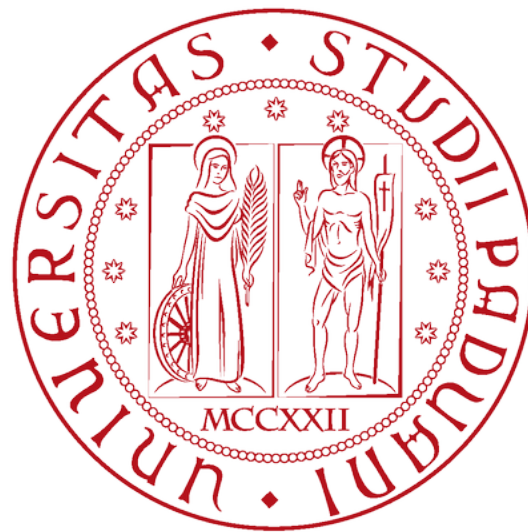


UNIVERSITÀ DEGLI STUDI DI PADOVA

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

TESI DI LAUREA TRIENNALE IN
INGEGNERIA DELL'INFORMAZIONE



Modellizzazione e progettazione
di un Cruise Control per automobili
in ambiente Matlab/Simulink

Laureando

Ettore Santoiemma

matr. 1201435

Relatore

Prof. Luca Schenato

ANNO ACCADEMICO 2021/22

Indice

Introduzione	5
1 Analisi del sistema	6
1.1 Configurazione fisica	6
1.2 Equazione del modello e Parametri del sistema	6
1.3 Funzione di trasferimento e Risposta impulsiva	7
1.3.1 Implementazione della FdT in Matlab	9
1.4 Requisiti progettuali	9
1.5 Analisi in catena aperta	10
1.5.1 Risposta al gradino	10
1.5.2 Rappresentazione di poli e zeri ad anello aperto	10
1.5.3 Diagramma di Bode	11
2 Progetto del controllore	12
2.1 Analisi in catena chiusa	12
2.2 Considerazioni preliminari	12
2.3 Controllori PID	13
2.3.1 Tipologia di controllore proporzionale P	14
2.3.2 Tipologia di controllore proporzionale PI	15
2.3.3 Tipologia di controllore proporzionale PID	18
2.4 Considerazioni conclusive	18
3 Analisi mediante luogo delle radici	20
3.1 Considerazioni preliminari	20
3.2 Luogo delle radici	20
3.3 Introduzione rete ritardatrice	23
3.4 Considerazioni conclusive	24
4 Analisi in frequenza	26
4.1 Considerazioni preliminari	26
4.2 Considerazioni catena aperta e catena chiusa	27
4.3 Rete ritardatrice	28
4.4 Considerazioni conclusive	29
5 Analisi in Simulink	30
5.1 Considerazioni preliminari	30
5.2 Creazione modello Simulink	30
5.3 Simulazione catena aperta	32

5.4	Simulazione catena chiusa	32
5.4.1	Controllore PI	32
5.4.2	Rete ritardatrice	33
5.5	Considerazioni conclusive	34
6	Conclusioni	36
	Riferimenti bibliografici	38

Introduzione

L'Automatic Cruise Control è un dispositivo elettronico presente in molti autoveicoli moderni. Si tratta di un perfetto esempio di un sistema controllato in retroazione (*feedback*) presente nella vita quotidiana, in quanto è responsabile della regolazione automatica della velocità. L'utente può selezionare una velocità di riferimento ed essa viene mantenuta costante nel tempo anche in presenza di disturbi esterni.

Esistono due tipologie: il **Cruise Control** e l'**Adaptive Cruise Control**.

Il *primo* modello permette al conducente di impostare una velocità in modo che rimanga costante, fino alla disattivazione del dispositivo o alla modifica della velocità. Il conducente può agire sull'acceleratore per incrementare la velocità, ma in seguito al rilascio del pedale il dispositivo riporta il veicolo alla velocità preimpostata; invece, nel caso in cui il conducente azioni il freno, la velocità memorizzata viene cancellata e il dispositivo smette di influenzare l'andatura. Successivamente sarà necessario impostare una nuova velocità di riferimento. Il *secondo* tipo può decrementare la velocità del veicolo nel caso in cui si presentino degli ostacoli di fronte alla vettura. Il conducente ha la possibilità di modificare la distanza (di conseguenza anche la velocità dell'auto) dalle vetture che lo precedono in modo da garantire la distanza di sicurezza.

Il Cruise Control non è solo un dispositivo che agevola la guida del conducente, ma è anche un ottimo strumento per risparmiare carburante nei viaggi con lunghi tratti rettilinei, come ad esempio le autostrade.

La tesi in questione ha l'obiettivo di analizzare il sistema Cruise Control e le sue caratteristiche, effettuando uno studio del modello matematico in forma di sistema lineare. Si costruiranno diversi modelli di controllore, usando alcune tecniche studiate durante il percorso universitario (*PID*, *Luogo delle Radici* e *Reti correttrici*). Le caratteristiche del dispositivo di controllo verranno delineate usando l'analisi dinamica in catena aperta (*open-loop*) e l'analisi dinamica in catena chiusa (*closed-loop*) per poter soddisfare le specifiche progettuali ipotizzate.

In parallelo allo studio teorico verranno utilizzati il software *Matlab* e l'ambiente *Simulink* per progettare, calcolare e simulare l'andamento del sistema, in risposta ad un riferimento e un controllore progettato con differenti metodi.

Capitolo 1

Analisi del sistema

1.1 Configurazione fisica

Il sistema Cruise Control ha l'obiettivo di mantenere la velocità del veicolo v costante bilanciando gli eventuali disturbi esterni, come il vento o l'aumento dell'inclinazione del piano stradale. Per ottenere una modellizzazione fisica del controllore si opera innanzitutto un'astrazione del problema e ne semplifichiamo la dinamica.

