# P1 - Problem n.1

Given the discrete-time system S (A, B, C, D) described below:

$$\begin{cases} x(k+1) = Ax(k) + Bu(k) \\ y(k) = Cx(k) + Du(k) \end{cases}$$

With:

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 1 & -1 & 0 \\ 3 & 0 & 0 \end{bmatrix} \qquad B = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \qquad C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad D = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

With the reference:

$$r(k) = \begin{bmatrix} 1(k) \\ 1(k) \\ 1(k) \end{bmatrix} \qquad 1(k) = 1 \quad \forall k > 0$$

Cost function:

$$V(k) = \sum_{i=H_W}^{H_P} [r(k+i) - y(k+i)]^T \begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} [r(k+i) - y(k+i)]$$

$$H_P = 2 \qquad H_w = 1 \qquad H_c = 2$$

**Questions:**

1. Compute the cost function V(k)
2. Compute u(k)
3. Compute u(k+1)
4. Considering u(k), what can be said about the stability of the closed-loop system?
5. By considering the cost function defined below, compute V(k), u(k), u(k+1)

$$V_1(k) = \sum_{i=H_W}^{H_P} [r(k+i) - y(k+i)]^T \begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} [r(k+i) - y(k+i)] + \sum_{i=0}^{H_C-1} [u(k+i)]^T \, [2] \, [u(k+i)]$$

## P1 - Problem n.2

Given the continuous-time system shown in the Figure and described by the following equations:

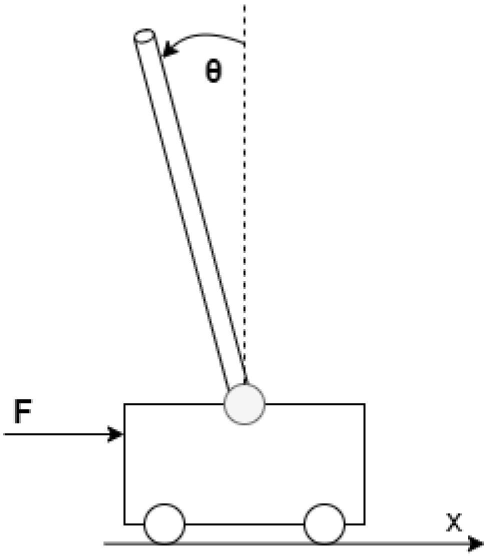
$$\dot{x_1} = x_2$$

$$\dot{x_2} = \frac{u - K_d x_2 - mLx_4^2 \sin x_3 + mg \sin x_3 \cos x_3}{M + m \sin^2 x_3}$$

$$\dot{x_3} = x_4$$

$$\dot{x_4} = \frac{\left(\frac{(u - K_d x_2 - mLx_4^2 \sin x_3) \cos \mathrm{x}_3}{M + m}\right) + g \sin x_3}{\left(L - \frac{mL \cos^2 x_3}{M + m}\right)}$$

$$x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix} \qquad y(t) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} x(t)$$




Where:

| Name | Description | Value [unit] |
|---|---|---|
| $u$ | Input Force on the cart | [N] |
| $x_1$ | Cart position (z) | [m] |
| $x_2$ | Cart velocity (v) | [m/s] |
| $x_3$ | Angle of the pendulum ($\theta$) | [rad] |
| $x_4$ | Angular velocity of the pendulum ($\omega$) | [rad/s] |
| $M$ | Mass of the cart | 1 [kg] |
| $m$ | Mass of the pendulum | 1 [kg] |
| $L$ | Pendulum length | 0.5 [m] |
| $K_d$ | Cart damping | 10 [N/m/s] |

**Questions:**

6. The settling time required is 1s. Choose the appropriate Ts and Hp. Consider Hc = 5.

7. Assume the following initial conditions for the system.
   -The cart is stationary at z=0.
   -The pendulum is in a downward equilibrium position where theta=-pi.
   The first control objective is:
   -Swing-up control: Initially swing the pendulum up to an inverted equilibrium position where z=0 and theta=0.
   - Initialize the MPC controller with the found parameters.
   - Constrain the system accordingly.
   - Run a simulation lasting 5s.
   - Observe and comment the simulation results.

8. The second control objective is the cart position reference tracking: move the cart to a new position using a step setpoint change, keeping the pendulum inverted.

