function values is near zero  
as measured by the value of the function tolerance, and  
the problem appears regular as measured by the gradient.
```

```
<stopping criteria details>
```

```
disp('Punto di equilibrio:')
```

```
Punto di equilibrio
```

```
Tabar=xSOL(1)
```

```
Tabar = 40.1140
```

```
Tbbar=xSOL(2)
```

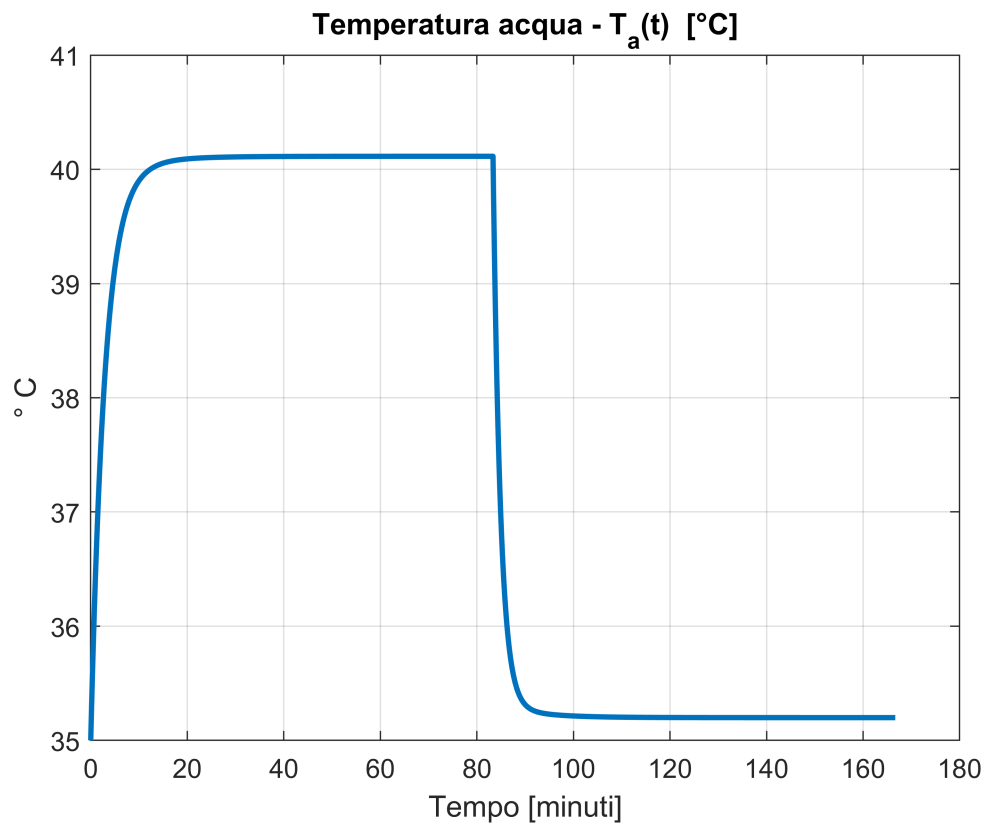
```
Tbbar = 60.3414
```

