

UNIVERSITÀ DEGLI STUDI DI PADOVA
DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

Tesi di Laurea Triennale in
INGEGNERIA DELL'INFORMAZIONE

Titolo Tesi

Progetto di un sistema di cruise control in
ambiente MatLab e Simulink

Relatore
Professor Alessandro Beghi

Laureando
Martini Giulio

Anno Accademico 2012/2013

Indice

Introduzione	2
1 Analisi del sistema fisico	3
1.1 Modellizzazione del sistema Cruise Control	3
1.1.1 Analisi fisica, definizione del modello e funzione di trasferimento	3
1.1.2 Definizione delle variabili e delle funzioni in ambiente MatLab	4
1.1.3 Determinazione dei requisiti progettuali	5
1.2 Analisi del sistema in catena aperta	5
1.2.1 Risposta al gradino	5
1.2.2 Mappa Zeri/Poli della funzione di trasferimento in catena aperta	6
1.2.3 Diagramma di Bode della funzione di trasferimento in catena aperta	7
2 Progetto del controllore	8
2.1 Analisi del sistema in retroazione e introduzione ai controllori	8
2.1.1 Analisi in catena chiusa	8
2.1.2 Considerazioni sui vantaggi dell'introduzione di un controllore	9
2.2 Progetto del controllore PID	9
2.2.1 Considerazioni preliminari sui controllori PID	9
2.2.2 Progetto di un controllore Proporzionale	10
2.2.3 Progetto di un controllore Proporzionale-Integrativo	12
2.2.4 Progetto del controllore Proporzionale-Integrativo-Derivativo	15
2.2.5 Considerazioni finali sulla progettazione di controllori di tipo PID	17
3 Analisi mediante luogo delle radici e studio in frequenza	18
3.1 Analisi mediante il luogo delle radici	18
3.1.1 Considerazioni preliminari	18
3.1.2 Analisi della risposta al variare del guadagno	19
3.1.3 Progetto della rete ritardatrice	23
3.1.4 Considerazioni finali sul luogo delle radici	26
3.2 Analisi in frequenza del controllore	27
3.2.1 Considerazioni preliminari	27
3.2.2 Diagramma di Bode e risposta in catena aperta	28
3.2.3 Studio in catena chiusa e progetto di una rete ritardatrice	28
3.2.5 Considerazioni finali sull'analisi in frequenza	31
4 Analisi del modello in Simulink	32
4.1 Modello e simulazione in catena aperta	32
4.1.1 Considerazioni preliminari	32
4.1.2 Creazione del modello in Simulink	32
4.1.3 Simulazione in catena aperta	33
4.2 Controllo e simulazione in retroazione	34
4.2.1 Considerazioni preliminari	34
4.2.2 Controllore Proporzionale-Integrativo	34
4.2.3 Controllore Proporzionale-Integrativo-Derivativo	35
4.2.4 Rete ritardatrice	36
4.2.5 Considerazioni finali sulla simulazione tramite Simulink	38
Conclusioni	39

Introduzione

Il Cruise Control è un dispositivo elettronico che si trova nella maggior parte dei veicoli di moderna progettazione. Esso ha il compito di regolare automaticamente la velocità della vettura in base a quanto impostato dal conducente e di mantenerla costante durante il tragitto, restando nei limiti di sicurezza posti dalle condizioni stradali, di assetto e di traffico presenti.

Il suo funzionamento si basa sull'azione di un sistema controllo retroazionato che regola la velocità del veicolo in base a quella impostata dal conducente. Esso valuta infatti la velocità corrente della vettura e la confronta con quella desiderata: a seconda che risulti minore o maggiore agisce di conseguenza sulla potenza erogata dal motore, aumentandola o diminuendola rispettivamente, fino a raggiungere quella richiesta.

Questo dispositivo introduce notevoli vantaggi nella guida, apprezzabili soprattutto durante lunghi tragitti su strade poco trafficate, dove è desiderabile viaggiare a velocità di crociera. Il mantenimento di una velocità relativamente costante permette infatti di risparmiare carburante e al tempo stesso di aumentare il comfort di viaggio grazie all'eliminazione delle continue accelerazioni e decelerazioni dovute alla regolazione manuale della velocità.

Questa tesi si propone di attuare uno studio di tale dispositivo attraverso una sua modellizzazione matematica e di simularne il comportamento in modo che soddisfi alcune condizioni di progettazione prestabilite. Una volta ottenuto tale modello si procede analizzandone le principali caratteristiche in catena aperta (risposta al gradino e analisi in frequenza) per poi passare alla progettazione di controllori seguendo diverse strategie (controllore PID, luogo delle radici, rete ritardatrice) e alla rispettive analisi in catena chiusa, volte al soddisfacimento delle condizioni poste.

Per lo svolgimento di tale studio si è fatto uso dei software MatLab e Simulink.

In conclusione viene effettuata un'osservazione volta a riassumere quanto osservato e a stabilire quale strategia di progetto sia preferibile tra le varie proposte, in base al soddisfacimento dei requisiti iniziali e ai criteri di semplicità di progettazione e di realizzazione.

1 Analisi del sistema fisico

1.1 Modellizzazione del sistema Cruise Control

1.1.1 Analisi fisica, definizione del modello e funzione di trasferimento

Per la modellizzazione fisica del nostro problema si considera per astrazione la vettura di massa m come puntiforme e si suppone che su di essa agiscano la forza motrice u da noi controllata e la forza antagonista di attrito proporzionale alla velocità bv (comprensiva dell'attrito volvente e viscoso).

Sono inoltre presenti la forza peso dell'autovettura e la corrispondente reazione vincolare della strada, come riportato in Figura 1.1.

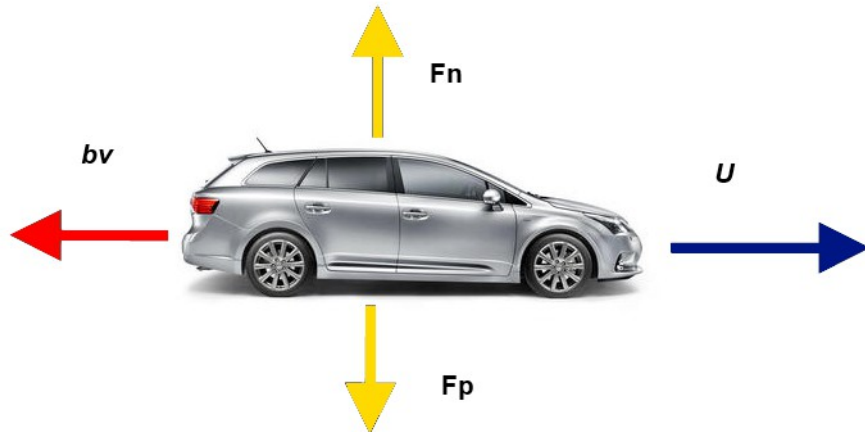


Figura 1.1: Schema del sistema fisico

Facendo la sommatoria delle forze in campo, la forza peso viene bilanciata dalla reazione vincolare, mentre per quanto riguarda forza motrice e attrito si ottiene:

$$F = u - bv = ma$$

Definendo poi la velocità v come derivata prima spostamento e l'accelerazione a come sua derivata seconda

$$v = \dot{x}$$

$$a = \dot{v} = \ddot{x}$$

si ottiene la seguente equazione differenziale di primo grado:

$$u = m\dot{v} + bv$$

Per giungere alla formulazione di un modello ingresso/uscita ed essendo la velocità la variabile che ci interessa poter controllare, si impone

$$y = v$$

ottenendo il seguente modello:

$$\begin{cases} \dot{x} = Fx + Gu \\ y = Hx + Du \end{cases}$$

In questo caso specifico diventa:

$$\begin{cases} \dot{v} = -\frac{b}{m}v + \frac{1}{m}u \\ y = v \end{cases}$$

Ora per trovare la funzione di trasferimento ad esso associata, si passa nel dominio delle trasformate di Laplace:

$$sV(s) = -\frac{b}{m}V(s) + \frac{1}{m}U(s)$$

Da cui:

$$U(s) = V(s)(ms + b)$$

La funzione di trasferimento ad anello aperto cercata è quindi:

$$G(s) = \frac{V(s)}{U(s)} = \frac{1}{ms + b}$$

Infine si attribuiscono i rispettivi valori numerici ai parametri di sistema considerati, come riportato in Tabella 1.1.

Tabella 1.1: Valori dei parametri di sistema

	Variabile	Valore numerico
Massa della vettura	m	1000 [kg]
Coefficiente d'attrito	b	50 [Ns/m]
Forza motrice	u	500 [N]
Velocità di riferimento	r	10 [m/s]

1.1.2 Definizione delle variabili e delle funzioni in ambiente MatLab

Per poter affrontare lo studio del problema in MatLab si rende ora necessario definire le variabili e le funzioni del modello ricavato in precedenza secondo il codice del programma.

Definizione delle variabili e attribuzione dei valori numerici stabiliti:

```
m = 1000; %Massa vettura  
b = 50; %Coefficiente d'attrito  
u = 500; %Forza motrice
```

Coefficienti nella rappresentazione in modello di stato:

```
F = -b/m;  
G = 1/m;  
H = 1;  
D = 0;
```

Definizione del modello e delle funzioni:

```
Sscruise=ss(F,G,H,D); %Crea il modello nello spazio di stato  
s=tf('s');  
W_Function=(1/(m*s+b)); %Definizione della funzione di trasferimento in anello aperto
```

1.1.3 Determinazione dei requisiti progettuali

Si stabiliscono ora quali siano i requisiti progettuali che il dispositivo dovrà soddisfare nella nostra analisi. Avendo attribuito alla forza motrice un valore di 500 N la velocità massima raggiungibile dalla vettura sarà di 10 m/s. Come specifiche di progetto si è stabilito che questa velocità debba essere raggiunta entro 5 s di tempo (partendo da una velocità iniziale nulla) e che la rispettiva sovralongazione non superi il 10%. Viene imposto inoltre che l'errore a regime sia al massimo del 2%.

Riassumendo, le specifiche da soddisfare sono:

- tempo di salita < 5 s
- sovralongazione < 10%
- errore a regime < 2%

1.2 Analisi del sistema in catena aperta

1.2.1 Risposta al gradino

Avendo completato la modellizzazione fisica del nostro sistema, si passa ora ad analizzarne il comportamento in catena aperta, ovvero in assenza di controllori. Il corretto funzionamento di un tale sistema, basandosi esclusivamente sulla corretta dell'individuazione del modello matematico e sulla precisione della conoscenza dei parametri in gioco, non è in grado di correggere gli errori dovuti all'inevitabile approssimazione introdotta dal modello stesso e alla non perfetta conoscenza dei parametri. Ciò comporta che nella pratica tale soluzione non funziona mai e di conseguenza va sempre evitata.

Si prosegue ora con una sua trattazione analitica, supponendo di avere condizioni iniziali nulle e di applicare alla vettura una forza motrice u a gradino di 500 N come stabilito in precedenza. Analizzando la risposta del sistema in MatLab tramite il comando

```
step(u*W_Function); %traccia la risposta al gradino
```

si ottiene il grafico riportato in Figura 1.2:

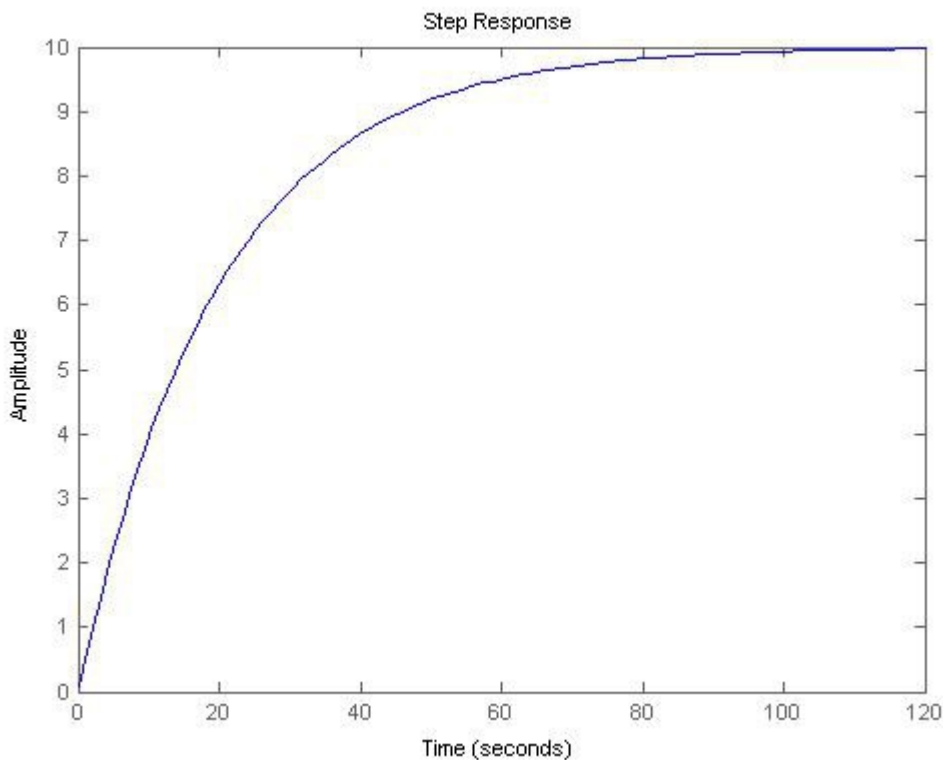


Figura 1.2: Risposta al gradino in catena aperta

Da questo si osserva come i requisiti di progettazione non siano tutti soddisfatti. Sebbene non ci sia sovraelongazione, infatti, si può notare come la risposta del sistema raggiunga la velocità preposta in un tempo decisamente superiore al tempo di salita stabilito. Di conseguenza è necessario passare alla fase di studio di un compensatore che sia in grado di soddisfare tutti i requisiti preposti modificando la funzione di trasferimento del sistema.

1.2.2 Mappa Zeri/Poli della funzione di trasferimento in catena aperta

Guardando la funzione di trasferimento del sistema si può notare come l'unico polo presente si trovi nel punto $s = -\frac{b}{m}$

Avendo parte reale negativa il sistema risulta essere BIBO-stabile. Grazie all'istruzione MatLab

```
pzmap(W_Function); axis([-1 1 -1 1]); %Grafico di zeri e poli
```

è possibile rappresentarlo nel piano di Gauss, come riportato in Figura 1.3.

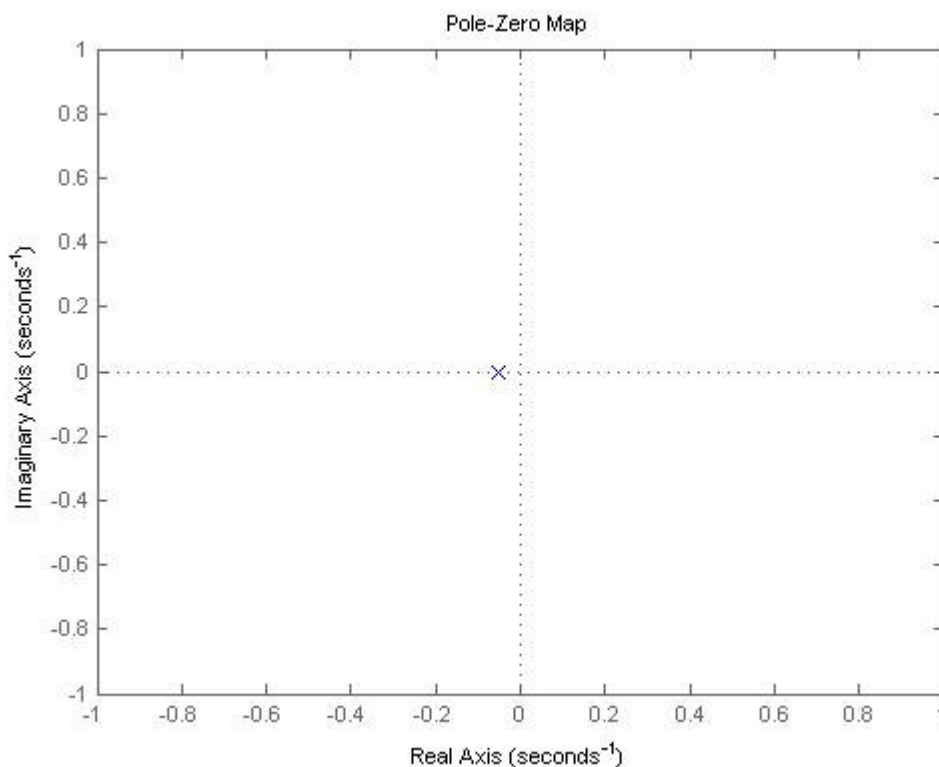


Figura 1.3: Mappa zeri/poli FDT in catena aperta

Dalla figura si ha conferma che essendo l'unico polo situato sul semiasse reale negativo, siamo in presenza di BIBO-stabilità; al tempo stesso il suo valore risulta essere molto piccolo (pari a -0,05) e questo comporta l'eccessiva lentezza nella risposta del sistema. Non potendo modificare i parametri fisici del sistema (la massa della vettura e il coefficiente d'attrito sono considerati come noti ma fissi) l'unica possibilità che si ha per migliorare il sistema risiede nel progetto di un controllore che permetta di raggiungere i requisiti.