   Implement the following cost function and run a simulation.
   Explain why this cost function helps in pursuing the objective and how does it affects the simulation.

$$V = \sum_{i=1}^{H_P} [r(i) - y(i)]^T \begin{bmatrix} 3 & 0 \\ 0 & 3 \end{bmatrix} [r(i) - y(i)] + \sum_{i=1}^{H_P - 1} \Delta u^T 0.1 \Delta u$$

9. At time t=5s a new reference is applied yref2=[8 0]'.
   - Run a simulation lasting 10s.
   - Observe and comment the simulation results.

10. The force F has a range between +-100N and the cart position is limited to the range +-10m.
    - Constrain the system accordingly.
    - Run a simulation lasting 10s.
    - Observe and comment the simulation results. Do the constraints operate during the simulation?

11. Analyze the computational time required for each move update. Using the parameters Ts and Hp found at point n.6, can you control the system in real time?

12. Upload your scripts calling it NAME_SURNAME.docx

# Problem n.1 - Soluzioni

## Domanda n.1

Il sistema è rappresentato tramite le matrici di stato in forma classica:

$$\begin{cases} x(k+1) = Ax(k) + Bu(k) \\ y(k) = Cx(k) + Du(k) \end{cases}$$

Con le matrici definite come:

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 1 & -1 & 0 \\ 3 & 0 & 0 \end{bmatrix} \qquad B = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \qquad C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad D = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Lo riscrivo come:

$$x_1(k+1) = u(k)$$
$$x_2(k+1) = x_1(k) - x_2(k)$$
$$x_3(k+1) = 3x_1(k)$$

La matrice di uscita C è una matrice identità di ordine 3 quindi:

$$y_1(k) = x_1(k)$$
$$y_2(k) = x_2(k)$$
$$y_3(k) = x_3(k)$$

Osservo la funzione di costo:

$$V(k) = \sum_{i=H_W}^{H_P} [r(k+i) - y(k+i)]^T \begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} [r(k+i) - y(k+i)]$$

Sostituisco quanto trovato in precedenza e la riscrivo come:

$$V(k) = \sum_{i=H_W}^{H_P} [r(k+i) - x(k+i)]^T \begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} [r(k+i) - x(k+i)]$$

$H_P = 2$ quindi avremo bisogno dei valori $x(k+2)$ che ricaviamo:

$$x_1(k+2) = u(k+1)$$
$$x_2(k+2) = x_1(k+1) - x_2(k+1) = u(k) - x_1(k) + x_2(k)$$
$$x_3(k+2) = 3x_1(k+1) = 3u(k)$$

Inizio a lavorare sul calcolo di V(k):

$$[r(k+i) - x(k+i)] = \left[ \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} x_1(k+i) \\ x_2(k+i) \\ x_3(k+i) \end{bmatrix} \right] = \begin{bmatrix} 1 - x_1(k+i) \\ 1 - x_2(k+i) \\ 1 - x_3(k+i) \end{bmatrix}$$

Riscrivo

$$V(k) = \sum_{i=1}^{2} [1 - x_1(k+i) \quad 1 - x_2(k+i) \quad 1 - x_3(k+i)] \begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} 1 - x_1(k+i) \\ 1 - x_2(k+i) \\ 1 - x_3(k+i) \end{bmatrix}$$

Da cui:

$$V(k)=\sum_{i=1}^{2}\begin{bmatrix}2\big(1-x_1(k+i)\big) & 1-x_2(k+i) & -\big(1-x_3(k+i)\big)\end{bmatrix}\begin{bmatrix}1-x_1(k+i)\\ 1-x_2(k+i)\\ 1-x_3(k+i)\end{bmatrix}=$$
$$=\sum_{i=1}^{2}\begin{bmatrix}2\big(1-x_1(k+i)\big)^2 & \big(1-x_2(k+i)\big)^2 & -\big(1-x_3(k+i)\big)^2\end{bmatrix}$$

Da cui:

$$V(k)=2\big(1-x_1(k+1)\big)^2+\big(1-x_2(k+1)\big)^2-\big(1-x_3(k+1)\big)^2+2\big(1-x_1(k+2)\big)^2+\big(1-x_2(k+2)\big)^2 - \big(1-x_3(k+2)\big)^2$$

Sostituisco i valori trovati in precedenza e ricavo V(k):

$$V(k)=2(1-u(k))^2+\big(1-x_1(k)+x_2(k)\big)^2-(1-x_1(k))^2+2(1-u(k+1))^2 + (1-u(k)+x_1(k)-x_2(k))^2-(1-3u(k))^2$$