```
CHECK=sistemaNL(xSOL)
```

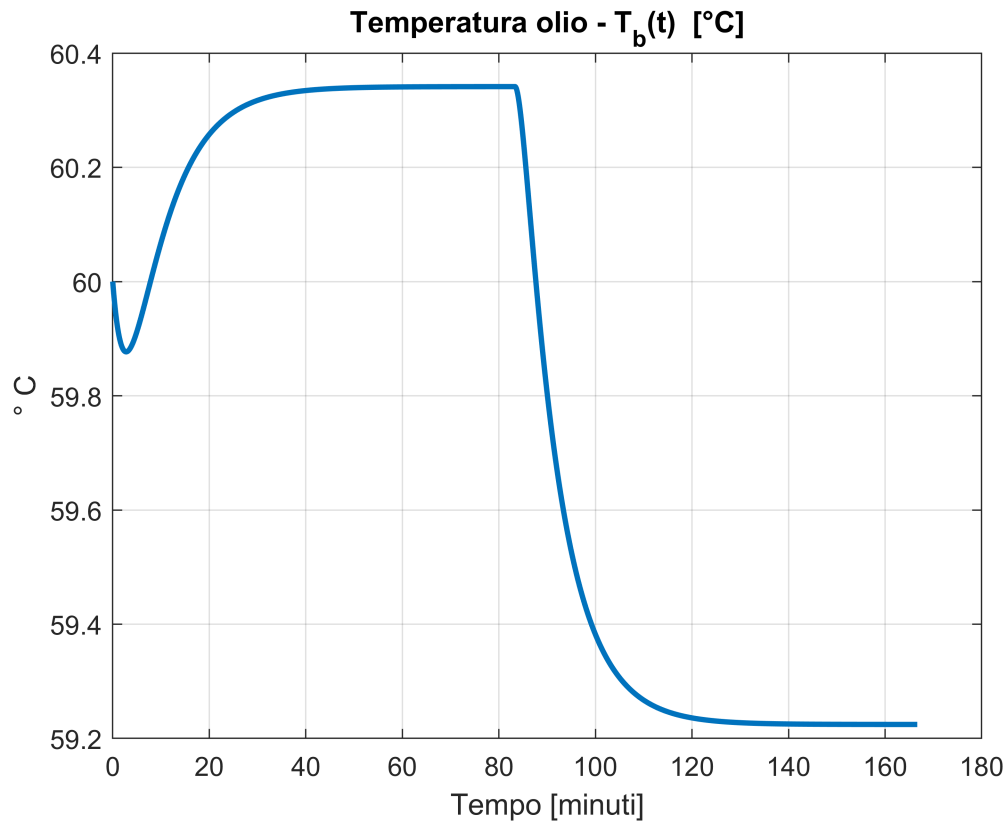
```
CHECK = 1×2  
0 0
```

simulazione tempo continuo

```
Tsim=1;  
  
%Ta0=30  
Ta0=35;  
%Tb0=100;  
Tb0=60;  
  
sim('scambiatore_dicalore_modello')  
  
disp('Modello a tempo continuo')  
figure(1)  
plot(Tamod.time/60,Tamod.data,'LineWidth',2),grid  
title('Temperatura acqua - T_a(t) [°C]')  
xlabel('Tempo [minuti]')  
ylabel('° C')
```



```
figure(2)
plot(Tbmod.time/60,Tbmod.data,'LineWidth',2),grid
title('Temperatura olio -  $T_b(t)$  [°C]')
xlabel('Tempo [minuti]')
ylabel('° C')
```



simulazione tempo continuo feedback

```

Tsim=1;

Ta0=Tabar*0.99;
Tb0=Tbbar*1.01;