Consideriamo una vettura di massa m puntiforme e ipotizziamo che su di essa agisca una forza motrice controllata u e una forza di attrito $F_a = bv$ che si oppone ad essa (l'attrito b è stato ipotizzato come la somma dell'attrito *volvente* e dell'attrito *viscoso*).

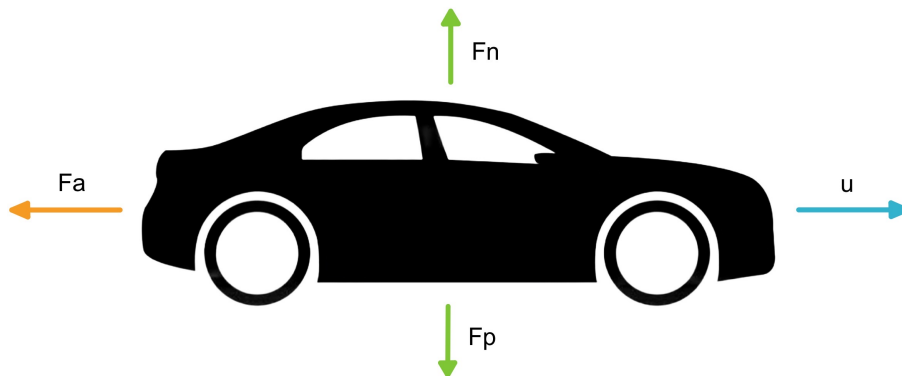


Figura 1.1: Modello grafico del sistema

Da si può anche notare la presenza della forza peso $F_p = mg$ e della reazione vincolare $F_n = -F_p$ che agiscono sul sistema. In questo modello semplificato si assume di poter controllare direttamente la forza motrice u generata dal motore che permette il movimento delle ruote sulla strada e di conseguenza il movimento del veicolo.

1.2 Equazione del modello e Parametri del sistema

Utilizzando il secondo principio della dinamica $F = ma$ è possibile descrivere il modello semplificato, infatti, sommando algebricamente le forze applicate lungo gli assi, si può notare che la forza peso viene contro bilanciata dalla reazione vincolare del piano di appoggio, di conseguenza non c'è spostamento verticale. Invece si ricava l'equazione

$$F = ma = u - bv \quad (1.1)$$

lungo l'asse orizzontale, o asse di marcia.

Al fine di ottenere il progetto si é interessati alla velocità dell'autoveicolo, di conseguenza si possono dedurre le equazioni del sistema definendo la velocità v come derivata prima dello spostamento x e l'accelerazione a come derivata seconda di x :

$$\begin{aligned} v &= \dot{x} \\ a &= \dot{v} = \ddot{x} \end{aligned} \quad (1.2)$$

Il modello matematico descrive il fenomeno in modo deterministico, in quanto dipende dall'evoluzione della velocità v nel tempo e il valore che essa assume in un determinato istante iniziale.

Per ottenere la formulazione di un modello ingresso-uscita a tempo continuo si effettua l'analisi su un sistema lineare ottenendo il seguente modello generale in spazio di stato:

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{F}\mathbf{x}(t) + \mathbf{G}u(t) \\ y(t) = \mathbf{H}\mathbf{x}(t) + \mathbf{J}u(t) \end{cases} \quad (1.3)$$

in cui nel caso specifico si impone $y = v$:

$$\begin{cases} m\dot{v} + bv = u \\ y = v \end{cases} \quad (1.4)$$

dove, nel caso generale \mathbf{F} (*matrice dinamica*) é una matrice quadrata $n \times n$, con n il numero delle variabili di stato del sistema, \mathbf{G} (*matrice di ingresso*) é una matrice $n \times m$, con m il numero di input del sistema, \mathbf{H} (*matrice di uscita*) é una matrice $p \times n$, con p il numero di output, e infine \mathbf{J} (*matrice di legame diretto ingresso-uscita*) una matrice $p \times m$.

Per poter proseguire con la modellizzazione del controllore é necessario fornire dei valori ai parametri, in modo da definire il sistema su cui verrà applicato il controllore. Per questo studio sono stati ipotizzati i seguenti dati:

\mathbf{m}	massa del veicolo	1000 [kg]
\mathbf{b}	coefficiente di attrito	50 [Ns/m]
\mathbf{u}	forza elettromotrice	500 [N]
\mathbf{r}	velocità di riferimento	10 [m/s]

Tabella 1.1: Parametri del sistema

1.3 Funzione di trasferimento e Risposta impulsiva

Prendendo in considerazione la soluzione dell'equazione differenziale del sistema (1.4), tramite risoluzione matematica si ottiene:

$$x(t) = \underbrace{e^{Ft}x(0)}_{\text{evoluzione libera}} + \underbrace{\int_0^t e^{F(t-\tau)}Gu(\tau)d\tau}_{\text{evoluzione forzata}} \quad (1.5)$$

Sostituendo la variabile $x(t)$ nell'equazione di output é possibile ricavare l'uscita come:

$$y(t) = He^{Ft}x(0) + \int_0^t [He^{F(t-\tau)}G + J\delta(t-\tau)]u(\tau)d\tau \quad (1.6)$$

dove l'integrale ottenuto rappresenta la convoluzione tra la risposta impulsiva e l'ingresso u :

$$w(t) = He^{Ft}G + J\delta(t) \quad (1.7)$$

Per semplificare i calcoli, é necessario passare dal dominio del tempo al dominio delle trasformate di Laplace s in modo da ricavare la **funzione di trasferimento** $W(s)$ associata alla risposta impulsiva $w(t)$:

$$W(s) : \mathbb{C} \rightarrow \mathbb{C} \quad W(s) = \mathcal{L}[w(t)](s) = \int_0^{+\infty} w(t)e^{-st}dt \quad (1.8)$$

