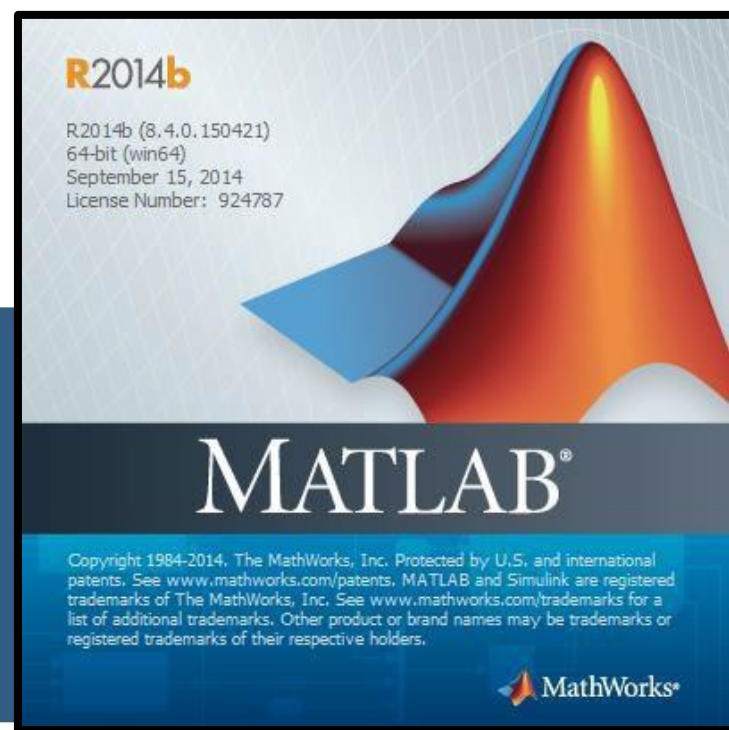




 POLITECNICO DI MILANO



# Esercitazione 4 - Soluzioni

Corso di Strumentazione e Controllo di impianti chimici

Prof. Davide Manca

PSE-Lab



# Problema



## Ese 4 – Serbatoio di stoccaggio

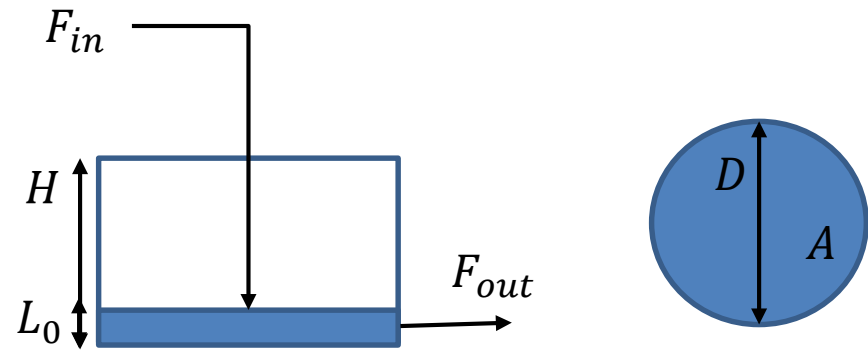
Si caratterizzi mediante il solo bilancio di massa il seguente problema relativo al riempimento di un serbatoio di stoccaggio.

$$H = 4 \text{ m}$$

$$D = 6 \text{ m}$$

$$F_{in} = F_{out} = 10 \text{ m}^3/\text{min}$$

$$L_0 = 0.5 \text{ m}$$



## Bilancio Materiale:

$$\frac{dm}{dt} = \dot{m}_{in} - \dot{m}_{out}$$

$$\dot{m} = \rho F$$

$$m = \rho V = \rho A h$$

$$\frac{\rho A dh}{dt} = \rho F_{in} - \rho F_{out}$$



$$\frac{A dh}{dt} = F_{in} - F_{out}$$



# Problema Base



```
function DinamicaSerbatoioMatStat % E4 problema base
clc
clear all
close all

global A Fin Fout

% Dati
D = 6; %m
H_vessel = 4; %m
A = pi*D^2/4; %m2

Fin = 10; % [m3/min]
Fout = 10; % [m3/min]
h0 = 0.5; %m condiz iniziale

tspan = [0:0.05:20];
% Comando
[t,H] = ode113(@eqLivello,tspan,h0);
.. Continua nella diapositiva successiva
```