1.2.3 Diagramma di Bode della funzione di trasferimento in catena aperta

È inoltre possibile analizzare il sistema in catena aperta tramite i diagrammi di Bode del modulo e della frequenza della sua funzione di trasferimento.

I due diagrammi si possono ottenere in MatLab tramite il comando

`bode(W_Function); %Comando che disegna i grafici di Bode`

che restituisce il grafico riportato in Figura 1.4:

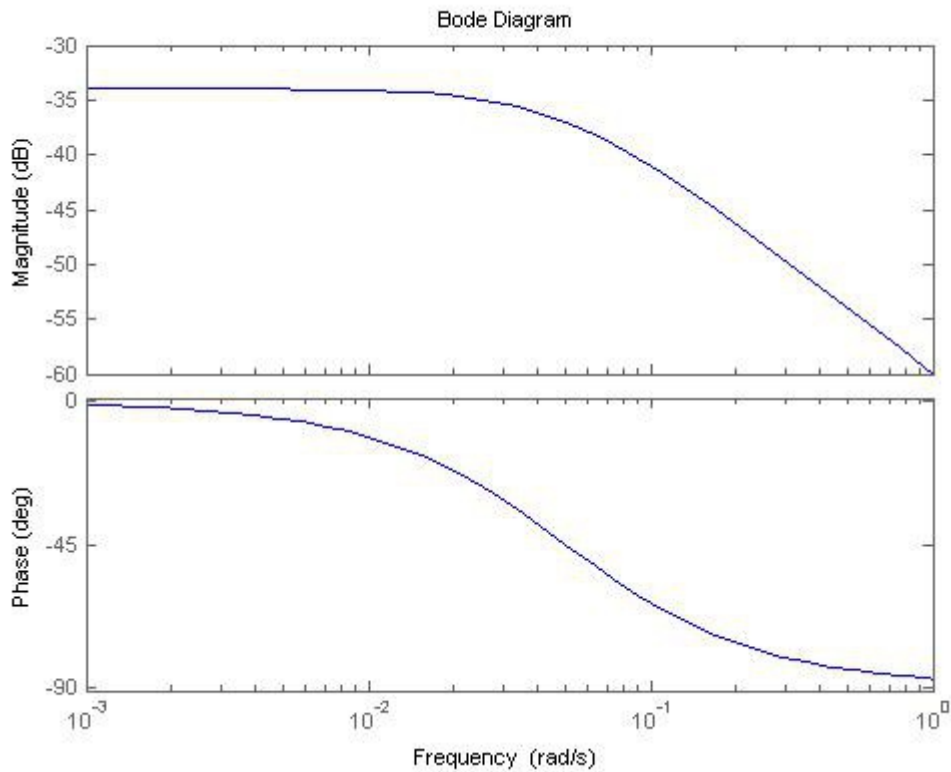


Figura 1.4: Diagrammi di Bode della FDT in catena aperta

dal quale è facile osservare come il modulo cali di 3 dB (banda passante) in corrispondenza della pulsazione $\omega = -\frac{b}{m} = 5 \cdot 10^2$, mentre la corrispondente fase scenda a -45° .

Per frequenze elevate la fase si mantiene costante a -90° mentre il modulo continua a calare con una pendenza di 20 dB per decade.

2 Progetto del controllore

2.1 Analisi del sistema in retroazione e introduzione ai controllori

2.1.1 Analisi in catena chiusa

In base a quanto visto in precedenza, si consideri ora di avere un sistema in retroazione unitaria negativa con controllore, come illustrato in Figura 2.1.

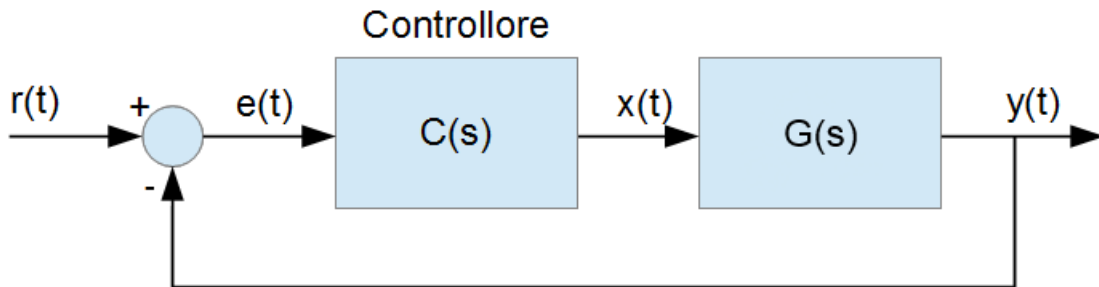


Figura 2.1: Schema a blocchi del sistema in retroazione unitaria negativa

In tale sistema l'uscita viene riportata all'ingresso non scalata e con segno invertito, in modo che al controllore arrivi un segnale dato dalla differenza tra il segnale di partenza e l'uscita stessa.

La figura rappresenta una schematizzazione del sistema descritto, in cui il blocco $G(s)$ rappresenta la funzione di trasferimento del sottosistema in catena aperta che si suppone essere nota con parametri non modificabili, mentre il blocco $C(s)$ rappresenta il controllore che da implementare per soddisfare i requisiti richiesti al dispositivo.

Il segnale

$$e(t) = r(t) - y(t)$$

rappresenta l'errore del sistema: nel caso in analisi deve essere minore del 2% per essere ritenuto accettabile.

Per ricavare la nuova funzione di trasferimento del sistema si passa quindi nel dominio di Laplace ottenendo:

$$E(s) = R(s) - Y(s)$$

Ora scrivendo $Y(s)$ e $X(s)$ come di seguito:

$$Y(s) = X(s)G(s)$$

$$X(s) = E(s)C(s)$$

si ottiene di conseguenza che:

$$Y(s) = E(s)C(s)G(s) = [R(s) - Y(s)]C(s)G(s) = R(s) \frac{C(s)G(s)}{1 + C(s)G(s)}$$

Da essa è infine possibile ricavare la funzione di trasferimento cercata, data da:

$$W(s) = \frac{Y(s)}{R(s)} = \frac{C(s)G(s)}{1 + C(s)G(s)}$$

2.1.2 Considerazioni sui vantaggi dell'introduzione di un controllore

L'introduzione di un blocco di controllo serve quindi a modificare il guadagno e ad introdurre zeri e poli modificando così la funzione di trasferimento in modo da ottenere un'uscita tale da soddisfare le specifiche richieste in partenza.

Nel caso in analisi si deve progettare un controllore facendo particolare attenzione - oltre al mantenimento della stabilità - all'errore al regime, al tempo di salita e alla sovraelongazione.

Per controllare l'errore a regime bisogna prestare attenzione al tipo del sistema: anche se le specifiche richiedono che l'errore sia piccolo, potrebbe essere preferibile che esso sia addirittura nullo. Per avere un errore nullo per la risposta al gradino occorre disporre di un sistema di tipo uno (ovvero con un polo nell'origine), mentre la funzione di trasferimento in catena aperta non ne presenta alcuno risultando essere di tipo 0. Ciò comporta un errore al gradino limitato ma non nullo.

Per quanto riguarda il tempo di salita, esso è collegato alla pulsazione di attraversamento. Si rende quindi necessaria un'azione sul guadagno della funzione di trasferimento volta ad aumentarlo.

Infine, per tenere conto del fenomeno della sovraelongazione, bisogna far sì di avere un adeguato margine di fase in modo da ridurre il fenomeno il più possibile.

2.2 Progetto del controllore PID

2.2.1 Considerazioni preliminari sui controllori PID

Alla categoria dei controllori PID appartengono dispositivi atti allo svolgimento di tre azioni distinte i cui effetti vengono poi sovrapposti. Essi prendono nome proprio dalla combinazione di queste tre azioni, che in ordine sono: Proporzionale, Integrativa e Derivativa.

La loro formulazione matematica è descritta dalla seguente funzione:

$$C(s) = K_p + \frac{K_i}{s} + K_d s$$

Si può facilmente constatare come a ciascuno corrisponda rispettivamente una delle tre azioni.

L'effetto del primo termine, chiamato proporzionale, è quello di aumentare o diminuire il guadagno della funzione di trasferimento in base al valore della costante influenzando di conseguenza la rispettiva pulsazione di attraversamento.

Il secondo termine, chiamato integrativo, serve invece ad introdurre un polo nell'origine, utile per innalzare il tipo del sistema ed eliminare o ridurre l'errore a regime.

Il terzo termine infine, ovvero quello derivativo, introduce uno zero in modo tale da migliorare la stabilità del sistema e a regolare pulsazione di attraversamento e margine di fase.

Non sempre si ricorre all'utilizzo di tutti e tre questi contributi: esistono infatti dei casi particolari in cui ne viene utilizzato solo uno o un'opportuna combinazione di due di essi.

Il primo caso è costituito dal controllore di tipo P, nel quale è presente solo il termine proporzionale (ponendo $K_i=0$ e $K_d=0$); il secondo caso è costituito dal controllore di tipo I, nel quale è solamente presente il termine integrativo (ponendo $K_p=0$ e $K_d=0$); il terzo caso è dato dalla combinazione dei primi due termini e prende il nome di controllore PI (ponendo $K_d=0$); il quarto caso è infine dato dalla combinazione del primo e dell'ultimo termine e prende quindi il nome di controllore PD (ponendo $K_i=0$).

Dal punto di vista dell'implementazione fisica del controllore bisogna inoltre ricordare come non sia possibile realizzare controllori la cui funzione di trasferimento sia impropria. In tali casi si agisce aggiungendo uno o più poli in alta frequenza in modo da renderla propria senza però interferire con l'azione del controllore stesso alle frequenze di interesse.

Tornando all'utilizzo di MatLab, esistono due metodi diversi per implementare un controllore PID.

Il primo consiste nel definire manualmente la funzione di trasferimento del controllore,

(attribuendo un valore provvisorio alle varie costanti):

```
Kp=1; %Costante proporzionale
Ki=1; %Costante di integrazione
Kd=1; %Costante di derivazione
s=tf('s');
C(s) = Kp + Ki/s + Kd*s;
```

Il secondo consiste nell'utilizzare il comando di MatLab prestabilito per la definizione di un controllore PID:

```
C(s)=pid(Kp,Ki,Kd); %Definizione controllore
```

Si procede ora con la fase di effettiva progettazione di controllori di tipo PID atti a migliorare la risposta al gradino del dispositivo in analisi, proponendo tre diverse soluzioni e discutendone la rispettiva efficacia.

2.2.2 Progetto di un controllore Proporzionale

Si inizia quindi la progettazione di un controllore di tipo PID partendo dal caso più semplice, ovvero da un controllore di tipo proporzionale, la cui funzione di trasferimento è data da:

$$C(s) = K_p$$

Grazie all'azione di tale controllore è possibile modificare il guadagno di Bode e la frequenza di attraversamento. Si riesce quindi a ridurre il tempo di salita della risposta e l'errore al gradino, senza però poter eliminare quest'ultimo in quanto non si innalza il tipo del sistema (non viene introdotto alcun polo nell'origine).

Considerando la funzione di trasferimento in catena chiusa in presenza di un controllore PID è ora possibile formulare il caso specifico che si ottiene con l'inserimento di un controllore puramente proporzionale.

Eseguendo alcuni passaggi si trova la rispettiva funzione di trasferimento:

$$W_{pro}(s) = \frac{Y(s)}{R(s)} = \frac{G(s)C(s)}{1+G(s)C(s)} = \frac{K_p}{ms+b+K_p}$$

Per proseguire l'analisi ora vengono attribuiti alcuni valori numerici alla variabile K_p , cercando di soddisfare i requisiti progettuali. Come primo valore, si stabilisce $K_p=100$

Si riporta di seguito il codice di MatLab in cui si attribuisce tale valore alla costante K_p per poi definire la funzione di trasferimento del compensatore:

```
m=1000;
b = 50;
r = 10; %Velocità di riferimento (m/s)
s=tf('s');
W_Function=1/(m*s + b); %Funzione di trasferimento
Kp=100; %Costante proporzionale
Cpid=pid(Kp); %Funzione di trasferimento PID ottenuta tramite comando diretto
```

```
W_prop = feedback(Cpid*W_Function,1); %Generazione della funzione di trasferimento in catena chiusa
                                     tramite comando diretto di Matlab
```

```
t = 0:0.1:20;
step(r*W_prop,t);
axis([0 20 0 10]);
```

Il quale restituisce il grafico riportato in Figura 2.2 per la risposta al gradino:

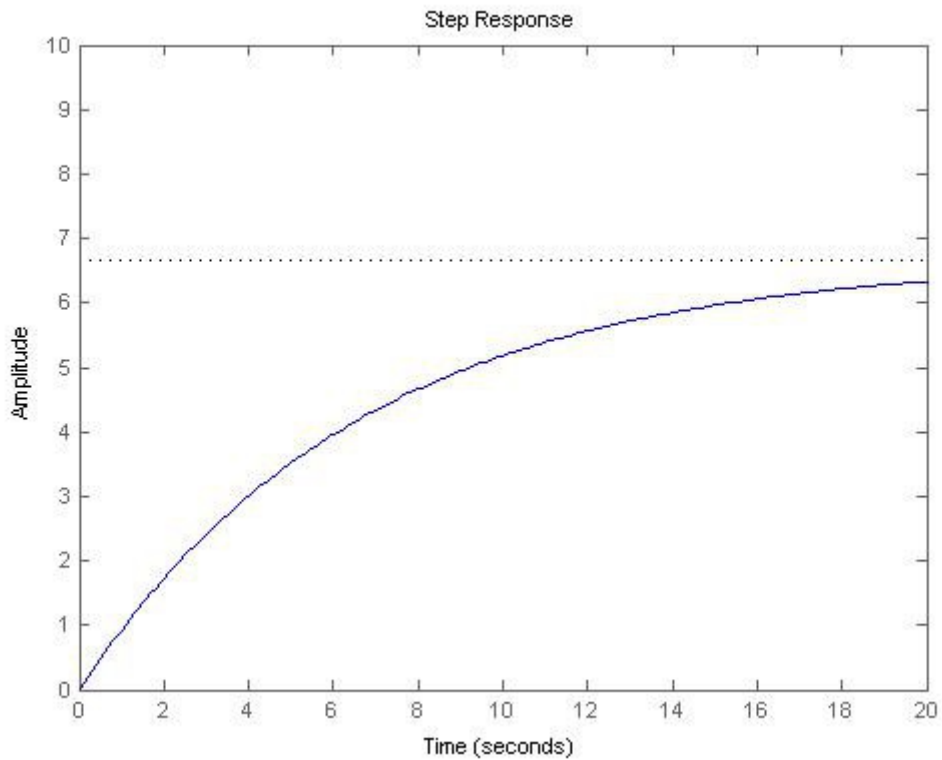


Figura 2.2: Risposta al gradino con controllore proporzionale ($K_p=100$)

Dal esso si osserva come, avendo aumentato il guadagno di Bode, la risposta sia effettivamente diventata più rapida rispetto al caso precedente. Restano tuttavia insoddisfatti i requisiti sia per quanto riguarda il tempo di salita sia per quanto riguarda l'errore a regime. Si nota invece l'assenza di sovraelongazione. Si opta quindi per un aumento di guadagno più considerevole per cercare di avvicinarsi maggiormente agli obiettivi preposti; scegliendo $K_p=5000$ si ottiene il grafico riportato in Figura 2.3 per la risposta:

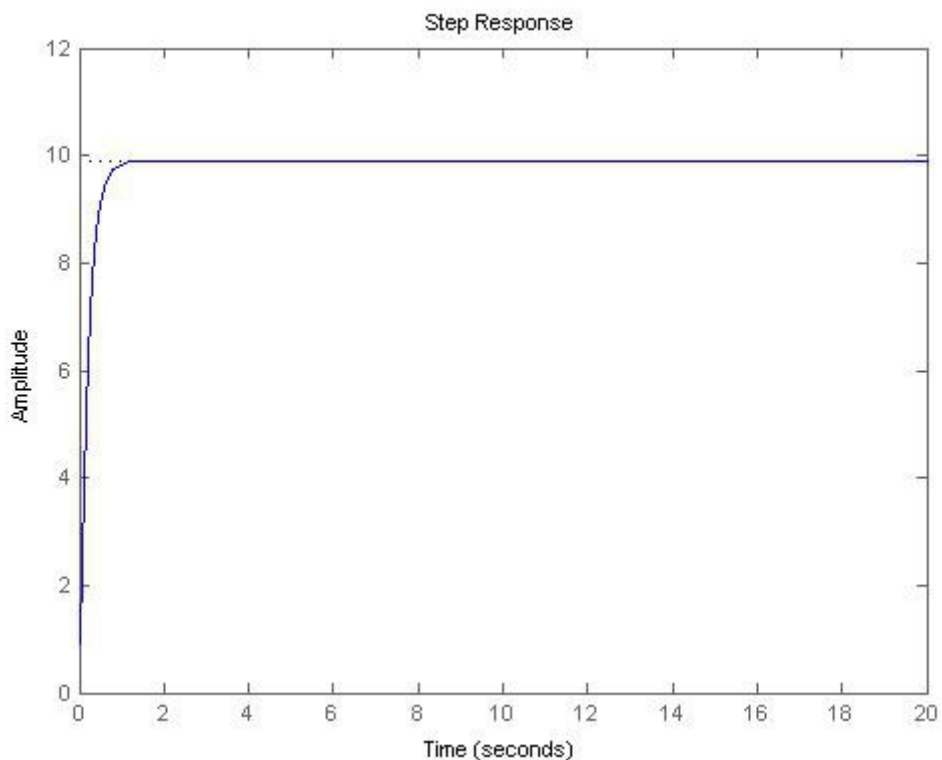


Figura 2.3: Risposta al gradino con controllore proporzionale ($K_p=5000$)

Si usa ora il seguente comando integrato in MatLab:

```
S = stepinfo(r*W_prop)
```

Esso restituisce i seguenti valori per alcuni parametri della risposta, tra i quali si trovano anche quelli richiesti dalle specifiche di progetto, ovvero tempo di salita (rise time), sovraelongazione (overshoot) ed errore a regime (ottenibile dalla differenza tra valore desiderato a regime e valore finale a regime).