## Domanda n.2

Voglio ora ricavarmi u(k). Calcolo la derivata di V(k) rispetto ad u(k)

$$\frac{\partial V(k)}{\partial u(k)}=-4\big(1-u(k)\big)-2\big(1-u(k)+x_1(k)-x_2(k)\big)+6\big(1-3u(k)\big)$$
$$=-4+4u(k)-2+2u(k)-2x_1(k)+2x_2(k)+6-18u(k)=-12u(k)-2x_1(k)+2x_2(k)$$

Pongo la derivata di V(k) pari a zero per ricavarmi u(k)

$$-12u(k)-2x_1(k)+2x_2(k)=0 \Leftrightarrow u(k)=\frac{x_2(k)-x_1(k)}{6}$$

## Domanda n.3

Voglio ora ricavarmi u(k+1). Calcolo la derivata di V(k) rispetto ad u(k+1)

$$\frac{\partial V(k)}{\partial u(k+1)}=-4\big(1-u(k+1)\big)$$

Pongo la derivata di V(k) pari a zero per ricavarmi u(k+1)

$$-4\big(1-u(k+1)\big)=0 \Leftrightarrow u(k+1)=1$$

## Domanda n.4

Vogliamo ora valutare la stabilità del sistema in anello chiuso utilizzando u(k).

Sostituisco il valore di u(k) trovato dentro l'equazione di stato:

$$x_1(k+1) = \frac{x_2(k) - x_1(k)}{6}$$
$$x_2(k+1) = x_1(k) - x_2(k)$$
$$x_3(k+1) = 3x_1(k)$$

Ricavo la matrice A di stato

$$A = \begin{bmatrix} -1/6 & 1/6 & 0 \\ 1 & -1 & 0 \\ 3 & 0 & 0 \end{bmatrix}$$

Per valutare la stabilità calcolo gli autovalori di A. C'è un autovalore esterno al cerchio unitario quindi il sistema è instabile. Non era necessario valutare il doppio polo nell'origine. Inoltre questo lo potevamo già immaginare in quanto Hp è minore di n, ovvero dell'ordine del sistema.

## Domanda n.5

Vogliamo calcolare $V_1(k)$,u(k) e u(k+1) per la nuova funzione di costo:

$$V_1(k) = \sum_{i=H_W}^{H_P} [r(k+i) - y(k+i)]^T \begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} [r(k+i) - y(k+i)] + \sum_{i=0}^{H_C-1} [u(k+i)]^T \, [2][u(k+i)]$$

Notiamo che:

$$V_1(k) = V(k) + \sum_{i=0}^{H_C-1} [u(k+i)]^2 \, [2] \, [u(k+i)]$$

Non c'è bisogno di calcolarsi tutta la funzione di costo da zero ma possiamo calcolare solo il termine aggiuntivo. Quindi:

$$V_1(k) = 2\big(1 - x_1(k+1)\big)^2 + \big(1 - x_2(k+1)\big)^2 - \big(1 - x_3(k+1)\big)^2 + 2\big(1 - x_1(k+2)\big)^2 + \big(1 - x_2(k+2)\big)^2 - \big(1 - x_3(k+2)\big)^2 + 2u^2(k) + 2u^2(k+1)$$

Alla stessa maniera ricavo u(k)

$$\frac{\partial V_1(k)}{\partial u(k)} = \frac{\partial V(k)}{\partial u(k)} + \frac{\partial(2u^2(k) + 2u^2(k+1))}{\partial u(k)} = -12u(k) - 2x_1(k) + x_2(k) + 4u(k)$$

$$-12u(k) - 2x_1(k) + 2x_2(k) + 4u(k) = 0 \Leftrightarrow u(k) = \frac{x_2(k) - x_1(k)}{4}$$

