

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/Simulink**

Relatore

Professor Alessandro Beghi

Laureando

Gianluca Caracino

Anno Accademico 2012/2013



# Indice

<b>1</b>	<b>Analisi del sistema</b>	<b>1</b>
1.1	Configurazione fisica . . . . .	1
1.2	Equazioni del modello . . . . .	2
1.2.1	Parametri del sistema . . . . .	3
1.3	Funzione di trasferimento e risposta impulsiva . . . . .	4
1.3.1	Derivazione matematica . . . . .	4
1.3.2	Implementazione della funzione di trasferimento mediante <i>MatLab</i> . . . . .	5
1.4	Specifiche da rispettare . . . . .	5
<b>2</b>	<b>Analisi in catena aperta</b>	<b>7</b>
2.1	Rappresentazione della risposta al gradino . . . . .	7
2.2	Rappresentazione di zeri e poli . . . . .	8
2.3	Diagrammi di Bode . . . . .	9
<b>3</b>	<b>Progetto del controllore</b>	<b>10</b>
3.1	Considerazioni preliminari . . . . .	10
3.2	Controllori <b>PID</b> . . . . .	11
3.2.1	Tipologia di controllo proporzionale <b>P</b> . . . . .	12
3.2.2	Tipologia di controllo proporzionale-integrativa <b>PI</b> . . . . .	14
3.2.3	Tipologia di controllo proporzionale-integrativa-derivativa <b>PID</b> . . . . .	18
3.2.4	Osservazioni conclusive sui controllori . . . . .	20
3.3	Analisi mediante luogo delle radici . . . . .	21
3.3.1	Introduzione delle rete ritardatrice (Phase Lag) . . . . .	26
3.3.2	Osservazioni conclusive sul luogo delle radici . . . . .	30
3.4	Analisi in frequenza . . . . .	30
3.4.1	Considerazioni preliminari . . . . .	30
3.4.2	Considerazioni in catena aperta e in catena chiusa . . . . .	31
3.4.3	Introduzione delle rete ritardatrice . . . . .	36
3.4.4	Osservazioni conclusive sull'analisi in frequenza . . . . .	38
<b>4</b>	<b>Analisi del modello mediante Simulink</b>	<b>39</b>
4.1	Osservazioni preliminari . . . . .	39
4.2	Costruzione del modello . . . . .	39

4.3	Simulazione in catena aperta . . . . .	41
4.4	Simulazione in catena chiusa . . . . .	42
4.4.1	Controllore <b>PI</b> . . . . .	42
4.4.2	Introduzione della rete ritardatrice . . . . .	43
4.5	Osservazioni conclusive sulla simulazione . . . . .	45
<b>5</b>	<b>Conclusioni</b>	<b>46</b>

# Introduzione

Il sistema elettronico cruise control é un dispositivo presente in ormai quasi tutte le moderne autovetture, responsabile della regolazione automatica della velocità. Il pilota seleziona la velocità desiderata ed essa viene mantenuta, compatibilmente con le condizioni di assetto della vettura stessa. Sostanzialmente esistono due tipologie di tale dispositivo:

- cruise control;
- adaptive cruise control;

Il primo mantiene la velocità impostata dal conducente, il quale può scegliere di aumentare e diminuire la velocità impostata azionando due pulsanti. Inoltre il guidatore può, ad esempio se decide di sorpassare un altro veicolo, premere il pedale dell'acceleratore e aumentare la velocità che tornerà a quella impostata in precedenza solo quando si smetterà di accelerare.

Il secondo inoltre può diminuire la velocità impostata in caso in cui degli ostacoli si trovino di fronte alla vettura. Il conducente può impostare la distanza (che varia automaticamente al variare della velocità) da mantenere dalle vetture che lo precedono entro un certo margine di sicurezza. Questo sistema prevede la presenza di un radar che, di solito, é montato anteriormente al veicolo.

Tale dispositivo rappresenta una caratteristica di comodità, che si apprezza soprattutto nei lunghi viaggi. Inoltre esso ha la capacità di mantenere la velocità costante per risparmiare carburante.

L'obiettivo di questo elaborato é quello di studiare le principali caratteristiche di tale sistema, effettuando un'analisi che porti alla derivazione di un modello matematico in forma di sistema lineare, del quale si studieranno varie caratteristiche come l'analisi dinamica in catena aperta (risposta indiciale ed in frequenza), progetto del controllore mediante diversi approcci (PID, luogo delle radici, reti correttive) e analisi dinamica in catena chiusa (soddisfacimento delle specifiche di controllo); ogni analisi terminerà con una breve osservazione conclusiva volta a confrontare i vari risultati ottenuti evidenziandone pro e contro. Come strumento di studio e analisi si é largamente impiegato il software *Matlab*.

# Capitolo 1

## Analisi del sistema

### 1.1 Configurazione fisica

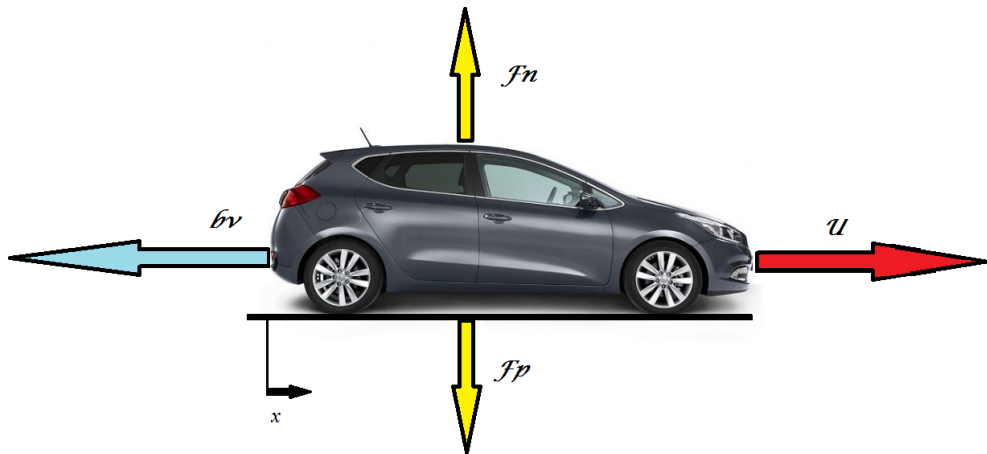


Figura 1.1: Modello grafico del sistema

Come spiegato nell'introduzione, l'intento principale del cruise control é quello di mantenere il veicolo ad una velocità costante indipendentemente da eventuali disturbi esterni come il vento oppure l'inclinazione della strada. Ciò avviene tramite una legge di controllo che regola automaticamente la quantità di carburante, sulla base di un confronto tra la velocità attuale del veicolo e quella di riferimento che si desidera ottenere.

L'immagine rappresenta un modello semplificato della dinamica dell'automobile in cui si può osservare la presenza della forza peso  $F_p = m \cdot g$  (con  $m$  che rappresenta la massa del veicolo e  $g$  accelerazione di gravità),

della reazione vincolare  $F_n = -F_p$ , della forza di controllo  $u$  e infine della forza di attrito  $F_a = b \cdot v$ .

La forza  $u$  é quella generata dal motore che si trasferisce sulla strada tramite gli pneumatici ed in grado di mettere in movimento il veicolo per il principio di rotolamento; in questo schema esemplificativo si assume di poter controllare direttamente tale forza, e di inglobare nella forza di attrito sia la componente dovuta all'attrito viscoso dell'aria che quella relativa alla resistenza dei pneumatici con l'asfalto. Queste ultime si oppongono al moto della macchina e sono proporzionali alla velocità del corpo tramite un opportuno coefficiente di attrito.

## 1.2 Equazioni del modello

Considerando il modello semplificato é possibile ricavarne le equazioni descrittive che rappresentano la modellizzazione matematica del nostro sistema. Esse si basano sulla seconda legge di Newton ( $F = m \cdot a$ ), secondo la quale l'interazione della vettura con la strada determina l'accelerazione della stessa ovvero la variazione della sua velocità nel tempo (con  $m$  che rappresenta la massa inerziale dell'automobile, ossia la resistenza a variare il suo stato di moto).

Ai fini del nostro progetto siamo interessati alla velocità assunta dall'auto per cui possiamo scrivere le equazioni del nostro sistema:

$$\begin{cases} m\dot{v} + bv = u \\ y = v \end{cases} \quad (1.1)$$

Ciò che si é ottenuto rappresenta un modello matematico in grado di descrivere il fenomeno in modo deterministico, dove l'evoluzione temporale della nostra variabile di interesse risulta univocamente individuata dal valore che essa assume in un opportuno istante (iniziale).

Esso descrive un fenomeno dinamico e a tempo continuo, in quanto le relazioni matematiche a cui fa riferimento sono proprio di tipo dinamico e la variabile temporale  $t$  é intesa come un numero reale.

In questo caso l'unica variabile di stato é rappresentata dalla velocità e grazie alla conoscenza delle sue condizioni iniziali, unitamente alla conoscenza dell'ingresso  $u$  applicato all'autovettura, permette la risoluzione delle equazioni differenziali che ne risultano.

Rimaneggiando le singole equazioni, ed essendo il sistema lineare, si può esplicitarlo in spazio di stato ottenendo:

$$\begin{cases} \dot{\mathbf{x}} = [\dot{v}] = \left[\frac{-b}{m}\right] [v] + \left[\frac{1}{m}\right] [u] \\ y = [1] [v] \end{cases} \quad (1.2)$$

Esso é composto dunque di un'equazione differenziale del primo ordine e di un'equazione di uscita (oltre che della condizione iniziale  $x(0)$ ) e rappresenta un caso particolare di quella piú generale di modello di stato lineare:

$$\begin{cases} \dot{\mathbf{x}}(t) = Fx(t) + Gu(t) \\ y(t) = Hx(t) + Ju(t) \end{cases} \quad (1.3)$$

dove, in generale,  $F$  é una matrice quadrata di dimensioni  $n \times n$  ( $n$  rappresenta il numero delle variabili di stato),  $G$  é una matrice di dimensioni  $n \times m$  ( $m$  rappresenta il numero di ingressi),  $H$  una matrice  $p \times n$  ( $p$  numero di uscite) e  $J$  una matrice  $p \times m$  (nel nostro caso nulla).

### 1.2.1 Parametri del sistema

Ora per proseguire lo studio é necessario fornire i valori ai nostri parametri incogniti in modo da poter determinare la funzione di trasferimento, a tal proposito consideriamo i seguenti dati:

<i>Parametri del sistema</i>		
<b>m</b>	<i>massa del veicolo</i>	1000 [kg]
<b>b</b>	<i>coefficiente di smorzamento</i>	50 [Ns/m]
<b>u</b>	<i>forza impressa al veicolo</i>	500 [N]
<b>r</b>	<i>velocitá di riferimento</i>	10 [m/s]



### 1.3 Funzione di trasferimento e risposta impulsiva

#### 1.3.1 Derivazione matematica

Considerando la soluzione dell'equazione differenziale del sistema (1.3), dopo alcuni passaggi matematici, si ottiene:

$$x(t) = \underbrace{e^{Ft}x(0)}_{\text{evoluzione libera}} + \underbrace{\int_0^t e^{F(t-\tau)} * Gu(\tau) d\tau}_{\text{evoluzione forzata}}$$

Sostituendo tale termine nell'evoluzione di uscita corrispondente alla seconda equazione del sistema (1.3) si può riscrivere:

$$y(t) = He^{Ft}x(0) + \int_0^t [He^{F(t-\tau)}G + J\delta(t-\tau)] * u(\tau) d\tau$$

dove l'ultimo integrale rappresenta la convoluzione tra la **risposta impulsiva**

$$w(t) = He^{Ft}G + J\delta(t). \quad (1.4)$$

e  $u(t)$ .

Considerando invece la trasformata di Laplace si ha che

$$\mathbf{Y}(s) = H\mathbf{X}(s) + J\mathbf{U}(s)$$

e ricordando che

$$\mathbf{X}(s) = (sI - F)^{-1}x(0) + (sI - F)^{-1}G\mathbf{U}(s)$$

si ottiene infine

$$\mathbf{Y}(s) = H(sI - F)^{-1}x(0) + [H(sI - F)^{-1}G + J]\mathbf{U}(s)$$

arrivando all'importante risultato:

$$\mathbf{W}(s) := H(sI - F)^{-1}G + J = \mathcal{L}[He^{Ft}G + J\delta(t)] = \mathcal{L}(w(t)) \quad (1.5)$$

che rappresenta la funzione di trasferimento del sistema.

Per ricavare la funzione di trasferimento del modello possiamo ricorrere più

semplicemente alla trasformata di Laplace dell'equazione differenziale ottenendo così, dopo alcuni semplici passaggi algebrici, dalla prima equazione del sistema (1.2):

$$\mathbf{U}(s) = ms\mathbf{V}(s) + b\mathbf{V}(s)$$

da cui risulta la fdt:

$$\mathbf{W}(s) = \frac{\mathbf{V}(s)}{\mathbf{U}(s)} = \frac{1}{ms + b} \quad (1.6)$$

### 1.3.2 Implementazione della funzione di trasferimento mediante *MatLab*

Il primo passo è rappresentato dalla definizione delle variabili necessarie al modello con i rispettivi valori assunti:

```
m=1000;           %Massa del veicolo
b=50;             %Coefficiente di attrito
u=500;           %Forza impressa
F=-b/m;          %Coefficiente dello spazio di stato
G=1/m;           %Coefficiente dello spazio di stato
H=1;             %Coefficiente dello spazio di stato
J=0;             %Coefficiente dello spazio di stato
```

Ora è possibile creare il nostro modello in spazio di stato mediante l'utilizzo del comando predefinito di *MatLab* `ss`, sarà così possibile introdurre la nostra funzione di trasferimento precedentemente ricavata (1.6) sulla base dei parametri qui introdotti.

```
space_state= ss(F,G,H,J);   %Definizione del modello in spazio di stato
s= tf('s');                %Variabile di Laplace
Fdt= 1/(m*s+b)              %Definizione della funzione di trasferimento
```

## 1.4 Specifiche da rispettare

Con l'obiettivo di realizzare una corretta simulazione è necessario introdurre alcune specifiche di progetto che devono essere soddisfatte dal nostro sistema. Abbiamo ipotizzato che il nostro veicolo, sotto la forza motrice di 500 N, raggiunga i 10 m/s (36 km/h) in un tempo inferiore ai 5 s e per tale applicazione risulta sufficiente considerare una sovraelongazione massima e

un errore a regime per la velocità rispettivamente del 10% e del 2%.  
Per cui le nostre specifiche richieste sono:

- Tempo di salita (Rise Time)  $< 5\text{s}$ ;
- Sovraelongazione massima (Overshoot)  $< 10\%$ ;
- Errore a regime (Steady-state error)  $< 2\%$ .