I valori restituiti sono:

```
RiseTime: 0.4351%tempo di salita  
SettlingTime: 0.7747%tempo di assestamento  
SettlingMin: 8.9164%valore minimo raggiunto dopo il tempo di salita  
SettlingMax: 9.8970%valore massimo raggiunto dopo il tempo di salita  
Overshoot: 0%sovraelongazione  
Undershoot: 0%ottoelongazione  
Peak: 9.8970%valore di picco  
PeakTime: 1.5478%tempo corrispondente al valore di picco
```

Si nota come grazie al considerevole aumento di guadagno si sia ora teoricamente riusciti a soddisfare le richieste progettuali: il tempo di salita è infatti $t_r=0.4351$, mentre l'errore a regime è minore del 2% ($e_{rp}=0.103$ ovvero circa dell'1 %).

Da un punto di vista pratico, però, una tale soluzione sarebbe impossibile da realizzare. Infatti essa comporterebbe un'accelerazione iniziale troppo elevata difficile da ottenere nonché sgradevole per il conducente, non trovandoci in condizioni di competizione su circuito.

Si è quindi costretti ad abbandonare tale soluzione e a concentrarsi su altri tipi di compensatore.

2.2.3 Progetto di un controllore Proporzionale-Integrativo

Si passa quindi ora allo studio di un controllore di tipo proporzionale integrativo, la cui funzione di trasferimento è data da:

$$C(s) = K_p + \frac{K_i}{s}$$

Da quanto osservato nel caso precedente discende la necessità di limitare il valore di K_p onde evitare accelerazioni troppo brusche. Per cercare di soddisfare le richieste è però possibile sfruttare l'introduzione di un polo nell'origine, in modo da innalzare così il tipo del sistema e da eliminare del tutto l'errore della risposta al gradino.

Si ricava ora la funzione di trasferimento del sistema retroazionato, sviluppando la funzione del caso generale per controllori PID e ottenendo:

$$W_{pi}(s) = \frac{Y(s)}{R(s)} = \frac{G(s)C(s)}{1 + G(s)C(s)} = \frac{K_p s + K_i}{ms^2 + (b + K_p)s + K_i}$$

Come nel caso precedente si procede per tentativi, partendo con $K_p=600$ e $K_i=1$. Passando alla stesura del codice in MatLab si ha quindi:

```
m = 1000;  
b = 50;  
r=10; %Velocità di riferimento  
  
s=tf('s');  
W_Function=1/(m*s + b);  
Kp=600; %Costante proporzionale  
Ki=1; %Costante di integrazione
```

```
Cpid=pid(Kp,Ki); %Funzione di trasferimento PID
```

```
W_pi=feedback(Cpid*W_Function,1); %Funzione di trasferimento in catena chiusa
```

```
t = 0:0.1:20;
```

```
step(r*W_pi,t); %Generazione grafico
```

```
axis([0 20 0 12]);
```

dal quale si ottiene la risposta al gradino riportata in figura 2.4:

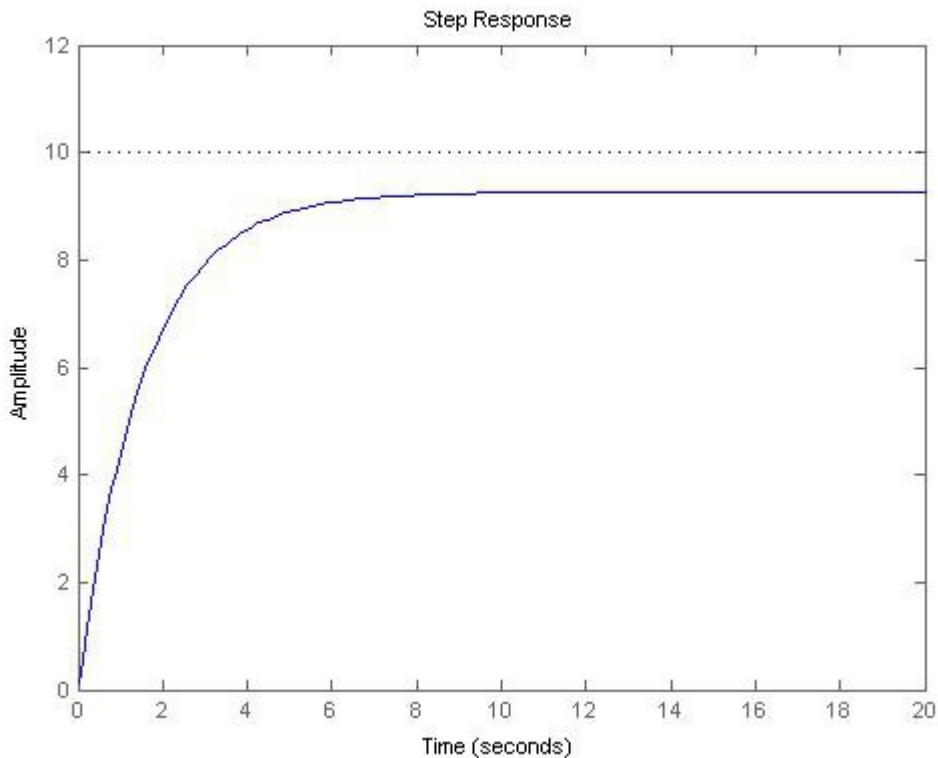


Figura 2.4: Risposta al gradino con controllore PI ($K_p=600$, $K_i=1$)

Dalla figura è facile vedere come le specifiche non siano affatto soddisfatte in termini di tempo di salita. È inoltre importante sottolineare che l'errore a regime è nullo: nella figura esso non risulta chiaro perché la finestra temporale risulta essere troppo ristretta. Ne si ha conferma però grazie al comando

```
S = stepinfo(r*W_pi)
```

che restituisce:

```
RiseTime: 5.3412  
SettlingTime: 856.2899  
SettlingMin: 9.0481  
SettlingMax: 9.9691  
Overshoot: 0  
Undershoot: 0  
Peak: 9.9691  
PeakTime: 2.0680e+03
```

Si procede pertanto modificando il valore numerico delle due costanti K_p e K_i nel codice MatLab fino a trovare dei valori ottimali.

Alcuni esempi di risposte ottenute modificando i valori delle costanti sono riportati di seguito in Figura 2.5.

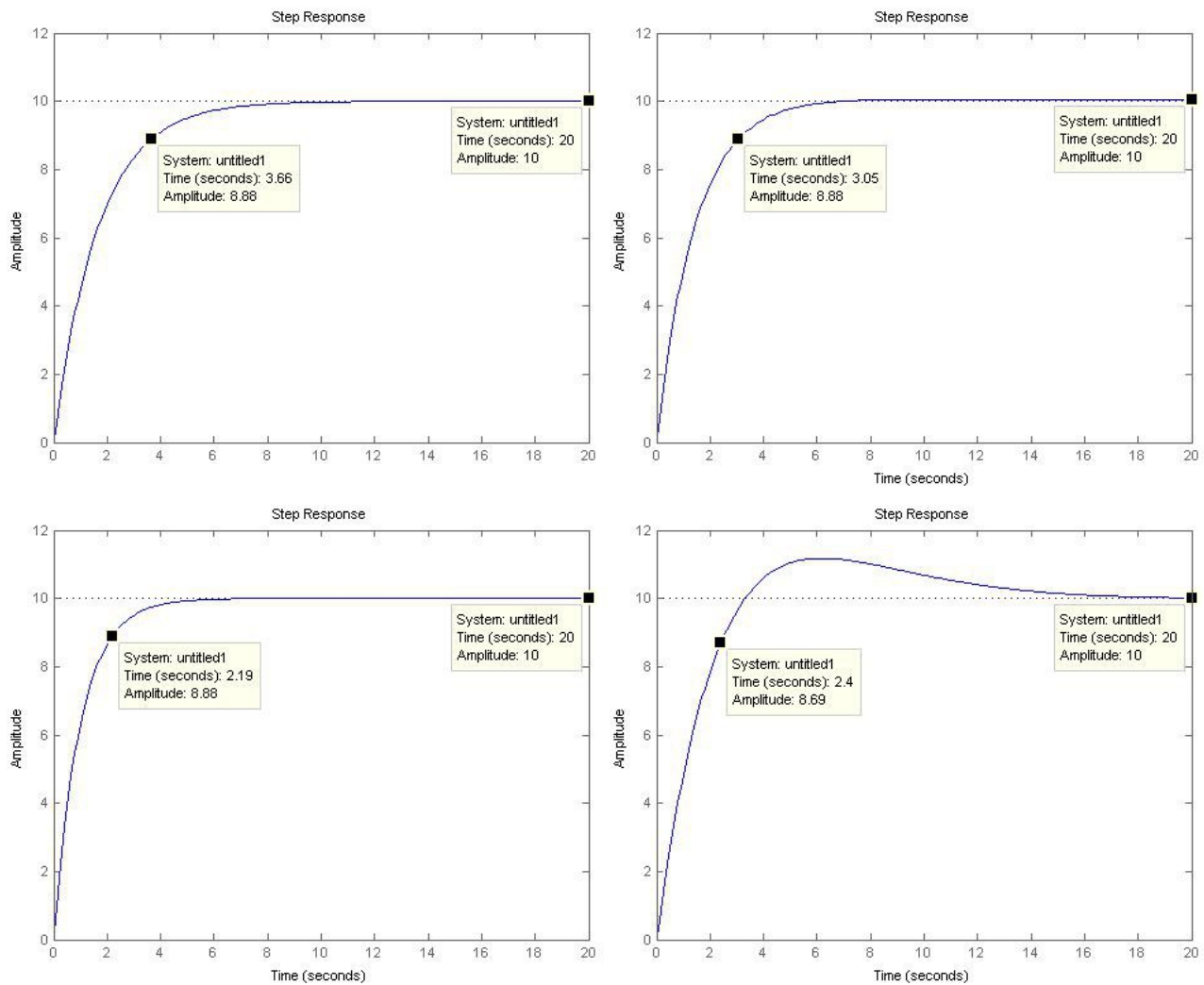


Figura 2.5: Confronto di risposte con controllo PI al variare di K_p e K_i .

Procedendo in ordine: 1) $K_p=600, K_i=30$ 2) $K_p=700, K_i=40$ 3) $K_p=1000, K_i=50$ 4) $K_p=600, K_i=130$

Da esse si può osservare come alle variazioni dei valori corrispondano notevoli cambiamenti nell'andamento delle diverse risposte: all'aumentare di K_p , come del resto già visto per il controllore di tipo P, si ha una riduzione del tempo di salita e un conseguente aumento dell'accelerazione iniziale, mentre all'aumentare di K_i si ha un incremento della sovralongazione. Se per valori troppo elevati di K_p ci si riconduce alle conclusioni discusse nel caso del controllore di tipo proporzionale, per valori elevati di K_i invece si ha una sovralongazione che supera le specifiche di progetto.

Tra i vari casi presi in analisi è possibile notare come nel terzo le specifiche siano tutte rispettate: infatti l'errore a regime è nullo (quest'ultimo lo è in tutti i casi), il tempo di salita è di 2.1973 secondi mentre la sovralongazione è sicuramente inferiore al 10% (Overshoot=0). È bene ricordare come i valori assegnati alle costanti in questa soluzione non siano necessariamente gli unici accettabili: non avendo specifiche troppo stringenti rimane certamente un margine di libertà nella loro scelta.

Si riporta più in grande in Figura 2.6 la risposta al gradino per i valori scelti.

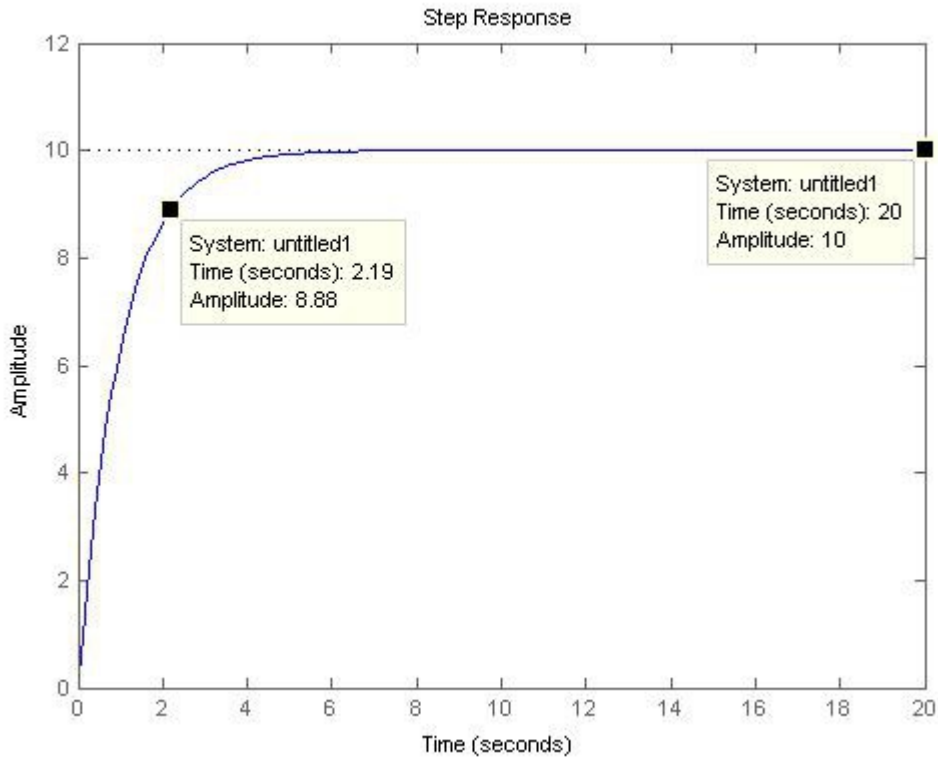


Figura 2.6: Risposta al gradino con controllore PI ($K_p=1000$, $K_i=50$)

Si riportano inoltre i valori restituiti dal comando:

`S = stepinfo(r*W_pi)`

RiseTime: 2.1973
 SettlingTime: 3.9124
 SettlingMin: 9.0056
 SettlingMax: 9.9960
 Overshoot: 0
 Undershoot: 0
 Peak: 9.9960
 PeakTime: 7.8163

2.2.4 Progetto del controllore Proporzionale-Integrativo-Derivativo

Avendo già trovato una soluzione semplice ed efficace per il problema, non sarebbe necessario proseguire con il progetto di ulteriori controllori. Tuttavia è interessante non limitare l'analisi a tale caso e proseguirla studiando il progetto di un controllore di tipo PID.

Ricordando l'equazione della funzione di trasferimento di un controllore di tipo PID:

$$C(s) = K_p + \frac{K_i}{s} + K_d s$$

risulta facile ricavare attraverso alcuni passaggi la funzione di trasferimento del sistema in catena chiusa. Essa è data da:

$$W_{pid}(s) = \frac{Y(s)}{R(s)} = \frac{G(s)C(s)}{1+G(s)C(s)} = \frac{K_d b s^2 + K_p s + K_i}{(m + K_d)s^2 + (b + K_p)s + K_i}$$

Scrivendo il tutto in codice MatLab si ottiene:

```

m = 1000;
b = 50;
r = 10; %Velocità di riferimento

s = tf('s');
W_Function = 1/(m*s + b);
Kp=600; %Costante proporzionale
Ki=40; %Costante integrativa
Kd=20; %Costante derivativa

Cpid = pid(Kp,Ki,Kd); %Funzione di trasferimento del PID tramite comando MatLab
W_pid= feedback(Cpid*W_Function,1); %Funzione di trasferimento in catena chiusa

t = 0:0.1:20;
step(r*W_pid,t);%Generazione grafico
axis([0 20 0 12]);

```

In Figura 2.7 sono riportati i grafici di alcune risposte ottenute facendo variare i valori assegnati alle costanti K_p , K_i e K_s .