## Grafico

```
figure(1)
plot(t,H,'b','LineWidth',1.3)
legend('Stationary condition')
set(gca,'fontsize',11)
title('Stationary Condition:  $F_{in} = F_{out}$ ','FontSize',18)
xlabel('Time [min]','FontSize',18)
ylabel('Level [m]','FontSize',18)

end % Chiudo la function DinamicaSerbatoioMatStat che
contiene il comando per la risoluzione della ODE, i dati, e
le istruzioni per il grafico.
```

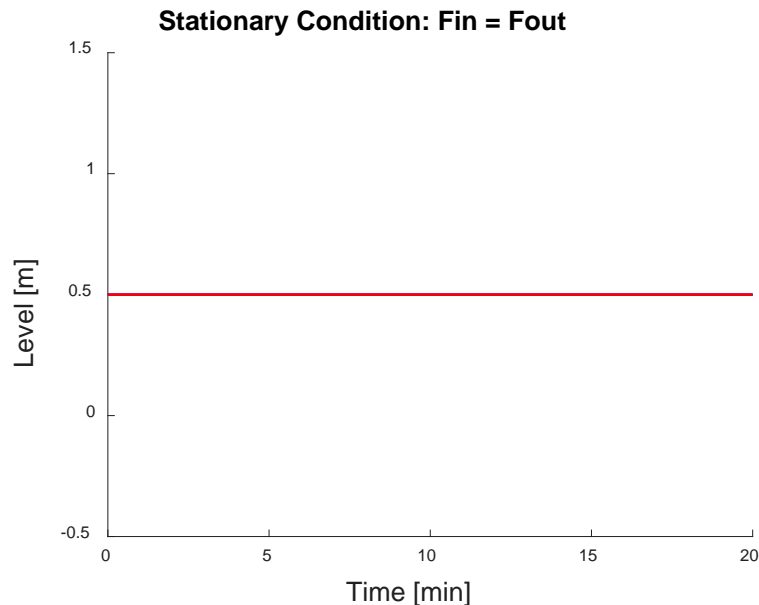


# Problema Base



```
function dh = eqLivello(t,h) % function che contiene
l'equazione che descrive la dinamica del livello
% Caso Base
global A Fin Fout
dh(1,:) = (Fin - Fout)/A;
end % Chiudo la function
```

Il modello è corretto : se  $F_{in} = F_{out}$  il livello rimane **COSTANTE**.





# Problema 1



Riduzione della portata uscente  $F_{out} = 0 \text{ m}^3/\text{min}$ . **NB.** Ci aspettiamo che il livello aumenti.  
**Azioni suggerite:** Usa la funzione **min** per correggere il risultato. Il livello non deve superare l'altezza del serbatoio. Usa un ciclo **if** per implementare il cambiamento delle condizioni.

```
function DinamicaSerbatoioMat1 % E4 Caso 1
clc
clear all
close all
global A Fin

% Dati
D = 6; %m
H_vessel = 4; %m
A = pi*D^2/4; %m2
Fin = 10; %m3/min
h0 = 0.5; %m
tspan = [0:0.05:20];

% Comando
[t,H] = ode113(@eqLivello,tspan,h0);
```



# Problema 1



```
figure(1)
hold on
plot(t,H,'b','LineWidth',1.3) % soluzione non corretta (blu)

H_max = H_vessel; % altezza del serbatoio
% Impongo un limite fisico al livello (che non deve superare
l'altezza del serbatoio) correggendo il vettore soluzione
H1 = min(H,H_max); % H1 è uguale al minimo tra Hmax e H
for i = 1:length(H)
    MaxHeight(i) = H_vessel; % costruisco una linea che
rappresenti il max livello sul grafico
end % chiudo il ciclo for
plot(t,H1,'r','LineWidth',1.3) % soluzione corretta (rossa)
plot(t,MaxHeight,'k--')

legend('No physical limit','Physical limit','Vessel Height')
set(gca,'fontsize',11)
title('Case 1 - Fout reduction','FontSize',18)
xlabel('Time [min]','FontSize',18)
ylabel('Level [m]','FontSize',18)
end
```