Analogamente ricavo u(k+1)

$$\frac{\partial V_1(k)}{\partial u(k+1)} = \frac{\partial V(k)}{\partial u(k+1)} + \frac{\partial(2u^2(k) + 2u^2(k+1))}{\partial u(k+1)} = -4\big(1 - u(k+1)\big) + 4u(k+1) = 8u(k+1) - 4$$

$$8u(k+1) - 4 = 0 \Leftrightarrow u(k+1) = \frac{1}{2}$$

# Problem n.2 - Soluzioni

## Domanda n.6

Banalmente Ts = 0.1, Hp=10, Hc=5

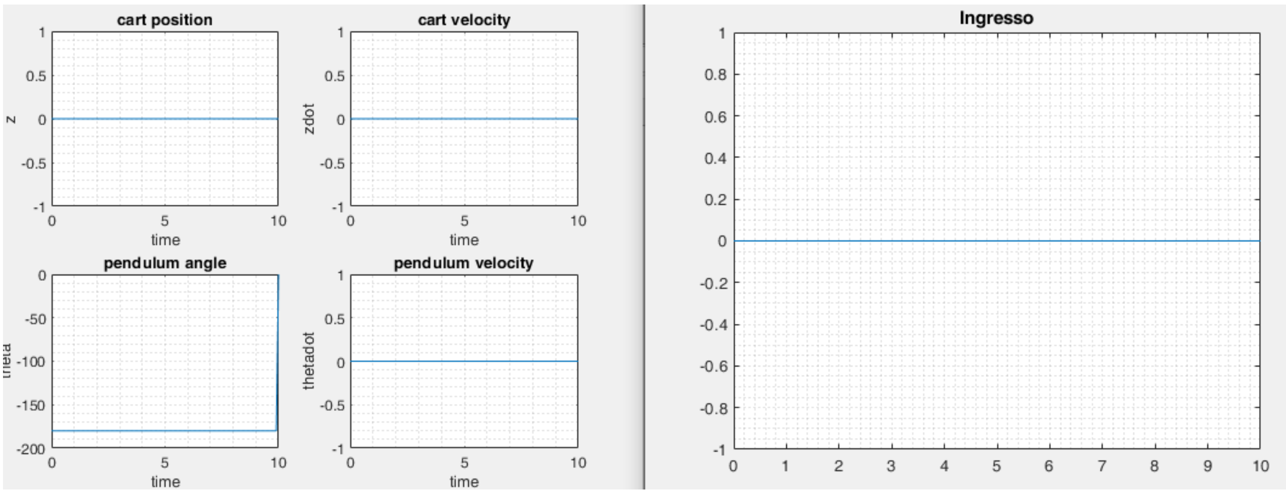
## Domanda n.7

Dalla domanda ricaviamo:

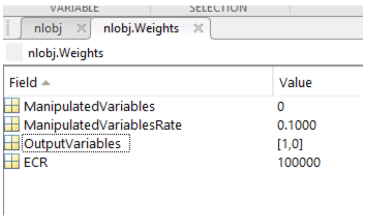
$$x_0 = [0 \quad 0 \quad -\pi \quad 0]^T$$

$$y_{ref} = [0 \quad 0]$$

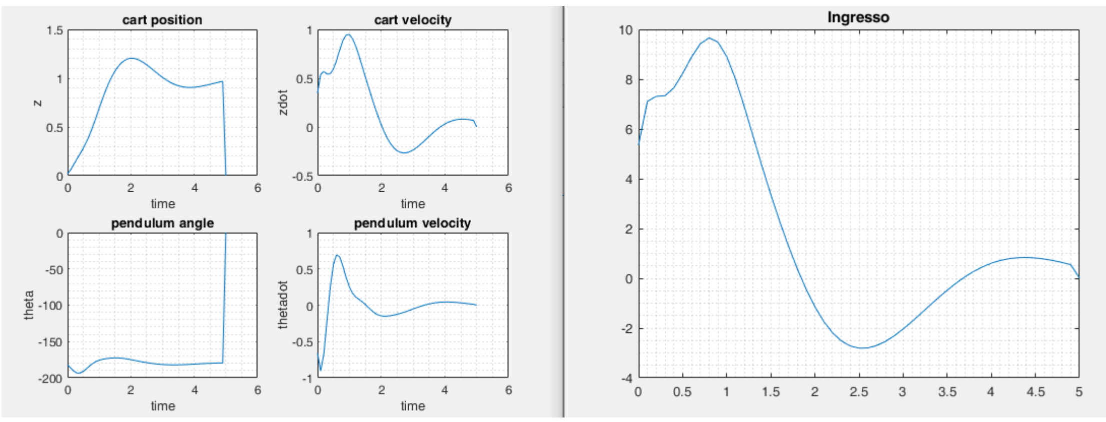
Effettuando una prima simulazione notiamo questa situazione. Il controllore non ha azionato il sistema. Tutti gli stati sono rimasti nelle condizioni iniziali.