## Capitolo 2

# Analisi in catena aperta

### 2.1 Rappresentazione della risposta al gradino

In questo paragrafo ci proponiamo di visualizzare il comportamento del nostro sistema mediante la risposta forzata in corrispondenza al gradino di ampiezza 500 N. In un sistema a catena aperta se l'output non é quello desiderato, il sistema di controllo non può in alcun modo accorgersi dell'errore, ne tanto meno modificare l'input del sistema. Per cui in tali casi la garanzia del risultato é affidata alla precisione del modello matematico e all'assenza di eventuali disturbi. I comandi *MatLab* per la rappresentazione della risposta al gradino sono:

```
step(u*Fdt)           %Ampiezza del gradino data da u=500N
```

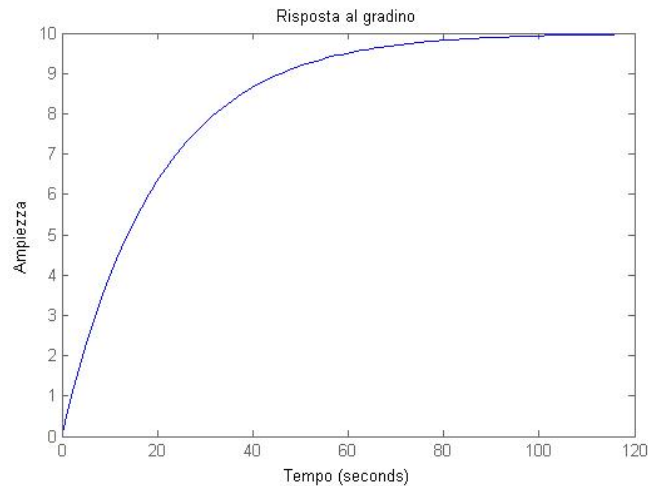


Figura 2.1: Risposta al gradino

Come é possibile osservare dalla Figura 2.1, il sistema raggiunge la velocità richiesta di 10m/s e le altre caratteristiche predominanti sono l'assenza di sovraelongazione ed un tempo di salita eccessivamente lungo rispet-

to alle specifiche richieste. Il tempo di salita  $t_r$  (al 10%) si può ottenere approssimativamente partendo dall'espressione della risposta al gradino  $w_{-1}(t) = W(0)[1 - e^{-pt}]\delta_{-1}(t)$  e risolvendo la seguente equazione

$$w_{-1}(t_r) = 0.9w_{+\infty} = 0.9W(0) \quad (2.1)$$

ovvero, in questo specifico esempio

$$t_r = \frac{1}{p} \ln\left(\frac{1}{0.1}\right) = \frac{\ln(10)}{p} = \frac{2.3}{p} = 46s \quad (2.2)$$

Si rende dunque necessario l'utilizzo di un controllo in retroazione al fine di ridurre sensibilmente il tempo di salita senza peggiorare le altre caratteristiche dinamiche del sistema.

## 2.2 Rappresentazione di zeri e poli

Il sistema cruise control possiede un unico polo posizionato nel punto  $-\frac{b}{m}$ , come si evince dall'equazione della funzione di trasferimento (1.6) e i comandi *Matlab* per la sua rappresentazione grafica sono :

```
pzmap(Fdt)           %istruzione che grafica zeri e poli
axis([-1 1 -1 1])    %range dell'asse reale e di quello immaginario
```

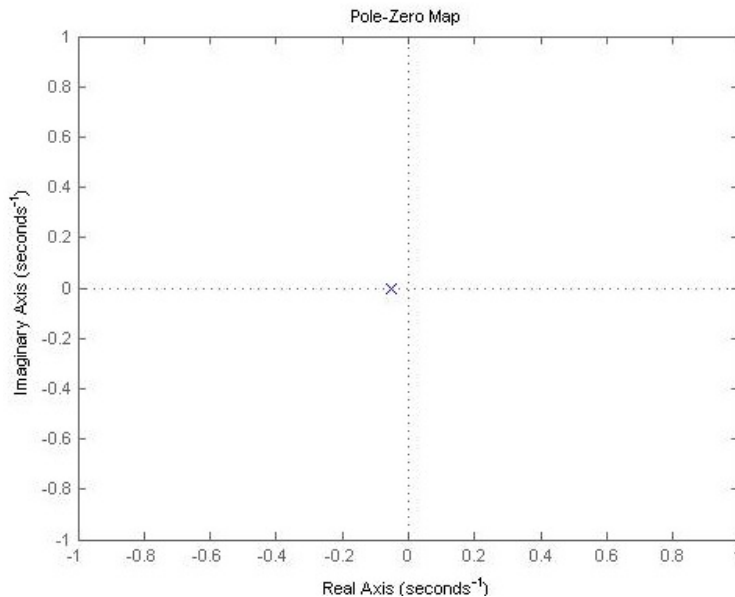


Figura 2.2: Rappresentazione di poli e zeri nel piano

Ciò che si osserva dalla Figura 2.2 é che essendo l'unico polo situato nel semipiano  $\Re(s) < 0$  siamo in presenza di *BIBO-stabilità*, ma essendo il suo valore molto piccolo (pari a -0,05) il nostro sistema risulta particolarmente lento ( $t_r \sim 46s$ ). Siccome non siamo in grado di poter modificare direttamente i parametri del sistema, siamo obbligati a progettare dei controllori in grado di alterare il posizionamento di zeri e poli al fine di raggiungere le performance richieste.

### 2.3 Diagrammi di Bode

Rappresentiamo ora l'andamento della risposta in frequenza mediante i diagrammi di Bode, i cui comandi identificativi in *MatLab* sono:

`bode(Fdt)`

Il diagramma di Bode della funzione di trasferimento risulta dunque:

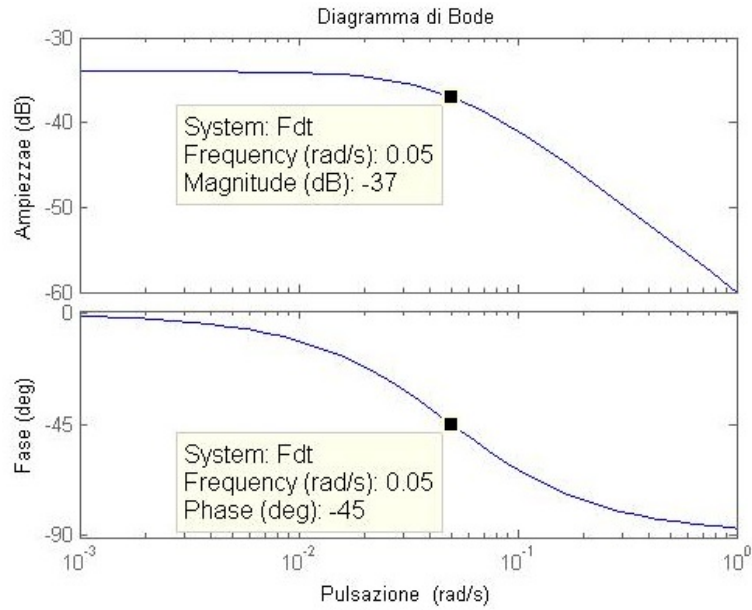


Figura 2.3: Diagramma di Bode della Fdt

Si osserva che in corrispondenza della pulsazione  $\omega = \frac{b}{m} = 5 \cdot 10^{-2}$  il modulo é calato di 3dB e la corrispondente fase é scesa a  $-45^\circ$ . A frequenze elevate il diagramma del modulo scende con una pendenza pari a -20dB/dec.

## Capitolo 3

# Progetto del controllore

### 3.1 Considerazioni preliminari

Considereremo ora il nostro sistema mediante retroazione unitaria, osservando che la retroazione da un lato diminuisce considerevolmente il tempo di salita, migliorando così le caratteristiche dinamiche del sistema, ma dall'altro introdurrà alcuni inconvenienti ragion per cui la soluzione migliore sarà un compromesso tra esigenze contrastanti. Lo schema di controllo a cui faremo riferimento è il seguente:

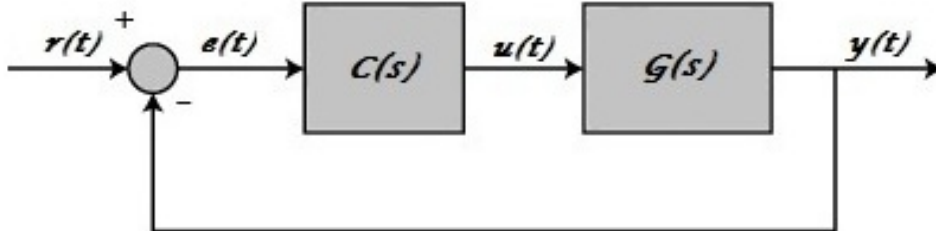


Figura 3.1: Schema di controllo in retroazione ad un grado di libertà

Al nostro processo noto rappresentato dal blocco  $G(s)$  (la nostra funzione di trasferimento in catena aperta 1.6) viene posto prima un ulteriore blocco chiamato compensatore  $C(s)$ , costituendo così un sistema ad un grado di libertà proprio perché  $C(s)$  rappresenta un sistema artificiale e libero del progettista affinché il sistema si comporti bene.

L'obiettivo naturale, legato a ragioni progettuali ed economiche, è quello di progettare compensatori  $C(s)$  che siano in grado di raggiungere gli obiettivi del controllo definiti a priori, e allo stesso tempo esibiscano una struttura che sia la più semplice possibile. L'introduzione di tale compensatore attribuisce allo schema di controllo a retroazione unitaria la funzione di trasferimento ad anello aperto  $W(s)=C(s)G(s)$  e la nuova fdt ad anello chiuso può essere

determinata con dei semplici conti.

Dove nello schema di figura (3.1):

- $r(t)$  rappresenta l'ingresso esterno di riferimento;
- $y(t)$  rappresenta il segnale di uscita;
- $e(t)$  rappresenta il segnale differenza  $r(t)-y(t)$  e costituisce l'errore che il sistema compie nel tentativo di far si che il segnale di uscita  $y(t)$  segua quello di riferimento  $r(t)$ ;

Al fine di determinare la funzione di trasferimento del sistema retroazionato, ricorriamo alla trasformata di Laplace.

$$E(s)=R(s)-Y(s) \quad (3.1)$$

e ricordando che

$$Y(s)=C(s)G(s)E(s) \quad (3.2)$$

si ottiene

$$Y(s)=C(s)G(s)(R(s)-Y(s)) \quad (3.3)$$

da cui segue immediatamente  $(1+C(s)G(s))Y(s)=C(s)G(s)R(s)$  e dunque la funzione di trasferimento del sistema retroazionato risulta

$$H(s) := \frac{C(s)G(s)}{1 + C(s)G(s)} \quad (3.4)$$

Vista la struttura del sistema di controllo, l'effetto di  $C(s)$  é essenzialmente quello di modificare il guadagno di Bode in catena aperta e di introdurre ulteriori poli e/o zeri che vanno ad aggiungersi a quelli del processo.

## 3.2 Controllori PID

Con il simbolo PID si é soliti indicare controllori che realizzano sul segnale al loro ingresso la combinazione lineare di tre azioni: un'azione proporzionale (P), un'azione integrativa (I) e un'azione derivativa (D) e vengono pertanto descritti dalla seguente funzione di trasferimento:

$$C_{PID}(s) = K_p + \frac{K_i}{s} + K_d s \quad (3.5)$$

con  $K_p, K_i, K_d$  reali, dove  $K_i$  serve per sistemare l'errore di regime permanente, con  $K_p$  e  $K_d$  é possibile introdurre due zeri reali negativi al fine di sistemare i problemi di pulsazione di attraversamento e margine di fase. Come casi particolari sono possibili controllori piú semplici quale il **PI** (per



$K_d=0$ ), il **PD** (per  $K_i=0$ ), il **P** (per  $K_i=K_d=0$ ) e l'**I** (per  $K_p=K_d=0$ ).

Un discorso a parte é la fisica realizzabilit  del controllore, in quanto non   possibile realizzare controllori a cui corrispondono funzioni di trasferimento non proprie (con pi  zeri che poli), nella pratica in una prima fase si trascura tale vincolo aggiungendo successivamente dei poli in alta frequenza in modo da rendere il controllore proprio.

I controllori PID possono essere implementati in codice *MatLab* secondo due metodologie differenti. La prima consiste nel definire direttamente la funzione di trasferimento del controllore:

```
Kp=1;           %Costante proporzionale
Ki=1;           %Costante integrale
Kd=1;           %Costante derivativa
s=tf('s');
C=Kp+\frac{Ki}{s}+Kd*s
```

Alternativamente   possibile utilizzare il comando integrato presente in *MatLab*:

```
C=pid(Kp,Ki,Kd)
```

### 3.2.1 Tipologia di controllo proporzionale P

Il nostro sistema ora pu  essere visto come in Figura (3.1) con il compensatore  $C(s)$  che esercita un controllo proporzionale. Il primo passo consiste nel determinare la nuova funzione di trasferimento del sistema retroazionato ottenuta tramite l'equazione (3.4) sostituendoci la fdt del controllore proporzionale (a sua volta ottenuta da (3.5) con  $K_i = 0$  e  $K_d = 0$ ).

$$H(s) = \frac{\frac{K_p}{ms+b}}{1 + \frac{K_p}{ms+b}} = \frac{\frac{K_p}{ms+b}}{\frac{ms+b+K_p}{ms+b}} = \frac{K_p}{ms + b + K_p} \quad (3.6)$$

Creiamo un m-file con il seguente codice *MatLab*:

```
m = 1000;
b = 50;
r = 10;                               %Ingresso di riferimento (m/s)

s = tf('s');
Fdt = 1/(m*s + b);

Kp = 100;
C = pid(Kp);                           %Funzione di trasferimento
```

```

%del PID
%ottenuta direttamente
%dal comando integrato

Fdt_catena_chiusa = feedback(C*Fdt,1) %Istruzione che
%permette la semplificazione
%dello schema a blocchi
%unendo le due fdt
%ed ottenendo quella
%in catena chiusa

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

```

La funzione di trasferimento del nostro sistema é dunque:

$$H(s) = \frac{100}{1000s + 150} \quad (3.7)$$

dove nel codice *MatLab*  $r$  rappresenta l'ingresso di riferimento mentre  $K_p$  la costante proporzionale scelta.