Consegue che la trasformata di Laplace dell'uscita Y è:

$$Y(s) = HX(s) + JU(s) \quad \text{dove} \quad X(s) = \mathcal{L}[x(t)] \quad , \quad U(s) = \mathcal{L}[u(t)] \quad (1.9)$$

Sapendo che:

$$X(s) = (sI - F)^{-1}x(0) + (sI - F)^{-1}GU(s) \quad (1.10)$$

si determina:

$$Y(s) = H(sI - F)^{-1}x(0) + \underbrace{[H(sI - F)^{-1}G + J]}_{\text{funzione di trasferimento}} U(s) \quad (1.11)$$

Da quest'ultima si ottiene la funzione di trasferimento:

$$W(s) := H(sI - F)^{-1}G + J = \mathcal{L}[He^{Ft}G + J\delta(t)] = \mathcal{L}[w(t)] \quad (1.12)$$

Nel caso specifico in esame, applicando la trasformata di Laplace a (1.4), dopo alcuni passaggi algebrici si ricava l'equazione:

$$U(s) = msV(s) + bV(s) \quad (1.13)$$

da cui si ricava la funzione di trasferimento del modello:

$$W(s) = \frac{V(s)}{U(s)} = \frac{1}{ms + b} \quad (1.14)$$

1.3.1 Implementazione della FdT in Matlab

Utilizzando il software *Matlab* si possono definire le variabili e le matrici del sistema con i valori definiti in precedenza nella tabella (1.1):

```
% Definizione variabili
m = 1000; % massa del veicolo
b = 50; % coefficiente di attrito
u = 500; % forza elettromotrice
r = 10; % velocità di riferimento

% Definizione delle matrici
F = -b/m;
G = 1/m;
H = 1;
J = 0;
```

Figura 1.2: Definizione variabili e matrici

Si esegue il comando predefinito in *Matlab* `ss` e si definisce la variabile complessa di Laplace s . Di conseguenza si ottiene il modello in spazio di stato (*state-space*) e la funzione di trasferimento:

```
state_space = ss(F,G,H,J); % inizializzazione modello
s = tf('s'); % variabile di Laplace
W = 1/(m*s + b) % funzione di trasferimento
```

Figura 1.3: Definizione modello

1.4 Requisiti progettuali

Per modellare il controllore, che dovrà soddisfare l'analisi e la simulazione del sistema di controllo in esame, bisogna definire dei limiti progettuali da rispettare.

Come definito nella Tabella (1.1), si ipotizza che il veicolo sia soggetto a una forza motrice $u = 500$ N che fa incrementare la velocità fino al valore 10 m/s.

Inoltre si impone che questo valore debba essere raggiunto in non più di 5 s, supponendo che l'abitacolo parta con velocità iniziale $v(0) = 0$ m/s.

In aggiunta è necessario definire un margine di errore sul valore che può assumere la velocità. Consideriamo una sovraelongazione (*overshoot*) del 10%, un errore a regime (*steady-state error*) del 2% e un tempo di assestamento (*settling time*) non superiore a 15 s:

- $t_r < 5$ s
- $M_p < 10\%$
- $e_r < 2\%$
- $t_s < 15$ s

1.5 Analisi in catena aperta

1.5.1 Risposta al gradino

Per poter osservare la risposta del sistema, si inserisce come riferimento una funzione gradino e si valuta successivamente l'evoluzione forzata in uscita. Nel caso in esame si definisce come input una forza $u = 500$ N e, sfruttando il comando *step* in *Matlab*, si ricava il grafico che segue:

```
step(u*W) % risposta al gradino con u = 500 N
```

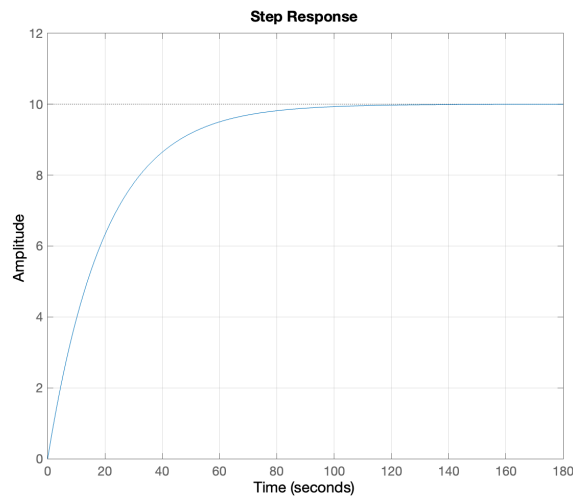


Figura 1.4: Risposta al gradino

In un sistema in catena aperta, al manifestarsi di errori in uscita o disturbi esterni, non c'è la possibilità di modificare né l'ingresso né l'uscita per correggerli. Perciò il corretto andamento del sistema in esame dipende esclusivamente dalla correttezza del modello matematico ricavato e dalla precisione dei dati utilizzati. Nella pratica questa soluzione non viene mai sfruttata, poiché non è possibile conoscere nel dettaglio le varie componenti e una modellizzazione accurata del sistema risulterebbe eccessivamente complessa e non lineare.

1.5.2 Rappresentazione di poli e zeri ad anello aperto

Analizzando la funzione di trasferimento (1.14) del modello in esame, si evidenzia che il sistema Cruise Control possiede un solo polo p_0 in corrispondenza del valore $-b/m$.