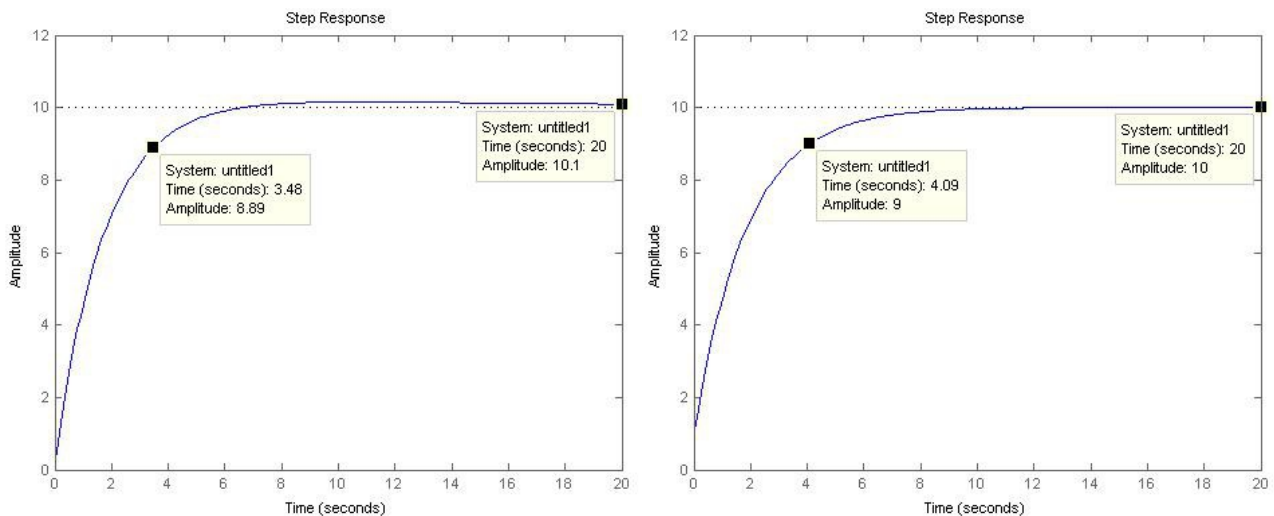


Figura 2.7: Confronto di risposte con controllo PID al variare di K_p , K_i e K_d

Procedendo in ordine: 1) $K_p=600$ $K_i=40$ $K_d=20$ 2) $K_p=600$ $K_i=30$ $K_d=110$

Tra le diverse terne di valori tali da soddisfare i requisiti si può scegliere ad esempio $K_p=1000$, $K_i=50$ e $K_d=30$ (per rimanere vicini a quelli scelti in precedenza e favorire un successivo confronto).

Essi generano la risposta riportata in Figura 2.8:

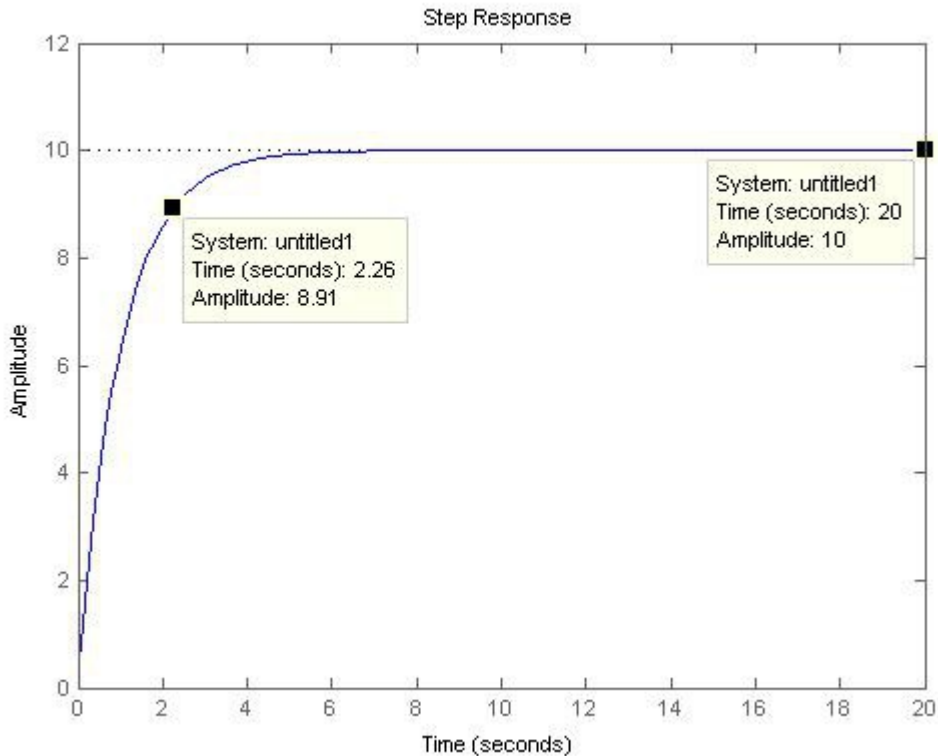


Figura 2.8: Risposta al gradino con controllore PID ($K_p=1000$, $K_i=50$, $K_d=30$)

Alla figura corrispondono infatti i seguenti parametri con i rispettivi valori:

RiseTime: 2.2662
SettlingTime: 4.0326
SettlingMin: 9.0352
SettlingMax: 9.9966
Overshoot: 0
Undershoot: 0
Peak: 9.9966
PeakTime: 8.0636

da cui si osserva immediatamente come le specifiche progettuali siano perfettamente soddisfatte.

2.2.5 Considerazioni finali sulla progettazione di controllori di tipo PID

Studiando la progettazione di tre differenti tipi di controllori PID, rispettivamente P, PI e PID, si è riscontrato come l'impiego di un controllore strettamente proporzionale sia in grado di risolvere il problema solo da un punto di vista teorico, in quanto l'eccessivo valore della rispettiva K_p renda una sua realizzazione non solo tecnologicamente difficile ma anche causa di un'accelerazione iniziale troppo elevata e quindi sgradevole.

Si è poi osservato come l'introduzione di un controllore proporzionale-integrativo, una volta attribuita una coppia congrua di valori alle tre costanti (nel nostro caso $K_p=1000$ e $K_i=50$), sia invece sufficiente per soddisfare in modo ottimale tutte le specifiche progettuali senza incorrere nel problema precedente.

Si è quindi proseguito con la progettazione di un controllore di tipo PID, notando che per la terna di valori $K_p = 1000$, $K_i = 50$ e $K_d = 30$ si ha una risposta simile a quella precedente e tale da costituire una valida soluzione al problema di controllo considerato.

3 Analisi mediante luogo delle radici e studio in frequenza

3.1 Analisi mediante il luogo delle radici

3.1.1 Considerazioni preliminari

Avendo concluso la trattazione della soluzione del problema tramite il progetto di un controllore di tipo PID, si prosegue ora l'analisi utilizzando il metodo del luogo delle radici.

Questo metodo consente di studiare nel piano complesso come gli zeri e i poli del sistema varino in corrispondenza dei valori assunti dal guadagno K_p , permettendo di fare importanti considerazioni non solo riguardo alla stabilità, ma anche al tipo di risposta che si otterrà, per esempio in termini di sovralongazione.

Trattandosi di uno studio ottenuto al variare del parametro K_p , per modellizzare il sistema si utilizzerà uno schema a blocchi in catena chiusa identico a quello ottenuto nel caso dello studio di un controllore di tipo puramente proporzionale.

Ne consegue che anche la rispettiva funzione di trasferimento sarà sempre data da:

$$W(s) = \frac{K_p G(s)}{1 + K_p G(s)}$$

Supponendo di conoscere tutti gli zeri e i poli della funzione di trasferimento del sistema in catena aperta e di poterla quindi scrivere analiticamente come:

$$G(s) = \frac{p(s)}{n(s)}$$

la funzione di trasferimento in retroazione diventa:

$$W(s) = \frac{K_p p(s)}{q(s) + K_p p(s)}$$

Sostituendo in essa l'espressione esplicita di $G(s)$

$$G(s) = \frac{1}{ms + b}$$

si ottiene infine:

$$W_{prop}(s) = \frac{K_p}{ms + b + K_p}$$

L'insieme dei poli della funzione in catena chiusa per un dato valore di $K_p \in \mathbb{R}$ sarà quindi costituito dalle n soluzioni (la cui esistenza è garantita dal teorema fondamentale) dell'equazione:

$$q(s) + K_p p(s) = 0$$

L'insieme delle soluzioni dell'equazione in $s \in \mathbb{C}$ al variare di K_p reale prende il nome di luogo delle radici. Si parla inoltre di luogo delle radici positivo per i punti che soddisfano l'equazione per valori positivi di K_p e di luogo negativo per i punti che la soddisfano per valori di K_p negativi. Studiando il grafico del luogo si osserva che esso è sempre caratterizzato da n rami, corrispondenti alle n soluzioni sopra citate.

Osservando invece il numeratore della funzione di trasferimento, si ha che esso ha m soluzioni corrispondenti agli zeri del sistema. Ricordando inoltre che la funzione di trasferimento è propria, si avrà necessariamente che $n \geq m$.

Nel luogo delle radici si ha quindi che gli n rami partono dai punti del grafico corrispondenti agli n poli di $U(s)$ per andare a chiudersi nei punti corrispondenti agli m zeri del sistema se $n=m$.
Se invece ci si trova nel caso di funzioni strettamente proprie con $n>m$, gli $n-m$ rami che non si chiudono negli zeri divergono all'infinito.

Analizzando il grafico ottenuto si possono immediatamente fare alcune importanti considerazioni.

Si può innanzitutto dedurre la stabilità del sistema per i diversi valori di K_p osservando che a poli stabili corrispondono i punti del luogo che si trovano a sinistra dell'asse immaginario, avendo essi parte reale negativa.

Ne consegue che più un polo si trova lontano dall'asse maggiore sarà il margine di stabilità associato ad esso.

Prima di tracciare il luogo delle radici in MatLab, è utile esprimere le specifiche progettuali in funzione di parametri direttamente osservabili dal luogo stesso.

Si passa quindi al calcolo del coefficiente di smorzamento ξ e della pulsazione naturale ω_n . Dato che la sovraelongazione massima tollerata (S_{max}) è del 10%, dalla formula che la lega al coefficiente di smorzamento ξ :

$$S_{max} \simeq e^{-\frac{\pi \xi}{\sqrt{1-\xi^2}}}$$

si ricava che:

$$\xi \geq \sqrt{\frac{\ln^2(S_{max})}{\pi^2 + \ln^2(S_{max})}} = 0.59 \simeq 0.6$$

Esprimendo il tempo di salita t_r in funzione della pulsazione naturale si può invece scrivere:

$$\omega_n \geq \frac{1.8}{t_r} = 0.36 \text{ rad/s}$$

3.1.2 Analisi della risposta al variare del guadagno

Ora si può proseguire scrivendo il codice in MatLab per poi tracciare il corrispondente luogo delle radici.

```
m = 1000;
```

```
b = 50;
```

```
r = 10;
```

```
s = tf('s');
```

```
W_Function = 1/(m*s + b); %Funzione di trasferimento
```

```
rlocus(W_Function) %Comando che traccia il luogo
```

```
axis([-0.6 0 -0.6 0.6]);
```

```
sgrid(0.6,0.36); %Comando che traccia i luoghi dove il fattore di smorzamento e le frequenza naturale sono costanti per i valori stabiliti
```

Si ottiene così il luogo delle radici, riportato in Figura 3.1:

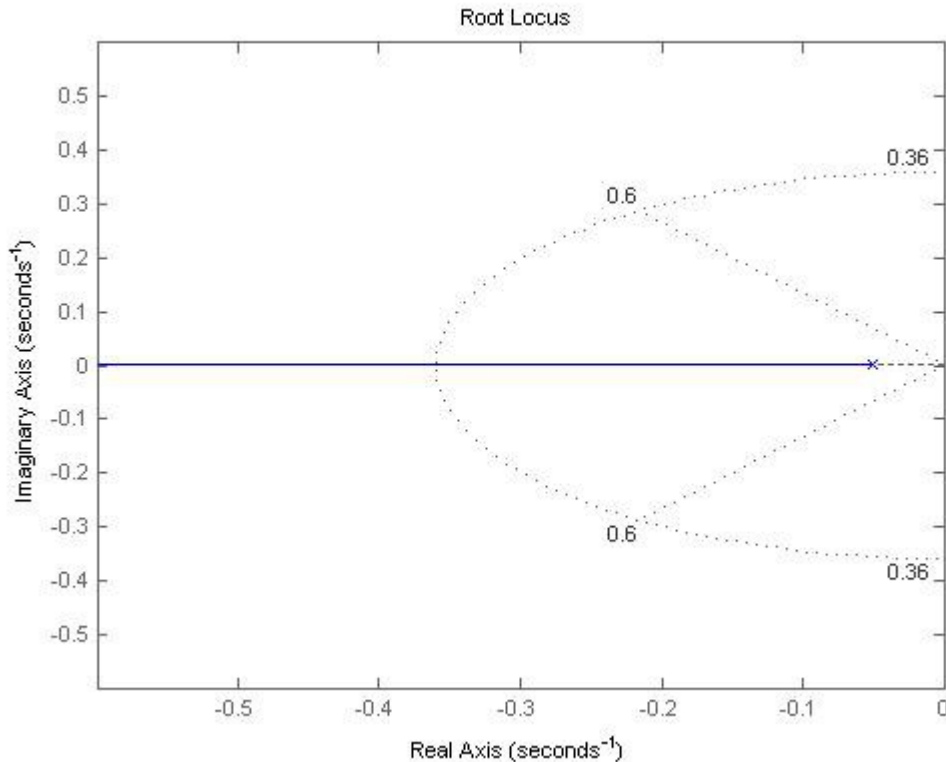


Figura 3.1: Grafico del luogo delle radici

Osservandolo si nota come per $K_p=1$ sia presente un unico polo in $(-0.05, 0)$ corrispondente all'unico polo della funzione di trasferimento in catena aperta. Non è inoltre presente alcuno zero, in accordo con quanto espresso dalla funzione di trasferimento stessa. Ne consegue che l'unico ramo che parte dal polo si sposta lungo l'asse reale negativo chiudendosi all'infinito e che quindi il sistema è stabile per ogni valore di K_p positivo.

L'ellisse e i due segmenti tratteggiati corrispondono rispettivamente ai luoghi dove la pulsazione naturale ω_n è uguale a 0.36 rad/s e il coefficiente di smorzamento ξ è di 0.6 .

Il valore di ω_n è maggiore fuori dall'ellisse e minore al suo interno, mentre il valore di ξ è maggiore per i punti compresi all'interno dell'angolo acuto formato due segmenti e minore per valori esterni ad esso.

Ora si fa ricorso al comando MatLab:

[Kp,poles]=rlocfind(W_Function)%Comando per ottenere il valore di K di un punto del luogo selezionato

esso permette di interagire con il luogo delle radici generato; consente infatti di selezionare un suo punto ottenendo come risposta il corrispettivo valore di K_p associato alla funzione di trasferimento in catena chiusa avente come polo il punto in questione.

Scegliendo ad esempio un punto, come riportato in Figura 3.2:

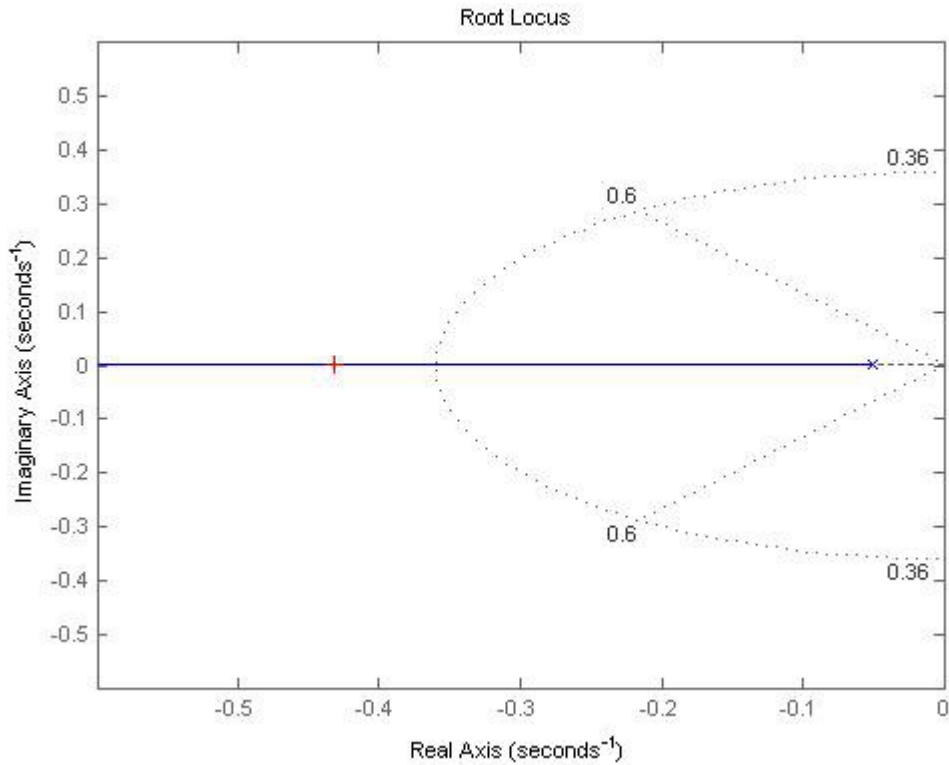


Figura 3.2: Scelta di un punto del luogo corrispondente a un polo della FDT

MatLab restituisce il seguente output:

```
selected_point = -0.4315 + 0.0019i
Kp = 381.5211
poles = -0.4315
```

Il valore di K_p ottenuto, se inserito nella funzione di trasferimento in catena chiusa, da origine a un polo nel punto specificato.