Rappresentando la risposta al gradino per tale funzione di trasferimento, si ottiene:

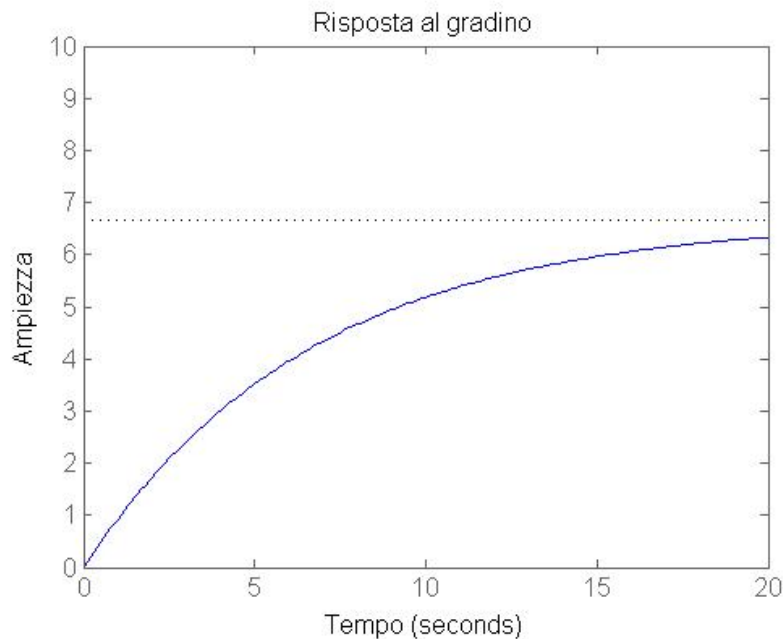


Figura 3.2: Risposta al gradino con  $K_p = 100$

Dalla figura (3.2) si osserva che l'introduzione del compensatore proporzionale con costante  $K_p = 100$  rende la risposta piú rapida rispetto alla figura

(2.1) in catena aperta, ma non risulta ad ogni modo soddisfacente per quanto riguarda le specifiche di *tempo di salita* ed *errore a regime*.

Possiamo però incrementare il guadagno proporzionale  $K_p$  in modo da ottenere un comportamento dinamico più vicino alle specifiche richieste, ponendo  $K_p = 5000$  si ottiene la seguente risposta:

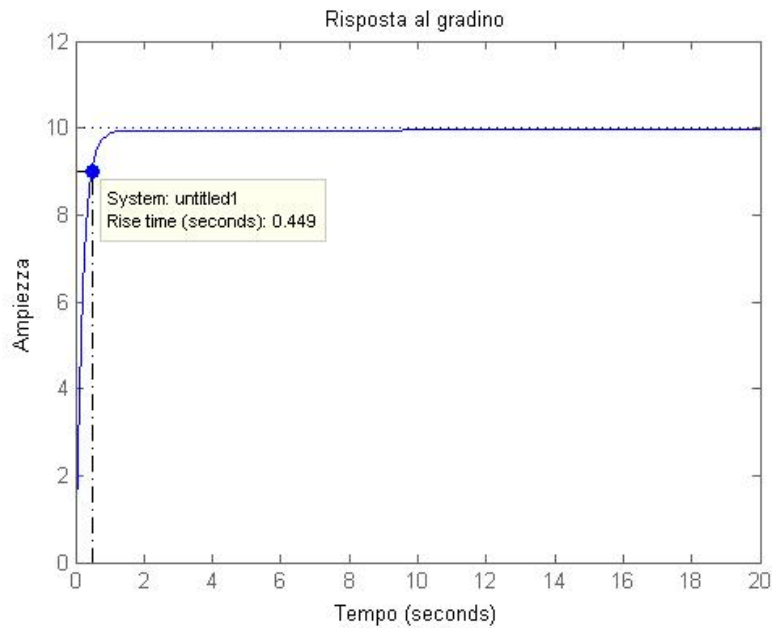


Figura 3.3: Risposta al gradino con  $K_p = 5000$

In questo particolare caso le specifiche richieste sarebbero perfettamente soddisfatte perché il nostro sistema presenta un *tempo di salita*  $t_p = 0,449s$  ed un *errore a regime*  $e_r \sim 0$ , ma riflettendo sui dati ottenuti significherebbe che il nostro veicolo dovrebbe passare dai  $0m/s$  ai  $9m/s$  (90% del valore di riferimento) in  $0,449s$ , il che è pressoché impossibile da realizzare. Ciò comporterebbe un'accelerazione spaventosa probabilmente non presente neanche nelle vetture da competizione.

### 3.2.2 Tipologia di controllo proporzionale-integrativa PI

Ripetiamo il procedimento svolto nel paragrafo precedente al fine di determinare la nuova funzione di trasferimento del sistema, dotato questa volta di un controllore **PI** (proporzionale integrativo). La Fdt del controllore **PI**

si ottiene dalla formulazione generica del **PID** (3.5) ponendo  $K_d = 0$ :

$$C(s) = K_p + \frac{K_i}{s} \quad (3.8)$$

Sostituendola nell'equazione (3.4) risulta:

$$H(s) = \frac{\frac{K_p s + K_i}{s(m s + b)}}{\frac{s(m s + b) + K_p s + K_i}{s(m s + b)}} = \frac{K_p s + K_i}{m s^2 + (b + K_p) s + K_i} \quad (3.9)$$

Rappresentiamo ora la risposta al gradino di tale funzione utilizzando come parametri  $K_p = 600$  e  $K_i = 1$  e sostituendoli nel precedente m-file si ottiene il seguente codice *MatLab* e la successiva rappresentazione grafica.

```

m = 1000;
b = 50;
r = 10;                                %Ingresso di riferimento (m/s)

s = tf('s');
Fdt = 1/(m*s + b);

Kp = 600;                               %Costante proporzionale
Ki=1;                                   %Costante integrativa
C = pid(Kp,Ki);                         %Funzione di trasferimento
                                        %del PID
                                        %ottenuta direttamente
                                        %dal comando integrato

Fdt_catena_chiusa = feedback(C*Fdt,1)    %Istruzione che
                                        %permette la semplificazione
                                        %dello schema a blocchi
                                        %unendo le due fdt
                                        %ed ottenendo quella
                                        %in catena chiusa

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

```

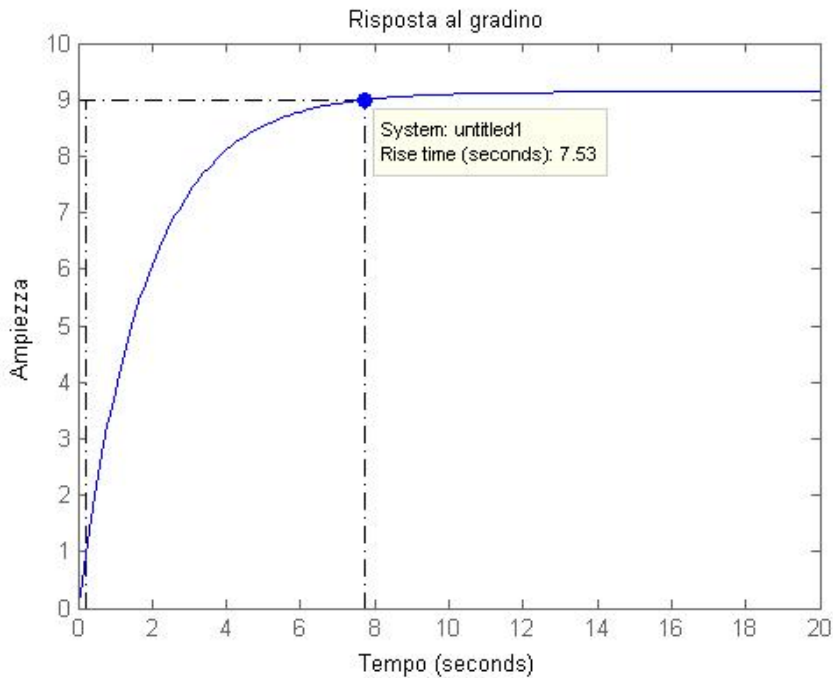


Figura 3.4: Risposta al gradino con  $K_p = 600$  e  $K_i = 1$

Osservando la figura soprastante si vede come le specifiche non siano minimamente soddisfatte, per cui è possibile modificare le varie costanti  $K_p$  e  $K_i$  nel nostro m-file, al fine di osservare un comportamento dinamico del sistema migliore. In figura (3.5) sono rappresentati gli andamenti della risposta al gradino al variare di tali parametri.

Modificandone i valori si possono notare dei cambiamenti nell'andamento della risposta; infatti aumentando il valore di  $K_p$  è possibile ridurre sensibilmente il *tempo di salita* stando comunque attenti che un valore troppo elevato potrebbe essere responsabile della irrealizzabilità del sistema (come precedentemente visto per il controllore  $\mathbf{P}$  con  $K_p = 5000$ ). Il valore della costante  $K_i$  è responsabile della sovralongazione, ponendo ad esempio  $K_i = 120$  come in figura 3.5 d) la sovralongazione supera il 10% non rispettando dunque le specifiche.

### CAPITOLO 3. PROGETTO DEL CONTROLLORE

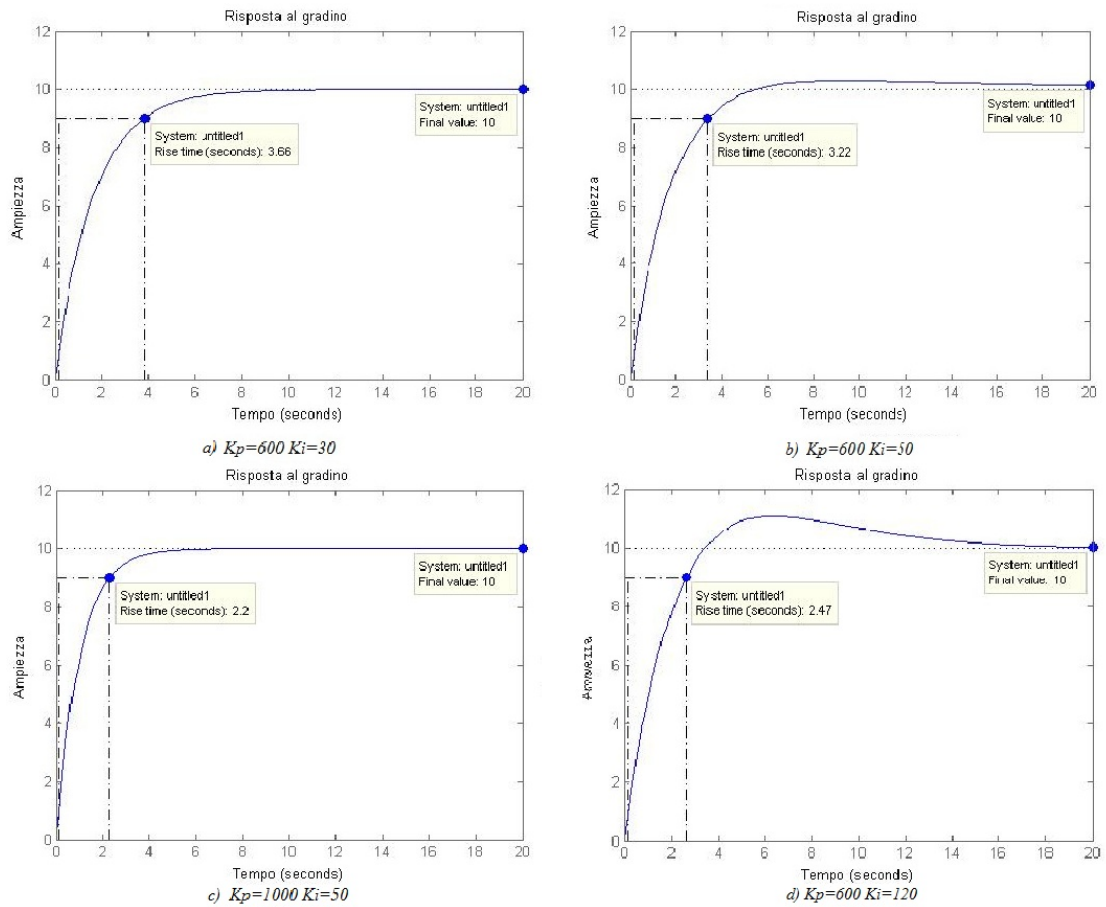


Figura 3.5: Risposte al gradino al variare dei parametri  $K_p$  e  $K_i$

Per ottenere una risposta che rispetti le specifiche in maniera ottimale, con un *tempo di salita* inferiore ai 5s, *sovraelongazione* inferiore al 10% ed un *errore a regime* al 2% (rispetto al valore di riferimento) sono stati utilizzati i valori  $K_p = 1000$  e  $K_i = 50$  ove come si vede dalla figura 3.5 c) non sono presenti l'*errore a regime* e la *sovraelongazione*, mentre il *tempo di salita* è pari a 2,2s riportata in figura (3.6).