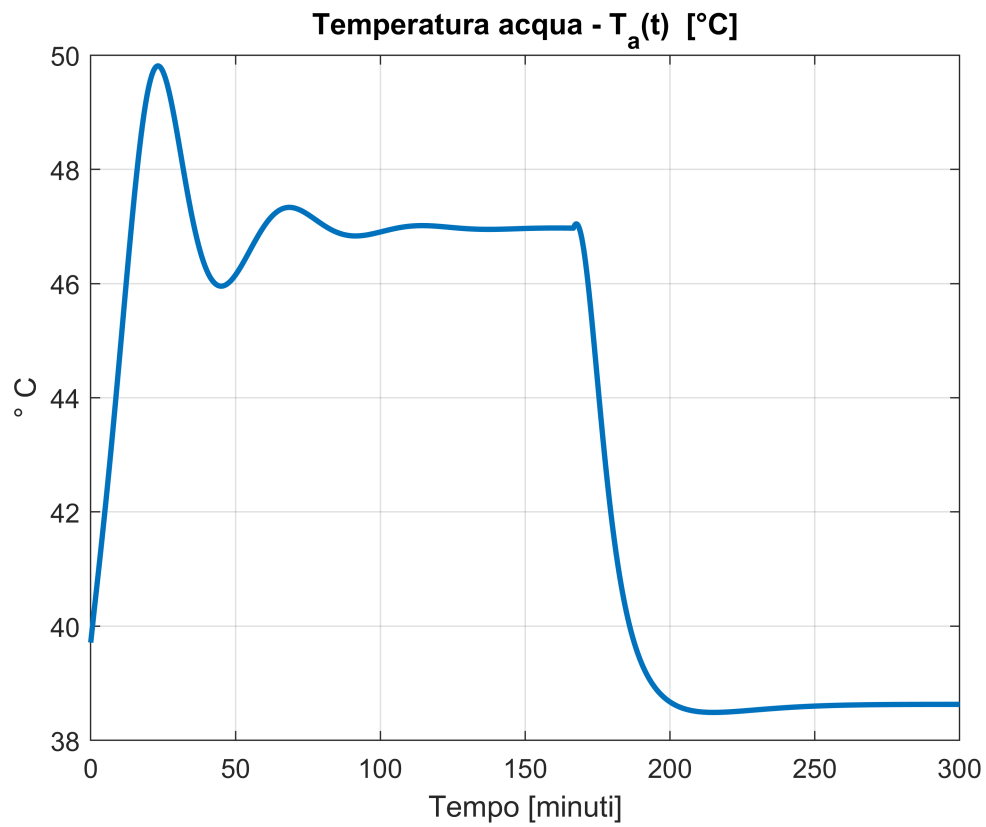
Tbin2=105;
Kp=0.02;
Ki=0.0001;

sim('scambiatore_dicalore_modello_feedback02')

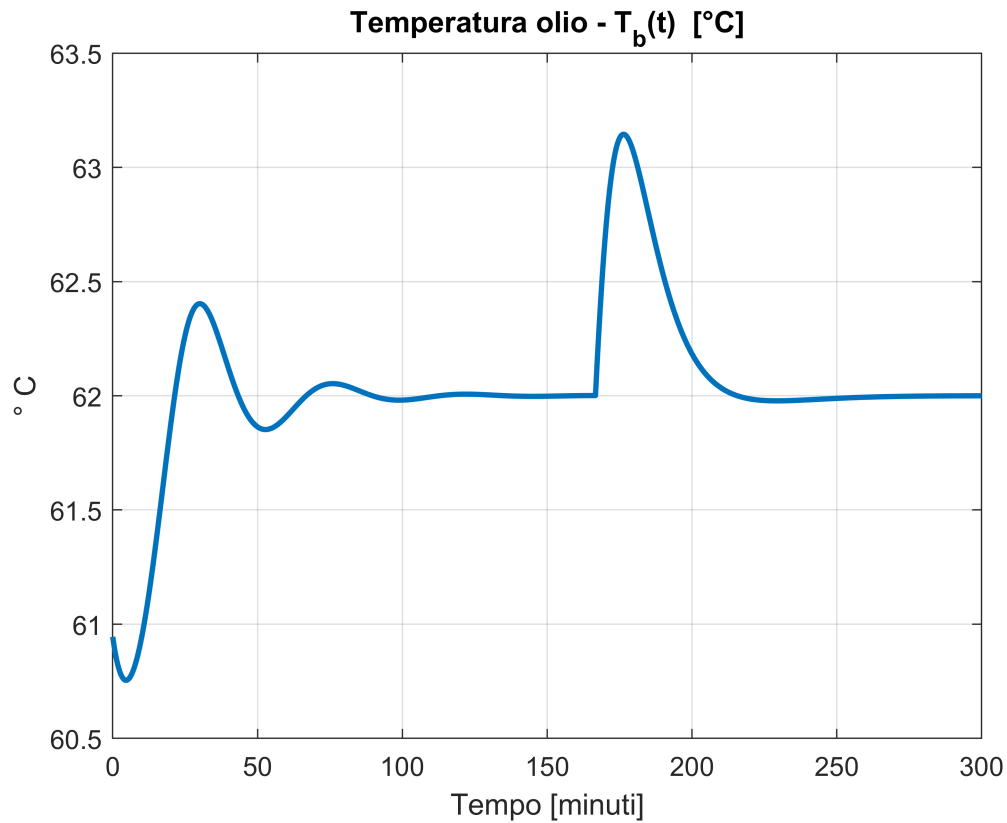
disp('Modello a tempo continuo con feedback')

