



UNIVERSITÀ DEGLI STUDI DI PADOVA
DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

TESI DI LAUREA TRIENNALE

**ANALISI E SINTESI DI UN
CONTROLLORE PER ROBOT
AUTOBILANCIANTI**

RELATORE: Prof. Luca Schenato

Autore: Luca Silvestri

Padova, 21 febbraio 2013

Indice

Introduzione	4
Il modello di un pendolo inverso	5
Funzione di trasferimento	7
Schema a blocchi	8
Studio del modello su Matlab.....	9
Controllore PID	11
Sintesi di un controllore PID	13
Luogo delle radici.....	16
Analisi in frequenza	21
Simulazioni.....	27
Sistema a catena aperta	28
Controllore PID	29
Luogo delle radici.....	30
Analisi in frequenza	31
Conclusioni	32

Introduzione

I robot autobilanciati sono particolari automi che riescono a stare in posizione verticale, e per farlo hanno bisogno di un certo numero di sensori per rilevare la loro posizione istante per istante oltre ad alcuni motori che possano reagire in modo da ristabilire l'equilibrio.

Si tratta naturalmente di una posizione completamente instabile, un po' come se noi volessimo tenere in posizione verticale una matita sulla punta del dito, e proprio per questo è necessaria un'analisi particolare per creare un controllore efficiente.

Questo genere di macchina è diventato celebre nella forma e nelle sembianze del Segway.



1 Lego NXT in versione Segway



2 Segway

Il Segway è un dispositivo per il trasporto personale inventato da Dean Kamen nel 2001, la cui particolarità è quella di stare in equilibrio su due ruote motorizzate disposte parallelamente.

Questo è possibile grazie ad un insieme di sensori che formano una IMU ("Inertial Measurement Unit"), grazie alla quale l'elettronica conosce istante per istante la posizione relativa del mezzo e ne permette il controllo pilotando adeguatamente i motori elettrici.

Il Segway, infatti, è un perfetto connubio di informatica, elettronica e meccanica, ma nonostante ciò il suo utilizzo risulta estremamente semplice, basta spostare il peso in avanti per farlo procedere e all'indietro per arretrare. Le svolte invece vengono controllate ruotando il manubrio o spostando il peso lateralmente.

Nonostante tutte queste premesse non fu un enorme successo commerciale, soprattutto a causa del costo e della ridotta autonomia, attualmente infatti il Segway pesa 38 kg, e percorre fino a 40 km con una velocità massima di 20 km/h.

Il vero successo dei Segway appare palese però se si osservano le varie comunità dei Makers i quali hanno riprodotto in casa (anche sotto forma di piccoli robot) questo incredibile mezzo di trasporto utilizzando anche materiali di scarto coniugati con elettronica a basso costo.

Perciò se è vero ciò che diceva Henry Ford, "*C'è vero progresso solo quando i vantaggi di una nuova tecnologia diventano per tutti*", si può affermare che ciò che vedremo non sarà nient'altro che espressione del *progresso*.



3 Segway DIY

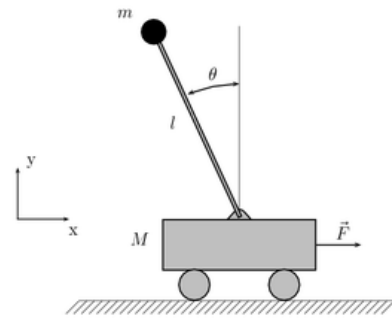
Il modello di un pendolo inverso

Per modellare il sistema lo tratteremo innanzitutto come un sistema SISO (Single Input Single Output) dove l'ingresso è l'angolo del robot con la verticale e l'uscita la forza da esercitare con i motori.

Trascureremo di fatto la posizione del carrello che renderebbe la modellizzazione più complicata e che esula dallo scopo di questo studio.

Per descrivere il modello e per costruire un controllore per il robot useremo questi dati ipotetici:

- Massa del carrello: $M = 0,5 \text{ Kg}$
- Massa del pendolo: $m = 0,2 \text{ Kg}$
- Coefficiente d'attrito del carrello: $b = 0,1 \text{ N/m/s}$
- Lunghezza del pendolo dal centro di massa: $l = 0,3 \text{ m}$
- Inerzia del pendolo: $I = 0,006 \text{ Kg}\cdot\text{m}^2$
- Forza applicata al carrello: F
- Posizione del carrello: x
- Angolo del pendolo: θ



4 Schema modello

Il nostro obiettivo progettuale sarà quello di fare in modo che il robot torni in posizione verticale dopo un disturbo con un tempo inferiore ai 5 secondi e in modo che l'angolo che forma con la verticale non superi mai i 0,05 radianti.

Per descrivere il sistema di controllo di un robot autobilanciante dobbiamo innanzitutto delineare un modello, e notiamo che si tratta sostanzialmente di un pendolo inverso.

In questo modello abbiamo un carrello sul quale è incernierata un'asta libera di muoversi, e il nostro scopo è di far scorrere il carrello affinché l'asta rimanga in equilibrio verticale.

Nella parte inferiore del modello otteniamo questa equazione differenziale

$$M\ddot{x} + b\dot{x} + N = F$$

Le forze che dobbiamo considerare nella parte superiore invece sono quelle orizzontali, per cui troviamo che

$$N = m\ddot{x} + ml\ddot{\theta} \cos \theta - ml\dot{\theta}^2 \sin \theta$$

Sostituendo quest'ultima equazione nella prima otteniamo la prima equazione di moto del sistema

$$(M + m)\ddot{x} + b\dot{x} + ml\ddot{\theta} \cos \theta - ml\dot{\theta}^2 \sin \theta = F$$

Per la seconda equazione consideriamo le forze perpendicolari al pendolo ed otteniamo quindi

$$P \sin \theta + N \cos \theta - mg \sin \theta = ml\ddot{\theta} + m\ddot{x} \cos \theta$$

e sapendo che

$$-Pl \sin \theta - Nl \cos \theta = I\ddot{\theta}$$

otteniamo la seconda equazione del moto

$$(I + ml^2)\ddot{\theta} + mgl \sin \theta = -ml\ddot{x} \cos \theta$$

Per una trattazione più semplice dobbiamo linearizzare le equazioni supponendo che θ sia un angolo molto piccolo, quindi $\sin \theta = \theta$ e $\cos \theta = 1$ e le equazioni

$$\begin{cases} (M + m)\ddot{x} + b\dot{x} + ml\ddot{\theta} \cos \theta - ml\dot{\theta}^2 \sin \theta = F \\ (I + ml^2)\ddot{\theta} + mgl \sin \theta = -ml\ddot{x} \cos \theta \end{cases}$$

diventano

$$\begin{cases} (I + ml^2)\ddot{\varphi} - mgl\varphi = ml\ddot{x} \\ (M + m)\ddot{x} + b\dot{x} - ml\ddot{\varphi} = u \end{cases}$$

Funzione di trasferimento

La funzione di trasferimento del sistema linearizzato viene trovata calcolando le trasformate di Laplace delle due equazioni, dove otteniamo

$$\begin{cases} (I + ml^2)\Phi(s)s^2 - mgl\Phi(s) = mlX(s)s^2 \\ (M + m)X(s)s^2 + bX(s)s - ml\Phi(s)s^2 = U(s) \end{cases}$$

Esplicitiamo $X(s)$ dalla prima equazione

$$X(s) = \left[\frac{(I + ml^2)}{ml} - \frac{g}{s^2} \right] \Phi(s)$$

e lo sostituiamo nella seconda fino ad ottenere

$$\frac{\Phi(s)}{U(s)} = \frac{\frac{ml}{q}s^2}{s^4 + \frac{b(I + ml^2)}{q}s^3 - \frac{(M + m)mgl}{q}s^2 - \frac{bmgl}{q}s}$$

che è la nostra funzione di trasferimento dove

$$q = [(M + m)(I + ml^2) - (ml)^2]$$

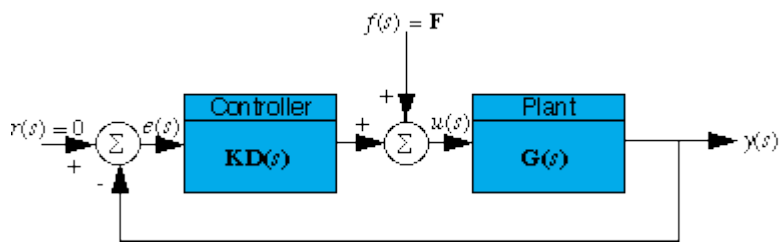
e dato che ci sono un polo e uno zero nell'origine li possiamo elidere ottenendo

$$\frac{\Phi(s)}{U(s)} = \frac{\frac{ml}{q}s}{s^3 + \frac{b(I + ml^2)}{q}s^2 - \frac{(M + m)mgl}{q}s - \frac{bmgl}{q}}$$

Schema a blocchi

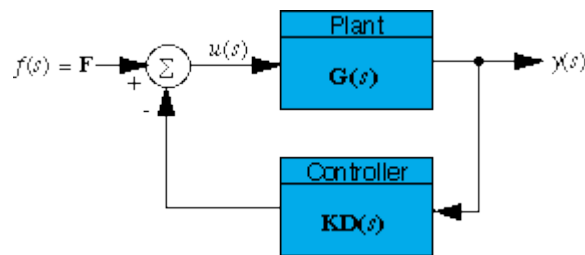
Nel controllare la posizione del pendolo dovremo fare in modo che, dopo un eventuale disturbo impulsivo, l'asta torni verticale e dato che il disturbo lo potrà spostare in due sensi opposti dobbiamo ottenere che il segnale di riferimento sia a zero.