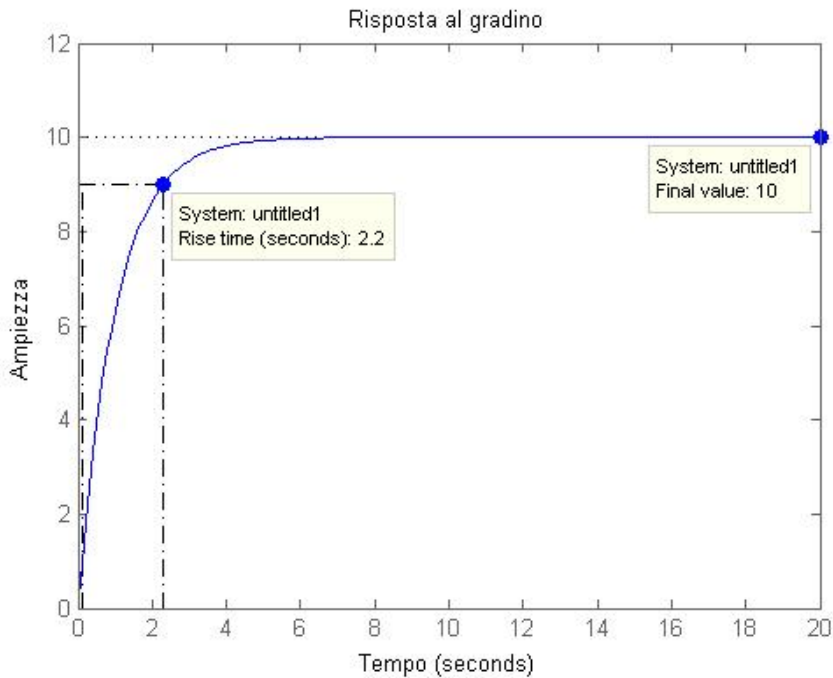


Figura 3.6: Scelta ottimale con  $K_p = 1000$  e  $K_i = 50$

### 3.2.3 Tipologia di controllo proporzionale-integrativa-derivativa PID

L'ultimo passo é rappresentato dal verificare cosa accade se aggiungiamo al controllore precedente una componente derivativa  $K_d$ , osservando (come fatto nei due paragrafi precedenti) l'andamento della risposta al gradino in funzione dei parametri selezionati. Ricordando l'equazione (3.5) generica per un controllore **PID** e sostituendola nell'equazione (3.4) si ottiene la funzione di trasferimento per il sistema retroazionato, che risulta essere dopo alcuni semplici passaggi algebrici:

$$H(s) = \frac{K_d s^3 + K_p s + K_i}{(m + K_d) s^2 + (b + K_p) s + K_i} \quad (3.10)$$

Scrivendo il tutto nell'm-file si ottiene:

m = 1000;

### CAPITOLO 3. PROGETTO DEL CONTROLLORE

```

b = 50;
r = 10;                                %Ingresso di riferimento (m/s)

s = tf('s');
Fdt = 1/(m*s + b);

Kp = 600;                                %Costante proporzionale
Ki=30;                                    %Costante integrativa
Kd=20;                                    %Costante derivativa
C = pid(Kp,Ki,Kd);                        %Funzione di trasferimento
                                        %del PID
                                        %ottenuta direttamente
                                        %dal comando integrato

Fdt_catena_chiusa = feedback(C*Fdt,1) %Istruzione che
                                        %permette la semplificazione
                                        %dello schema a blocchi
                                        %unendo le due fdt
                                        %ed ottenendo quella
                                        %in catena chiusa

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

```

e facendo variare tali parametri si ottengono le seguenti risposte al gradino:

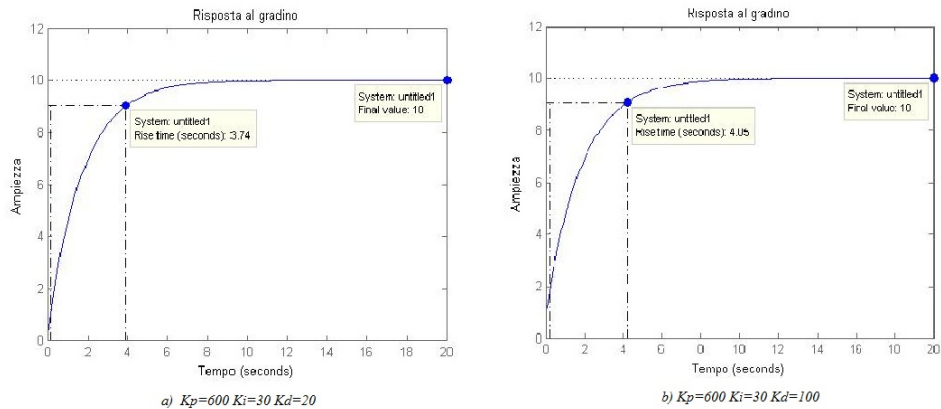


Figura 3.7: Risposte al gradino al variare dei parametri  $K_p, K_i$  e  $K_d$

Anche in questo caso é possibile trovare un determinato valore per ciascun parametro affinché le specifiche richieste vengano rispettate, ad esempio con  $K_p = 1000, K_i = 50$  e  $K_d = 30$  si ottiene la seguente risposta:



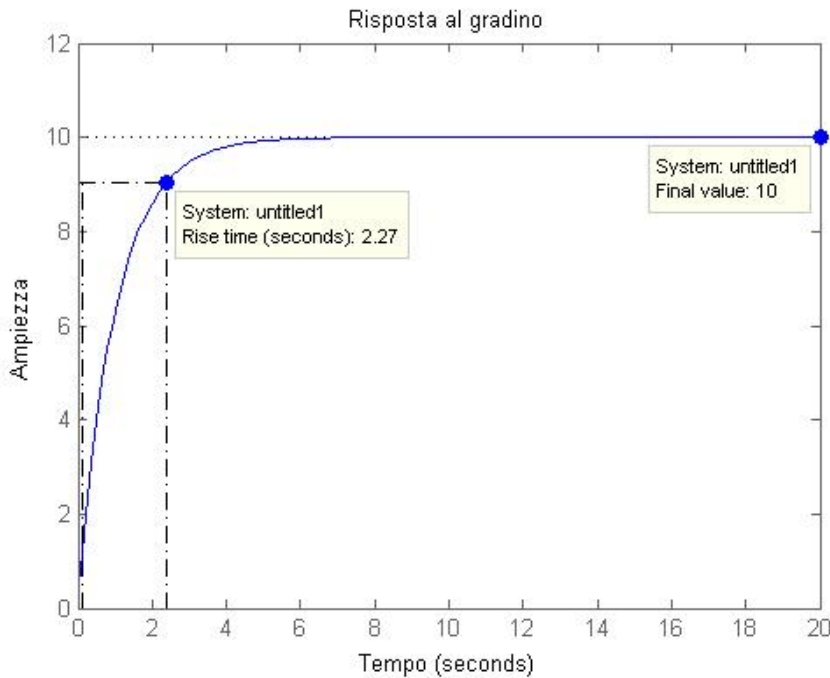


Figura 3.8: Risposta al gradino con  $K_p = 1000, K_i = 50$  e  $K_d = 30$

### 3.2.4 Osservazioni conclusive sui controllori

Come discusso nel paragrafo (3.2) l'utilizzo di un compensatore puramente proporzionale (**P**), sebbene in grado di ridurre sensibilmente il *tempo di salita*, permette di rispettare le specifiche richieste solo per via teorica, essendo però non realizzabile nella pratica per impossibilità fisica. L'utilizzo di un controllore proporzionale integrativo (**PI**) permette di raggiungere buoni risultati sotto il profilo del comportamento dinamico del sistema, in particolare con  $K_p = 1000$  e  $K_i = 50$  in cui si ha un *tempo di salita* pari a 2,2s e un *errore a regime* nullo.

Per quanto riguarda il (**PID**) si può concludere che pure esso è in grado di soddisfare le regole progettuali, ad esempio utilizzando i valori  $K_p = 1000, K_i = 50$  e  $K_d = 30$ , ma è preferibile utilizzare il compensatore **PI** per la sua maggior semplicità costruttiva.

Bisognerebbe poi aggiungere ai controllori, come ultimo passo, uno o più poli in alta frequenza (cioè a pulsazioni molto maggiori di quella di attraversamento) in modo da renderli propri, senza alterare le caratteristiche in bassa frequenza.

### 3.3 Analisi mediante luogo delle radici

Tra le varie metodologie possibili per l'analisi della stabilità (non considerate in questa relazione) vi sono il criterio di Routh e il criterio di Nyquist. Il primo permette di stabilire la collocazione dei poli della funzione di trasferimento del sistema retroazionato ( $H(s)$ ) rispetto all'asse immaginario, mentre il secondo permette di concludere qualcosa in più rispetto alla sola stabilità BIBO, come la robustezza della stabilità, valutabile mediante la distanza del diagramma di Nyquist dal punto critico  $-1+j0$ .

La funzione di trasferimento del sistema retroazionato é:

$$H(s) = \frac{C(s)G(s)}{1 + C(s)G(s)} \quad (3.11)$$

con  $C(s) = K_p$  in quest'analisi, e  $G(s)$  che rappresenta la funzione di trasferimento del sistema in catena aperta:

$$G(s) = \frac{p(s)}{q(s)} \quad (3.12)$$

ove  $p(s)$  e  $q(s)$  sono polinomi monici e coprime tra loro.

Il metodo del luogo delle radici, ora considerato, permette la determinazione dei singoli poli di  $H(s)$  e l'individuazione di determinati valori critici del parametro  $K_p$  responsabili di alcune transizioni nel comportamento del sistema, ad esempio da BIBO stabile ad instabile.

Per cui tale metodo assume la conoscenza esatta dei polinomi a numeratore e denominatore di  $G(s)$  permettendo dunque di dedurre la posizione di tutti i poli di  $H(s)$ .

Sostituendo l'equazione (3.12) in (3.11) si ottiene:

$$H(s) = \frac{K_p p(s)}{q(s) + K_p p(s)} \quad (3.13)$$

Il luogo delle radici é rappresentato dall'insieme di punti che annullano l'equazione  $q(s) + K_p p(s)$  al variare di  $K_p$  nei reali (escluso  $K_p = 0$ ). Ricordando che  $G(s)$  rappresenta la funzione di trasferimento ad anello aperto:

$$G(s) = \frac{1}{ms + b} \quad (3.14)$$

e sostituendola in (3.13) si ottiene finalmente:

$$H(s) = \frac{K_p}{ms + (b + K_p)} \quad (3.15)$$

Prima di rappresentare il luogo delle radici mediante le funzioni di *MatLab* é necessario calcolare il fattore di smorzamento  $\xi$  e la pulsazione naturale  $\omega_n$ .

Ricordando che la *sovraelongazione massima*  $M_p$  può essere approssimata dalla seguente funzione:

$$M_p \approx e^{-\frac{\pi\xi}{\sqrt{1-\xi^2}}} \quad (3.16)$$

tramite formula inversa, essendo la *sovraelongazione massima* consentita pari al 10% ossia 0.1, si può determinare il *fattore di smorzamento*:

$$\xi \geq \sqrt{\frac{\ln^2(M_p)}{\pi^2 + \ln^2(M_p)}} = 0.59 \implies 0.6 \quad (3.17)$$

Per quanto riguarda la pulsazione naturale é possibile utilizzare l'approssimazione:

$$\omega_n \geq \frac{1.8}{t_r} = 0.36 \text{rad/s} \quad (3.18)$$

dove riepilogando:

- $\xi$  rappresenta il fattore di smorzamento;
- $\omega_n$  rappresenta la pulsazione naturale;
- $t_r$  é il tempo di salita;
- $M_p$  la sovraelongazione massima.

Creando un nuovo m-file ed introducendo il seguente codice *MatLab*:

```
m = 1000;
b = 50;
r = 10;
s = tf('s');
Fdt = 1/(m*s + b);
rlocus(Fdt) %Istruzione che grafica il luogo delle radici
axis([-0.6 0 -0.6 0.6]);
sgrid(0.6,0.36) %Coefficiente di smorzamento e
                %pulsazione naturale
```

si ottiene il seguente luogo delle radici:

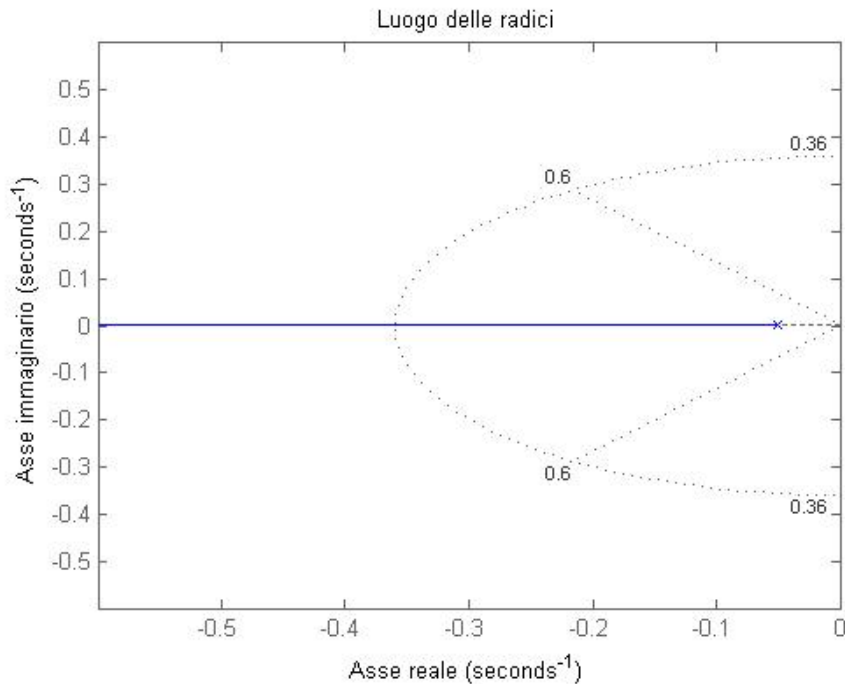


Figura 3.9: Rappresentazione del luogo delle radici

Essendo presente un unico polo in  $-0,05$  esso é rappresentato correttamente da una  $\times$  alla sinistra dell'asse immaginario e non essendoci zeri in  $G(s)$  effettivamente tale ramo si chiude all'infinito. Tale figura rappresenta inoltre i due segmenti (tratteggiati) lungo i quali il coefficiente di smorzamento vale esattamente  $0,6$  ed analogamente il semi-ellisse rappresenta i punti aventi pulsazione naturale pari a  $0,36$ . Mantenendosi all'interno dei due segmenti il valore di  $\xi$  aumenta e allo stesso modo allontanandosi dall'ellisse  $\omega_n$  aumenta.

Introducendo l'istruzione *MatLab* alla fine del nostro m-file

```
[Kp,poles]=rlocfind(Fdt)
```

ci é consentito di "interagire" con il luogo delle radici, nel senso che risulta possibile specificare dove voler posizionare il polo ed automaticamente *MatLab* fornisce il valore della costante proporzionale  $K_p$  necessario per ottenere tale risultato.

Vediamo meglio con un esempio. Facendo partire il programma dall'editor di *MatLab* ci viene chiesto di selezionare sul luogo un determinato punto, che noi prenderemo nell'intorno di  $-0,5$  e ciò che otteniamo in output é:

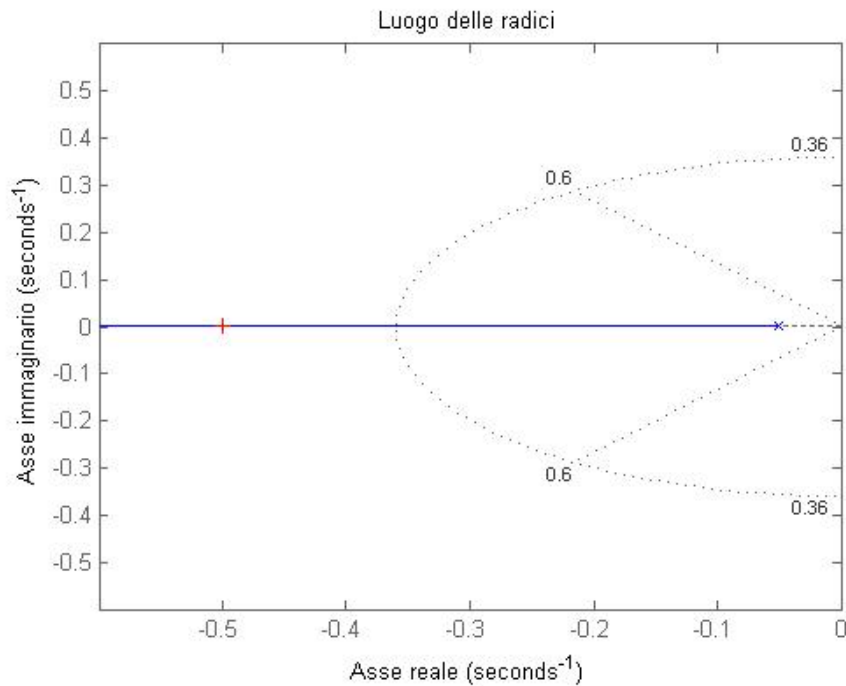


Figura 3.10: Punto del luogo selezionato

ed automaticamente il command window di *MatLab* fornirà la seguente risposta

Select a point in the graphics window

selected\_point =

-0.4998 + 0.0019i

Kp =

449.7669

poles =

-0.4998

tale valore di  $K_p$  potrà essere inserito direttamente nella funzione di trasferimento  $H(s)$  per visualizzare l'andamento della risposta al gradino.