figure(1)
plot(TaFb.time/60,TaFb.data,'LineWidth',2),grid
title('Temperatura acqua - T_a(t) [°C]')
xlabel('Tempo [minuti]')
ylabel('° C')

```



```
figure(2)
plot(TbFb.time/60,TbFb.data,'LineWidth',2),grid
title('Temperatura olio -  $T_b(t)$  [°C]')
xlabel('Tempo [minuti]')
ylabel('° C')
```



simulazione tempo discreto - feedback

```

Tsim=1; % passo di integrazione del modello a tempo continuo [s]
Tc=5;   % periodo di campionamento del modello a tempo discreto [s]

Ta0=Tabar*0.99;
Tb0=Tbbar*1.01;

Tbin2=105;
Kp=0.02;
Ki=0.0001;

sim('scambiatore_dicalore_modello_feedback02_DiscrTimeActivated')

disp('Modello a tempo discreto con feedback')

```

Modello a tempo discreto con feedback

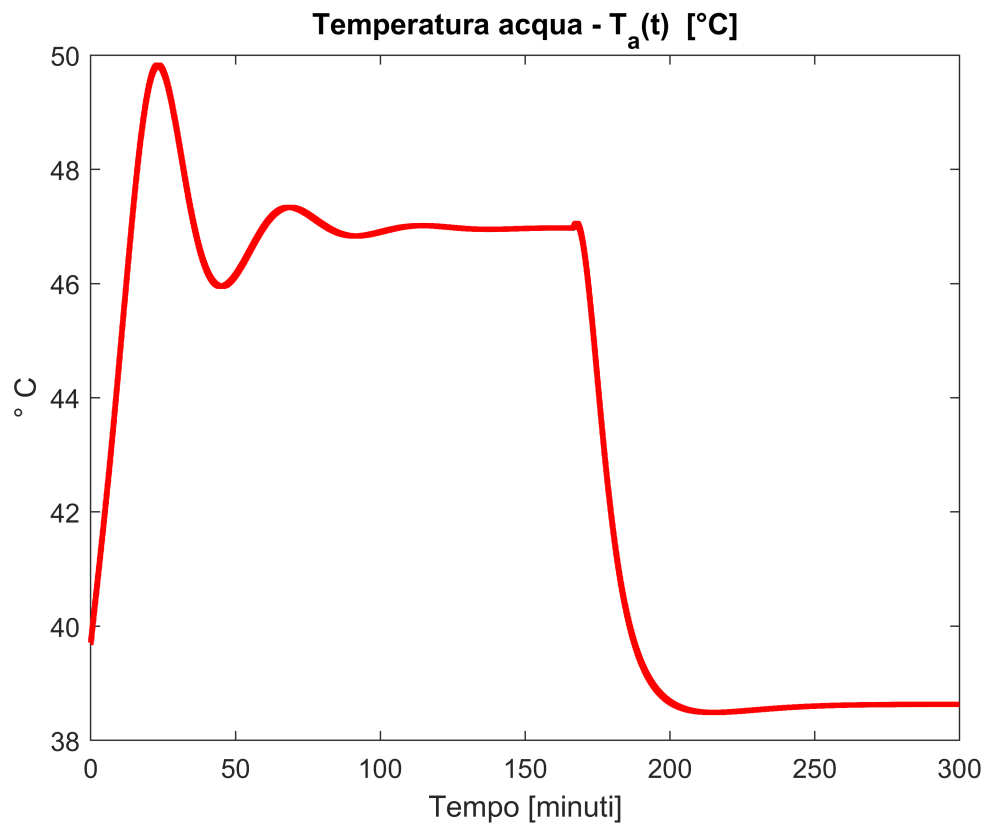
```

figure(1)
plot(TaTC.time/60,TaTC.data,'k','LineWidth',2),grid
hold on
stairs(Tak.time/60,Tak.data,'r','LineWidth',2),grid
hold off

title('Temperatura acqua - T_a(t) [°C]')
xlabel('Tempo [minuti]')