Lo schema a blocchi di questo problema è rappresentabile in questo modo



5 Schema a blocchi

che per comodità possiamo riorganizzare così



6 Schema equivalente

Da questo schema risulta che la funzione di trasferimento del sistema a catena chiusa sarà

$$Y(s) = \frac{G(s)}{1 + G(s) \cdot K \cdot D(s)} \cdot F(s)$$

dove $G(s)$ è la funzione di trasferimento del sistema a catena aperta, K il guadagno del controllore e $D(s)$ la funzione di trasferimento del controllore.

Studio del modello su Matlab

Per calcolare la funzione di trasferimento del sistema a catena aperta su Matlab usiamo il seguente codice

```
M = 0.5;
m = 0.2;
b = 0.1;
i = 0.006;
g = 9.8;
l = 0.3;

q = (M+m)*(i+m*l^2)-(m*l)^2; % semplifica l'input

num = [m*l/q 0];
den = [1 b*(i+m*l^2)/q -(M+m)*m*g*l/q -b*m*g*l/q];

pend=tf(num,den) % calcola la funzione di trasferimento
```

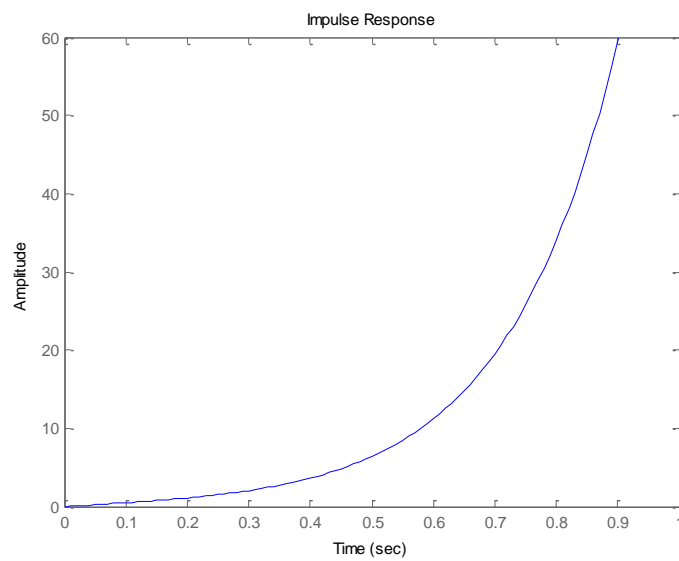
Il quale darà come risultato

```
Transfer function:
          4.545 s
-----
s^3 + 0.1818 s^2 - 31.18 s - 4.455
```

A questo punto possiamo vedere la risposta impulsiva del sistema scrivendo

```
t=0:0.01:5;
impz(pend,t) % risposta impulsiva
axis([0 1 0 60])
```

e otteniamo il seguente grafico



7 Risposta impulsiva

Come era ragionevole pensare è evidente che il sistema ad anello aperto non è stabile e richiede quindi lo studio di un controllore.

Ci concentreremo ora su diversi metodi per trovare la funzione di trasferimento e il guadagno del controllore quali il controllo PID, il luogo delle radici e l'analisi in frequenza.

Controllore PID

Il controllore PID (Proporzionale, Integrale, Derivativo) è un sistema in retroazione negativa che valutando un segnale di ingresso e uno di riferimento permette di agire in modo preciso e di ottimizzare la risposta del regolatore in base alle tre azioni sopra citate.

La funzione di trasferimento di questo controllore è

$$C_{PID}(s) = K_p + K_d s + \frac{K_i}{s} = \frac{K_d s^2 + K_p s + K_i}{s}$$

e la nostra sintesi consiste nel trovare la tripletta (K_p, K_d, K_i) che soddisfa i requisiti progettuali.

Consideriamo la funzione di trasferimento del sistema in forma estesa

$$\frac{\Phi(s)}{U(s)} = \frac{m l s}{[(M + m)(I + m l^2) - (m l)^2] s^3 + b(I + m l^2) s^2 - (M + m) m g l s - b m g l}$$

e applichiamo il criterio di Routh al denominatore:

3	$(M + m)(I + m l^2) - (m l)^2$	$-(M + m) m g l$
2	$b(I + m l^2)$	$-b m g l$
1	$-\frac{m^3 l^3 g b}{b(I + m l^2)}$	
0	$-b m g l$	

Possiamo vedere che abbiamo una variazione e due permanenze di segno, perciò il polinomio ha due radici a parte reale negativa e una radice reale positiva.

La nostra funzione di trasferimento ha uno zero nell'origine, due poli a parte reale negativa ed un polo reale positivo. Di conseguenza, come era semplice intuire, non è BIBO stabile.

A questo punto riscriviamo la funzione di trasferimento nella sua corrispettiva forma di Bode, ottenendo:

$$G(s) = K_B(G) \frac{s}{(1 + sT_1)(1 + sT_2)(1 - sT_3)}$$

con T_1, T_2, T_3 positivi reali e guadagno negativo.

Se riscriviamo il controllore PID nella forma

$$C_{PID}(s) = K_i \frac{\frac{K_d}{K_i} s^2 + \frac{K_p}{K_i} s + 1}{s}$$

e assumiamo che

$$\frac{K_d}{K_i} s^2 + \frac{K_p}{K_i} s + 1 = (1 + sT_1)(1 + sT_2)$$

otteniamo come funzione di trasferimento in catena aperta

$$\tilde{G}(s) := C_{PID}(s)G(s) = K_B(G)K_i \frac{1}{1 - sT_3}.$$

la quale ha un solo polo reale positivo pari a 1 se assumiamo $K_i > 0$ e tale che $|K_B(G)K_i| > 1$, per cui il diagramma di Nyquist di $\tilde{G}(j\omega)$, $\omega \in \mathfrak{R}$, è una curva chiusa che compie un giro attorno al punto $-1+j0$.

La funzione di trasferimento $W(s)$ ottenuta per retroazione unitaria negativa da $\tilde{G}(s)$ è certamente BIBO stabile.

Infatti si trova

$$W(s) = \frac{\tilde{G}(s)}{1 + \tilde{G}(s)} = \frac{K_B(G)K_i}{(1 + K_B(G)K_i)(1 + s \frac{-T_3}{1 + K_B(G)K_i})}$$

con

$$\frac{-T_3}{1 + K_B(G)K_i} > 0.$$

Osserviamo che scegliendo $K_i > 0$ e tale che $|K_B(G)K_i| \gg 1$ assicuriamo che il sistema retroazionato (di tipo 0), presenti comunque un errore di regime permanente al gradino molto piccolo, tanto più vicino a zero quanto più grande è $|K_B(G)K_i|$.

Inoltre l'unico polo reale negativo di $W(s)$ diventa estremamente grande, assicurando una grande banda passante e quindi piccoli tempi di salita nella risposta al gradino.

Sintesi di un controllore PID

Costruiamo un controllore PID con $K_p = K_i = K_d = 1$ e vediamo la risposta del sistema all'impulso con il seguente codice:

```
M = 0.5;
m = 0.2;
b = 0.1;
i = 0.006;
g = 9.8;
l = 0.3;

q = (M+m)*(i+m*l^2)-(m*l)^2; % semplifica l'input

num = [m*l/q 0];
den = [1 b*(i+m*l^2)/q -(M+m)*m*g*l/q -b*m*g*l/q];

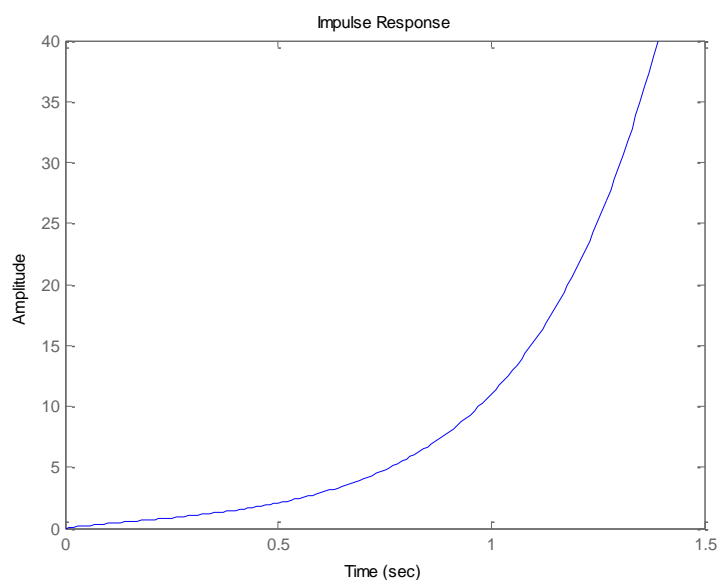
pend=tf(num,den) % calcola la funzione di trasferimento

Kd = 1;
Kp = 1;
Ki = 1;

contr=tf([Kd Kp Ki],[1 0]); % FDT del controllore

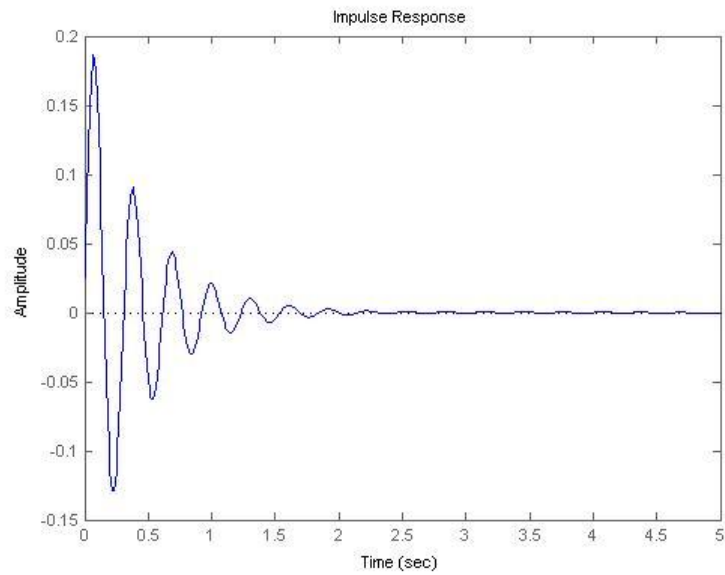
sys_cl=feedback(pend,contr); % controllore in retroazione
t=0:0.01:5;
impulse(sys_cl,t)
axis([0 1.5 0 40])
```