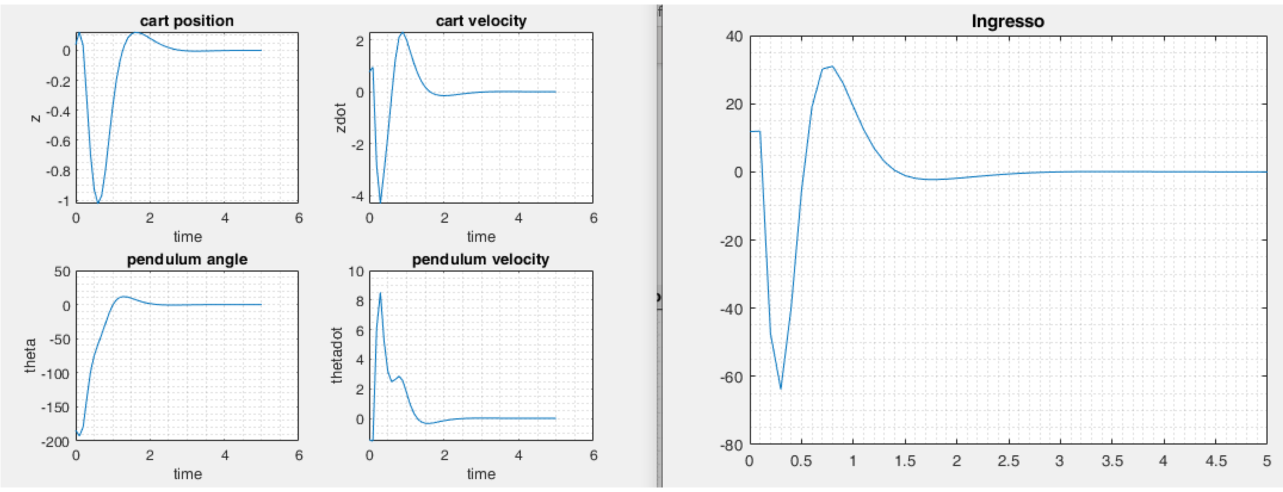

Questo accade perché di default MATLAB inizializza solo i primi nu pesi delle uscite. Ispezionando nlobj ho la conferma:




Avendo un peso solo sul riferimento della posizione del cart, il controllore non ritiene necessario azionare il sistema (x0(1) = yref(1)) => errore nullo. Potevamo anche fermarci qui. Se avessimo provato a impostare un peso anche al tracking dell’angolo del pendolo, avremmo ottenuto la seguente situazione:




In questo caso notiamo come il sistema non riesca a compiere lo swing-up ed assestarsi nel tempo di simulazione. Avremmo potuto cercare di rendere l’azione di controllo più “aggressiva” portando i pesi sulle uscite a 2 ottenendo il seguente risultato:



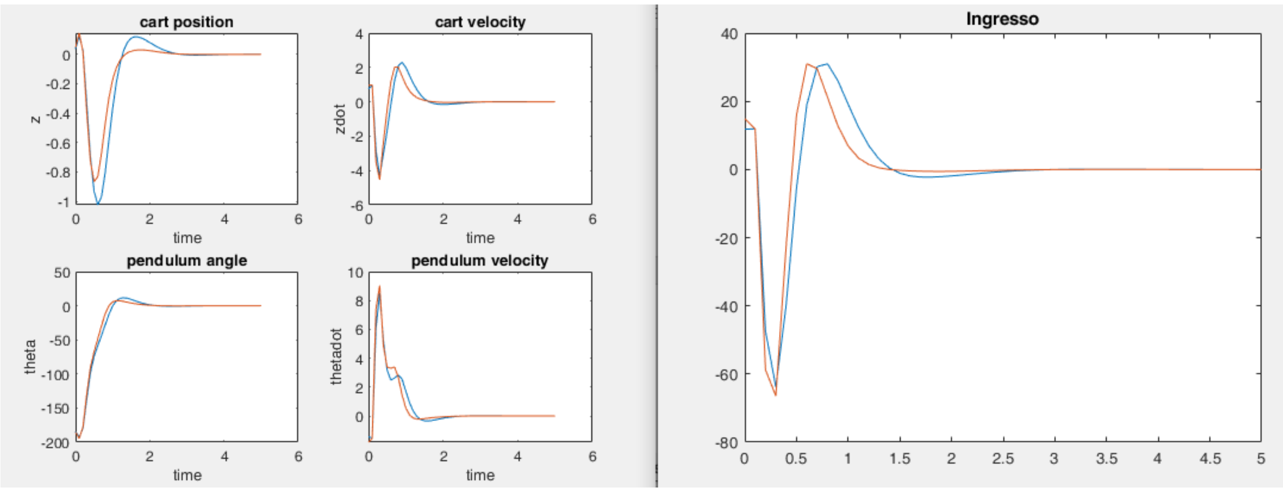

In questo caso il sistema ha una risposta molto migliore, riuscendosi ad assestare in poco più del tempo richiesto.

## Domanda n.8

Ci viene chiesto di implementare la seguente funzione di costo, effettuare una simulazione e spiegare il perché essa aiuti nel perseguire i nostri obiettivi.

$$V = \sum_{i=1}^{H_P} [r(i) - y(i)]^T \begin{bmatrix} 3 & 0 \\ 0 & 3 \end{bmatrix} [r(i) - y(i)] + \sum_{i=1}^{H_P - 1} \Delta u^T 0.1 \Delta u$$

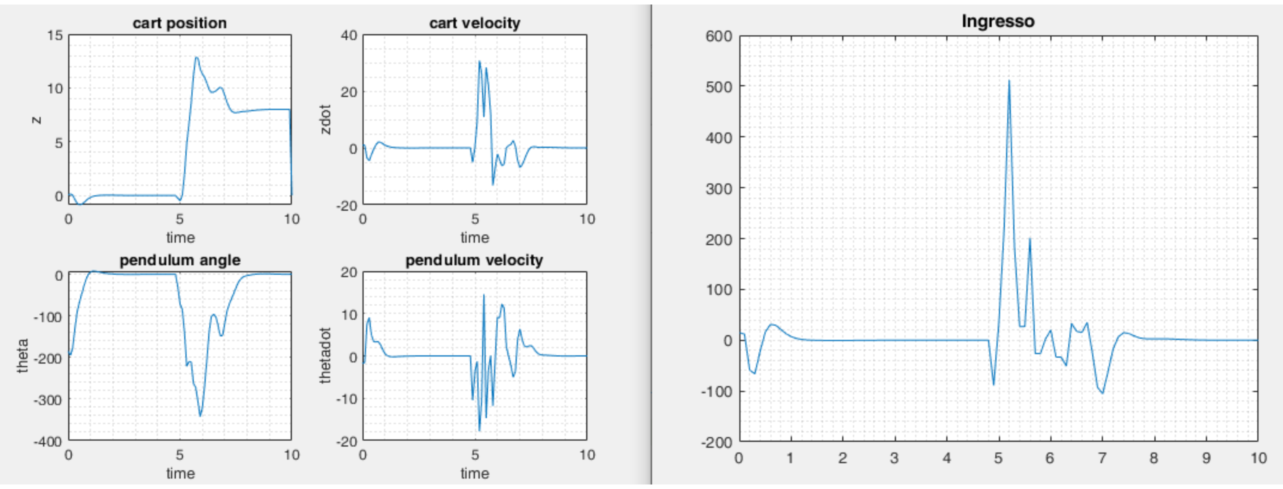
I pesi sulle uscite (3) rendono l'azione di controllo ancora più aggressiva ed equilibrata (stesso peso sulla posizione del cart e l'angolo del pendolo). Il peso sulla variazione dell'ingresso ci aiuta a rendere l'azione di controllo più smooth. N.B. 0.1 è il valore di default per il peso su $\Delta u$. Impostando questi pesi ed effettuando la simulazione con il caso precedente, notiamo come effettivamente l'azione del sistema sia più veloce e con oscillazioni ridotte.



## Domanda n.9

Ci viene chiesto di fare una simulazione di 10s aggiungendo una variazione di riferimento al tempo t=5s.