Kp=449.7669;

H= feedback(Kp\*Fdt,1);

```
t=0:0.1:20;
step(r*H,t)
```

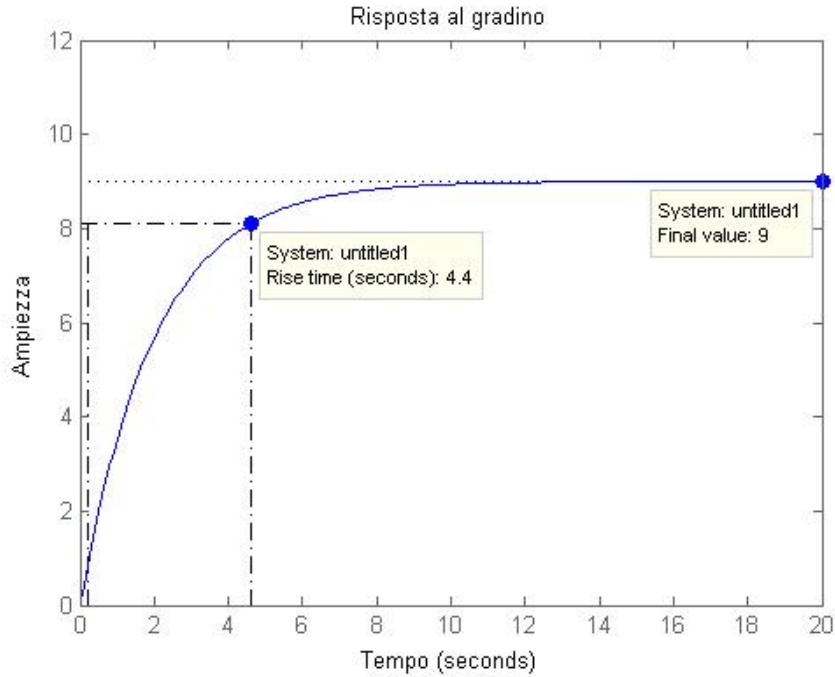


Figura 3.11: Risposta al gradino con il valore  $K_p = 449.7669$

Il vincolo progettuale relativo al *tempo di salita* viene rispettato in quanto  $4,4s < 5s$ , ma per quanto riguarda l'*errore a regime* le specifiche non vengono rispettate perché esso risulta essere del 10% mentre viene richiesto che sia inferiore al 2%.

Verifichiamo come varia la risposta in funzione del punto del luogo selezionato, cioè sostanzialmente ripetiamo l'analisi precedentemente condotta selezionando diversi punti nel luogo e rappresentandone la corrispondenti risposte al gradino.

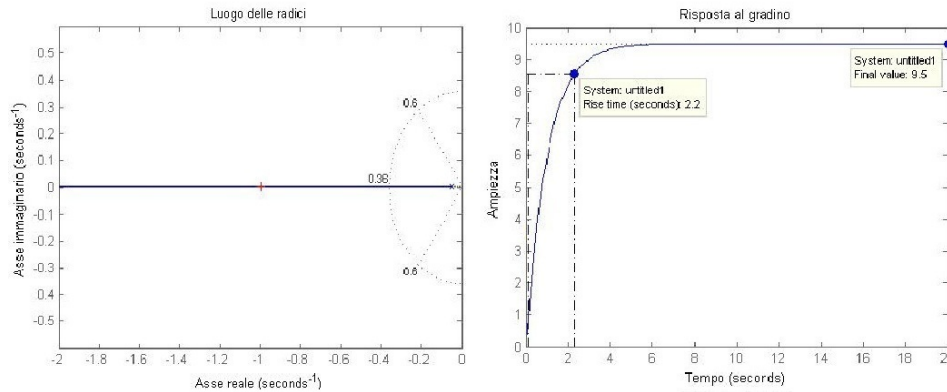


Figura 3.12: Luogo delle radici e risposta al gradino per il punto  $-0.9976 + 0.0019i$  con  $K_p = 947.6322$

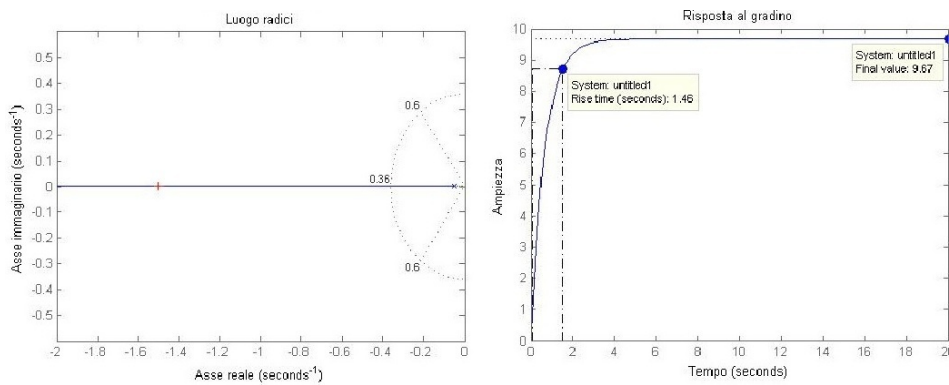


Figura 3.13: Luogo delle radici e risposta al gradino per il punto  $-1.5000 + 0.0019i$  con  $K_p = 1450$

Allontanando il polo dall'asse immaginario si osserva un incremento del guadagno  $K_p$  che si ripercuote in una diminuzione del *tempo di salita* e dell'*errore a regime*, ricadendo nell'analogo caso del controllore proporzionale  $\mathbf{P}$  in cui il limite é ancora una volta rappresentato dalla fisica realizzabilità.

### 3.3.1 Introduzione delle rete ritardatrice (Phase Lag)

Al fine di ridurre l'*errore a regime* viene aggiunta al sistema una rete ritardatrice la cui funzione di trasferimento é:

$$C(s) = \frac{s + z_0}{s + p_0} \quad (3.19)$$

costituita da uno zero e da un polo, dove la pulsazione del polo  $\omega_p$  é inferiore a quella dello zero  $\omega_z$  ( $\omega_p < \omega_z$ ).

Per cui la rete ritardatrice non introduce attenuazioni quando é  $\omega < \omega_p$ , introduce un'attenuazione di 20dB per decade quando si ha  $\omega_p < \omega < \omega_z$ , mentre

alle alte frequenze ( $\omega > \omega_z$ ) l'attenuazione rimane costante (perché i contributi di polo e zero si compensano). Considerando ora il sistema in retroazione unitaria composto dalla rete ritardatrice, dal controllo proporzionale e dalla nota funzione di trasferimento  $G(s)$  si trova dopo alcuni passaggi algebrici:

$$H(s) = \frac{K_p s + K_p z_0}{m s^2 + (b + m p_0 + K_p) s + (b p_0 + K_p z_0)} \quad (3.20)$$

Scegliendo di piazzare il polo  $p_0$  in  $\omega_p = 0.03 \text{ rad/s}$  e lo zero  $z_0$  una decade dopo cioè  $\omega_z = 0.3 \text{ rad/s}$  ed utilizzando il seguente codice *MatLab* e rappresentando graficamente il tutto si ottiene:

```

z0=0.3;
p0=0.03;
Rete_rit=(s+z0)/(s+p0);
rlocus(Rete_rit*G)           %Rappresentazione grafica del luogo
axis([-0.6 0 -0.4 0.4])
sgrid(0.6,0.36);           %Rappresenta i parametri
                             %di smorzamento e pulsazione
[Kp,poles]=rlocfind(Rete_rit*G) %Istruzione che permette
                             %di selezionare il punto nel luogo
    
```

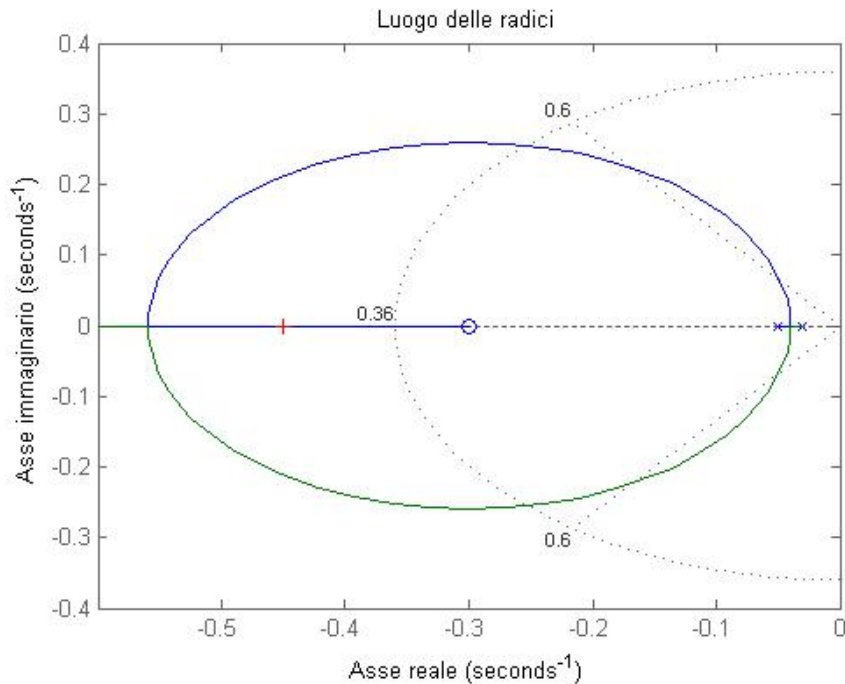


Figura 3.14: Luogo delle radici per  $H(s)$



e selezionando un punto del luogo si ha:

Select a point in the graphics window

selected\_point =

-0.4457 - 0.0012i

Kp =

1.1289e+03

poles =

-0.7631

-0.4457

che ci fornisce immediatamente il valore della costante  $K_p = 1128.9$ , utilizzabile per rappresentarne la risposta al gradino.

```
Kp=1.1289e+03;
H=feedback(Rete_rit*Kp*Fdt,1);    %Istruzione che permette
                                  %il calcolo della Fdt retroazionata
t=0:0.1:20;
step(r*H,t)                       %Rappresentazione grafica
                                  %della risposta al gradino
axis([0 20 0 12])
```

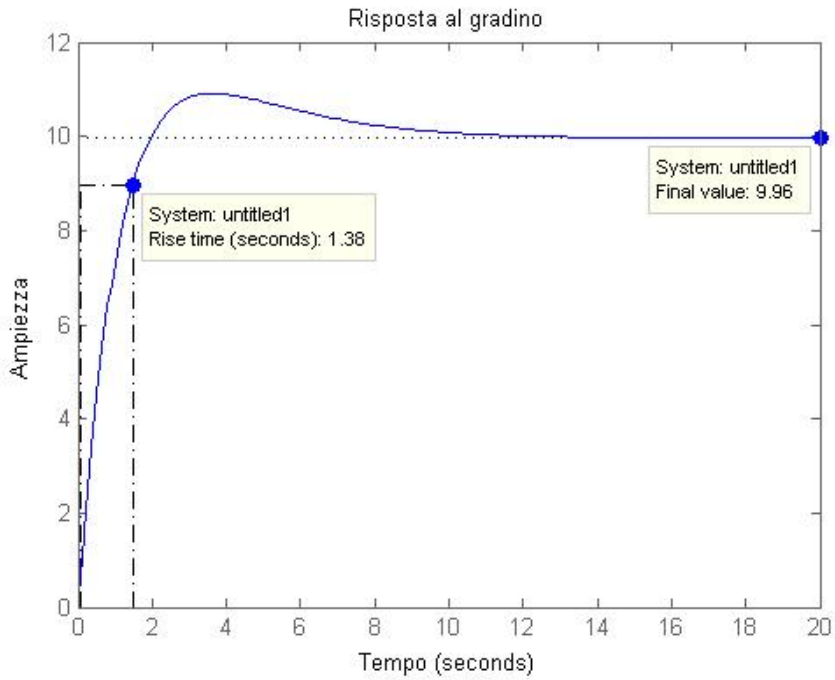


Figura 3.15: Risposta al gradino in funzione del punto selezionato

La rete ritardatrice ha permesso di migliorare il comportamento dinamico del sistema, infatti sia il *tempo di salita* che l'*errore a regime* rispettano i vincoli progettuali, ma bisogna fare sempre attenzione a  $t_r$  in quanto un suo valore troppo basso impedisce la realizzabilità meccanica del sistema cruise control.

La presenza di una *sovraelongazione* in prossimità dei 3,5 s, dovuta allo zero della rete ritardatrice, non crea nessun problema perché anch'essa rispetta le specifiche assegnate (<10%). Viene ora rappresentato il luogo delle radici e la risposta al gradino variando la posizione dello zero e del polo.