Dal quale otteniamo la risposta impulsiva:



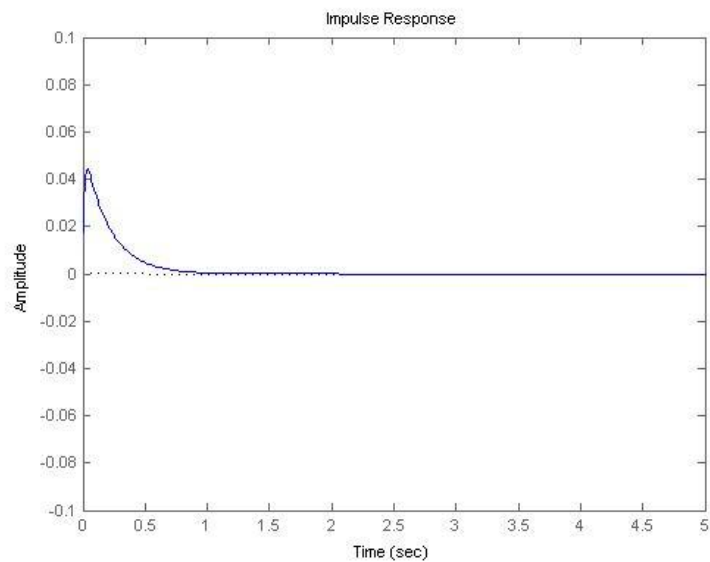
8 Risposta impulsiva PID 1,1,1

Notiamo che il sistema non è assolutamente stabile e perciò proviamo ad aumentare K_p fino al valore 100 ottenendo:



9 Risposta impulsiva PID 100,1,1

Ora il nostro sistema diventa stabile in meno di 3 secondi ma presenta delle forti oscillazioni che proviamo a rimediare ponendo $K_d = 20$



10 Risposta impulsiva PID 100,1,20

Possiamo notare che ora il transitorio è decisamente inferiore e compatibile con un sistema fisico reale.

Le specifiche del progetto sono state soddisfatte, perciò il controllore è sufficiente nonostante non si sia rivelato necessario l'aumento di K_i .

Il controllore che useremo sarà dunque

$$C_{PID} = \frac{20s^2 + 100s + 1}{s}$$

Luogo delle radici

Il metodo del luogo delle radici è un procedimento grafico che permette di studiare lo spostamento dei poli di una funzione di trasferimento ad anello chiuso al variare di un parametro reale K.

Usando la funzione di Matlab *rlocus* otteniamo il luogo delle radici del sistema aperto:

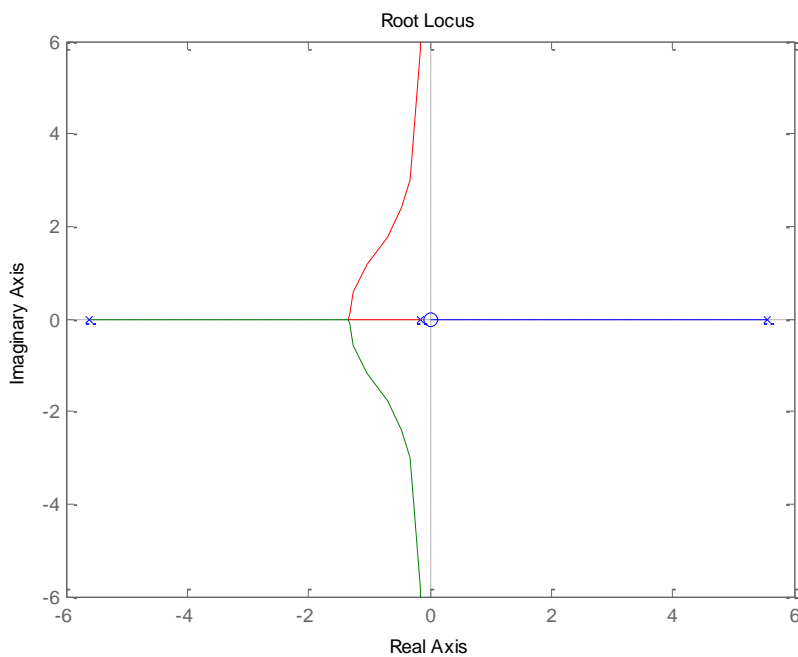
```
M = 0.5;
m = 0.2;
b = 0.1;
i = 0.006;
g = 9.8;
l = 0.3;

q = (M+m) * (i+m*l^2) - (m*l)^2; % semplifica l'input

num = [m*l/q 0];
den = [1 b*(i+m*l^2)/q - (M+m)*m*g*l/q -b*m*g*l/q];

pend=tf(num,den) % calcola la funzione di trasferimento

rlocus(pend)
siggrid(0.92)
axis([-6 6 -6 6])
```



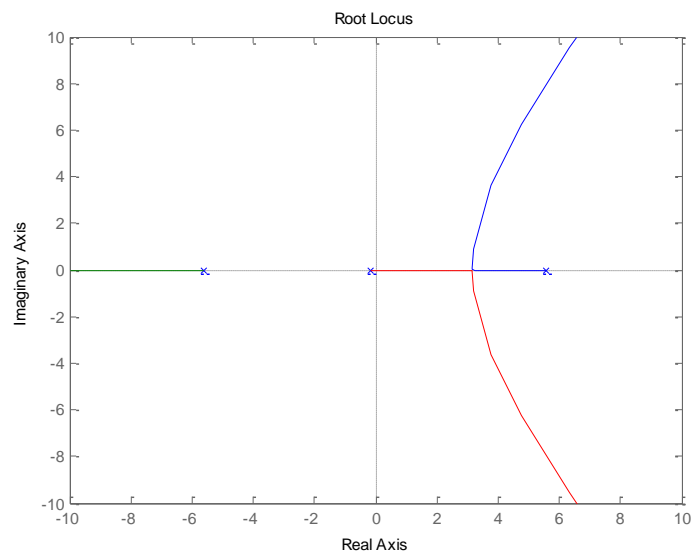
11 Luogo delle radici

Come si vede dal grafico, una delle radici si trova nel semipiano positivo delle ascisse rendendo il sistema instabile.

Per risolvere questo problema dobbiamo innanzitutto eliminare lo zero nell'origine, altrimenti avremo sempre un polo che punta a questo zero nel semipiano destro o ad infinito.

Una volta aggiunto un termine $1/s$ al controllore otterremo questo luogo:

```
contr=tf(1,[1 0]);  
rlocus(contr*pend)  
sigrid(0.92)  
axis([-10 10 -10 10])
```



12 Luogo delle radici con integratore

A questo punto è possibile rendere il sistema stabile aggiungendo un polo a sinistra il più lontano possibile e due zeri relativamente piccoli, in modo da portare il polo a parte reale positiva nel semipiano stabile ed avere un numero di asintoti opportuno.