Ricorrendo nuovamente ai comandi MatLab già visti, è possibile ottenere la corrispondente risposta al gradino in catena chiusa, come mostrato in Figura 3.3:

```
W=feedback(Kp*W_Function,1); %Creazione della FDT in retroazione con il Kp trovato
t=0:0.1:20;
step(r*W,t)
axis([0 20 0 12])
```

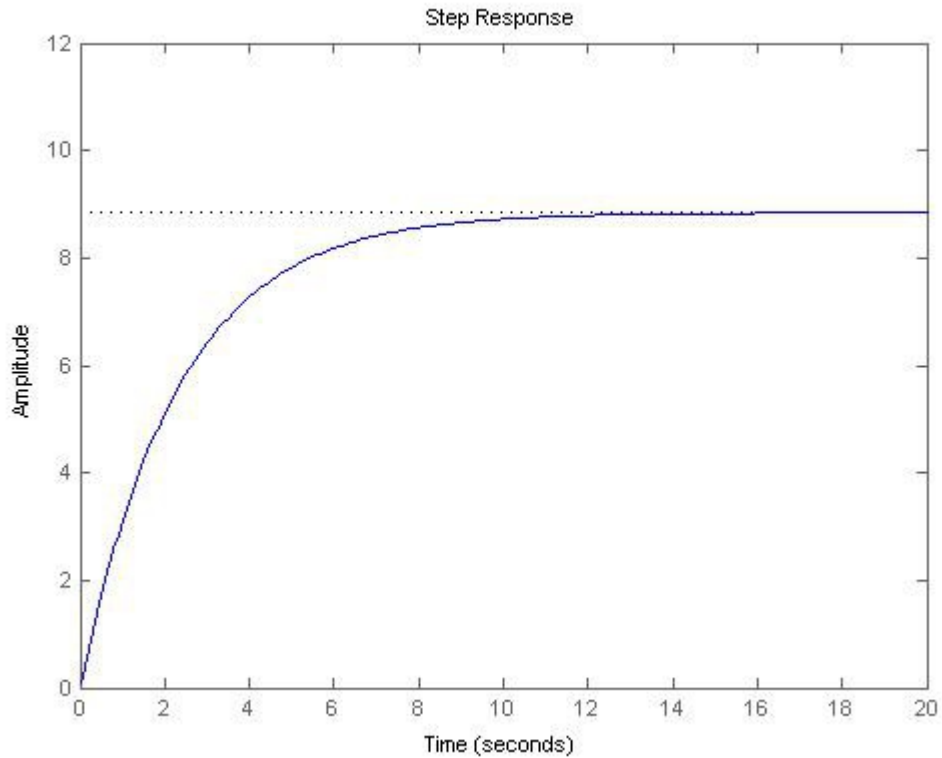


Figura 3.3: Risposta al gradino corrispondente al polo selezionato

Tramite il comando

$S = \text{stepinfo}(r*W)$

si ottengono i seguenti valori associati ai parametri della risposta:

RiseTime: 5.0921
SettlingTime: 9.0664
SettlingMin: 7.9621
SettlingMax: 8.8377
Overshoot: 0
Undershoot: 0
Peak: 8.8377
PeakTime: 18.1134

È possibile vedere come il vincolo sul tempo di salita non sia affatto rispettato, né tanto meno quello sull'errore a regime.

Si prosegue quindi decidendo di osservare come cambia la risposta del sistema al variare del punto selezionato nel luogo delle radici, per poi riportare in Figura 3.4 e in Figura 3.5 i risultati ottenuti per il vari casi.

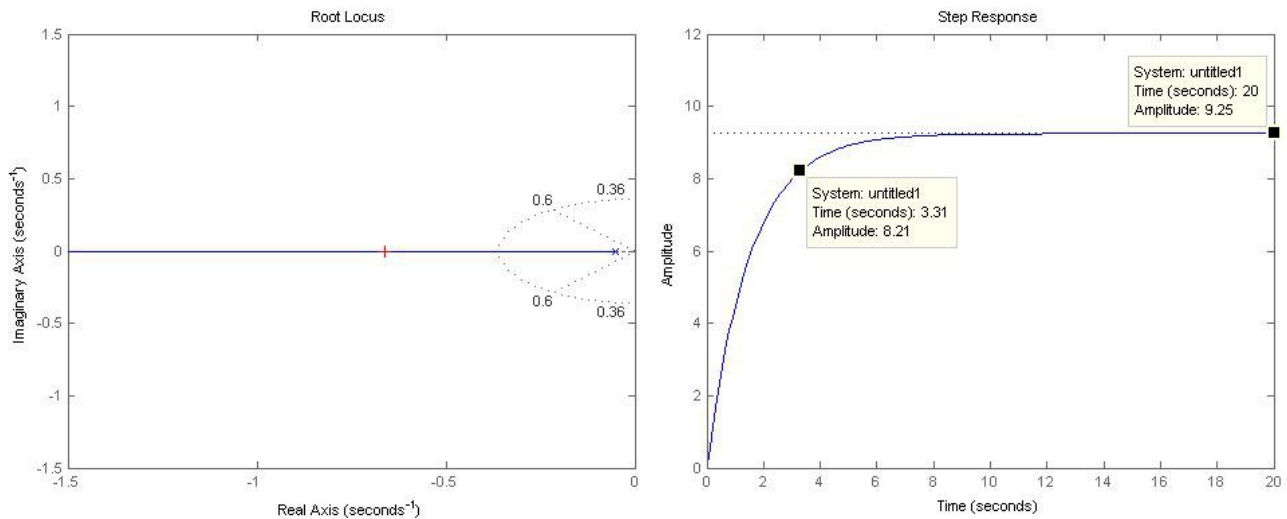


Figura 3.4: Luogo delle radici e corrispondente risposta al gradino per polo=-0.6629 e $K_p=612.9324$

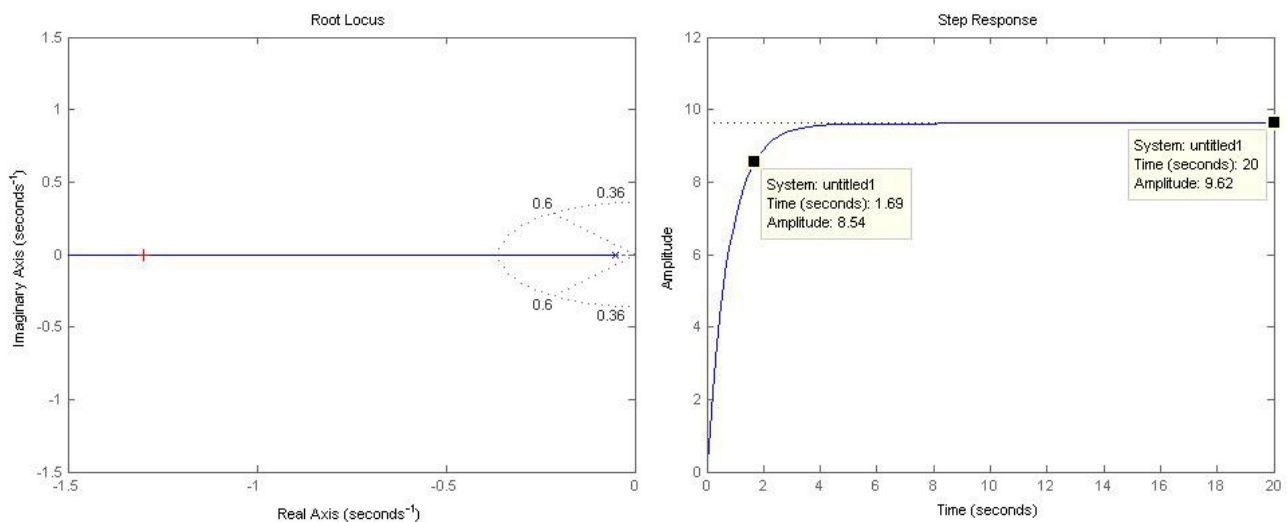


Figura 3.5: Luogo delle radici e corrispondente risposta al gradino per polo=-1.2992 e $K_p=1249.2$

Si constata come all'allontanarsi del polo dall'origine lungo l'asse reale negativo corrisponda un aumento del valore di K_p e una conseguente diminuzione del tempo di salita e dell'errore al gradino, come peraltro già osservato nella progettazione del controllore di tipo puramente proporzionale. Ciò comporta che le considerazioni fatte riguardo la sua effettiva realizzabilità valgono anche in questo caso.

3.1.3 Progetto della rete ritardatrice

Si procede quindi lo studio introducendo un diverso tipo di controllore chiamato rete ritardatrice, la cui funzione di trasferimento è data da:

$$C(s) = \frac{s + z_0}{s + p_0}$$

Trattandosi di una rete ritardatrice, si ha che lo zero z_0 ha pulsazione ω_z maggiore di quella corrispondente al polo p_0 , ovvero che $\omega_z > \omega_p$.

Facendo riferimento ai diagrammi di Bode della risposta complessiva, questo comporta che la

rete non influisca sul comportamento della risposta per frequenze più basse di quella del polo, inducendo invece un abbassamento del guadagno di 20dB/dec per valori compresi tra ω_p e ω_z . Per frequenze più elevate invece lo zero compensa l'effetto del polo e il guadagno torna ad essere costante. Per quanto riguarda la fase, si osserva come essa si abbassi per valori compresi tra ω_p e ω_z . La distanza tra le due pulsazioni risulta essere inoltre un parametro importante da determinare in base a quanto interessi far calare il guadagno (variando di conseguenza la pulsazione di attraversamento).

Si riporta in Figura 3.6 il diagramma di Bode e il luogo delle radici della rete per illustrare quanto appena affermato, avendo stabilito per questo esempio che $p_0=0.03$ e $z_0=0.3$.

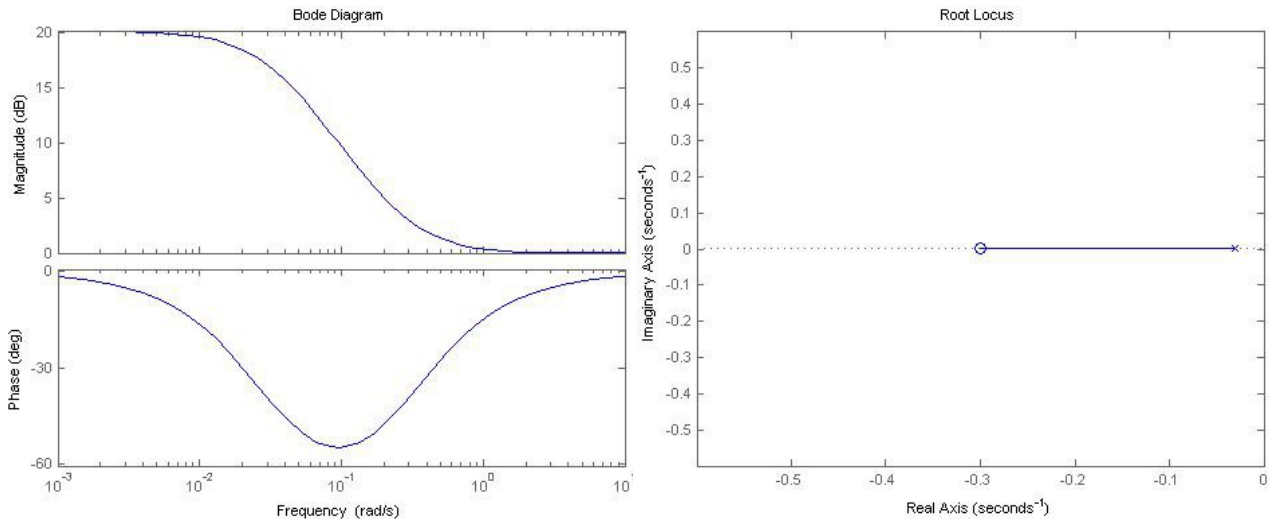


Figura 3.6: Diagramma di Bode e luogo delle radici della rete ritardatrice

Il codice MatLab usato per generare le due figure è dato da:

```
p0 = 0.03
z0 = 0.3
```

```
s = tf('s');
C_rit = (s+z0)/(s+p0);
```

```
bode(C_rit);%Comando che traccia i diagrammi di Bode
rlocus(C_rit) %Comando che traccia il luogo
axis([-0.6 0 -0.6 0.6]);
sgrid(0.6,0.36);%Comando che traccia i luoghi dove il fattore di smorzamento e le frequenza naturale sono
costanti
rlocus(C_rit) %Comando che traccia il luogo
axis([-0.6 0 -0.6 0.6]);
```

Si procede ora considerando il sistema in catena chiusa, comprensivo della cascata composta dal controllore proporzionale, dalla rete ritardatrice e dalla funzione di trasferimento del sistema in catena aperta.

Essa è data da:

$$W(s) = \frac{K_p s + K_p z_0}{ms^2 + (b + mp_0 + K_p)s + (bp_0 + K_p z_0)}$$

Passando al rispettivo codice MatLab e mantenendo gli stessi valori per il polo e per lo zero, si ha il seguente codice:

```
s = tf('s');
```

```

W_Function = 1/(m*s + b);
C_rit=(s+z0)/(s+p0); %Funzione di trasferimento della rete ritardatrice

rlocus(C_rit*W_Function); %Comando che traccia il luogo

axis([-0.6 0 -0.6 0.6]);
sgrid(0.6,0.36);

```

grazie al quale si ottiene il luogo delle radici per il sistema considerato riportato in Figura 3.7:

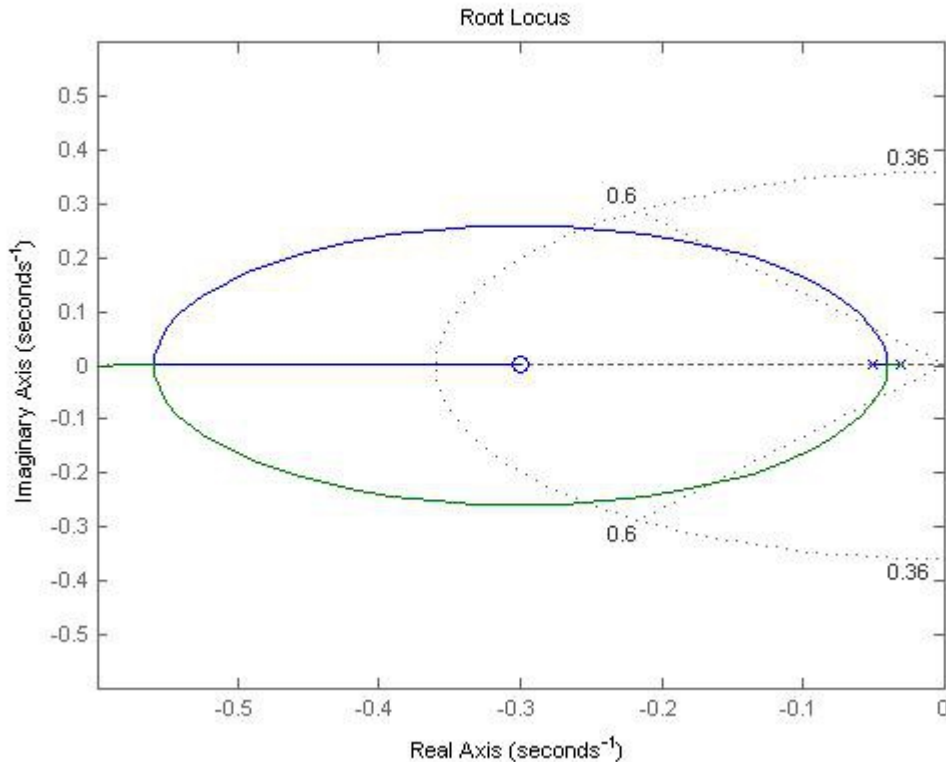


Figura 3.7: Luogo delle radici del sistema retroazionato con rete ritardatrice

Disponendo ora del luogo delle radici in catena chiusa, si procede come nel caso precedente andando a selezionare un suo punto per trovare il corrispondente valore di K_p ad esso associato.