È possibile visualizzarlo nella mappa zeri-poli usando il comando in *Matlab*:

```
pzmap(W) % mappa zeri-poli nel piano complesso
```

Si può osservare, da Figura (1.5), che p_0 è situato alla sinistra dell'asse immaginario, di conseguenza si trova nella parte di piano tale per cui $\Re(s) < 0$. Da questo primo approccio si nota che il modello è BIBO-stabile (*Bounded Input Bounded Output*) e risulta poco reattivo nella risposta al gradino ($t_r = 46s$) in quanto il polo presenta il modulo molto piccolo (uguale a 0,05).

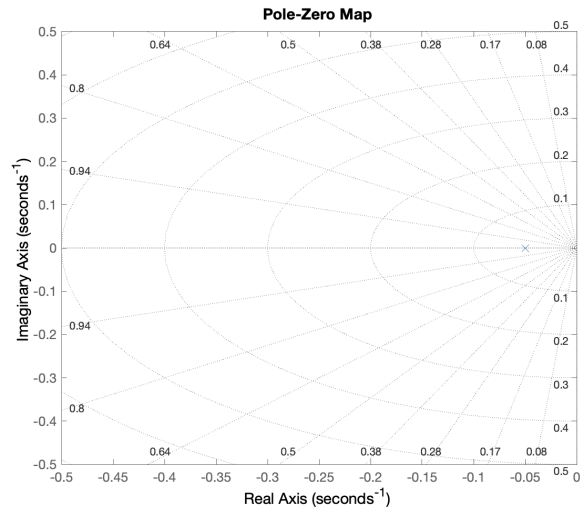


Figura 1.5: Mappa di poli e zeri ad anello aperto

Non essendo possibile agire direttamente sui parametri del sistema osservato, per raggiungere le specifiche progettuali imposte è necessario progettare un sistema di controllo capace di modificare la posizione dei poli e degli zeri della $W(s)$.

1.5.3 Diagramma di Bode

Parallelamente alla risposta nel dominio del tempo è possibile rappresentare l'andamento del sistema (a catena aperta) nel dominio della frequenza.

Tramite *Matlab* è possibile visualizzare il grafico:

```
bode(W) % diagramma di Bode
```

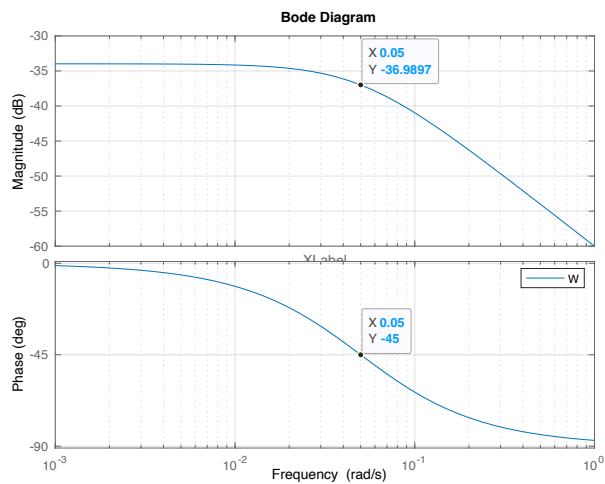


Figura 1.6: Diagramma di Bode

Si osserva che nel punto a frequenza $\omega = b/m = 5 \cdot 10^{-2}$ il modulo diminuisce di 3 dB, e la corrispondente fase è -45 gradi. Inoltre si osserva che ad alte frequenze il modulo scende con una pendenza di -20 dB/dec.

Capitolo 2

Progetto del controllore

2.1 Analisi in catena chiusa

La retroazione, o retroregolazione (*feedback*), è la capacità di un sistema dinamico di modificare le sue stesse caratteristiche, tenendo conto dei risultati. Per poter ridurre notevolmente il tempo di salita, ovvero perfezionare le caratteristiche dinamiche, si confrontano l'output e l'input del sistema, inoltre bisogna aggiungere un compensatore che anticipi il processo. Questa modifica può creare alcuni problemi, come la formazione di una sovraelongazione (*overshoot*), ma solitamente si tratta di un adeguato compromesso per il problema della realizzazione di un controllore.

Si effettua uno studio del sistema in catena chiusa e si progetta un controllore in grado di modificare zeri e poli della funzione di trasferimento per raggiungere i limiti imposti all'inizio.

2.2 Considerazioni preliminari

Si analizza un sistema con retroazione unitaria negativa, rappresentato da un processo $G(s)$ e preceduto da un controllore $C(s)$ con un grado di libertà, come quello in Figura (2.1):

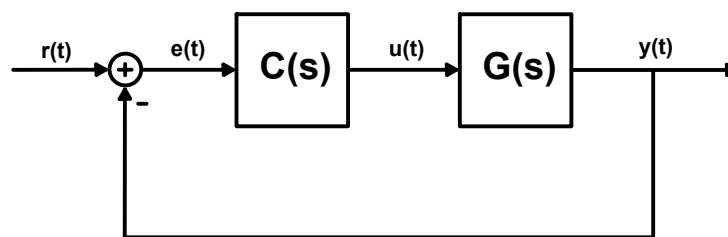


Figura 2.1: Schema di controllo in retroazione unitaria

L'obiettivo della componente $C(s)$ è modificare le caratteristiche del sistema per poter ottenere le specifiche richieste necessarie a risolvere il problema.