Usiamo quindi il seguente codice:

```
M = 0.5;
m = 0.2;
b = 0.1;
i = 0.006;
g = 9.8;
l = 0.3;

q = (M+m)*(i+m*l^2)-(m*l)^2; % semplifica l'input

num = [m*l/q 0];
den = [1 b*(i+m*l^2)/q -(M+m)*m*g*l/q -b*m*g*l/q];

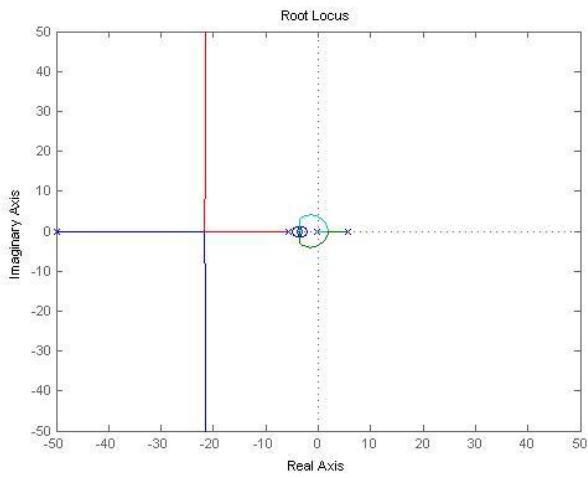
pend=tf(num,den) % calcola la funzione di trasferimento

% zeri e poli
z1 = 3;
p1 = 0;
z2 = 4;
p2 = 50;
numlag = [1 z1];
denlag = [1 p1];
numlead = [1 z2];
denlead = [1 p2];
numc = conv(numlead, numlag); % num controllore
denc = conv(denlead, denlag); % den controllore
contr=tf(numc,denc); % FDT controllore

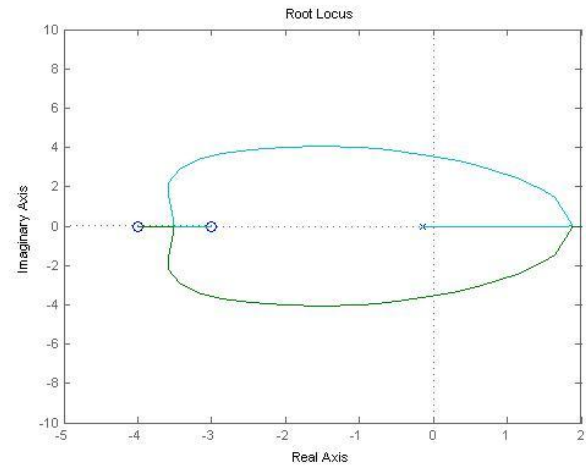
rlocus(contr*pend)
sigrid(0.92)
axis([-50 50 -50 50])
figure
rlocus(contr*pend)
sigrid(0.92)
axis([-10 10 -10 10])
[k,poles]=rlocfind(contr*pend)
sys_cl=feedback(pend,k*contr);

figure
impulse(sys_cl)
axis([0 2 -0.05 0.05])
```

Ci mostrerà il luogo e un suo ingrandimento nella zona dell'origine:



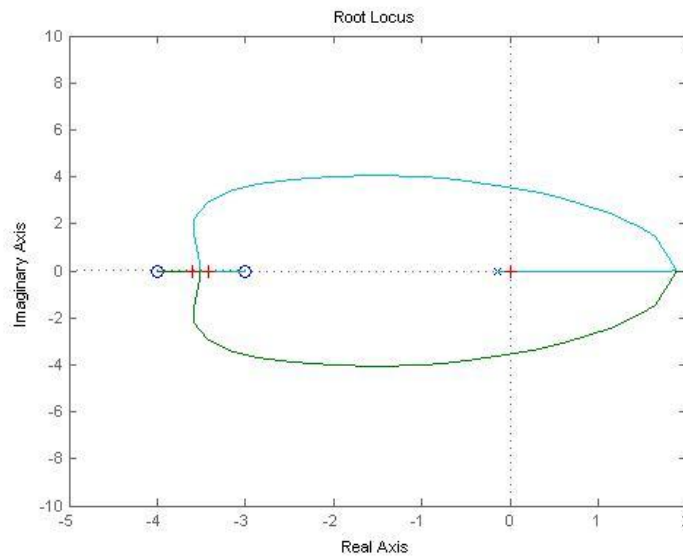
13 Luogo con compensatore



14 Dettaglio dell'origine

Ora si tratta di scegliere dove posizionare il polo in modo tale da rispettare i requisiti del progetto.

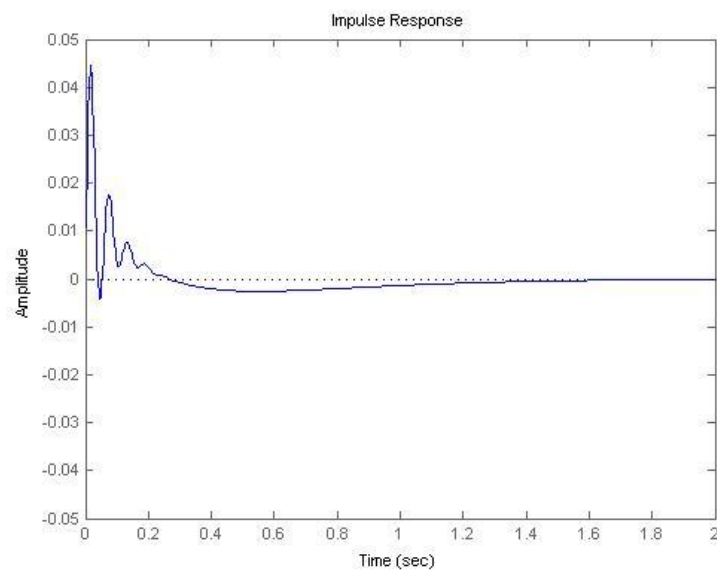
Se ad esempio posizioniamo in questo modo i poli



15 Scelta dei poli

otteniamo il seguente valore di K e i seguenti poli:

```
k =  
  
2.7027e+003  
  
poles =  
  
1.0e+002 *  
  
0  
  
-0.2158 + 1.0715i  
  
-0.2158 - 1.0715i  
  
-0.0360  
  
-0.0342
```



16 Risposta impulsiva sistema controllato

e la risposta impulsiva soddisfa le nostre specifiche.

Il controllore che useremo sarà dunque

$$C_{RL} = K \frac{(s + 3)(s + 4)}{s(s + 50)} = K \frac{s^2 + 7s + 12}{s^2 + 50s}$$

Analisi in frequenza

L'analisi in frequenza si avvale di strumenti quali il diagramma di Bode e quello di Nyquist per la sintesi del controllore.

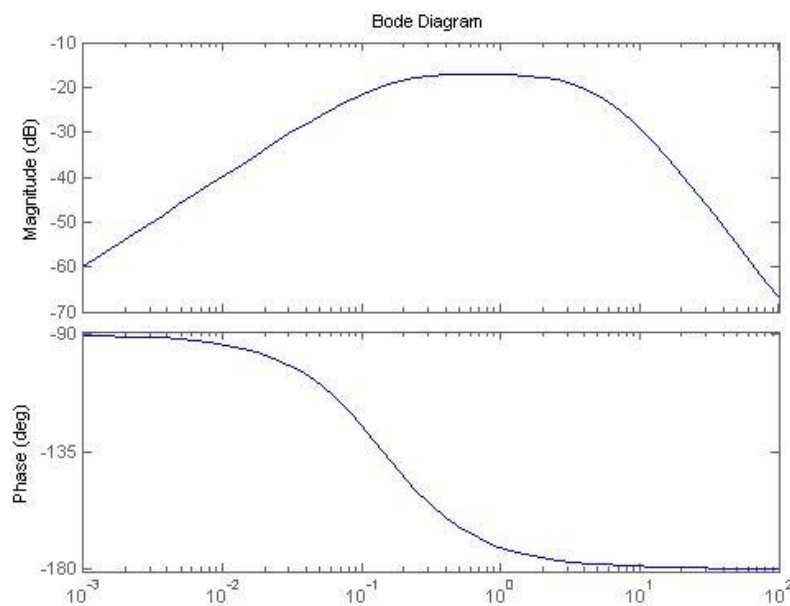
Le specifiche del progetto, cioè la prontezza del sistema e la capacità smorzante, sono riferite al dominio del tempo, ma quando passiamo a quello della frequenza i nostri requisiti diventano rispettivamente avere una banda passante elevata ed un picco di risonanza abbastanza piccolo.

Il diagramma di Nyquist di $W(s)$, cioè del sistema in retroazione unitaria, sarà particolarmente d'aiuto in questo compito grazie al suo criterio

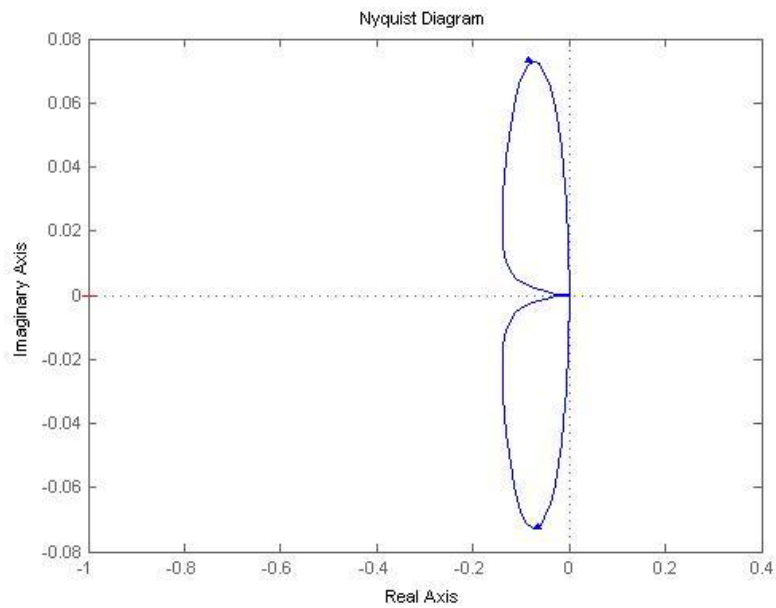
$$Z = P - N$$

dove Z è il numero di zeri instabili di $1+W(s)$, P è il numero di poli instabili di $1+W(s)$ e N è il numero di giri in senso antiorario che il diagramma di Nyquist fa attorno al punto critico -1 .