Il comando MatLab

```
[Kp,poles]=rlocfind(C_rit*W_Function)
```

restituisce i seguenti valori:

```

selected_point = -0.3775 + 0.0019i
Kp = 1.4682e+03
poles = -1.1707    -0.3775

```

permettendo di tracciare la corrispondente risposta al gradino del sistema retroazionato, data dal seguente codice e rappresentata in Figura 3.8.

```

W_rit = feedback(Kp*C_rit*W_Function,1); %Funzione di trasferimento in retroazione
t=0:0.1:20;
step(r*W_rit,t) %Traccia la risposta al gradino
axis([0 20 0 12])

```

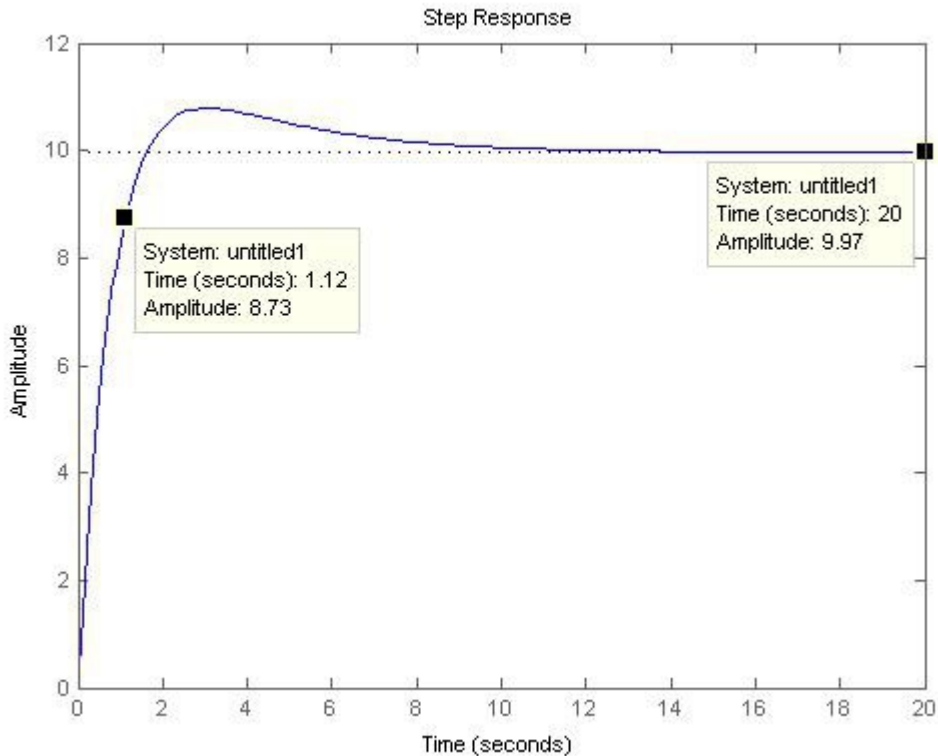


Figura 3.8: Risposta al gradino del sistema con rete ritardatrice

Si utilizza ora il seguente comando per ricavare i valori per i parametri della risposta:

```
S = stepinfo(r*W_rit)
```

```
RiseTime: 1.1179
SettlingTime: 7.7893
SettlingMin: 9.0337
SettlingMax: 10.7805
Overshoot: 8.1724
Undershoot: 0
Peak: 10.7805
PeakTime: 3.0471
```

Si può notare come le specifiche progettuali siano tutte rispettate: il tempo di salita è infatti di 1.1179 secondi (inferiore a 5 secondi), la sovraelongazione, seppur presente, è inferiore al 10% come richiesto e l'errore a regime è trascurabile.

3.1.4 Considerazioni finali sul luogo delle radici

Studiando il problema attraverso l'utilizzo del luogo delle radici si è avuto conferma di come un controllo di tipo prettamente proporzionale sia in grado di risolvere il problema solo da un punto di vista strettamente teorico, risultando di difficile realizzazione nonché di scarsa utilità applicativa.

Si è quindi deciso di introdurre in cascata una rete ritardatrice in modo da ridurre sensibilmente il tempo di salita e l'errore a regime della risposta senza dover incrementare eccessivamente il guadagno.

Ciò comporta la comparsa di una certa sovraelongazione, la quale però si mantiene entro il valore limite richiesto. Si è così arrivati ad ottenere una valida soluzione al problema, come nel caso dei controllori PI e PID.

3.2 Analisi in frequenza del controllore

3.2.1 Considerazioni preliminari

Questo studio è stato fin'ora orientato a soddisfare alcuni parametri progettuali riguardanti la risposta al gradino caratteristici del dominio del tempo; l'attenzione è stata in particolare rivolta al tempo di salita, alla sovraelongazione e all'errore a regime.

Lo studio del comportamento del sistema nel dominio del tempo non costituisce però l'unica strada percorribile: definendo alcuni parametri riferiti alla risposta in frequenza del sistema stesso è infatti possibile studiarlo nel dominio della frequenza, giungendo a conclusioni analoghe a quelle viste in precedenza.

La risposta in frequenza del sistema è innanzitutto definita dalla funzione:

$$G(j\omega): \mathbb{C} \rightarrow \mathbb{C} \text{ per } \omega \in [0, +\infty]$$

e si ottiene valutando la funzione di trasferimento del sistema in $s=j\omega$.

La risposta in frequenza individua i seguenti parametri, utili per definire in un secondo momento quelli che serviranno durante l'analisi:

- B_p = banda passante (a 3dB); è costituita dall'intervallo $[0, B_p]$ in cui si ha che:

$$|G(j\omega)_{dB}| \geq |G(0)_{dB}| - 3_{dB}$$

- ω_r = pulsazione di risonanza; se è presente essa è maggiore di zero e unica.
È definita come la pulsazione tale che:

$$|G(j\omega_r)| = \max\{|G(j\omega)|, \omega \in \mathbb{R}, \omega > 0\}$$

- M_{rel} = picco di risonanza; è definito come il rapporto (espresso in dB) tra il modulo della risposta in frequenza per $\omega=\omega_r$ e il modulo della stessa per $\omega=0$, espresso da:

$$M_{rel} = \frac{|G(j\omega_r)|}{|G(0)|}_{dB}$$

Lo studio della risposta in frequenza del sistema si rivela spesso molto utile, in quanto da esso è possibile dedurre la stabilità del sistema sia in catena aperta che in catena chiusa, nonché la sua sensibilità a variazioni parametriche e altre sue caratteristiche.

Per poter rappresentare tale funzione si ricorre di solito a due rappresentazioni grafiche: la prima è costituita dai diagrammi di Bode per ampiezza e fase, già utilizzati in precedenza nel corso del nostro studio; la seconda dal diagramma di Nyquist.

Il diagramma di Nyquist è costituito dalla rappresentazione nel piano complesso dei valori assunti dalla $G(j\omega)$ al variare della variabile ω reale.

Come già introdotto, per studiare la risposta in frequenza risulta utile definire alcuni parametri legati ad essa: la pulsazione di attraversamento ω_A , la fase di attraversamento φ_A e il margine di fase m_φ .

Essi sono definiti come di seguito:

- ω_A = pulsazione di attraversamento; se esiste è unica ed è definita come la pulsazione positiva per la quale il diagramma di Bode del modulo attraversa l'asse delle ascisse, ovvero:

$$|G(j\omega_A)_{dB}| = 0$$

- φ_A = fase di attraversamento; è definita come la fase corrispondente alla pulsazione di attraversamento, ovvero:

$$\varphi_A = \arg(G(j\omega_A))$$

- m_φ = margine di fase; è definita dalla relazione:

$$m_\varphi = \pi + \varphi_A$$

3.2.2 Diagramma di Bode e risposta in catena aperta

Si inizia ora l'analisi in frequenza partendo dalla risposta del sistema in catena aperta. La funzione di trasferimento del sistema è perciò data dalla già nota:

$$G(s) = \frac{1}{ms + b}$$

Come già visto il sistema ha un unico polo a parte reale negativa; di conseguenza c'è stabilità. La sua risposta al gradino è quindi quella già osservata durante l'analisi in catena aperta, ed effettuando la sostituzione $s = j\omega$ risulta possibile studiare la risposta in frequenza tramite i diagrammi di Bode.

Si riportano quindi in Figura 3.9 per completezza il grafico della risposta al gradino e i diagrammi di Bode in catena aperta, già discussi in precedenza. Nei diagrammi di Bode sono indicati rispettivamente la banda passante e la fase corrispondente.

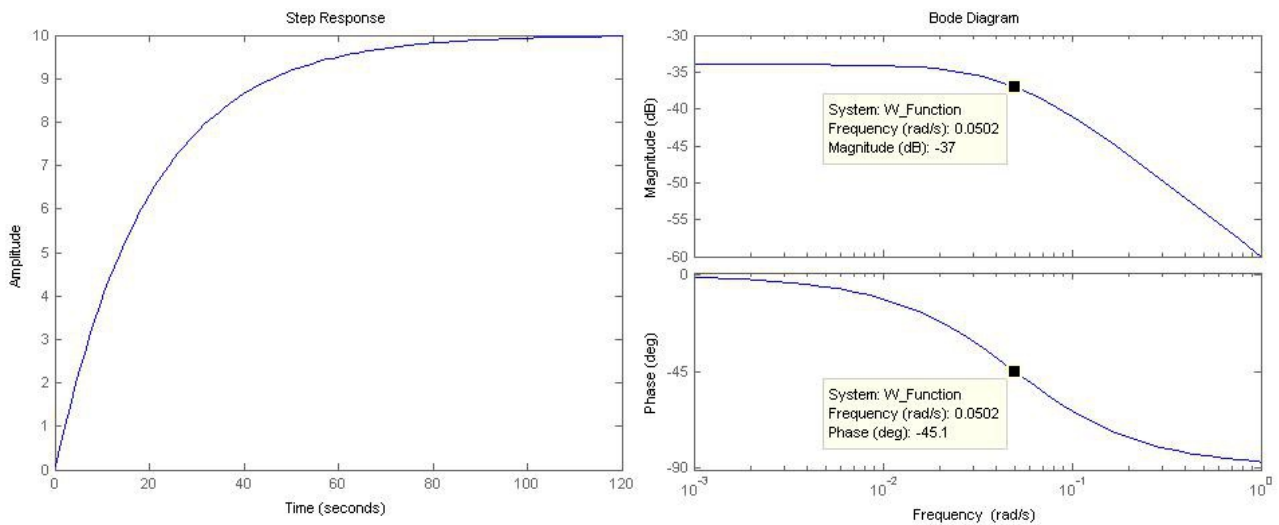


Figura 3.9: Risposta al gradino e diagramma di Bode del sistema in catena aperta

Come già concluso all'inizio di questo studio, la risposta in catena aperta, pur rispettando i vincoli di sovraelongazione ed errore a regime, risulta avere un tempo di salita molto maggiore di quello richiesto.

Si passa quindi a studiare la risposta in frequenza del sistema retroazionato.

3.2.3 Studio in catena chiusa e progetto di una rete ritardatrice

Considerando il sistema in retroazione unitaria negativa, la sua funzione di trasferimento è data da:

$$W_{pro}(s) = \frac{1}{ms + b + 1}$$

Dopo averla definita in MatLab seguendo le ormai note istruzioni, si ottiene in Figura 3.10 la corrispondente risposta al gradino. I rispettivi diagrammi di Bode sono gli stessi riportati in figura 3.9.

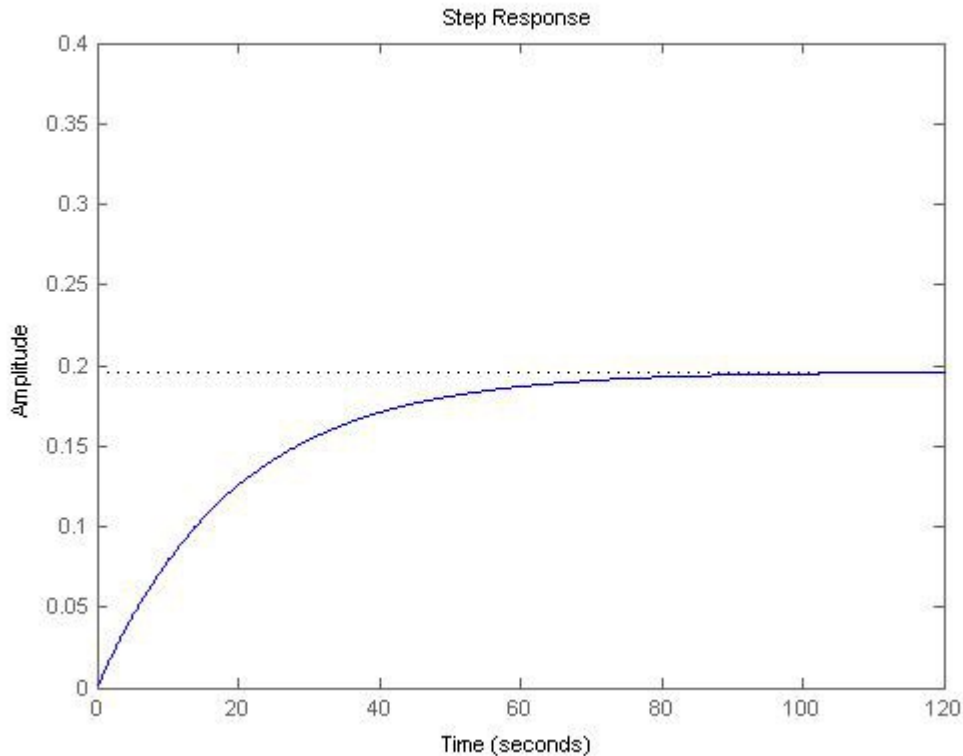


Figura 3.10: Risposta al gradino del sistema in retroazione unitaria negativa

Si evince subito che la risposta è lontanissima dal rispettare i requisiti di errore a regime e tempo di salita.

È necessario procedere al progetto di un controllore. Si tralascia lo studio di un controllore di tipo puramente proporzionale per non ripetere ulteriormente le stesse considerazioni fatte in precedenza. Si passa quindi direttamente al progetto di una rete ritardatrice che soddisfi le specifiche assegnate.

Come già visto nella sezione relativa allo studio tramite il luogo delle radici, l'introduzione di una rete ritardatrice aumenta il guadagno a basse frequenze diminuendo al tempo stesso la pulsazione di attraversamento, permettendo di ottenere una risposta non eccessivamente rapida. Questa rete inoltre non altera eccessivamente il margine di fase, lasciando così invariata la stabilità del sistema.

La funzione di trasferimento della rete è sempre data da:

$$C(s) = \frac{s + z_0}{s + p_0}$$

Attraverso il metodo per tentativi si stabilisce rispettivamente che $z_0=0.1$ e che $p_0=0.02$, in modo da avere una riduzione dell'errore a regime pari a $\frac{z_0}{p_0}=5$.

È inoltre necessario anteporre alla rete un blocco proporzionale per aumentare il guadagno in modo da diminuire il tempo di salita; questo aumento si manterrà però limitato senza creare i problemi di realizzazione visti per guadagni troppo elevati.

Posto $K_p=500$, dopo aver scritto il codice MatLab tramite i comandi:

```
bode(Kp*Fdt); %Comando che traccia il diagramma di Bode
margin(Kp*Fdt); %Comando che mette in evidenza il margine di fase
```

si ottiene il diagramma di Bode della funzione di trasferimento e la risposta al gradino del

sistema, riportati di seguito in Figura 3.13 e in Figura 3.14.