Il feedback descritto in Figura (2.1) è negativo, in quanto il segnale $y(t)$ viene retroazionato con segno opposto. Infine si confronta con il segnale $r(t)$ (segnale di riferimento), per ottenere l'errore $e(t)$ del sistema:

$$e(t) = r(t) - y(t) \quad (2.1)$$

Passando nel dominio delle frequenze otteniamo:

$$E(s) = R(s) - Y(s) \quad (2.2)$$

mentre, usando il precedente schema:

$$Y(s) = C(s)G(s)E(s) \quad (2.3)$$

si ottiene l'equazione:

$$Y(s) = C(s)G(s)(R(s) - Y(s)) \quad (2.4)$$

Infine, dall'equazione (2.4) risulta la funzione di trasferimento:

$$H(s) := \frac{C(s)G(s)}{1 + C(s)G(s)} \quad (2.5)$$

L'effetto del compensatore $C(s)$ è quello di cambiare il guadagno di Bode in catena aperta e di introdurre zeri e poli nel sistema.

2.3 Controllori PID

I controllori **PID** sono dei dispositivi che realizzano la combinazione lineare di tre trasformazioni sul segnale al loro ingresso (nel caso in esame l'errore $e(t)$). Le azioni sono: proporzionale (**P**), integrativa (**I**) e derivativa (**D**).

La loro funzione di trasferimento è la seguente:

$$C_{PID}(s) = K_p + \frac{K_i}{s} + K_d s \quad (2.6)$$

dove K_p , K_i e K_d sono i guadagni di ognuna delle tre componenti del compensatore.

Ciascuna parte induce un diverso tipo di effetto sul sistema:

- **proporzionale**: aumentandolo si riduce il tempo di salita t_r del sistema;
- **integrale**: si usa per eliminare l'errore a regime e_r in risposta ad un ingresso a gradino, ma può rallentare la risposta (decrementa il tempo di salita t_r);
- **derivativo**: è usato per aumentare la stabilità del sistema, riducendo le sovraelongazioni M_p e migliorando la risposta nel regime transitorio;

I compensatori più semplici che si possono realizzare sono il **PI** ($K_d = 0$), il **PD** ($K_i = 0$), il **P** ($K_i = K_d = 0$) e l'**I** ($K_p = K_d = 0$), mentre il più complesso è il **PID**.

I compensatori possono essere simulati in Matlab come funzioni di trasferimento:

```
Kp = 1;      % guadagno proporzionale
Kd = 1;      % guadagno derivativo
Ki = 1;      % guadagno integrativo
C = Kp + Ki/s + Kd*s
```

o alternativamente si può usare il comando già presente in *Matlab*:

$$C = \text{pid}(K_p, K_i, K_d)$$

Nella pratica è fondamentale prestare attenzione alla realizzabilità fisica del controllore, ad esempio il derivatore ideale \mathbf{D} non è realizzabile. Nello specifico, non si possono costruire controllori con più zeri che poli, poiché nascono da funzioni razionali improprie.

2.3.1 Tipologia di controllore proporzionale P

Si consideri ora la funzione di trasferimento del modello in catena chiusa di Figura (2.1). Implementando il controllore \mathbf{P} si ricava una nuova funzione di trasferimento:

$$H(s) = \frac{K_p}{ms + (b + K_p)} \quad (2.7)$$

dove $H(s)$ rimane una funzione del primo ordine e perciò non presenterà sovraelongazioni.

Con *Matlab* è possibile progettare il controllore con la seguente modalità:

```
m = 1000; % massa del veicolo
b = 50; % coefficiente di attrito
r = 10; % velocità di riferimento

s = tf('s');
W = 1/(m*s + b); % funzione di trasferimento

Kp = 100; % guadagno proporzionale
C = pid(Kp); % controllore P

H = feedback(C*W,1); % comando per retroazionare
% il sistema (1 perchè è unitaria)

t = 0:0.1:20; % vettore del tempo
step(r*H,t); % risposta al gradino
```

In questo caso si ottiene la funzione di trasferimento:

$$H(s) = \frac{100}{1000s + 150} \quad (2.8)$$

e il relativo grafico della risposta al gradino:

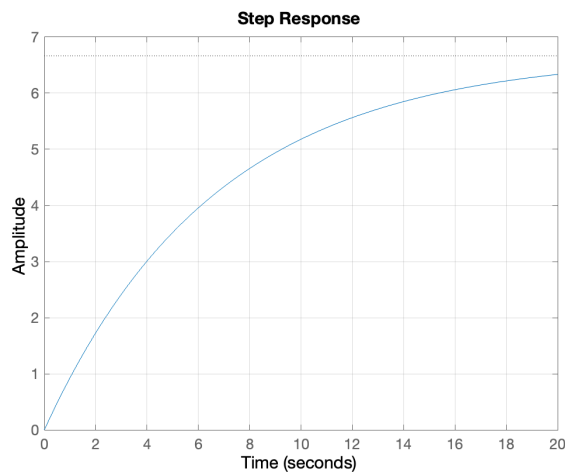


Figura 2.2: Risposta al gradino con $K_p = 100$

Dalla Figura (2.2) é possibile osservare che l'aggiunta di un controllore proporzionale diminuisce il tempo di salita, rendendo la risposta al gradino piú veloce.

In questo specifico caso il guadagno $K_p = 100$ non soddisfa i requisiti richiesti del tempo di salita t_r e dell'errore a regime e_r . L'unica soluzione è aumentare il valore.