Sappiamo dall'analisi precedente che il sistema non è stabile in catena aperta, e ciò ovviamente è confermato anche dall'analisi in frequenza, infatti usando questo `bode(pend)` e `nyquist(pend)` otteniamo i seguenti grafici:



17 Diagramma di Bode del sistema



18 Diagramma di Nyquist del sistema

Dal diagramma di Nyquist notiamo che il grafico non compie un giro in senso antiorario attorno a -1, causandone l'instabilità.

Il nostro controllore dovrà quindi modificare il grafico per rendere il sistema stabile rispettando le specifiche del progetto. Per far ciò useremo il seguente codice:

```
M = 0.5;
m = 0.2;
b = 0.1;
i = 0.006;
g = 9.8;
l = 0.3;

q = (M+m)*(i+m*l^2)-(m*l)^2; % semplifica l'input

num = [m*l/q 0 0];
den = [1 b*(i+m*l^2)/q-(M+m)*m*g*l/q-b*m*g*l/q 0];

pendulum = tf(num,den);
figure(1)

% chiede i dati all'utente
numc = input('Numeratore controllore: ');
denc = input('Denominatore controllore: ');
k = input('Guadagno K: ');

% Diagramma di bode del sistema controllato
contr = k*tf(numc,denc);
loop = contr*pendulum;
bode(loop)
```

```

% diagramma di nyquist del sistema controllato
figure(2)
subplot (2,1,1)
nyquist(loop)

% risposta impulsiva sistema controllato
subplot(2,1,2)
sys_cl = feedback(pendulum,contr);
impulse(sys_cl)

```

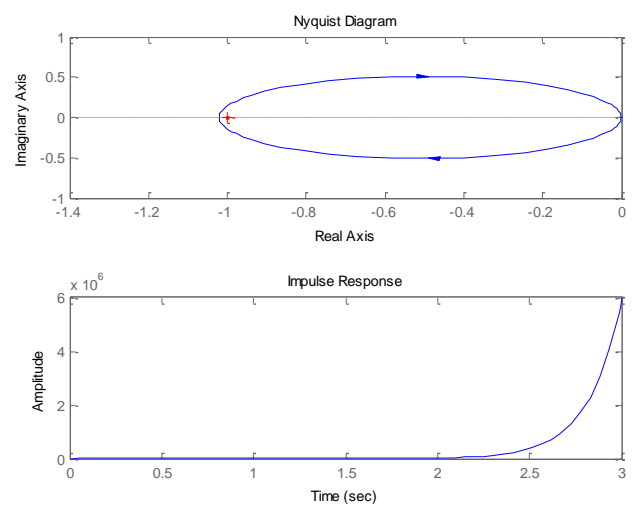
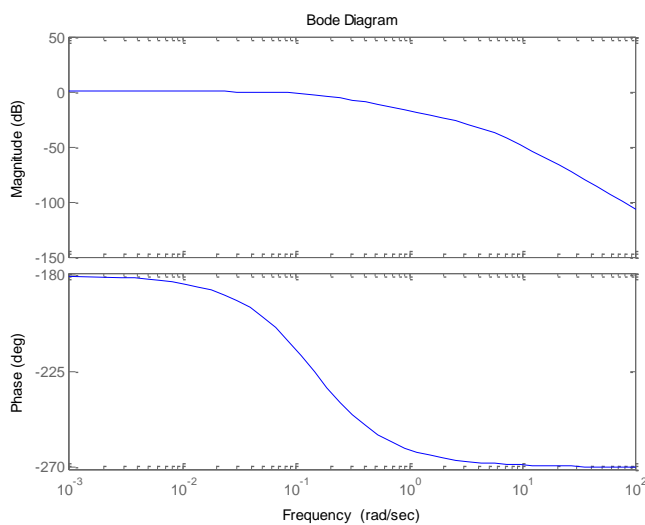
Ora dobbiamo inserire un integratore per eliminare lo zero che si trova nell'origine, quindi digitiamo nella console:

```

Numeratore controllore: 1
Denominatore controllore: [1 0]
Guadagno K: 1

```

da cui otteniamo

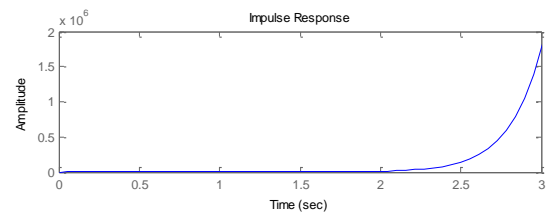
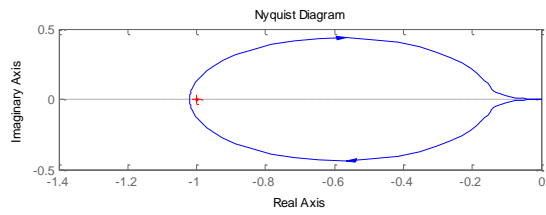
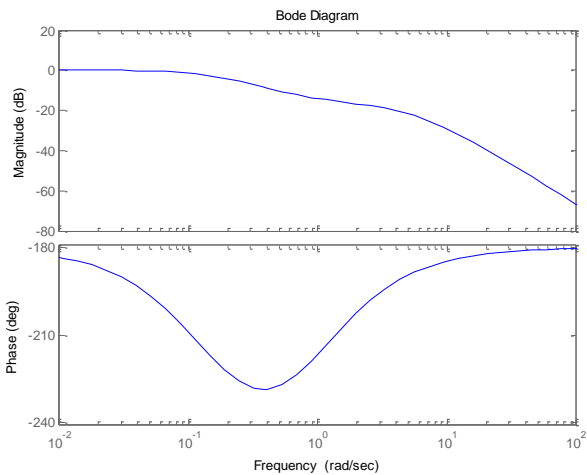


Ora notiamo che abbiamo circondato il punto -1 ma in senso opposto. Dobbiamo aggiungere fase mettendo uno zero al controllore in -1

```

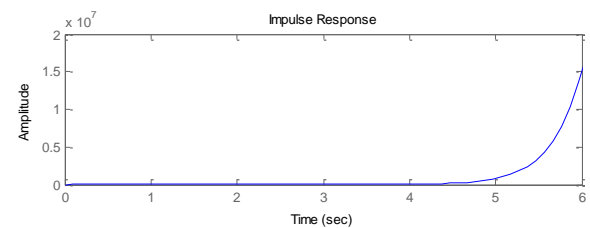
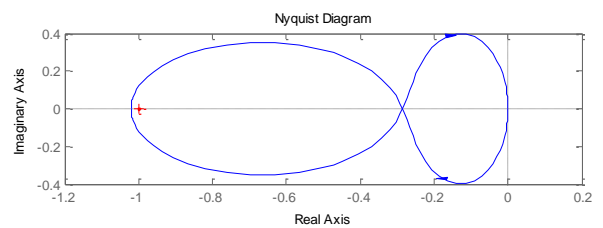
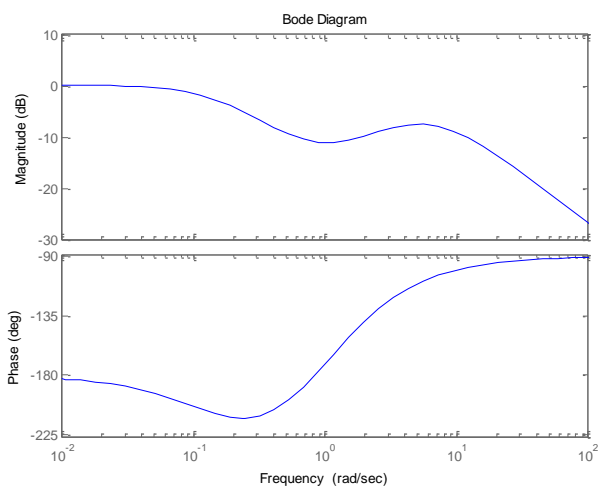
Numeratore controllore: [1 1]
Denominatore controllore: [1 0]
Guadagno K: 1

```



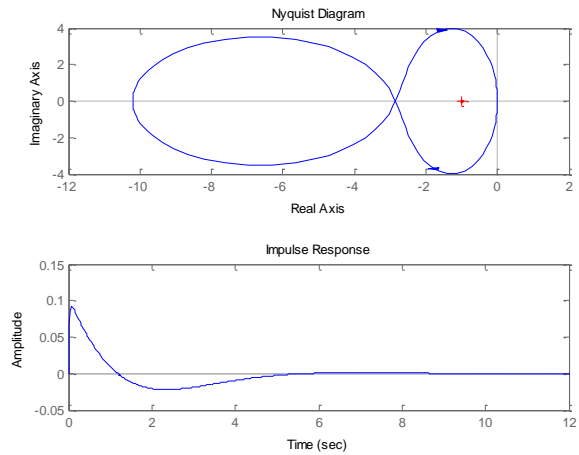
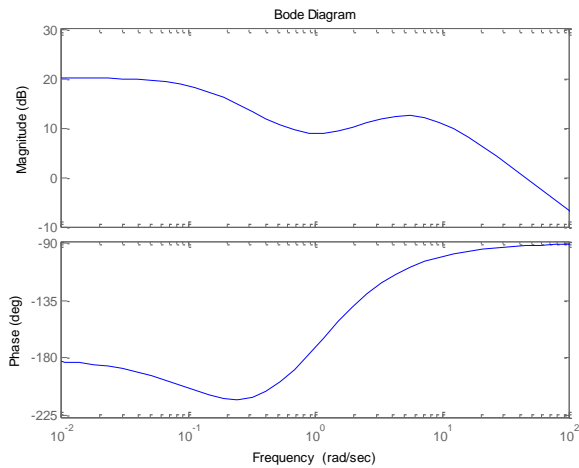
La fase che abbiamo aggiunto non è ancora sufficiente, quindi mettiamo un altro zero sempre in -1