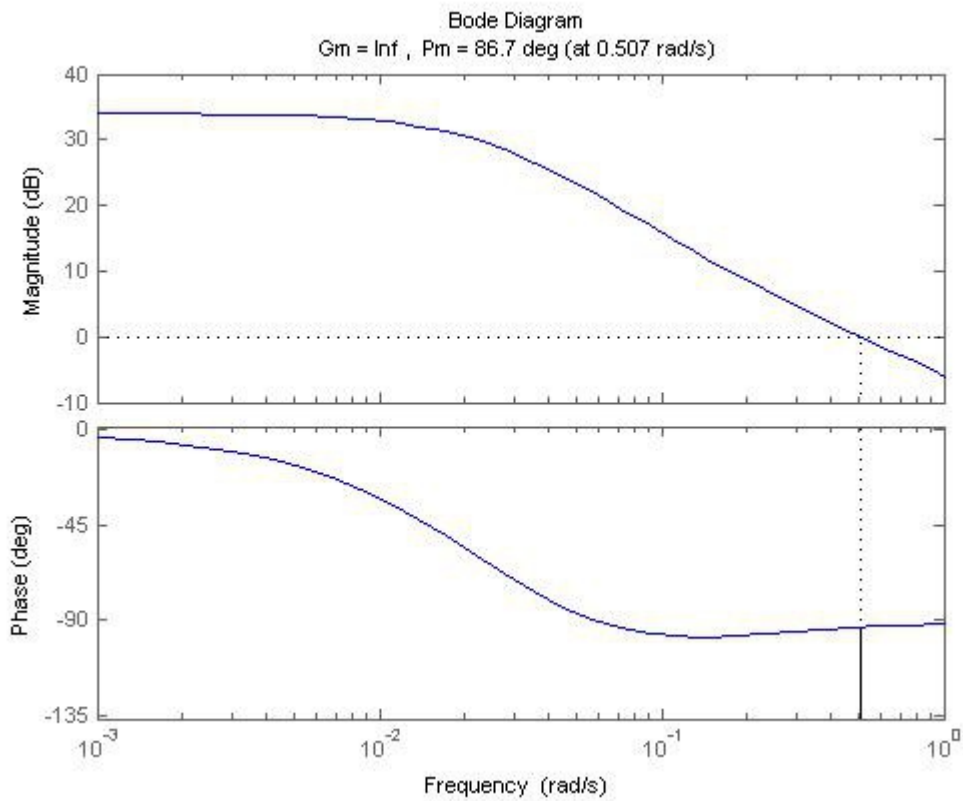


Figura 3.13: Diagramma di Bode del sistema in retroazione con rete ritardatrice

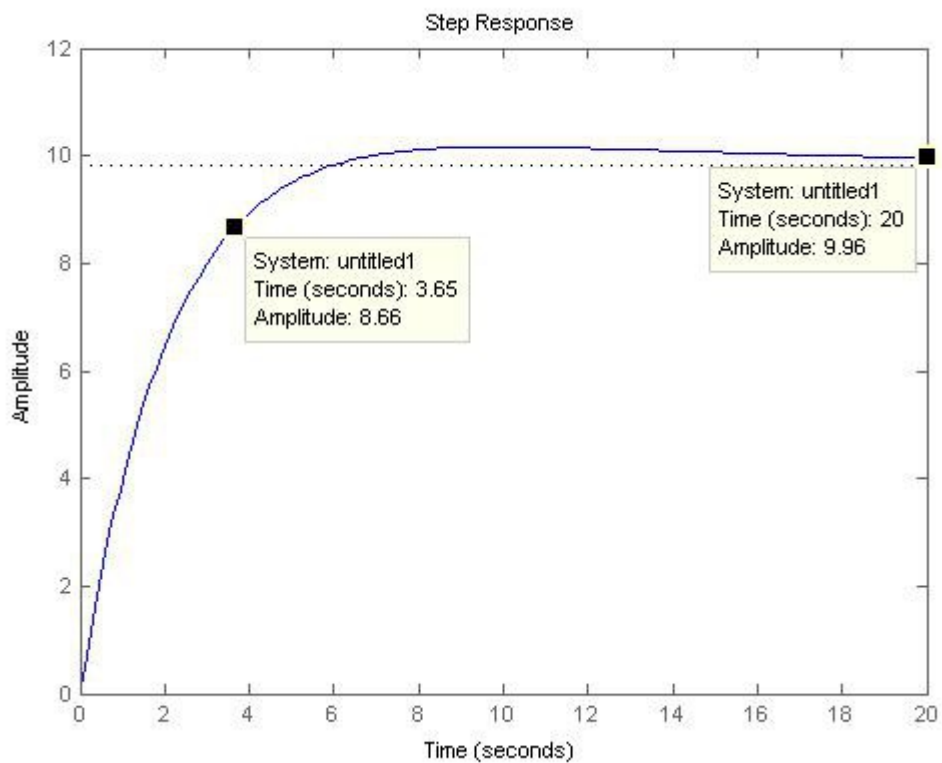


Figura 3.14: Risposta al gradino del sistema in retroazione con rete ritardatrice

Dal diagramma di Bode si nota facilmente come la frequenza di attraversamento sia diminuita, come il guadagno a basse frequenze sia pari a quello desiderato e come il margine di fase sia rimasto tale da non minacciare la stabilità del sistema.

La risposta al gradino conferma invece come le specifiche di sistema siano state tutte soddisfatte: il tempo di salita è di circa 3.65 secondi, la sovraelongazione è minore del 10% e l'errore a regime è trascurabile.

3.2.4 Considerazioni finali sull'analisi in frequenza

Studiando la risposta in frequenza del sistema si è avuto conferma di quanto già visto in precedenza durante il progetto dei controllori di tipo PID e durante l'analisi mediante il luogo delle radici. Per poter rispettare le specifiche richieste, infatti, si è dovuto far ricorso all'introduzione di una rete ritardatrice, in quanto un controllo di tipo puramente proporzionale, pur funzionando da un punto di vista teorico, richiede valori di K_p troppo elevati.

Grazie alla rete ritardatrice la risposta al gradino ottenuta rispetta tutti i requisiti; dall'analisi della risposta in frequenza si ha inoltre riscontro di come sia stato mantenuto un buon margine di stabilità.

4 Analisi del modello in Simulink

4.1 Modello e simulazione in catena aperta

4.1.1 Considerazioni preliminari

Al fine di comprendere meglio il funzionamento del sistema da un punto di vista grafico, si procede nello studio facendo ricorso a Simulink.

Questo programma permette di rappresentare il sistema con un uno schema a blocchi collegati tra loro (ingressi di segnali, funzioni di trasferimento, guadagni, sommatore, integratori, derivatori, blocchi di visualizzazione) e di simularne il funzionamento su un determinato intervallo temporale.

Nell'analisi seguente si suppone di disporre dello stesso sistema fisico trattato all'inizio di questa tesi e di dover soddisfare le stesse specifiche di progetto.

Per brevità sono nuovamente riportate solo la Tabella 1.1 dei parametri di sistema e tali specifiche, omettendo le equazioni di modello supposte già note.

Tabella 1.1: Valori dei parametri di sistema

	Variabile	Valore numerico
Massa della vettura	m	1000 [kg]
Coefficiente d'attrito	b	50 [Ns/m]
Forza motrice	u	500 [N]
Velocità di riferimento	r	10 [m/s]

Specifiche di progetto:

- tempo di salita < 5 s
- sovraelongazione < 10%
- errore a regime < 2%

4.1.2 Creazione del modello in Simulink

Si incomincia ora a costruire il modello in Simulink definendo le varie parti dello schema a blocchi.

Si ricorda che si dispone di un ingresso dato dalla forza motrice e che l'uscita è costituita da una velocità; il sistema è descritto dalle seguenti equazioni:

$$F = ma = m \dot{v} = u - bv \Rightarrow \dot{v} = \frac{1}{m}(u - bv)$$

$$v = \int \frac{dv}{dt} dt$$

Per modellizzare il sistema in Simulink si fa quindi ricorso ad un ingresso a gradino per rappresentare la forza motrice. Questo ingresso viene poi diviso da un blocco di guadagno rappresentante la massa della vettura, fornendo così la sua accelerazione. Il blocco successivo è costituito da un integratore che ci dà in uscita la velocità: in cascata si inserisce poi un blocco visualizzatore per poter studiare la risposta al gradino del sistema. Si aggiunge inoltre un altro blocco di guadagno che preleva la velocità in uscita e la moltiplicandola per il coefficiente d'attrito; la forza d'attrito così ottenuta viene riportata in ingresso e sottratta a quella motrice. Il risultato di questo procedimento è riportato Figura 4.1, rappresentante il sistema in catena aperta.

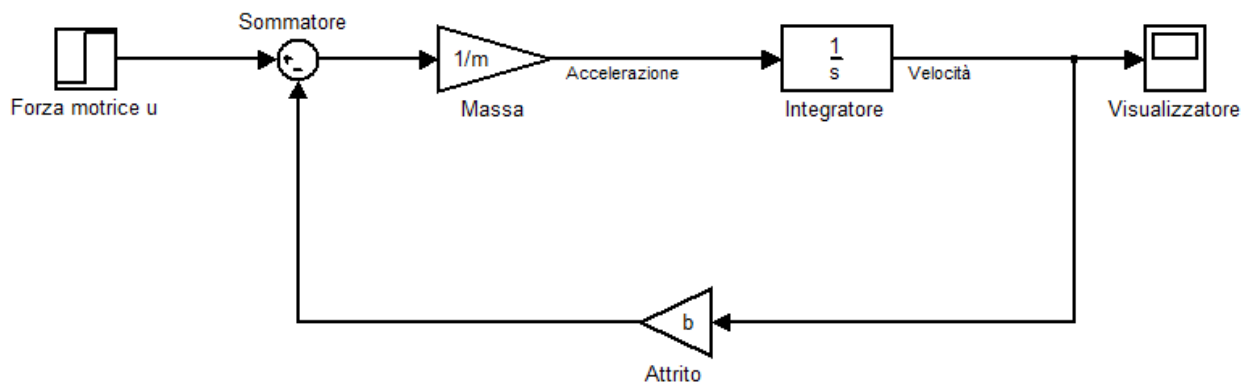


Figura 4.1: Schema a blocchi in Simulink del sistema in catena aperta

4.1.3 Simulazione in catena aperta

Avendo ultimato lo schema rappresentante il sistema, prima di iniziare la simulazione è necessario assegnare i valori numerici prestabiliti ai parametri utilizzati dai blocchi e stabilire l'intervallo di tempo durante il quale si farà girare la simulazione stessa.

Pertanto si definiscono in MatLab le variabili corrispondenti ai parametri di sistema e si imposta per questa prima simulazione un intervallo di 120 secondi.

Facendo infine partire la simulazione e aprendo la finestra corrispondente al blocco visualizzatore, si ottiene l'andamento riportato in Figura 4.2 per la velocità del sistema, corrispondente alla sua risposta al gradino in catena aperta.

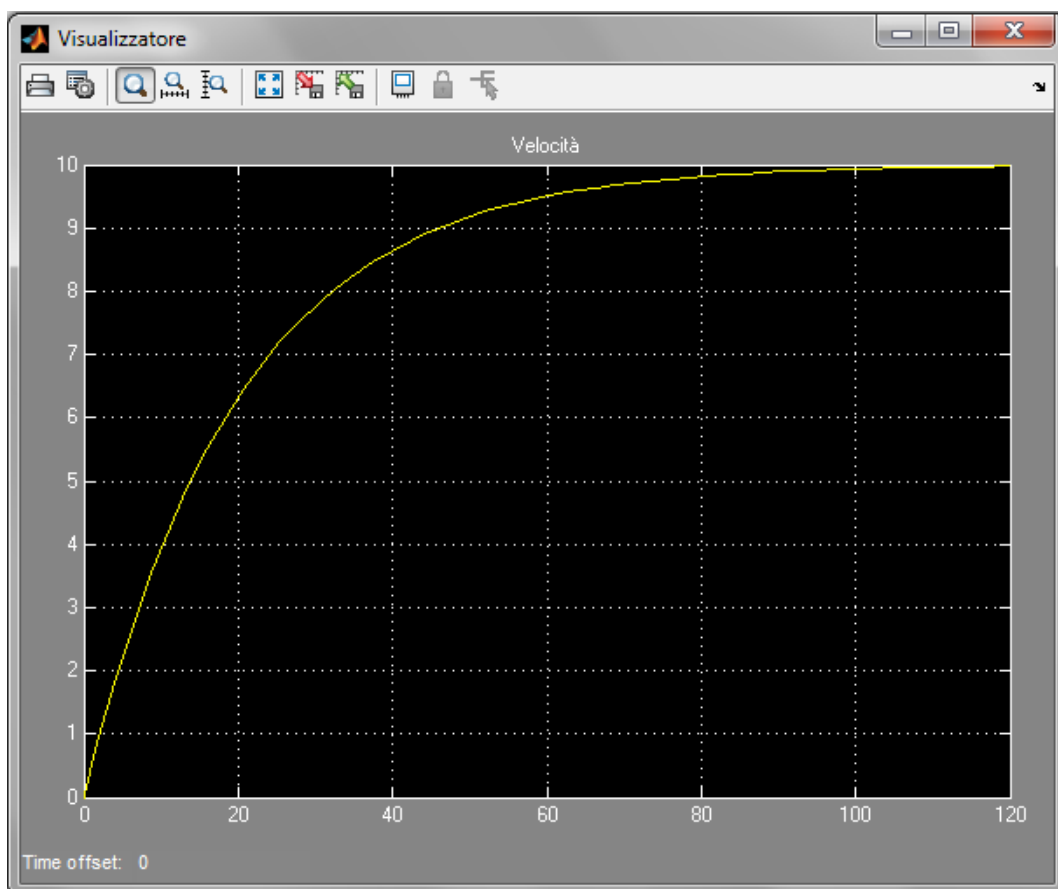


Figura 4.2: Risposta al gradino del sistema in catena aperta in Simulink

Osservando la figura si ha conferma dei risultati visti in precedenza per l'analisi in catena aperta. Sebbene la risposta non presenti sovraelongazione e l'errore a regime sia nullo, il tempo di salita risulta essere decisamente superiore ai 5 secondi richiesti, rendendo così necessaria l'analisi in catena chiusa previa l'introduzione di un controllore.

4.2 Controllo e simulazione in retroazione

4.2.1 Considerazioni preliminari

Avendo già studiato diverse soluzioni di controllo in ambiente MatLab, in questo capitolo della tesi non si ripercorreranno tutti i passi svolti; ci si limiterà piuttosto a realizzare in Simulink solamente le soluzioni funzionanti. Per questo motivo viene omesso lo studio del controllore puramente proporzionale, riportando invece le risposte al gradino del sistema corrispondenti ai valori ottimali già individuati per i parametri dei controllori.

Lo schema a blocchi rappresentante il sistema in catena aperta appena visto, ad esclusione del blocco di ingresso e del blocco visualizzatore, viene inoltre racchiuso in un unico blocco di sottosistema per semplicità di visualizzazione.

Si procede quindi analizzando la risposte ottenute rispettivamente con l'introduzione di un controllore di tipo PI, di un controllore di tipo PID e di una rete ritardatrice (in quest'ultimo caso con i valori individuati durante l'analisi in frequenza del sistema).

Per le successive simulazioni si imposta un intervallo temporale di 20 secondi e un ingresso di riferimento di 10 m/s.

4.2.2 Controllore Proporzionale-Integrativo

Per poter studiare la risposta in retroazione ottenuta introducendo un controllore di tipo PI, si deve innanzitutto andare a modificare il modello in Simulink.

Scegliendo di rappresentare il controllore attraverso un blocco rappresentante la sua funzione di trasferimento, data da:

$$C(s) = K_p + \frac{K_i}{s}$$

Si antepone tale blocco ad un blocco riassuntivo di tutto nostro sottosistema in catena aperta. Successivamente si introduce un ingresso a gradino rappresentante la velocità di riferimento, al quale si sottrae la velocità ottenuta in uscita dal blocco di sottosistema utilizzando un blocco sommatore. Infine si inserisce un blocco di visualizzazione per osservare la velocità in uscita dal sistema.

Lo schema a blocchi così ottenuto è riportato in Figura 4.3.

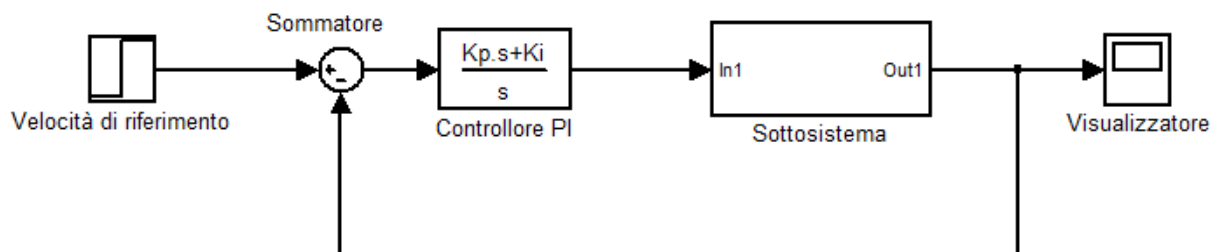


Figura 4.3: Schema a blocchi in Simulink del sistema in retroazione con controllore PI

Avendo terminato la costruzione del modello, prima di passare alla simulazione è necessario attribuire tramite MatLab i valori ottimali individuati alle costanti del controllore; in questo caso si avrà $K_p=1000$ e $K_i=50$.

Si fa quindi girare la simulazione commentandone il grafico della risposta al gradino riportato in Figura 4.4, ottenuto aprendo la finestra del blocco di visualizzazione.

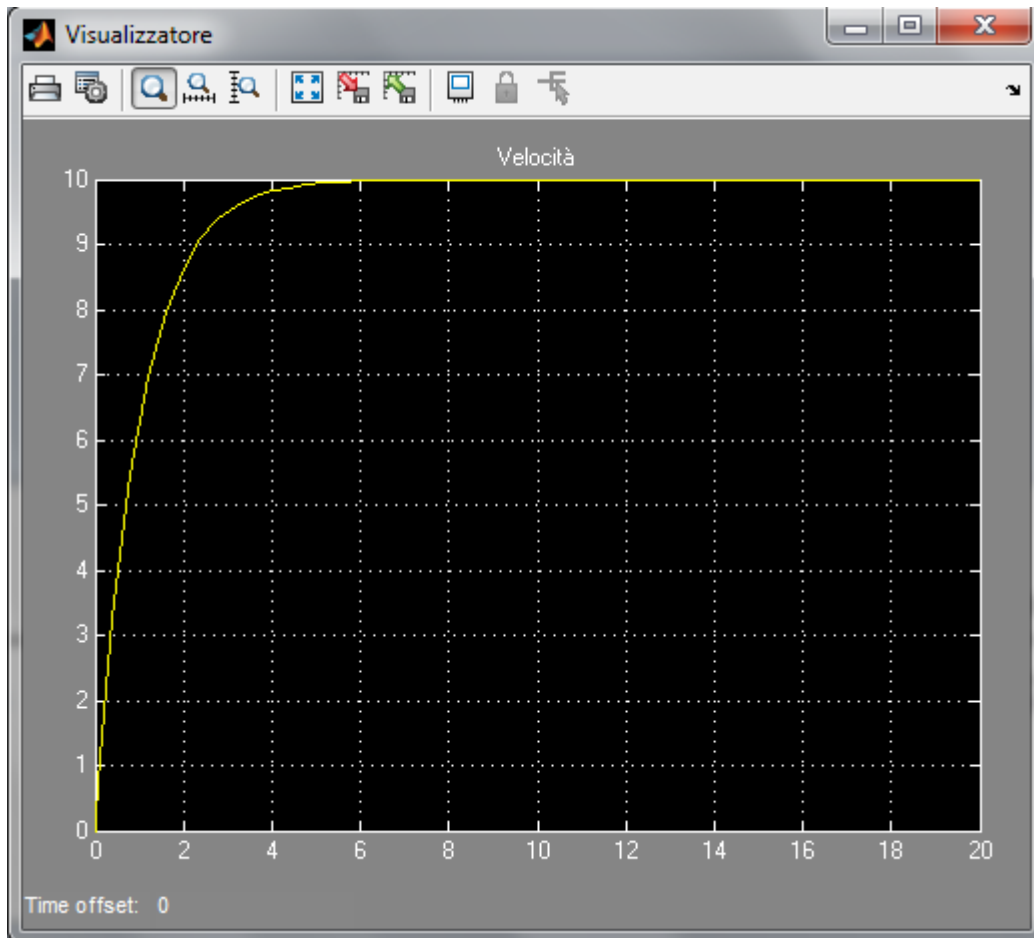


Figura 4.4: Risposta al gradino del sistema in retroazione con controllore PI in Simulink

Anche in questo caso il risultato è in accordo con quello ottenuto nell'analisi mediante MatLab. La risposta non presenta sovraelongazione, ha errore a regime nullo e un tempo di salita sicuramente inferiore a 5 secondi.