Scegliendo come valore di $K_p = 5000$ si ottiene la risposta al gradino:

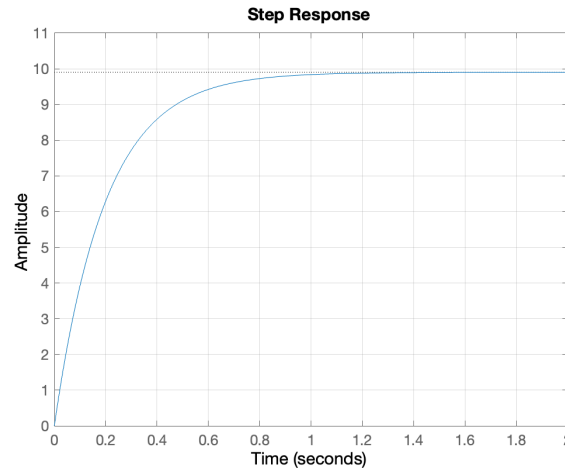


Figura 2.3: Risposta al gradino con $K_p = 5000$

In Figura (2.3) si evidenzia che i limiti imposti vengono perfettamente soddisfatti, ricavando $e_r \sim 0$ e $t_r = 0,435$ s.

Tuttavia nel caso reale é necessario evidenziare alcuni particolari. Questa soluzione non può essere accettabile a livello fisico, poiché corrisponde a variare la velocità del veicolo da 0 m/s a 9 m/s in un tempo molto breve (t_r), quindi con un'accelerazione elevatissima e con costi decisamente insostenibili.

2.3.2 Tipologia di controllore proporzionale PI

Nella precedente analisi é risultato che l'unico limite fisico é mantenere il guadagno K_p entro determinati limiti. Al tempo stesso si vuole eliminare l'errore a regime per ottenere un corretto tracciamento del gradino. Per renderlo possibile si aggiunge al compensatore **P** la componente integrale.

La funzione di trasferimento del controllore **PI** si ricava imponendo $K_d = 0$:

$$C(s) = K_p + \frac{K_i}{s} \quad (2.9)$$

Calcolando di nuovo la funzione di trasferimento del sistema in catena chiusa risulta:

$$H(s) = \frac{K_p s + K_i}{m s^2 + (b + K_p) s + K_i} \quad (2.10)$$

Per rappresentare la nuova risposta al gradino si utilizza il seguente codice *Matlab*, simile a quello usato in precedenza:


```

m = 1000; % massa del veicolo
b = 50; % coefficiente di attrico
r = 10; % velocità di riferimento

s = tf('s');
W = 1/(m*s + b); %funzione di trasferimento open-loop

Kp = 500; % guadagno proporzionale
Ki = 5; % guadagno integrativo
C = pid(Kp,Ki);

H = feedback(C*W,1); % comando per retroazionare
% il sistema (1 perchè unitaria)

t = 0:0.1:20; % vettore del tempo
figure;
step(r*H,t); % risposta al gradino

```

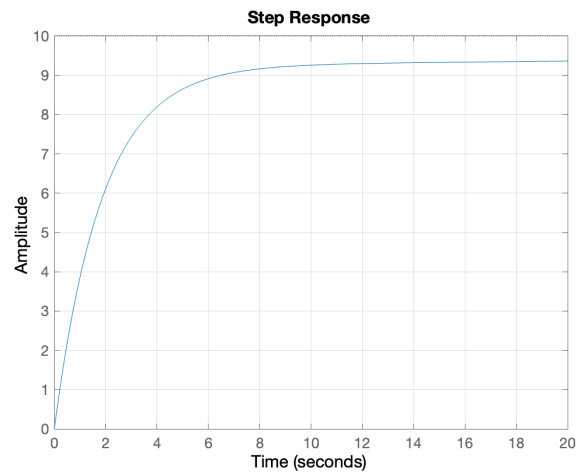
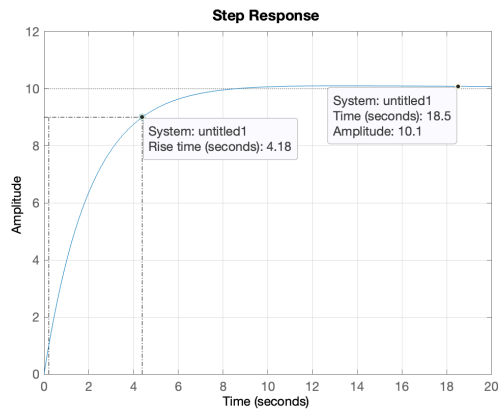


Figura 2.4: Risposta al gradino con $K_p = 500$ e $K_i = 5$

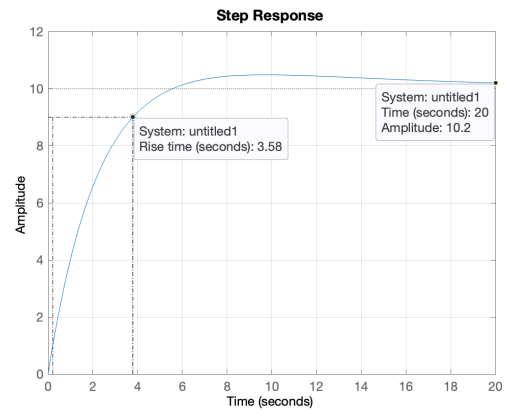
Osservando ora la Figura (2.4) é possibile vedere che le specifiche non vengono rispettate, di conseguenza bisogna modificare i valori di K_p e K_i .

Incrementando il K_p si può ridurre il tempo di salita t_r , prestando sempre attenzione alla realizzabilità fisica, mentre aumentando K_i si generano sovraelongazioni M_p , che secondo le specifiche iniziali non devono superare il 10% del valore del gradino in ingresso.