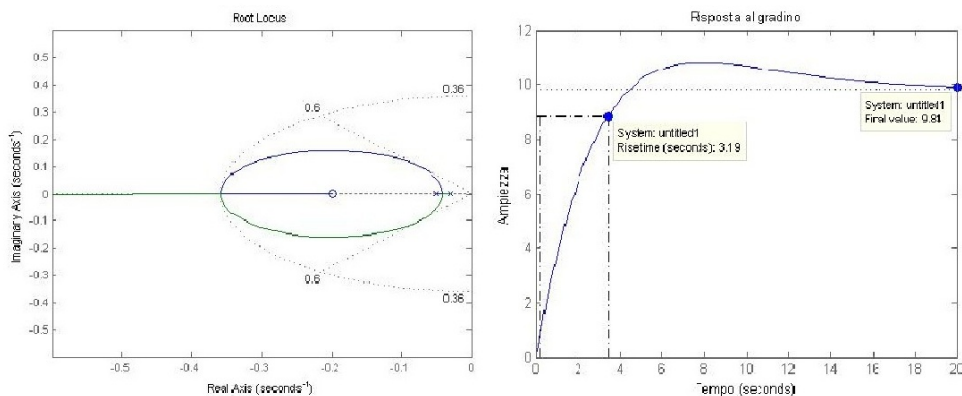


Figura 3.16:  $K_p = 450$  con  $z_0 = 0.2$ ,  $p_0 = 0.005$

### 3.3.2 Osservazioni conclusive sul luogo delle radici

Utilizzando il metodo del luogo delle radici abbiamo osservato che l'introduzione di un solo guadagno proporzionale non risulta sufficiente al fine di rispettare le specifiche richieste, in modo particolare l'*errore a regime*; incrementando tale valore  $K_p$  si osservava la contemporanea diminuzione dell'errore stesso, ma anche una riduzione così drastica del *tempo di salita* tale da comprometterne la realizzabilità meccanica. Abbiamo allora introdotto una rete ritardatrice nel nostro sistema con l'obiettivo sostanziale di ridurre l'*errore a regime* e di ottenere un *tempo di salita* ragionevole per la realizzazione fisica.

La presenza dello zero nella nostra *Phase Lag* introduce nella risposta una leggera *sovraelongazione*, che a seconda di dove allochiamo  $z_0$  risulterà più o meno accentuata, ma nei casi da noi considerati e rappresentati risulta comunque compresa nel range dei vincoli di progetto (ossia  $<10\%$ ). La figura 3.16 presenta una risposta al gradino accettabile in quanto la *sovraelongazione* è  $\sim 8\%$ ,  $t_r=3,19$  s e l'*errore a regime*  $\sim 1.9\%$ .

## 3.4 Analisi in frequenza

### 3.4.1 Considerazioni preliminari

Finora abbiamo considerato la nostra analisi dinamica in funzione di alcuni parametri caratteristici nel tempo, quali *tempo di salita*  $t_r$  e la *sovraelongazione*  $s$ .

È possibile fornire una caratterizzazione del comportamento del sistema anche nel dominio della frequenza, definendo alcuni parametri che fanno riferimento alla risposta in frequenza del sistema  $W(j\omega)$  (ottenuta direttamente dalla  $W(s)$  tramite il cambio di variabile  $s \rightarrow j\omega$ ). Tali parametri sono:

- $B_p$  = banda passante (a 3dB);
- $\omega_r$  = pulsazione di risonanza;
- $M_{rel}$  = picco di risonanza relativo.

La banda passante (a 3dB)  $B_p$  individua l'intervallo di pulsazioni  $[0, B_p]$  in cui si ha che

$$|W(j\omega)|_{dB} \geq |W(0)|_{dB} - 3dB \quad (3.21)$$

mentre la pulsazione di risonanza (se presente è unica) è quella pulsazione  $\omega_r > 0$  in corrispondenza alla quale il modulo di  $W(j\omega)$  assume il valore massimo, ovvero

$$|W(j\omega_r)| = \max_{(\omega > 0)} |W(j\omega)|_{dB}. \quad (3.22)$$

Il picco di risonanza relativo  $M_{rel}$  rappresenta il rapporto tra il modulo delle risposta in frequenza alla pulsazione  $\omega_r$  e  $|W(0)|$ , ossia

$$M_{rel} = \left[ \frac{|W(j\omega_r)|}{|W(0)|} \right]_{dB}. \quad (3.23)$$

La risposta in frequenza compendia in sé tutte le informazioni relative alla *BIBO stabilità* e permette di studiare la stabilità del cosiddetto sistema ad anello chiuso. Al fine di rappresentare la risposta in frequenza, solitamente si utilizzano due rappresentazioni grafiche piuttosto che quella analitica:

- il diagramma di Nyquist che propone una rappresentazione della funzione complessa  $W(j\omega)$  nel piano complesso, parametrizzata dalla variabile reale  $\omega$ ;
- il diagramma di Bode, uno per l'ampiezza ed uno per la fase che riportano rispettivamente il logaritmo del modulo della risposta in frequenza in funzione del logaritmo della pulsazione, e l'argomento della risposta in frequenza (in radianti o gradi) sempre in funzione del logaritmo della pulsazione.

Con l'obiettivo di studiare la risposta in frequenza, si rivela fondamentale introdurre alcuni parametri per una generica funzione di trasferimento  $G'(s)$ :

- **pulsazione di attraversamento** di  $G'(s)$ , indicata col simbolo  $\omega_A$ , è quella pulsazione positiva in corrispondenza alla quale il diagramma di Bode delle ampiezze di  $G'(s)$  attraversa l'asse delle ascisse, ossia

$$|G'(j\omega_A)|_{dB} = 20 \log |G'(j\omega_A)| = 0; \quad (3.24)$$

- **fase di attraversamento** di  $G'(s)$ , la grandezza

$$\varphi_A = \arg(G'(j\omega_A)); \quad (3.25)$$

- **marginale di fase** di  $G'(s)$ , la grandezza

$$m_\varphi = \pi + \varphi_A \quad (3.26)$$

### 3.4.2 Considerazioni in catena aperta e in catena chiusa

Come quanto svolto per il luogo delle radici, effettueremo la nostra analisi in frequenza del sistema utilizzando un controllore proporzionale. Similmente a quanto svolto nel capitolo 2 si ricorda la funzione di trasferimento in catena aperta risulta (con la stessa notazione di tale capitolo)

$$W(s) = C(s)G(s) = \frac{K_p}{ms + b} \quad (3.27)$$

e per  $C(s) = K_p = 1$  il diagramma di Bode è identico a quello di figura (2.3) così come la risposta al gradino risulta analoga a quella di figura (2.1).

Per completezza riportiamo qui di seguito il diagramma di Nyquist, che si ottiene con il seguente codice *MatLab*:

```
Kp=1;
nyquist(Kp*Fdt)           %Istruzione che mi fornisce
                           %la rappresentazione
                           %del diagramma di Nyquist
```

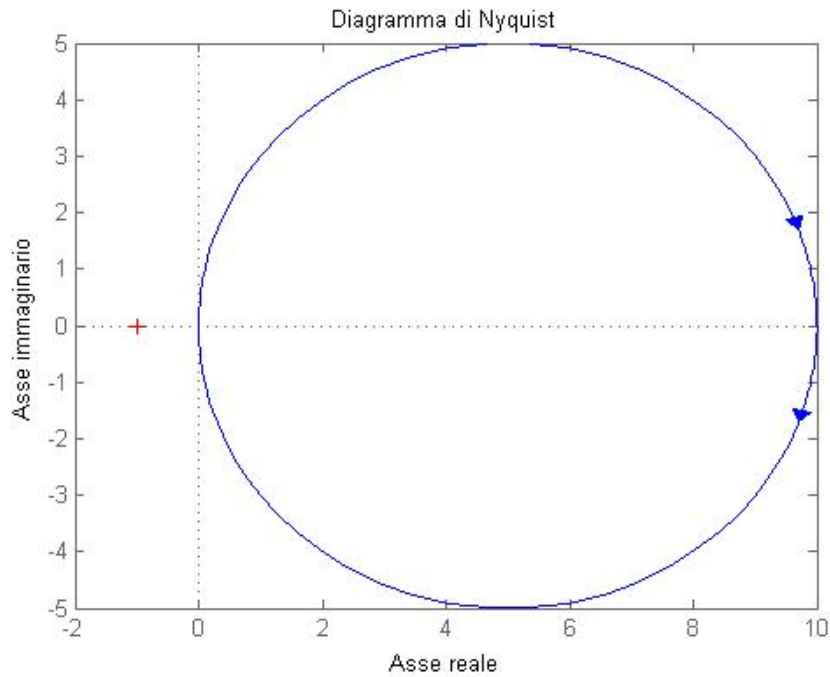


Figura 3.17: Diagramma di Nyquist

Nel capitolo 2 avevamo già discusso la stabilità di tale sistema in catena aperta, mentre per quanto riguarda il sistema in catena chiusa si può applicare il criterio ridotto di Nyquist, infatti vale che:

$$N = n_{W+} - n_{H+} \quad (3.28)$$

dove  $N$  rappresenta il numero di giri che il diagramma compie attorno al punto critico  $-1+j0$ ,  $n_{W+}$  e  $n_{H+}$  rappresentano rispettivamente il numero di poli a parte reale positiva della funzione di trasferimento in catena aperta  $W(s)$  ed il numero di poli a parte reale positiva del sistema retroazionato con fdt  $H(s)$ . Dalla figura (3.17) si evince che tale diagramma non circonda il punto critico e dunque  $N=0$ , a priori sappiamo anche che  $W(s)=C(s)G(s)$  non esibisce poli a parte reale positiva e dunque segue immediatamente che anche  $n_{H+}=0$  e dunque il sistema retroazionato risulta *BIBO stabile*.

Per maggior completezza e per alcune considerazioni riportiamo qui di seguito la figura (2.3)

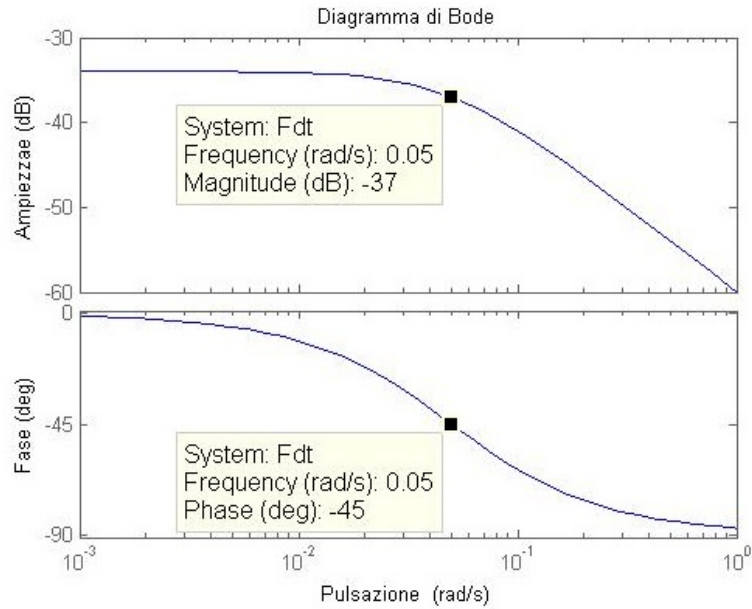


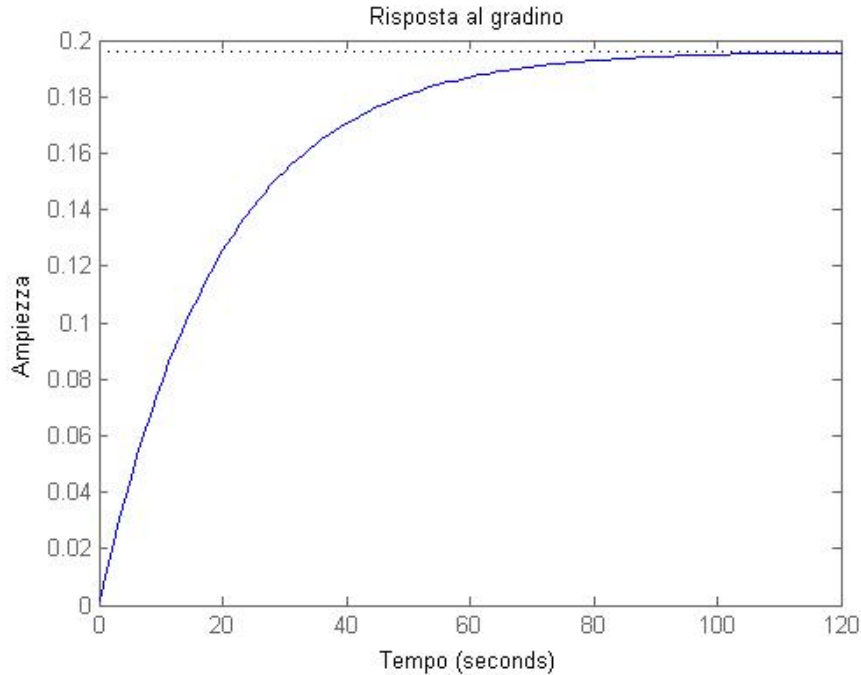
Figura 3.18: Diagramma di Bode della Fdt  $W(s)=C(s)G(s)$

L'errore a regime può essere determinato mediante la seguente formula:

$$e_{rp} = \frac{1}{1 + M_{\omega \rightarrow 0}} \cdot 100 = \frac{1}{1 + 0,02} \cdot 100 \simeq 98\% \quad (3.29)$$

in quanto  $M_{\omega \rightarrow 0} = -34\text{dB}$  che in lineare corrispondono circa a 0,02. Questo ragionamento può essere confermato dalla seguente risposta al gradino:

```
r=10;
W=feedback(Kp*Fdt,1);
step(r*W);
```



Per cui resta ora da migliorare l'errore a regime in modo che venga rispettato il vincolo progettuale.