Numeratore controllore: $\text{conv}([1 \ 1],[1 \ 1])$
 Denominatore controllore: $[1 \ 0]$
 Guadagno K: 1



Si forma dunque un'altra circonferenza che gira in senso antiorario, ma che si trova a destra del punto -1, quindi per traslare il grafico verso sinistra proviamo ad aggiungere un guadagno K=10

Numeratore controllore: $\text{conv}([1 \ 1],[1 \ 1])$
 Denominatore controllore: $[1 \ 0]$
 Guadagno K: 10

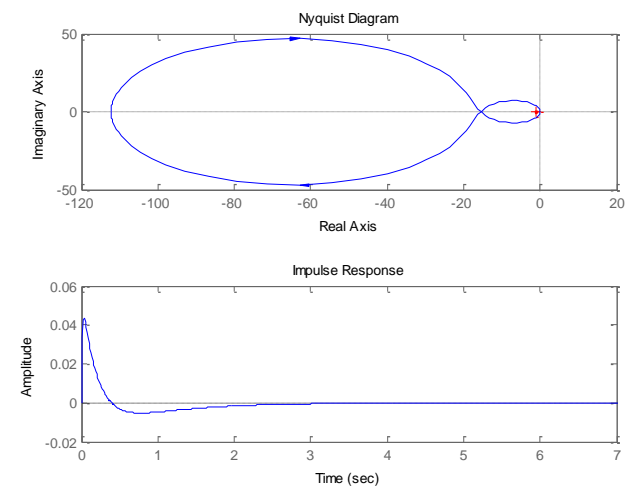
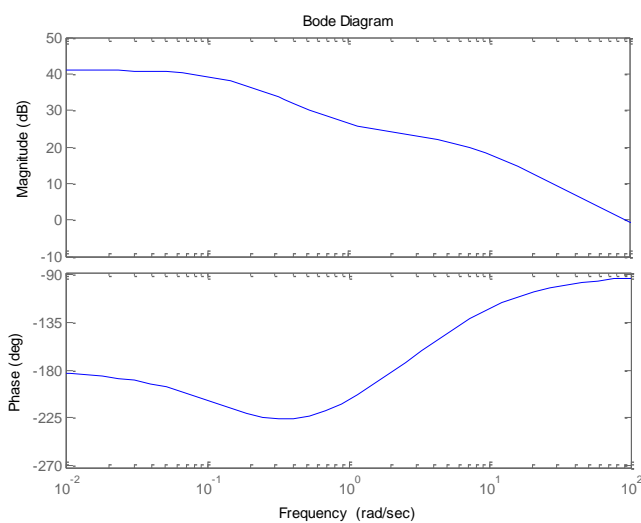


Ora finalmente il sistema è stabile, ma dalla risposta impulsiva notiamo che le specifiche del progetto non sono soddisfatte.

Per migliorare le prestazioni sappiamo che poli vicino all'origine modificheranno la risposta a piccole frequenze, mentre i poli grandi le grandi frequenze.

Un esempio di controllore che soddisfa i requisiti lo otteniamo inserendo

Numeratore controllore: $\text{conv}([1 \ 1.1],[1 \ 5])$
 Denominatore controllore: $[1 \ 0]$
 Guadagno K: 20

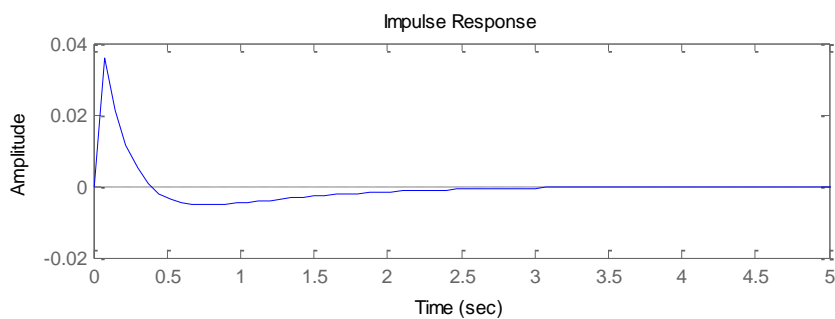
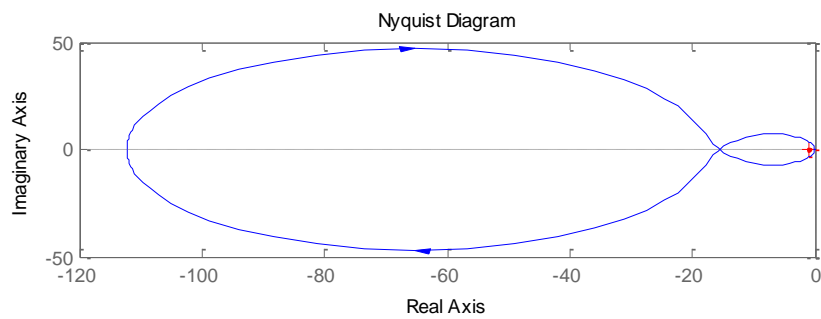
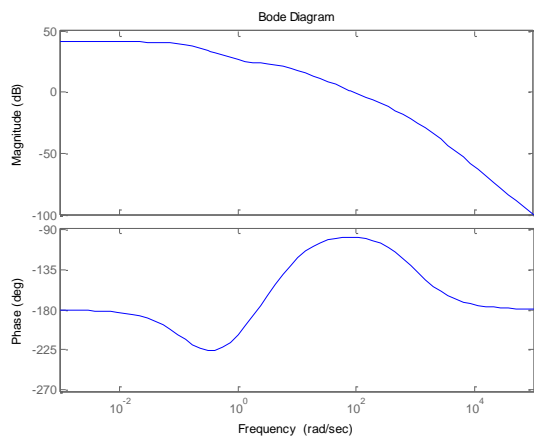


Notiamo però che la funzione di trasferimento è impropria, quindi aggiungiamo un polo ad altissima frequenza che non modificherà di tanto il grafico ma che ci permetterà di avere una corretta funzione di trasferimento.

Numeratore controllore: `conv([1 1.1],[1 5])`

Denominatore controllore: `[0.001 1 0]`

Guadagno K: 20



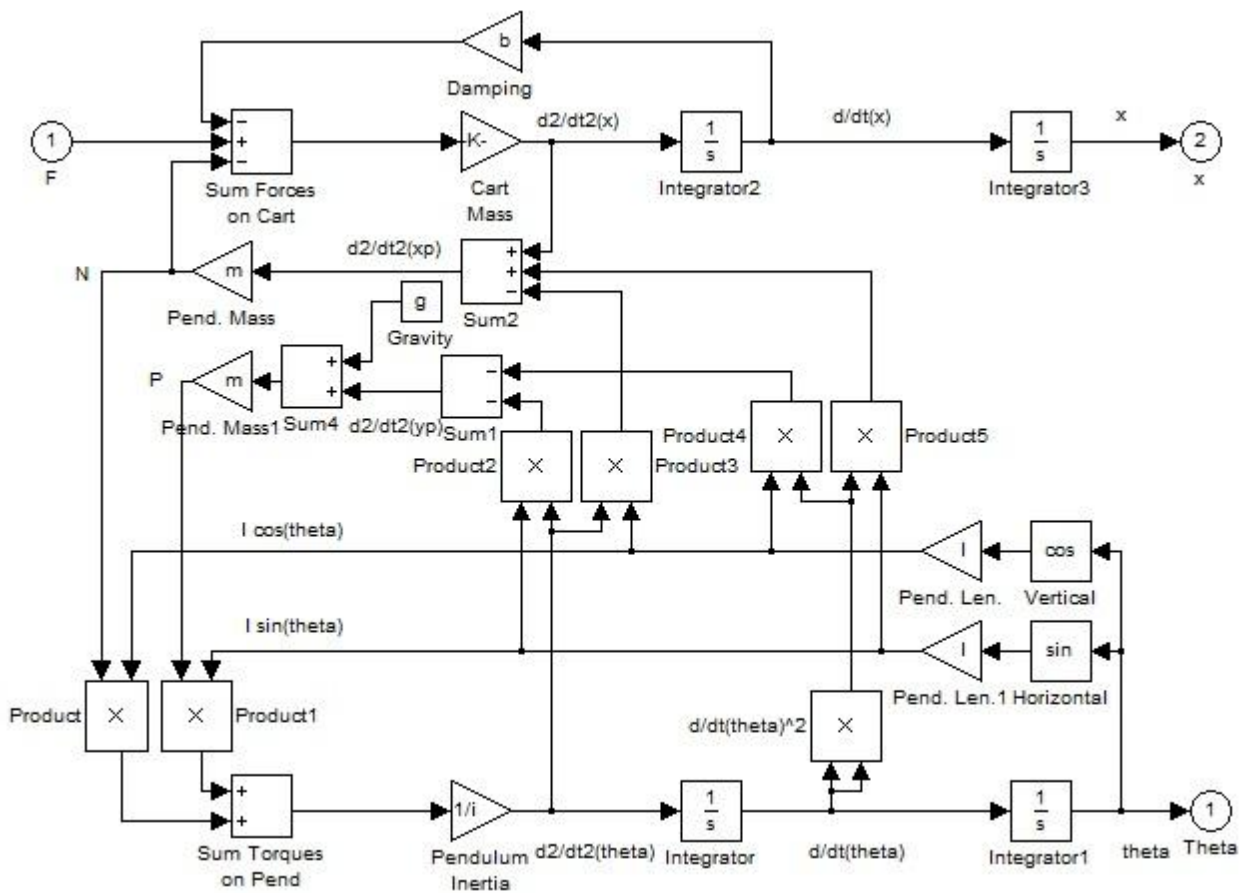
Il controllore che useremo sarà dunque

$$C_{FRQ} = 20 \frac{(s + 1,1)(s + 5)}{s} = \frac{20s^2 + 122s + 110}{0,001s^2 + s}$$

Simulazioni

Matlab mette a disposizione un importante strumento di simulazione chiamato Simulink in grado di lavorare direttamente con i modelli non lineari che ci permetterà di verificare con esattezza se i nostri controllori sono effettivamente in grado di rispettare i requisiti permettendoci anche di stabilire quale si comporta meglio.