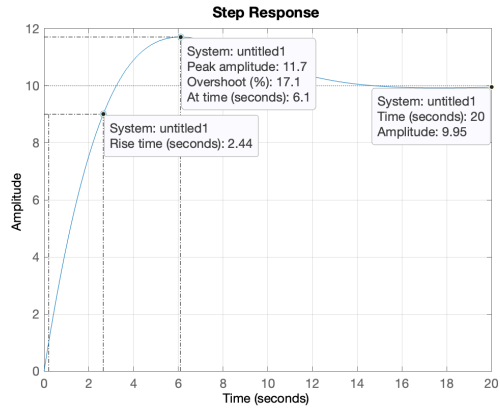
Per rispettare le richieste al meglio sono stati scelti i valori $K_p = 1000$ e $K_i = 50$, come mostrato in Figura (2.5d). Si noti che non sono presenti errore a regime e sovraelongazione, inoltre il tempo di salita, che risulta pari a 2,2 s come riportato in Figura (2.6), permette la realizzazione fisica del sistema.



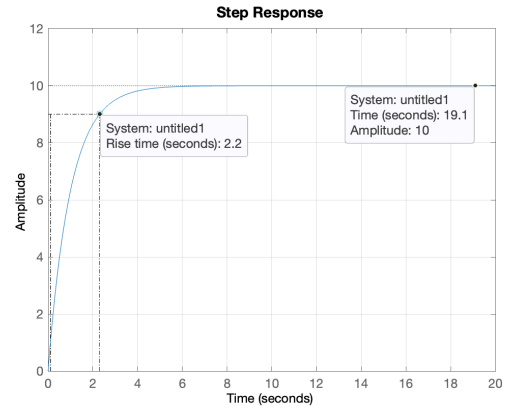
(a) $K_p = 500$ $K_i = 30$



(b) $K_p = 500$ $K_i = 50$



(c) $K_p = 500$ $K_i = 150$



(d) $K_p = 1000$ $K_i = 50$

Figura 2.5: Risposta al gradino con diversi K_p e K_i

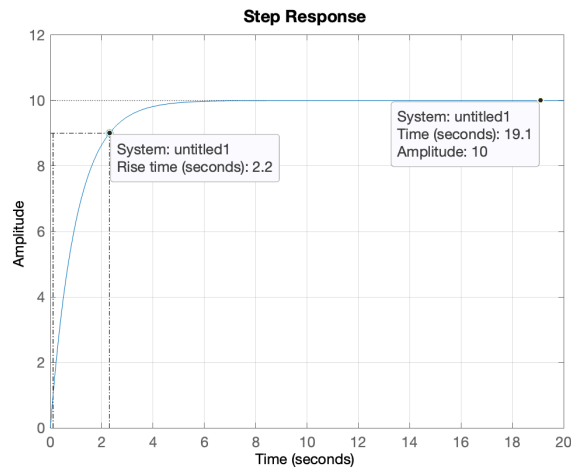


Figura 2.6: Risposta al gradino con $K_p = 1000$ e $K_i = 50$

2.3.3 Tipologia di controllore proporzionale PID

Per migliorare il comportamento in risposta al gradino può essere implementata anche l'azione derivativa con un guadagno K_d , salvo che l'aggiunta di questa componente non faccia venire meno il rispetto per i limiti progettuali.

Modificando nuovamente l'equazione (2.6) si ricava la funzione di trasferimento del sistema in catena chiusa:

$$H(s) = \frac{K_d s^3 + K_p s + K_i}{(m d + K_i) s^2 + (b + K_p) s + K_i} \quad (2.11)$$

e utilizzando di nuovo i comandi in Matlab:

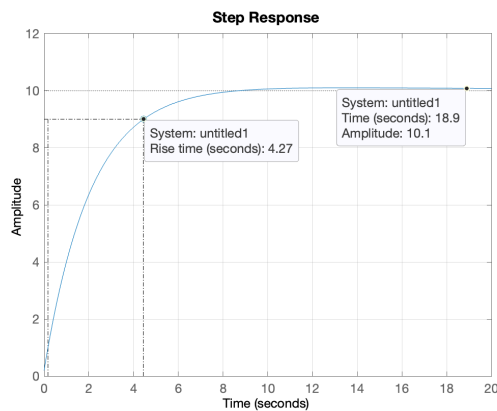
```
m = 1000; % massa del veicolo
b = 50; % coefficiente di attrico
r = 10; % velocità di riferimento

s = tf('s');
W = 1/(m*s + b); %funzione di trasferimento open-loop

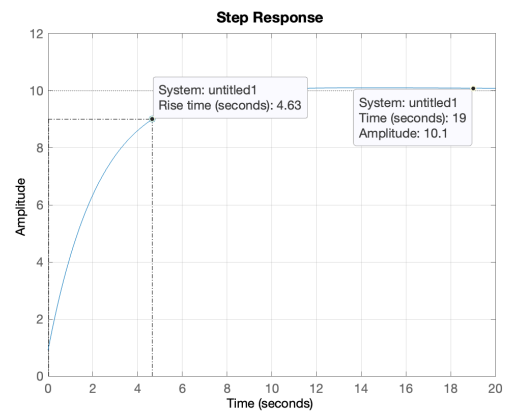
Kp = 500; % guadagno proporzionale
Ki = 30; % guadagno integrativo
Kd = 20; % guadagno derivativo
C = pid(Kp,Ki,Kd);

H = feedback(C*W,1); % comando per retroazionare
% il sistema (1 perchè unitaria)

t = 0:0.1:20; % vettore del tempo
figure;
step(r*H,t); % risposta al gradino
```



(a) $K_p = 500$ $K_i = 30$ $K_d = 20$



(b) $K_p = 500$ $K_i = 30$ $K_d = 100$

Figura 2.7: Risposta al gradino con diversi K_p , K_i , K_p

É possibile determinare un valore per ciascun guadagno, nel caso in esame i parametri che rispettano le specifiche sono: $K_p = 1000$, $K_i = 50$ e $K_p = 30$.