# Problema 1



```
function dh = eqLivello(t,h)
```

```
% Casol
```

```
global Fin A
```

```
if t <= 5 % Scelgo un certo istante a partire dal quale la  
portata uscente non è più uguale a quella entrante
```

```
Fout = 10;
```

```
else
```

```
Fout = 0;
```

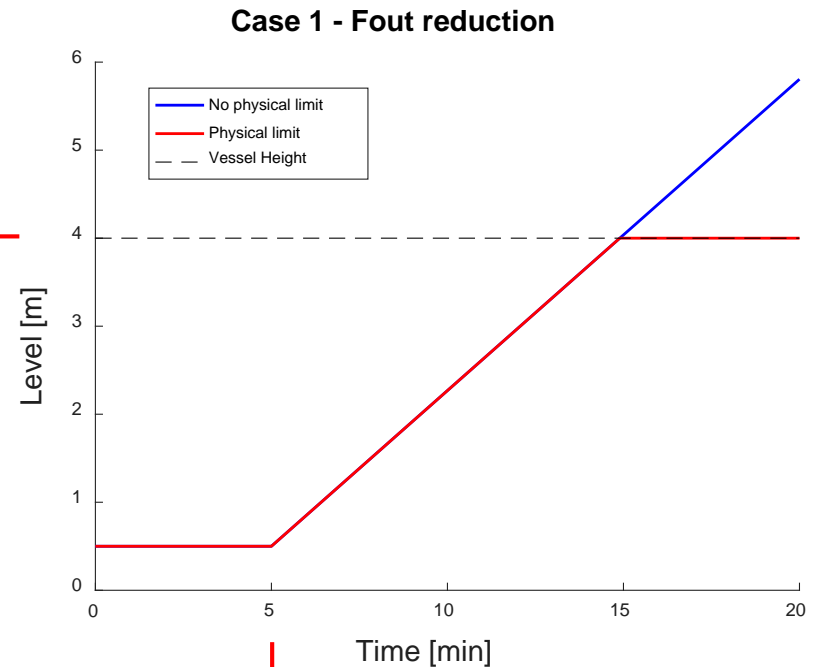
```
end
```

```
dh(1,:) = (Fin - Fout)/A;
```

```
end
```



Limite massimo del livello



Riduzione  $F_{out}$





# Problema 2



Riduzione della portata entrante  $F_{in} = 5 \text{ m}^3/\text{min}$ . **NB.** Ci aspettiamo che il livello diminuisca.  
**Azioni suggerite:** Usa la funzione **max** per correggere il risultato. Il livello non deve superare il limite fisico cioè zero. Usa un ciclo **if** per implementare il cambiamento delle condizioni.

```
function DinamicaSerbatoioMat2 % E4 Caso 2
clc
clear all
close all
global A Fout1
% Dati
D = 6; %m
H_vessel = 4; %m
A = pi*D^2/4; %m2
Fout1 = 10; %m3/min
h0 = 0.5; %m
tspan = [0:0.05:20];

% Comando
[t,H2] = ode113(@eqLivello2,tspan,h0);
```



## Problema 2



% Impongo un limite fisico al livello (che non può diventare negativo)

```
H2 = max(0, H2); % H2 non può mai essere negativo; al massimo 0
```

```
for i = 1:length(H2)
    MinHeight(i) = 0.;
end
```

```
figure(2)
hold on
plot(t, H2, 'b', 'LineWidth', 1.3)
plot(t, MinHeight, 'k--')
set(gca, 'fontsize', 11)
title('Case 2 - Fin reduction', 'FontSize', 18)
xlabel('Time [min]', 'FontSize', 18)
ylabel('Level [m]', 'FontSize', 18)
xlim([0 8])
ylim([-0.2 0.8])
end
```



# Problema 2



```
function dh2 = eqLivello2(t,h)
```

```
global A Fout1
```

```
if t <= 2 % Scelgo un certo istante a partire dal quale la  
portata entrante non è più uguale a quella uscente
```

```
    Fin1 = 10;
```

```
else
```

```
    Fin1 = 5;
```