Ciò che vogliamo ottenere é un *errore a regime*  $e_{rp} < 2\%$ , dunque:

$$e_{rp} < 2\% \implies \frac{1}{1 + M_{\omega \rightarrow 0}} \cdot 100 < 2\% \implies M_{\omega \rightarrow 0} > 49 \quad (3.30)$$

ed esprimendo tale rapporto in dB si ottiene:

$$M_{\omega \rightarrow 0}|_{dB} = 33.8dB. \quad (3.31)$$

Per cui il nostro incremento dovrà risultare pari a:

$$|33.8 - (-34)|_{dB} = 67.8dB \implies K_p > 2454,71 \quad (3.32)$$

ossia utilizzeremo un guadagno pari a  $K_p = 2500$ . Tale procedimento equivale a spostare in alto il diagramma di Bode delle ampiezze di 67,8dB infatti possiamo ora rappresentare il tutto e mettere in evidenza la *pulsazione di attraversamento*, il *margin di fase* e la *fase di attraversamento*.

```
Kp=2500;           %Valore calcolato
bode(Kp*Fdt);     %Istruzione che rappresenta
                  %graficamente il diagramma di Bode
margin(Kp*Fdt);   %Istruzione che permette
                  %la visualizzazione
                  %dei vari parametri
```

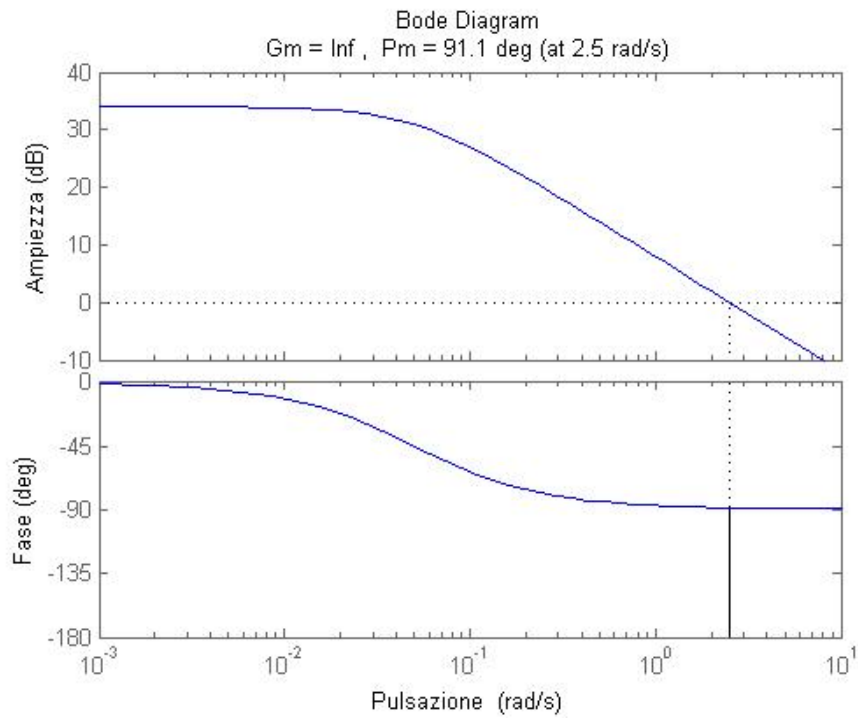


Figura 3.19: Diagramma di Bode con  $K_p=2500$

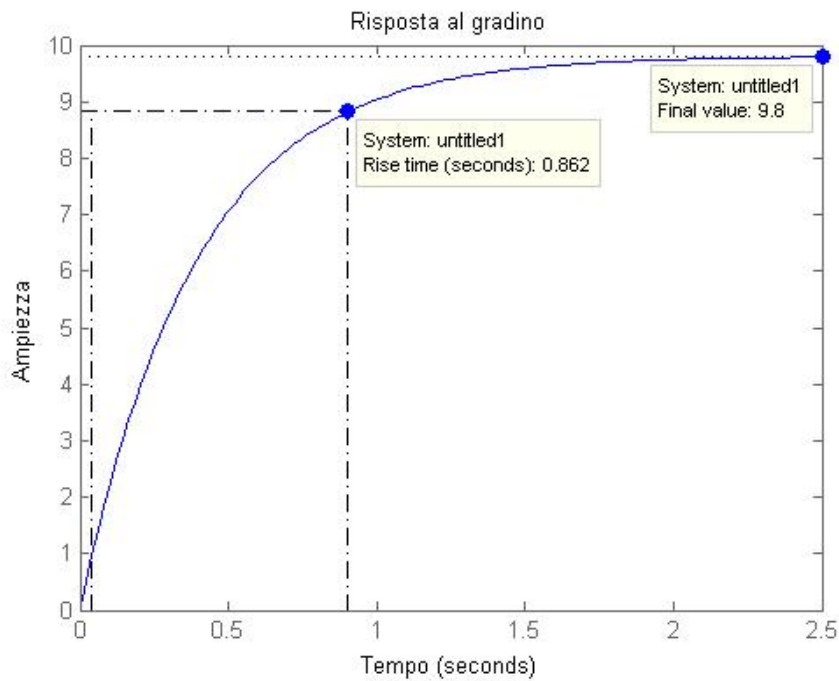


Figura 3.20: Risposta al gradino con  $K_p=2500$  ed  $e_{rp} = 2\%$



Quello che si può ancora una volta osservare é che il sistema ora presenta tutte le caratteristiche soddisfacenti, un *errore a regime* pari al 2% ed un *tempo di salita*  $< 5$  s. Tale valore di  $t_r$  é troppo piccolo e come negli analoghi casi precedenti ne comprometterebbe la realizzabilità meccanica, per cui si rende necessario l'utilizzo di una rete ritardatrice. La figura (3.19) evidenzia un margine di fase ampio di  $91,1^\circ$ , e siccome nelle vicinanze della pulsazione di attraversamento  $\omega_A = 2,5 \text{ rad/s}$  il diagramma di Bode dell'ampiezza scende con una pendenza sufficiente e non si riavvicina piú all'asse delle ascisse, esso garantisce una distanza sufficiente dal punto critico nel diagramma di Nyquist.

### 3.4.3 Introduzione delle rete ritardatrice

Come per il luogo delle radici, al fine di migliorare il comportamento dinamico del sistema soprattutto per quanto riguarda il *tempo di salita*, risulta necessario l'utilizzo di una Phase Lag. Come già visto nella sezione (3.4) tale rete ci permette di ridurre l'errore a regime e di diminuire la pulsazione di attraversamento così da ottenere una risposta meno rapida. Ricordando l'equazione di una rete ritardatrice

$$C(s) = \frac{s + z_0}{s + p_0} \quad (3.33)$$

ed allocando lo zero  $z_0 = 0.1$  e il polo  $p_0 = 0.02$  si avrà una riduzione dell'*errore a regime* pari ad un fattore  $\frac{z_0}{p_0} = 5$ .

```
Kp=500;           %Costante proporzionale
                  %prima calcolata
z0=0.1;           %zero
p0=0.02;          %polo
Rete_rit=(s+z0)/(s+p0);
bode(Kp*Rete_rit*Fdt);
```

Il diagramma di Bode e la risposta al gradino risultano:

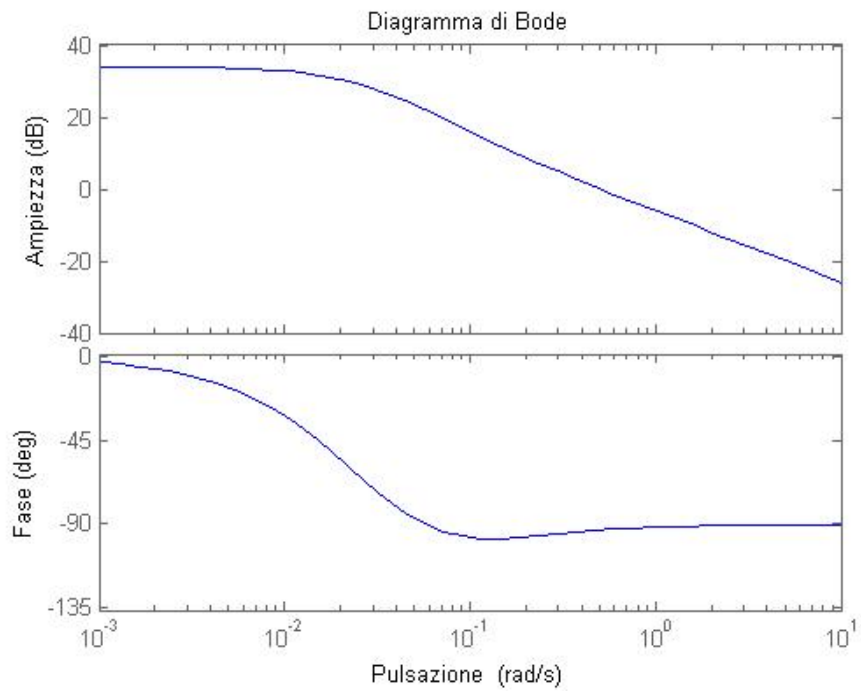


Figura 3.21: Diagramma di Bode con  $K_p=500$  e  $z_0 = 0.1$  e  $p_0 = 0.02$

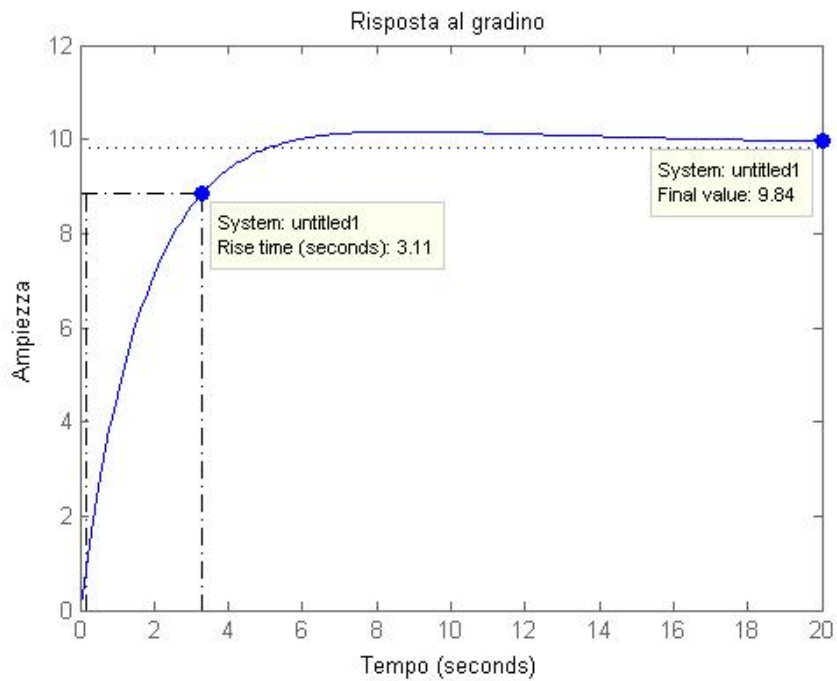


Figura 3.22: Risposta al gradino con  $K_p=500$

ora tutte le specifiche risultano soddisfatte infatti l'*errore a regime* é pari al 1.6% e il *tempo di salita* pari a 3,11 s.

#### 3.4.4 Osservazioni conclusive sull'analisi in frequenza

Come già visto precedentemente in questa relazione il controllore proporzionale permette di soddisfare le specifiche solo per via teorica, non essendo però possibile realizzare una tale sistema cruise control nella pratica a causa delle forze impulsive che si genererebbero a seguito di un *tempo di salita* così breve. Con l'inserimento della rete ritardatrice si sono ottenuti buoni risultati sia per quanto riguarda  $t_r=3,11$  s ed un  $e_{rp}=1.6\%$  , migliorando il comportamento dinamico rispetto all'utilizzo del solo controllore proporzionale.

## Capitolo 4

# Analisi del modello mediante Simulink

### 4.1 Osservazioni preliminari

In questo capitolo tratteremo brevemente l'analisi del sistema mediante l'utilizzo di *Simulink*.

Tale programma viene utilizzato per la modellizzazione, la simulazione e l'analisi di sistemi dinamici descritti da equazioni differenziali, mediante l'utilizzo di funzioni di trasferimento e sistemi in forma di stato. La simulazione avviene sostanzialmente in due passi:

- **descrizione del sistema mediante schema a blocchi:** *Simulink* possiede un'ampia libreria di blocchi che descrivono elementi algebrici e dinamici elementari, come funzioni di trasferimento, guadagni, saturazioni ecc. L'utente compone lo schema a blocchi del sistema da simulare mediante l'interconnessione dei blocchi elementari;
- **simulazione del sistema:** attraverso l'integrazione numerica delle equazioni differenziali in un certo intervallo di tempo.

Per l'analisi che segue si farà riferimento allo stesso identico sistema introdotto all'inizio di questa tesi e rappresentato dalla figura (1.1) del capitolo 1. I parametri a cui si farà riferimento sono gli stessi introdotti nella tabella del paragrafo (1.2.1) a pagina 3, ossia massa  $\mathbf{m}=1000$  kg, coefficiente di attrito  $\mathbf{b}=50$  Ns/m, forza motrice  $\mathbf{u}=500$  N e velocità di riferimento  $\mathbf{r}=10$ m/s. Per non appesantire ulteriormente il discorso nel ripetere il tutto, si suppongono note tutte le leggi derivate nei capitoli precedenti.

### 4.2 Costruzione del modello

Al fine di costruire il nostro sistema per poi poterlo simulare dobbiamo comporre lo schema a blocchi partendo dal blocco finale, ossia il blocco

che permette l'integrazione dell'accelerazione al fine di ottenere la velocità. Partendo dall'equazione che regola il moto

$$m\dot{v} = u - bv \quad (4.1)$$

si ottiene che la nostra variabile di interesse si può calcolare mediante l'equazione:

$$v = \int \frac{u(t) - bv}{m} = \int \frac{dv}{dt} dt. \quad (4.2)$$

Cercando nella libreria di *Simulink* si trova il blocco integratore che fa proprio il caso nostro e rappresentando in tale blocco l'input (accelerazione  $\dot{v}$ ) e l'output (velocità  $v$ ) di nostro interesse si ottiene la seguente rappresentazione:

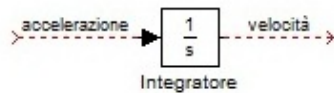


Figura 4.1: Blocco integratore

L'ingresso applicato a tale blocco rappresenta l'accelerazione del nostro veicolo ottenuta tramite la seconda legge di Newton dal rapporto tra la differenza della forza motrice e della forza di attrito e la massa, ossia

$$\frac{dv}{dt} = \frac{u(t) - bv(t)}{m}. \quad (4.3)$$

Proseguiamo ora nel completamento dello schema a blocchi introducendo il blocco per creare come input il gradino, i due blocchi di "guadagno" (rispettivamente  $\frac{1}{m}$  e  $b$ ) ed infine il blocco sommatore (a cui verrà cambiato uno dei due segni) che permette di realizzare la differenza tra  $u(t)$  e  $bv$ .

Come ultimo passo viene introdotto un blocco visualizzatore che permette di rappresentare l'andamento della nostra risposta al gradino.

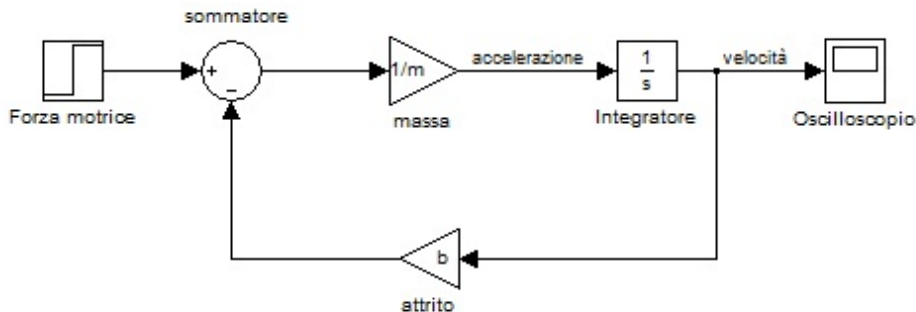


Figura 4.2: Schema a blocchi completo

Una volta ultimato lo schema a blocchi ed inserito in ogni blocco le caratteristiche che ci interessano, come ad esempio il fatto che il gradino sia ampio esattamente 500, si può procedere con l'analisi in catena aperta.