Otteniamo la seguente situazione:

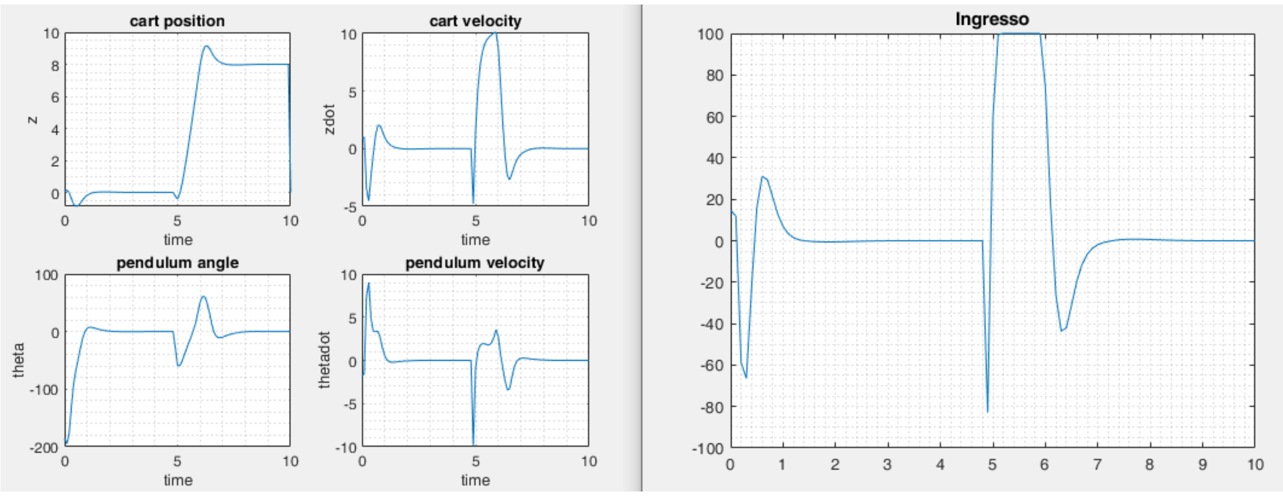



Si nota chiaramente la perturbazione nel sistema che avviene a partire da t=5s. Ci sono notevoli variazioni sull'ingresso. Si noti come il pendolo per stabilizzarsi compia quasi un giro su sé stesso. Il tempo di assestamento è di circa 2 secondi.

## Domanda n.10

Ci viene chiesto di impostare dei vincoli sulla variabile manipolabile e sulla prima uscita. Effettuando una simulazione notiamo chiaramente come entrambi i vincoli entrino in funzione (confrontati con la risposta precedente).

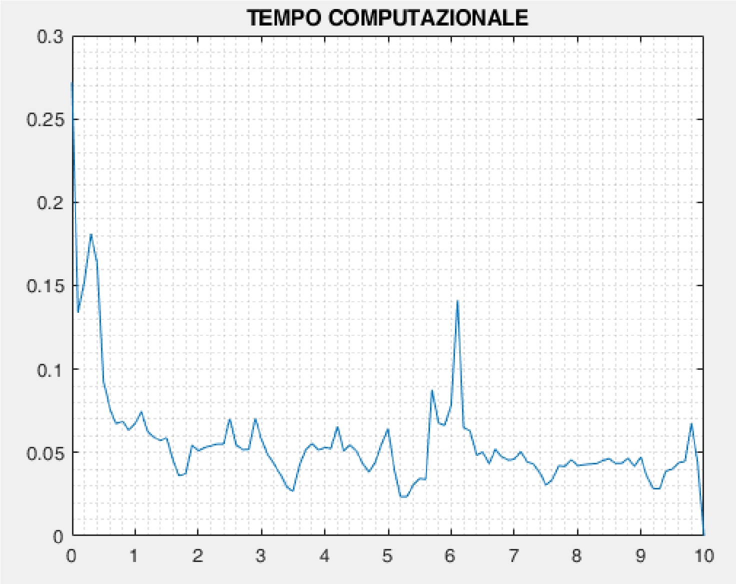
Sull'ingresso l'effetto di questi vincoli è molto più marcato sull'ingresso che satura in corrispondenza del vincolo.



## Domanda n.11

Ci viene chiesto di valutare se il sistema fosse in grado di essere eseguito per controllare il sistema in tempo reale.

Plottando il tempo di calcolo su step otteniamo il seguente grafico:




Ad eccezione di brusche variazioni del riferimento (come durante l'inizializzazione e al sec 5), l'ottimizzatore riesce a fornire la variabile di controllo ottimale in meno di 0.1s. Tuttavia gli spike oltre 0.1s non ci consentono di fare in maniera robusta un controllo in real-time.

Ovviamente la risposta a questa domanda era soggettiva in quanto da PC a PC i tempi variano (un PC più potente ha dei tempi computazionali minori).