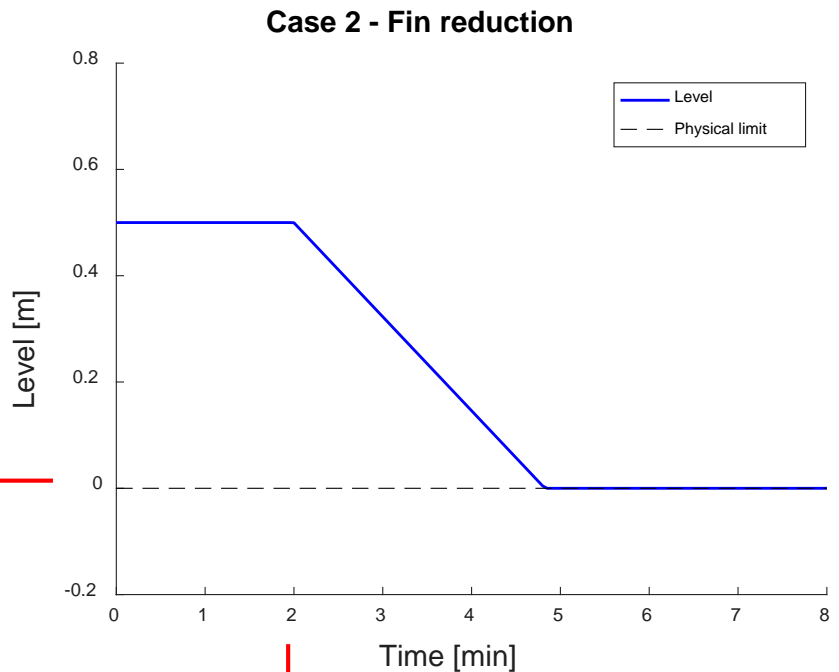
```
end
```

```
dh2(1,:) = (Fin1 - Fout1)/A;
```

```
end
```



Limite fisico del livello



Riduzione  $F_{in}$



# Problema 3



Ripetere l'esercizio per un serbatoio di  $D = 22$  m e  $H = 8$  m.

**Azioni suggerite:** Usa un ciclo **for** per confrontare la dinamica dei due serbatoi di diversa dimensione.

```
function DinamicaSerbatoioConfronto
```

```
clc
```

```
clear all
```

```
close all
```

```
global A Fin Fout1
```

```
Fin = 10; % [m3/min]
```

```
Fout1 = 10; % [m3/min]
```

```
h0 = 0.5; %m
```

```
D = [6 22]; %m
```

```
H_vessel = [4 8]; %m
```

```
for indiceCaso = 1:2
```

```
d = D(indiceCaso); % in questo modo considero un valore per  
volta di diametro e altezza
```

```
h_vessel = H_vessel(indiceCaso);
```

```
A = pi*d^2/4; %m2
```



# Problema 3



```
tspan = [0:0.05:360];  
% Caso 1  
[t,H] = ode113(@eqLivello,tspan,h0);  
H_max = h_vessel;  
H1 = min(H,H_max);  
  
figure(1)  
hold on  
if indiceCaso == 1  
p1 = plot(t,H1,'b','LineWidth',1.3); % limite  
else  
p2 = plot(t,H1,'m','LineWidth',1.3);  
end  
legend('Caso 1','Caso2')  
set(gca,'fontsize',11)  
title('Case_1 - F_o_u_t reduction','FontSize',18)  
xlabel('Time [min]','FontSize',18)  
ylabel('Level [m]','FontSize',18)  
xlim([0 360])  
ylim([0 10])
```



# Problema 3



```
% Caso 2
[t,H2] = ode113(@eqLivello2,tspan,h0);
H2 = max(0,H2);

figure(2)
hold on
if indiceCaso == 1
p1 = plot(t,H2,'b','LineWidth',1.3);
else
p2 = plot(t,H2,'m','LineWidth',1.3);
legend([p1 p2],'Caso1','Caso2')
end % Chiudo il ciclo if
set(gca,'fontsize',11)
title('Case_2 - F_i_n reduction','FontSize',18)
xlabel('Time [min]','FontSize',18)
ylabel('Level [m]','FontSize',18)
ylim([-0.05 0.6])
xlim ([0 350])
end % Chiudo il ciclo for
end % Chiudo la function
```