### 4.3 Simulazione in catena aperta

Considerando lo schema a blocchi precedentemente rappresentato al fine di visualizzare l'andamento della risposta al gradino bisogna impostare alcuni parametri tramite il comando "simulation" → "configuration simulation", come il tempo di stop (fissato qui pari a 120 s come nell'analisi iniziale di figura 2.1).

L'ultimo step consiste nell'inserire i vari parametri di *massa*, *coefficiente di attrito* e *ampiezza del gradino* tramite il prompt dei comandi di *MatLab*.

L'oscilloscopio o visualizzatore posizionato alla fine del nostro schema a blocchi permette la visualizzazione dell'andamento della risposta al gradino, e tramite il comando *Start Simulation* si ottiene:

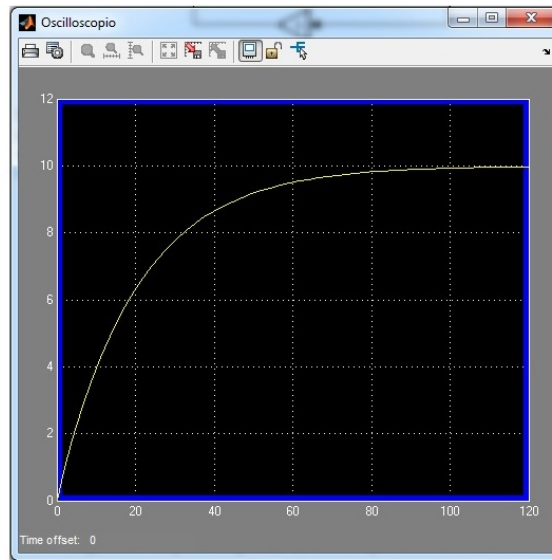


Figura 4.3: Risposta al gradino ad anello aperto

È immediato rendersi conto che la risposta non soddisfa tutte le specifiche richieste (v. paragrafo 1.4), infatti sebbene non sia presente *sovraelongazione* e l'*errore a regime* risulti nullo, il *tempo di salita* non è assolutamente inferiore ai 5 s ( $\simeq 46s$ ).

Si rende dunque necessario l'utilizzo di un controllo in retroazione.

## 4.4 Simulazione in catena chiusa

Per migliorare la rappresentazione grafica dello schema a blocchi é stato semplificato lo schema precedentemente utilizzato in un specie di "black box" in modo da non appesantire le successive immagini.

Tale *scatola nera* rappresenta ciò che é stato derivato in figura (4.2), con gli stessi parametri precedentemente utilizzati.

Unitamente alla conoscenza dei risultati ottenuti nei precedenti capitoli, verranno qui proposte solamente le metodologie di controllo proporzionale-integrativo (**PI**) e tramite l'utilizzo di una *phase lag*.

### 4.4.1 Controllore PI

Il primo passo da svolgere ora é rappresentato dal costruire lo schema a blocchi per il sistema retroazionato. Utilizzando la semplificazione dello schema in "black box" introdotta nel paragrafo precedente bisogna aggiungere i blocchi relativi alla funzione di trasferimento del **PI**, al sommatore, all'oscilloscopio e alla velocità di riferimento. Lo schema a blocchi che ne risulta é rappresentato dalla seguente figura:

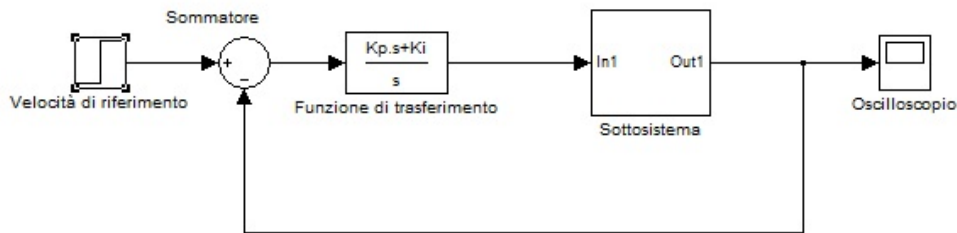


Figura 4.4: Schema a blocchi del sistema retroazionato

Facendo doppio *click* sul blocco relativo alla funzione di trasferimento é possibile implementare la fdt del controllore **PI**, e dal *prompt* dei comandi *MatLab* sará possibile modificare tutti i parametri del sistema.

Il codice a cui faremo riferimento é:

```
m = 1000;
b = 50;
r = 10;           %Velocit\'a di riferimento
Kp = 1000;       %Costante proporzionale
Ki = 50;         %Costante integrativa
```

come costante proporzionale ed integrativa si sono utilizzati i valori ottimali ottenuti con l'analisi in *MatLab*, che sono rispettivamente  $K_p = 1000$  e  $K_i = 50$ . Facendo partire la simulazione e facendo un doppio *click* sul visualizzatore si ottiene il seguente andamento della risposta:

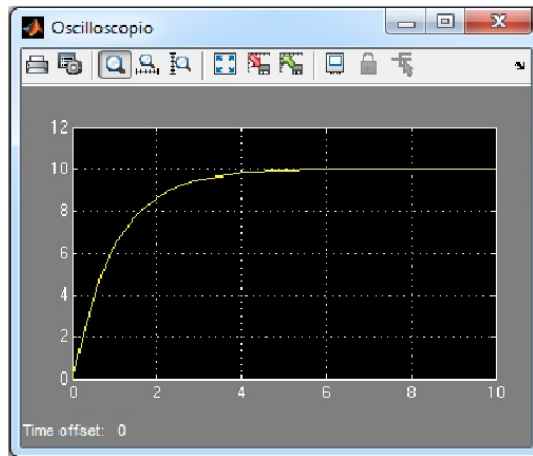


Figura 4.5: Risposta al gradino a catena chiusa con controllore PI

da tale figura si evince che ogni vincolo progettuale é soddisfatto in quanto non vi é né *sovraelongazione* né *errore a regime* ed il *tempo di salita* risulta inferiore ai 4 s.

Per non essere ridondanti visto che con il controllore proporzionale integrativo abbiamo raggiunto gli obiettivi prefissati, non viene riportata l'analisi mediante controllore **PID**.

#### 4.4.2 Introduzione della rete ritardatrice

Per quest'analisi richiamiamo la funzione di trasferimento della rete *phase lag* e quella corrispondente al sistema retroazionato (paragrafo 3.3.1)

$$C(s) = \frac{s + z_0}{s + p_0} \quad (4.4)$$

$$H(s) = \frac{K_p s + K_p z_0}{m s^2 + (b + m p_0 + K_p) s + (b p_0 + K_p z_0)}. \quad (4.5)$$

Sostanzialmente lo schema a blocchi non risulta essere tanto diverso da quello di figura 4.4, infatti risulta sufficiente aggiungere dopo il sommatore un guadagno (che rappresenta il guadagno proporzionale  $K_p$  come svolto nelle precedenti analisi) seguito in cascata dal blocco rappresentante la funzione di trasferimento della rete ritardatrice, eliminando poi il blocco relativo al controllore **PI**.



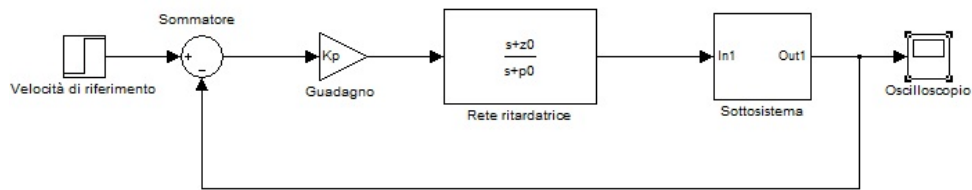


Figura 4.6: Schema a blocchi del sistema con phase lag

Dal *prompt* dei comandi *MatLab* è sufficiente introdurre i valori di  $K_p$  del polo e dello zero e far partire la simulazione per vedere l'andamento della risposta; facendo tesoro dell'analisi precedentemente svolta un buon andamento della risposta si ottiene in corrispondenza a questi valori:

```
Kp=750;
z0=0.03;
p0=0.003;
```

ottenendo la seguente risposta:

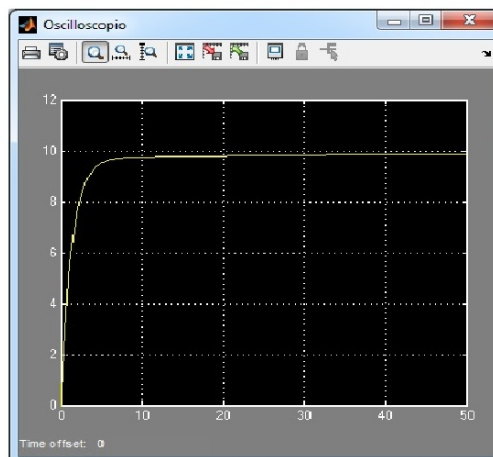


Figura 4.7: Risposta al gradino con  $K_p = 750, z_0 = 0.03, p_0 = 0.003$

dove sia il *tempo di salita* ( $\sim 3,4 s$ ) che la sovralongazione (assente), sia l'*errore a regime* ( $<1,3 \%$ ) rispettano le specifiche richieste. Variando leggermente i parametri, ad esempio  $K_p = 500, z_0 = 0.2$  e  $p_0 = 0.02$  si ottiene la seguente risposta

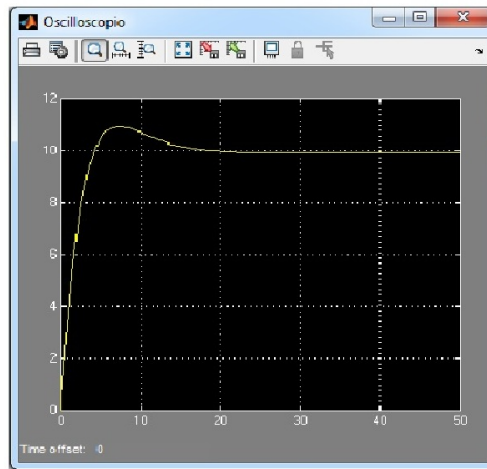


Figura 4.8: Risposta al gradino con  $K_p = 500, z_0 = 0.2, p_0 = 0.02$

avendo aumentato il valore dello zero siamo in presenza di un overshoot che comunque rispetta i vincoli di progetto ( $\sim 9\%$ ) così come  $t_r \sim 3,2$  s ed un errore a regime  $\sim 1\%$ .

## 4.5 Osservazioni conclusive sulla simulazione

Tramite l'utilizzo di *Simulink* siamo riusciti a raggiungere pressoché tutti i risultati ottenuti nell'analisi capitoli precedenti. Come già fatto il semplice controllo proporzionale non è stato riportato a causa del fatto che valori troppo elevati di tale costante impediscono la realizzabilità meccanica di tale sistema (eccessiva prontezza).

Per raggiungere buone prestazioni è sufficiente utilizzare un controllore di tipo **PI**, senza la necessità di aggiungere la componente derivativa (perché i risultati sono già buoni). Anche l'introduzione della rete ritardatrice (con i parametri ricavati dal luogo delle radici) permette di ottenere un buon comportamento dinamico anche in presenza di una piccola sovraelongazione presente per via dell'introduzione dello zero (più grande è il valore dello zero, maggiore la sovraelongazione), ma comunque i valori considerati mantengono la risposta entro i vincoli progettuali.

## Capitolo 5

# Conclusioni

Riepilogando il tutto, in questa tesi si é voluto modellizzare, analizzare e simulare un sistema *cruise control*. Tra i vari risultati ottenuti cerchiamo ora di condurre un'analisi sistematica basata su alcune caratteristiche che si riferiscono piú ad un aspetto pratico rispetto a quello prettamente teorico condotto fino a qui.

Tra le varie caratteristiche da considerare vi sono:

- **prestazioni:** ogni controllore deve soddisfare i requisiti per i quali é necessario il suo impiego;
- **semplicitá di progetto:** legata ai costi di progettazione del controllore;
- **semplicitá realizzativa:** legata ai costi di produzione e alla sua affidabilitá.

Per quanto riguarda il primo punto nella nostra analisi complessiva abbiamo dedotto che l'unico controllore non adatto al soddisfacimento delle nostre specifiche é rappresentato dal compensatore proporzionale **P**. Esso infatti permetteva di rispettare i vincoli progettuali solamente per via teorica, in quanto si riscontrava un'eccessiva prontezza nella risposta a cui sarebbero seguite delle sollecitazioni troppo elevate per il nostro veicolo; pertanto tale tipo di controllo é stato omesso anche nelle successive analisi.

I controllori **PI** e **PID** permettevano di ottenere un buon comportamento dinamico con la peculiaritá che il secondo permette di ottenere una risposta piú "morbida" rispetto al primo. Analogamente l'introduzione della rete ritardatrice ha portato al soddisfacimento dei requisiti, fornendo un modo alternativo alla realizzazione del compensatore.

La semplicitá di progetto ha un'importanza non trascurabile per quanto concerne il costo finale del controllore. Pertanto le metodologie che solitamente si preferiscono sono quelle del controllo *proporzionale-integrativo* e *proporzionale integrativo derivativo*: quest'ultimo presenta rispetto al primo una componente derivativa la quale richiede sicuramente una fase in piú e dunque il controllore di tipo **PI** é da preferire. Anche la rete ritardatrice gode di una

buona semplicitá di progetto anche se solitamente comporta qualche osservazione in piú per cui sulla base delle considerazioni fatte in questa tesi risulta piú appropriato l'utilizzo di un compensatore *proporzionale-integrativo*, anche dal fatto che essendo piú semplice ha meno probabilitá di essere esposto all'inconvenienza dei guasti.

# Bibliografia

- [1] M.Bisiacco, S.Braghetto, *Teoria dei sistemi dinamici*, Esculapio collana Progetto Leonardo, Bologna, ottobre 2010;
- [2] M.Bisiacco, M.E. Valcher, *Controlli automatici*, Libreria Progetto Padova, Padova, 2008;
- [3] Cruise control <http://ctms.engin.umich.edu/CTMS/index.php?example=CruiseControl&section;>
- [4] Adaptive Cruise Control [http://www.daf.eu/SiteCollectionDocuments/Infosheets/2012/Safety-comfort/ACC'\\_IT.pdf;](http://www.daf.eu/SiteCollectionDocuments/Infosheets/2012/Safety-comfort/ACC'_IT.pdf)
- [5] A.De Santis, M.Cacciaglia,C.Sagrese, *Sistemi, automazione e organizzazione della produzione*, Calderini, Milano, gennaio 2004;