Il modello non lineare del pendolo inverso assume questo aspetto su Simulink

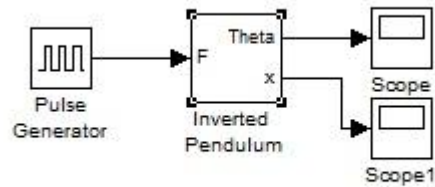


19 Modello in Simulink

e una volta salvato in un unico blocco con un'entrata e due uscite (x e θ) possiamo passare ad implementare i controllori.

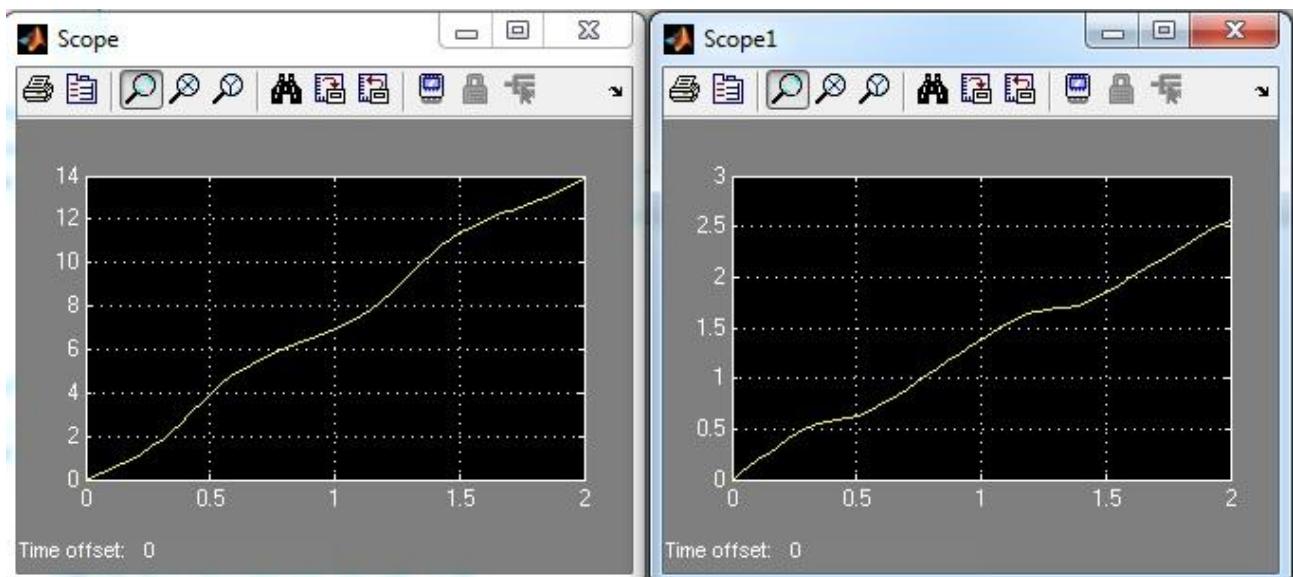
Sistema a catena aperta

Possiamo rappresentare il sistema senza alcuna correzione in questo modo



20 Modello senza correzione

e avviando la simulazione otteniamo questi grafici

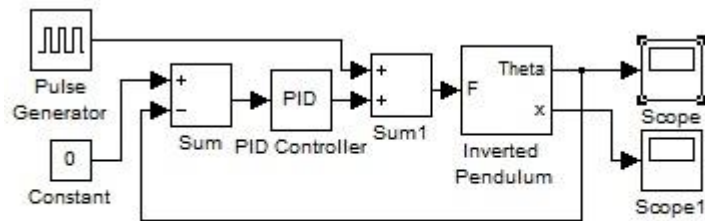


21 Angolo e posizione simulati

Il sistema è evidentemente instabile come già ampiamente discusso.

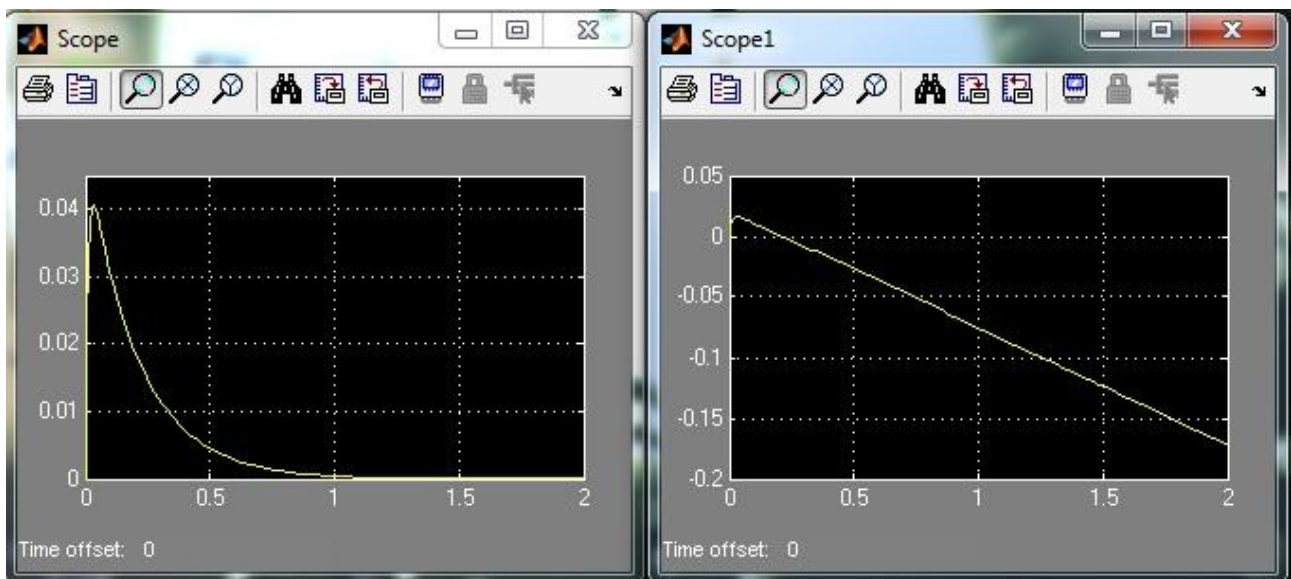
Controllore PID

Simulink offre direttamente un blocco per applicare un controllo PID, quindi possiamo aggiungerlo inserendo i parametri studiati in precedenza



22 Controllore PID

La simulazione restituisce questi risultati

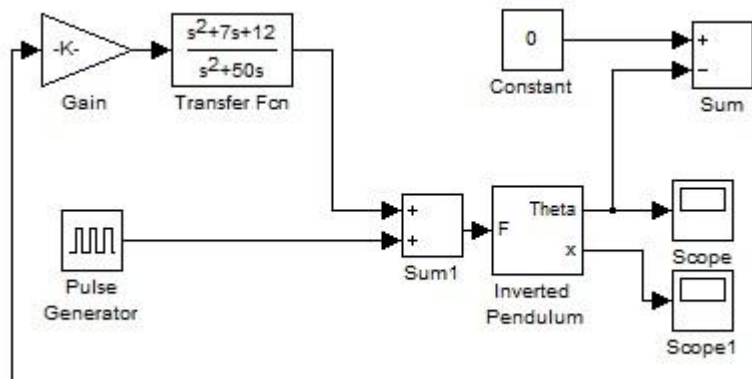


23 Angolo e posizione simulati PID

compatibili con i precedenti e quindi soddisfano i requisiti.