# Problema 3



```
function dh = eqLivello(t,h)
```

```
% Caso1
```

```
global A Fin
```

```
if t <= 50
    Fout = 10;
else
    Fout = 0;
end
```

```
dh(1,:) = (Fin - Fout)/A;
end
```

```
function dh2 = eqLivello2(t,h)
```

```
% Caso 2
```

```
global A Fout1
```

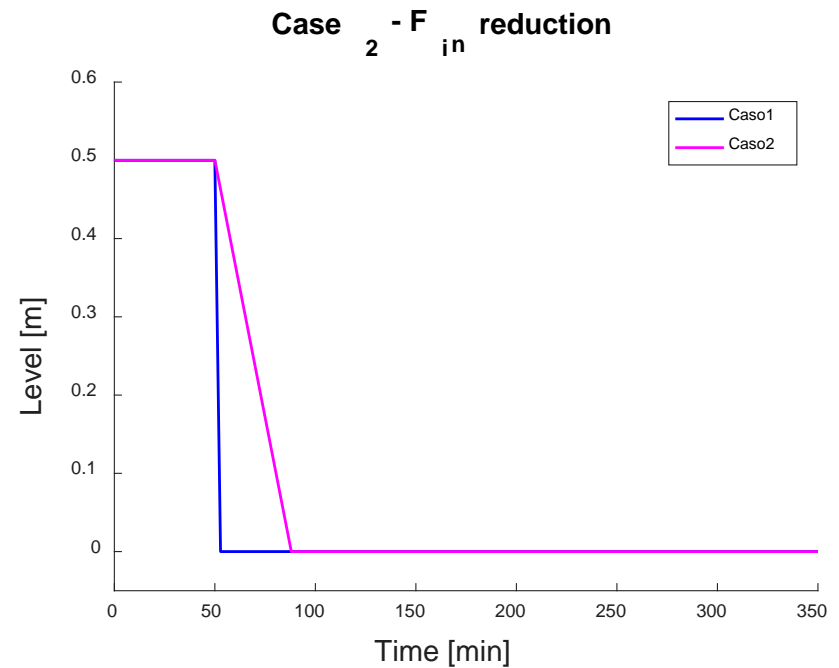
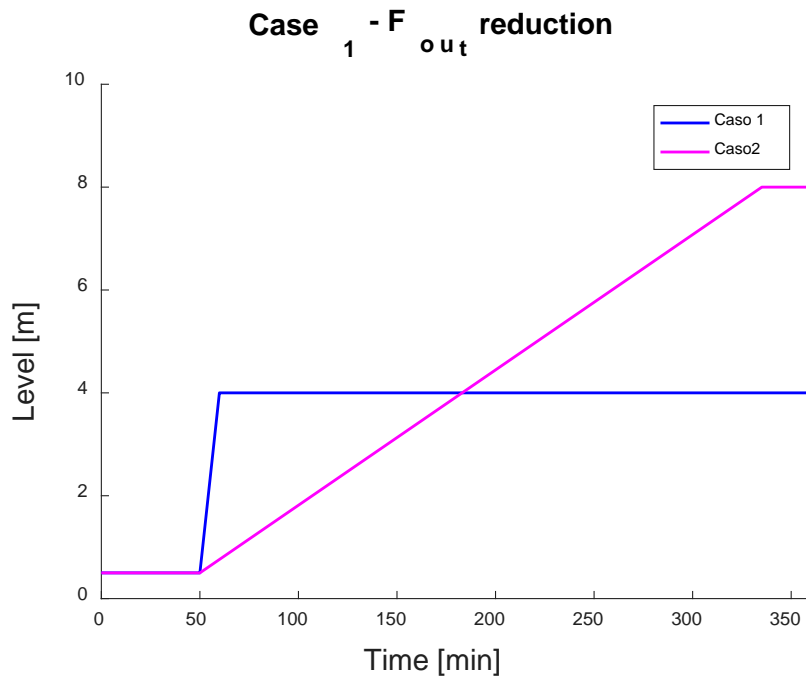
```
if t <= 50
    Fin1 = 10;
else
    Fin1 = 5;
end
```

```
dh2(1,:) = (Fin1 - Fout1)/A;
end
```

**NB.** Le tre functions sono nello stesso script



# Soluzione



La dinamica di un serbatoio di dimensioni maggiori è più lenta.