Oltre a constatare che i vincoli progettuali imposti sono tutti soddisfatti si può anche concludere che la risposta non presenta un'accelerazione iniziale troppo elevata, la quale comprometterebbe la realizzazione pratica di questa soluzione.

4.2.3 Controllore Proporzionale-Integrativo-Derivativo

Si prosegue l'analisi trattando il caso relativo all'introduzione di un controllore di tipo PID, la cui funzione di trasferimento è impropria e data da:

$$C(s) = K_p + \frac{K_i}{s} + K_d s$$

Il blocco di funzione di trasferimento di Simulink non permette però di definire una funzione di trasferimento impropria. Per rappresentare il controllore PID si fa quindi ricorso al corrispondente blocco PID della libreria del programma.

Il nuovo modello Simulink è riportato di seguito in Figura 4.5.

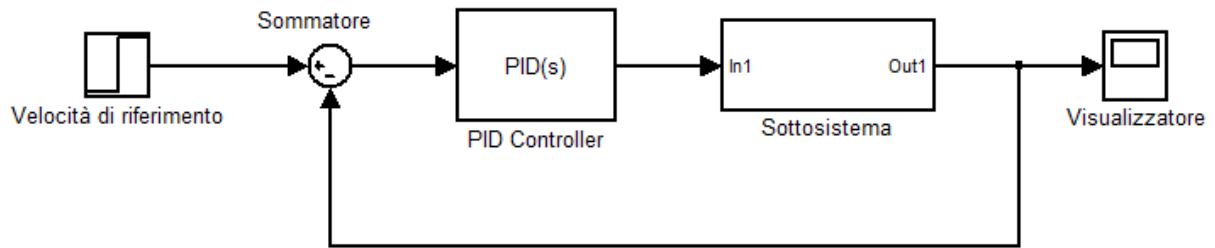


Figura 4.5: Schema a blocchi in Simulink del sistema in retroazione con controllore PID

Come già fatto per il caso precedente, si attribuiscono in MatLab i valori ottimali corrispondenti alle tre costanti utilizzate dal blocco PID; in questo caso si avrà $K_p=1000$, $K_i=50$ e $K_d=30$. Si passa ora alla simulazione per poi studiare la risposta al gradino ottenuta in figura 4.6.

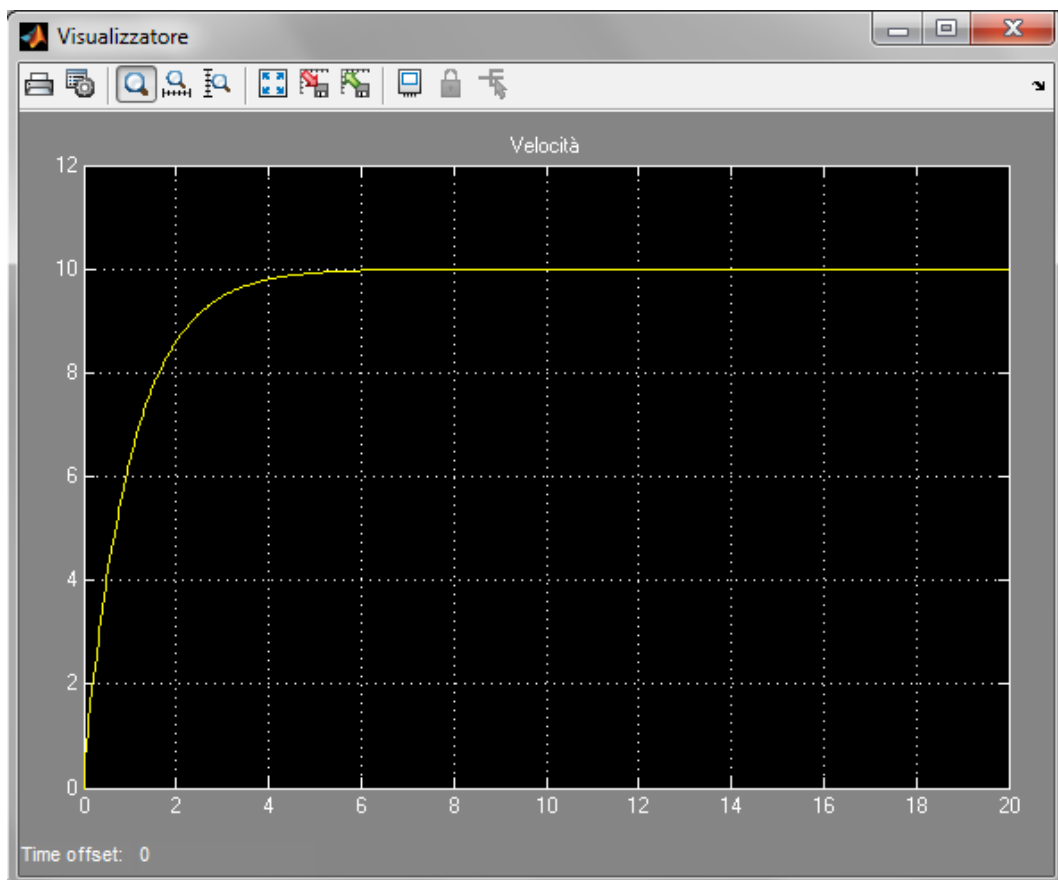


Figura 4.6: Risposta al gradino del sistema in retroazione con controllore PID in Simulink

Si osserva come essa sia molto simile a quella ottenuta grazie all'introduzione del controllore di tipo PI. Di conseguenza anche in questo caso valgono le considerazioni fatte in precedenza; le specifiche di sistema sono tutte soddisfatte e tale soluzione risulta essere inoltre anche realizzabile nella pratica.

4.2.4 Rete Ritardatrice

Come ultimo caso si considera l'introduzione di una rete ritardatrice, la cui funzione di trasferimento, come già visto, è data da:

$$C(s) = \frac{s + z_0}{s + p_0}$$

Si modifica quindi ancora una volta lo schema a blocchi, reintroducendo il blocco di funzione di trasferimento corrispondente alla funzione della rete in questione. Lo schema a blocchi ottenuto è riportato in Figura 4.7.

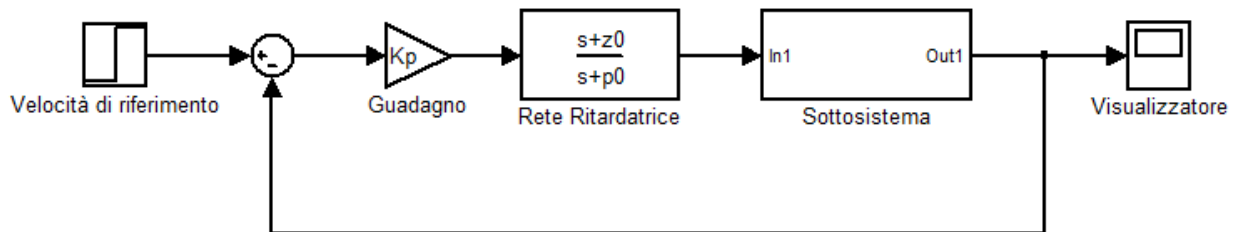


Figura 4.7: Schema a blocchi in Simulink del sistema in retroazione con rete ritardatrice

Anche in quest'ultimo caso si attribuiscono in MatLab i valori individuati per lo zero e per il polo durante lo studio in frequenza del sistema e per la costante K_p ; si avrà $K_p=500$, $z_0 = 0.1$ e $p_0=0.02$.

Si fa girare la simulazione per l'intervallo temporale stabilito per poi studiare la risposta del sistema, osservabile in Figura 4.8:

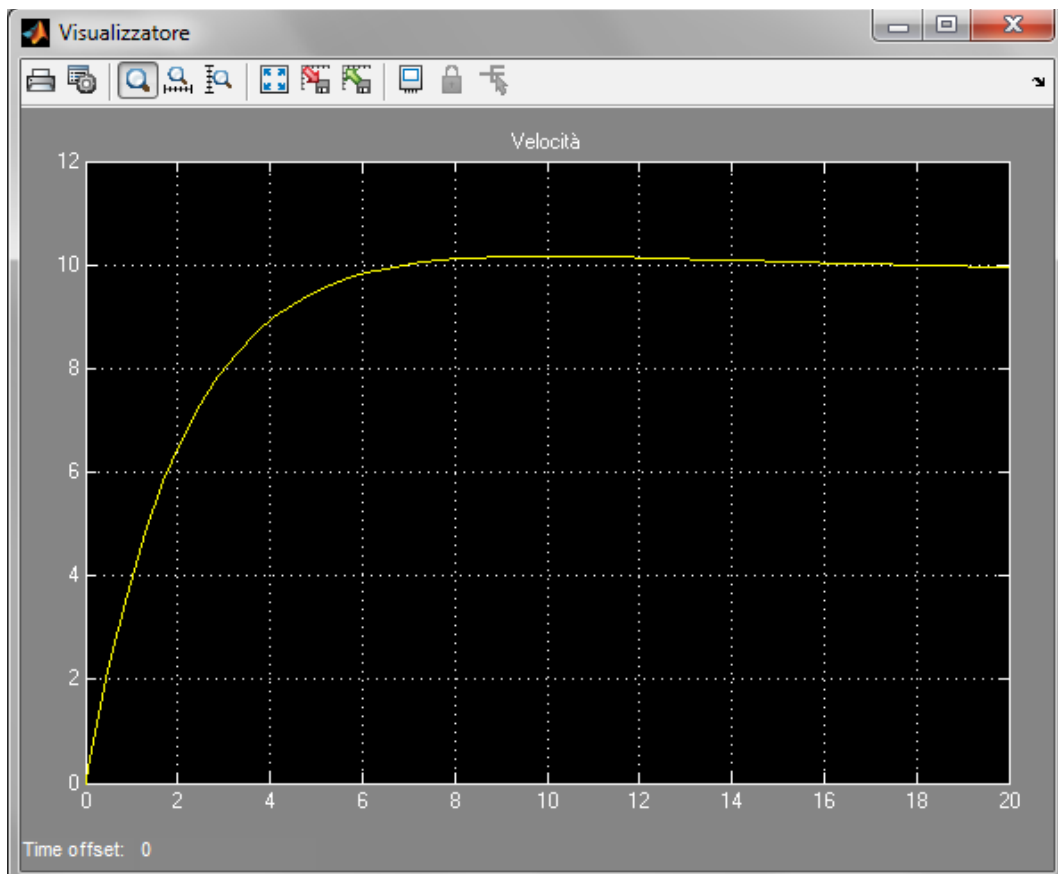


Figura 4.8: Risposta del sistema in retroazione con rete anticipatrice in Simulink

Analizzando la figura si nota come sia presente una lieve sovralongazione; ciò concorda appieno con quanto osservato durante l'analisi in frequenza effettuata in MatLab. Tale sovralongazione non costituisce affatto un problema, in quanto risulta essere

chiaramente inferiore al limite progettuale del 10%. Il tempo di salita inferiore al vincolo dei 5 secondi e l'errore a regime è praticamente nullo comportano che tutte le specifiche progettuali sono soddisfatte. L'assenza di un'accelerazione iniziale troppo elevata non compromette inoltre la realizzazione pratica di tale soluzione, rendendola quindi una valida alternativa alle due soluzioni già presentate.

4.2.5 Considerazioni finali sulla simulazione tramite Simulink

Rianalizzando il problema di controllo in ambiente Simulink si è inizialmente simulato il comportamento del sistema in catena aperta, per poi passare alla simulazione in retroazione per i casi di maggior rilievo visti in precedenza.

Si è deciso di tralasciare lo studio relativo all'introduzione di un controllore puramente proporzionale, in quanto si era già visto come questa soluzione non sia tecnologicamente implementabile per elevati valori di K_p .

Sono invece stati presi in considerazione i casi relativi all'introduzione di un controllore di tipo PI, di un controllore di tipo PID e di una rete ritardatrice.

Tutte e tre queste soluzioni si sono rivelate valide a patto di utilizzare i valori ottimali per i parametri dei controllori trovati in precedenza nell'analisi in MatLab.

Rispetto alla risposta ottenuta con l'impiego di un controllore di tipo PI, quella relativa al controllore di tipo PID risulta essere più morbida, mentre quella relativa alla rete ritardatrice introduce una lieve sovraelongazione che si mantiene però entro i vincoli richiesti.

Conclusioni

In questa tesi si è realizzato lo studio di un modello di Cruise Control, affrontandolo in ambiente MatLab e Simulink.

Si è partiti dalla modellizzazione del sistema fisico di una vettura spinta da una forza motrice costante a cui si oppone una forza lineare di attrito proporzionale alla sua velocità.

Dopo aver individuato le equazioni di sistema, si sono attribuiti dei valori realistici ai suoi parametri fisici per poter proseguire l'analisi numerica in MatLab.

Si è imposto che la risposta al gradino del sistema dovesse rispettare tre specifiche progettuali: tempo di salita <5 secondi, errore al regime $<2\%$, sovraelongazione $<10\%$.

Si è quindi passati ad analizzare il sistema in MatLab incominciando dall'osservazione del suo comportamento in catena aperta.

Avendo verificato come le specifiche non fossero rispettate, si è proseguito studiando la risposta in catena chiusa attraverso diverse strategie, come l'introduzione di controllori PID e l'analisi mediante il luogo delle radici e della risposta in frequenza con successivo progetto di una rete ritardatrice.

Infine si è simulato il comportamento del sistema in ambiente Simulink, prendendo in considerazione il suo comportamento in catena aperta e in retroazione per le soluzioni di controllo rivelatesi più valide.

Confrontando i risultati ottenuti, si è visto come la risposta al gradino del sistema ottenuta grazie all'introduzione di un controllore di tipo P sia in grado di soddisfare tutti i requisiti solo per un valore della rispettiva costante proporzionale eccessivamente alto. Tale valore infatti darebbe origine ad un'accelerazione iniziale troppo elevata, tale da impedirne la realizzazione pratica e da risultare in ogni caso di scarsa utilità applicativa.

Introducendo un controllore di tipo PI, una volta individuati dei valori appropriati per le costanti del controllore, si è ottenuta una risposta meno impulsiva e tale da rispettare tutte le specifiche senza dare origini ai problemi legati al caso precedente.

La risposta relativa all'impiego di un controllore di tipo PID è simile a quella del controllore PI, risultando appena più dolce in salita. Per valori prossimi a quelli ottimali si può notare anche una lieve sovraelongazione, che si mantiene però entro i limiti prestabiliti.

Anche questa strategia non presenta un'accelerazione iniziale eccessivamente impulsiva.

Si è infine considerato il progetto di una rete ritardatrice; dall'analisi della risposta ottenuta si evince che anche in quest'ultimo caso tutti i requisiti progettuali siano rispettati. Si può notare come sia presente una piccola sovraelongazione dovuta all'introduzione dello zero; essa si mantiene però sempre al di sotto del 10% , non costituendo quindi alcun problema.

Dovendo stabilire quale strategia sia preferibile scegliere tra quelle proposte per risolvere questo problema di controllo si scarterà quindi immediatamente quella relativa all'utilizzo di un controllore di tipo puramente proporzionale. Le tre rimaste rimangono tutte percorribili, ma basandosi sulla semplicità di realizzazione e sulla robustezza del controllore si può osservare come il controllore di tipo proporzionale-integrativo non introduca mai alcuna sovraelongazione, dia errore a regime nullo e inoltre non ricorra all'introduzione di zeri nella funzione di trasferimento del sistema. Questo ultimo fatto può quindi costituire un vantaggio in termini di semplicità di implementazione ed affidabilità, rendendo la scelta di un controllore di tipo PI quella preferibile.