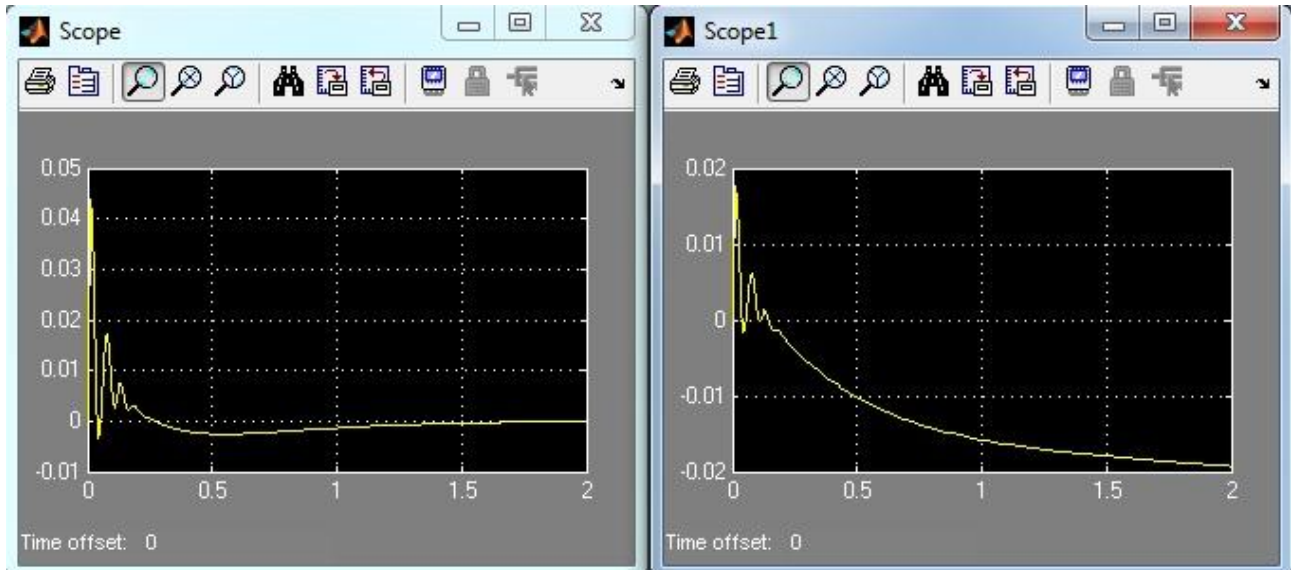
Luogo delle radici

Per implementare questo controllore su Simulink dobbiamo inserire un blocco guadagno K e un blocco funzione di trasferimento con i valori che abbiamo trovato prima



24 Controllore RL

Avviando la simulazione otteniamo questi grafici

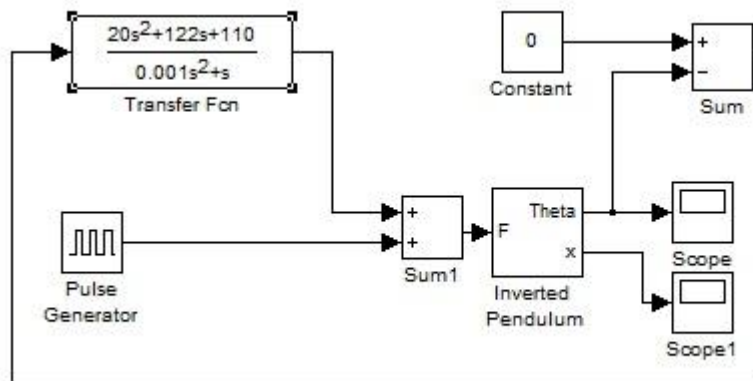


25 Angolo e posizione simulati RL

soddisfando ancora i requisiti progettuali.

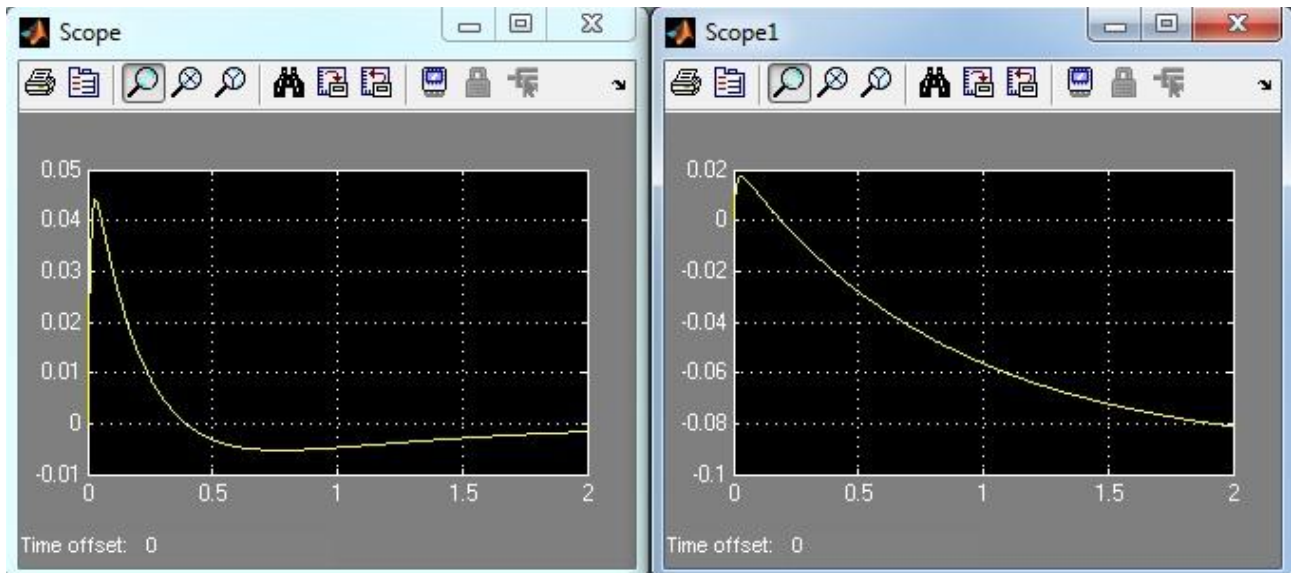
Analisi in frequenza

Per implementare questo controllore su Simulink basterà inserire un blocco funzione di trasferimento con all'interno quella trovata



26 Controllore in frequenza

Avviando la simulazione otteniamo questi grafici



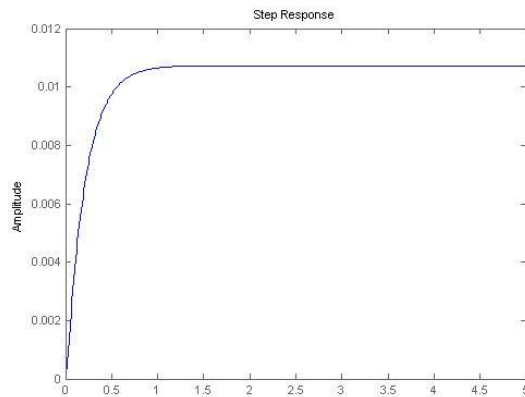
27 Angolo e posizione simulati in frequenza

sempre in linea con quanto richiesto.

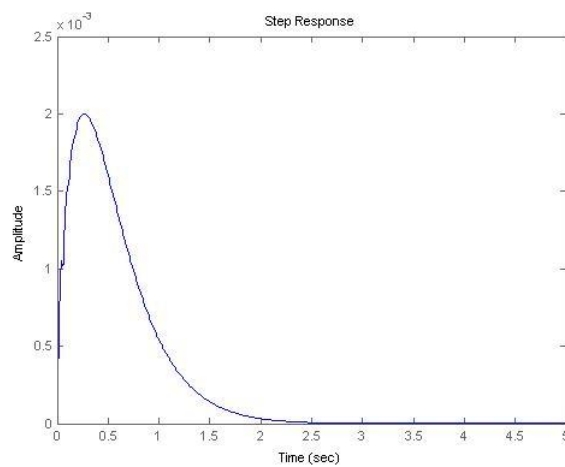
Conclusioni

Abbiamo visto come con metodi molto diversi siamo riusciti a creare tre controllori in grado di rispettare le nostre specifiche e quindi funzionanti. Mettendoli a confronto possiamo stabilire quale si comporta meglio e perché.

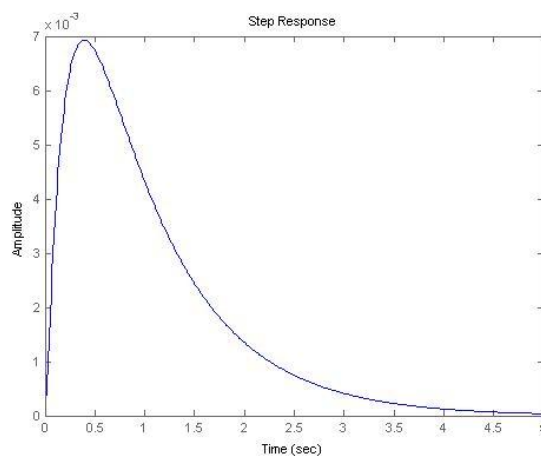
Come metro di paragone utilizzeremo la risposta al gradino del sistema controllato ottenendo rispettivamente quella del PID, del luogo delle radici e dell'analisi in frequenza.



30 PID



29 Luogo delle radici



28 Analisi in frequenza

Il controllore che si comporta peggio è il PID che conserva un errore a regime seppur di piccola entità, mentre vediamo che il luogo delle radici si comporta meglio rispetto all'analisi in frequenza per il fatto che viene smorzato più velocemente.

Se guardiamo i risultati delle simulazioni dei controllori notiamo che la posizione del carrello non è stabile in nessuno dei casi, questo è dovuto allo studio del modello come SISO, con il quale abbiamo controllato il modello solo in base all'angolo con la verticale. Per questo motivo ci rendiamo conto che una realizzazione concreta di questo modello è fisicamente irrealizzabile, si rende quindi necessaria un'analisi MIMO.