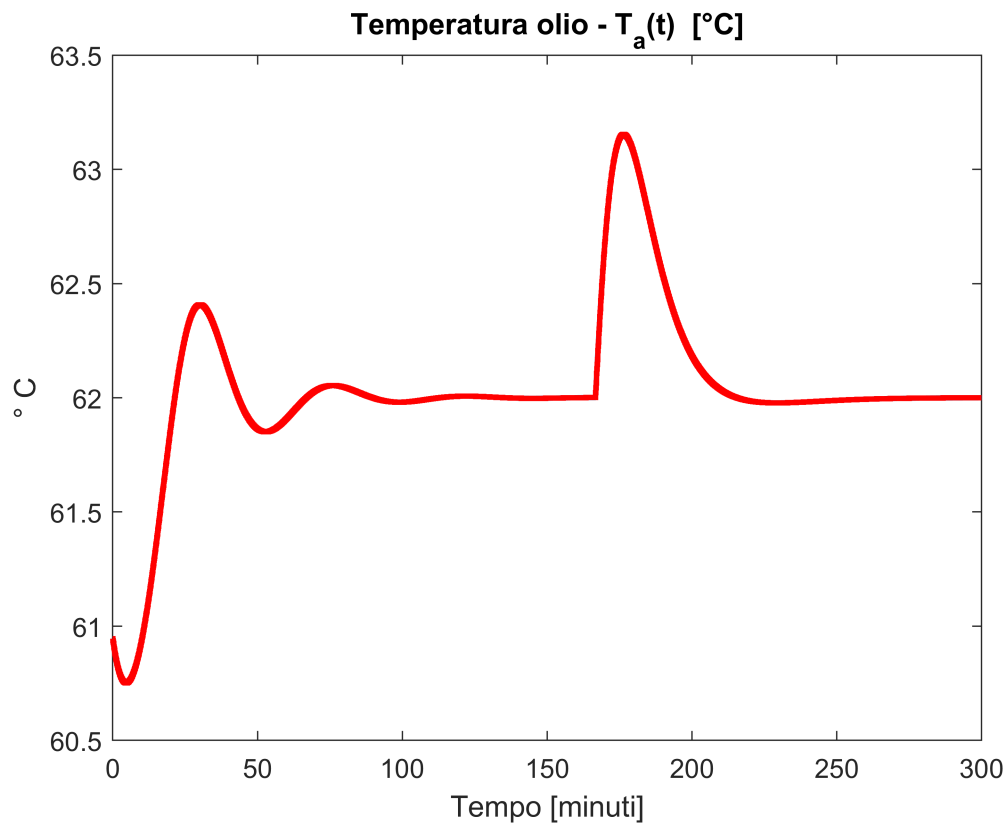
```

```
ylabel('° C')
```



```
figure(2)
%plot(TbaTC.time/60,TbTC.data,'k','LineWidth',2),grid
hold on
stairs(Tbk.time/60,Tbk.data,'r','LineWidth',2),grid
hold off

title('Temperatura olio - T_a(t) [°C]')
xlabel('Tempo [minuti]')
ylabel('° C')
```



Function per il calcolo del punto di equilibrio

```
function F = sistemaNL(x)

global wa wb Tain Tbin ca cb Rba

DT1=Tbin-x(1);
DT2=x(2)-Tain;

F(1) = ca*wa*(Tain-x(1))+1/Rba*(DT1-DT2)/log(DT1/DT2);
F(2) = cb*wb*(Tbin-x(2))-1/Rba*(DT1-DT2)/log(DT1/DT2);

end
```