## Domanda n.12

**file main.m**

```
clc;
clear all;
close all;

nx = 4;
ny = 2;
nu = 1;
nlobj = nlmpc(nx, ny, nu);

Tf = 10;
Ts = 0.1;
Hp = 10;
Hc = 5;

nlobj.Ts = Ts;
nlobj.PredictionHorizon = Hp;
nlobj.ControlHorizon = Hc;

%% Specify Nonlinear Plant Model
nlobj.Model.StateFcn = "stateFun";
nlobj.Model.OutputFcn = 'outFun';

%% Define Weights and Constraints
% nlobj.Weights.OutputVariables = [2 2];          % INIZIALE
nlobj.Weights.OutputVariables = [3 3];            % domanda 8
nlobj.Weights.ManipulatedVariablesRate = 0.1;    % domanda 8

nlobj.OutputVariables(1).Min = -10;     % domanda 10
nlobj.OutputVariables(1).Max = 10;      % domanda 10

nlobj.ManipulatedVariables.Min = -100;  % domanda 10
nlobj.ManipulatedVariables.Max = 100;   % domanda 10

x0 = [0;0;-pi;0];
u0 = 0;
validateFcns(nlobj,x0,u0);

x = [0;0;-pi;0];
y = [x(1);x(3)];

mv = u0;

yref1 = [0 0];
yref2 = [8 0]; % Domanda 9

xk = x0;

hbar = waitbar(0,'Simulation Progress');
% History vectors
T = (Tf/Ts)+1;
ref_history = zeros(T,ny);
x_history = zeros(T,nx);
u_history = zeros(T,nu);
y_history = zeros(T,ny);
J_history = zeros(T,1);
cTime_history = zeros(T,1);
ref_history(1,:) = y;

tic;
for k = 1:(Tf/Ts)
    % Domanda 9
    if k*Ts<5
```

```
        yref = yref1;
    else
        yref = yref2;
    end
    ref_history(k,:) = yref;

    % Compute optimal control moves.
    [mv,nloptions,info] = nlmpcmove(nlobj,xk,mv,yref);
    xk = info.Xopt(2,:);
    y = [info.Yopt(1,:)];

    x_history(k,:) = xk;
    y_history(k,:) = y;
    J_history(k,:) = info.Cost;
    u_history(k,:) = mv';

    waitbar(k*Ts/20,hbar);
    cTime_history(k,:) = toc; % Domanda 11
    tic;
end
close(hbar)
disp("Tempo di esecuzione: ");
disp(sum(cTime_history));

x_history = x_history';

% Plot the closed-loop response.
titles = ["cart position", "cart velocity", "pendulum angle", "pendulum velocity"];
ylabels = ["z", "zdot", "theta", "thetadot"];
figure(1)
for i=1:4
    subplot(2,2,i)
    plot(0:Ts:Tf,x_history(i,:))
    xlabel('time')
    ylabel(ylabels(i))
    title(titles(i))
    grid minor
    hold on
end

figure(2)
plot(0:Ts:Tf, u_history)
title("Ingresso")
grid minor
hold on

figure(3)
plot(0:Ts:Tf,cTime_history);
grid minor
title("TEMPO COMPUTAZIONALE")

figure(2)
plot(0:Ts:Tf, u_history)
title("Ingresso")
grid minor
hold on

figure(3)
plot(0:Ts:Tf,cTime_history);
grid minor
title("TEMPO COMPUTAZIONALE")
```

**file stateFun.m**

```
function dxdt = stateFun(x,u)
    %% parameters
    M = 1;  % cart mass
    m = 1;  % pendulum mass
    g = 9.81;   % gravity of earth
    L = 0.5;    % pendulum length
    Kd = 10;    % cart damping

    %% Compute dxdt
    dxdt = zeros(4,1);
    dxdt(1) = x(2);
    dxdt(2) = (u - Kd*x(2) - m*L*x(4)^2*sin(x(3)) + m*g*sin(x(3))*cos(x(3))) / (M +
m*sin(x(3))^2);
    dxdt(3) = x(4);
    dxdt(4) = ((u - Kd*x(2) - m*L*x(4)^2*sin(x(3)))*cos(x(3))/(M + m) + g*sin(x(3))) / (L -
m*L*cos(x(3))^2/(M + m));
end
```

**file outFun.m**

```
function y = outFun(x,~)
    y = [x(1); x(3)];
end
```