Da questi ultimi si ottiene la seguente risposta al gradino:

2.4 Considerazioni conclusive

Analizzando i tre tipi di compensatori **PID**, nello specifico **P**, **PI** e **PID**, si riscontra che incrementare semplicemente il guadagno del controllore proporzionale (**P**), nonostante

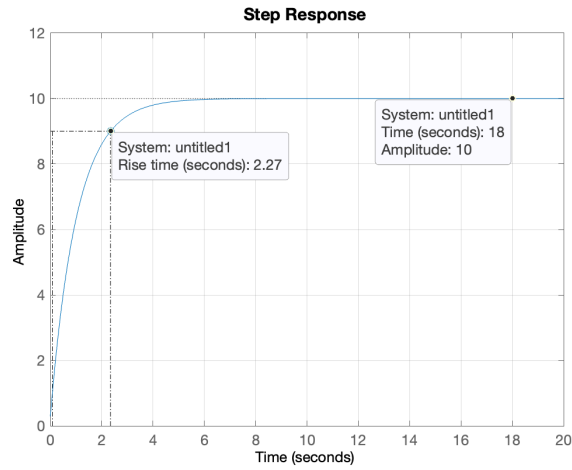


Figura 2.8: Risposta al gradino con $K_p = 1000$, $K_i = 50$, $K_d = 30$

diminuisca considerevolmente il tempo di salita, risolve il problema solamente dal punto di vista teorico. Infatti, nel caso si ottenga un guadagno eccessivamente grande, risulta impossibile l'implementazione pratica.

È sufficiente aggiungere ad esso la componente integrativa (**PI**) in modo da ricavare una risposta al gradino ottimale, nonostante compaiano delle sovraelongazioni.

Si può inoltre osservare che implementando anche il compensatore derivativo (**PID**) le limitazioni progettuali rimangono pienamente soddisfatte, tuttavia è preferibile un controllore **PI** per la semplicità di realizzazione e per il costo inferiore.

Capitolo 3

Analisi mediante luogo delle radici

3.1 Considerazioni preliminari

Nel Capitolo (2) viene esaminato il sistema partendo dall'analisi in catena aperta, per poi passare alla catena chiusa con una retroazione unitaria negativa.

In questo capitolo si analizza il problema mediante il tracciamento del luogo delle radici (*root locus*). Questa modalità consente di individuare con precisione la posizione dei poli e degli zeri della funzione di trasferimento in catena chiusa $H(s)$, con le rispettive molteplicità, e l'evoluzione in funzione dell'aumento del guadagno K_p .

Il luogo delle radici traccia l'andamento del sistema partendo dai poli e raggiungendo asintoticamente gli zeri. Secondo i criteri di Nyquist e Routh si può ricavare la stabilità del sistema e il valore di K_p per il quale esso garantisce la BIBO stabilità del sistema. Inoltre, il criterio di Nyquist consente di ottenere delle informazioni riguardo la **robustezza** della stabilità.

3.2 Luogo delle radici

Si prenda in considerazione la funzione di trasferimento di un sistema retroazionato (2.5). Sostituendo $C(s) = K_p$ (controllore proporzionale) e $G(s) = p(s)/d(s)$ (funzione di trasferimento in catena aperta), con $p(s)$ e $d(s)$ **monici** e **coprimi** tra di loro, si ottiene la seguente formula:

$$H(s) = \frac{K_p \cdot p(s)}{q(s) + K_p \cdot p(s)} \quad (3.1)$$

Il luogo delle radici rappresenta i punti che, al variare di $K_p \neq 0$, annullano il denominatore della funzione di trasferimento $H(s)$:

$$q(s) + K_p \cdot p(s) \quad (3.2)$$

Tramite questo metodo è possibile ricavare alcune informazioni molto importanti come:

- se uno o più poli di $H(s)$ appartengono alla parte destra del piano complesso, sicuramente il sistema risulterà instabile;
- minore è la distanza di un polo dall'asse immaginario, maggiore sarà la sua rilevanza nella risposta del sistema;

- i punti del luogo delle radici che intersecano l'asse immaginario vengono chiamati punti critici, in quanto il sistema oscilla tra la stabilità e l'instabilità

Tornando al sistema (1.14) ipotizzato nella Sezione (1.3), sostituendo l'espressione della catena aperta $G(s) = 1/(ms + b)$ nella funzione di trasferimento in catena chiusa (3.1), si ottiene:

$$H(s) = \frac{K_p}{ms + (b + K_p)} \quad (3.3)$$

Si impostando le limitazioni progettuali ricavate analizzando i limiti imposti a livello teorico e si inseriscono nella implementazione in *Matlab*.

Prendendo in considerazione un overshoot $M_p = 10\% = 0,1$ e un rise time massimo $t_r = 5$ s, si può ricavare un coefficiente di smorzamento ξ e una pulsazione naturale ω_n pari a:

$$\xi \geq \sqrt{\frac{\ln^2(M_p)}{\pi^2 + \ln^2(M_p)}} = 0,59 \sim 0,6 \quad \text{e} \quad \omega_n \geq \frac{1,8}{t_r} = 0,36 \text{rad/s} \quad (3.4)$$

Si può che inoltre scrivere la sovraelongazione M_p in funzione dello smorzamento ξ tramite la formula:

$$M_p \approx e^{-\frac{\pi\xi}{\sqrt{1-\xi^2}}} \quad (3.5)$$

Definendo i valori calcolati come parametri in *Matlab* si può utilizzare il seguente codice:

```
m = 1000; % massa del veicolo
b = 50; % coefficiente di attrito
r = 10; % velocità di riferimento

s = tf('s');
W = 1/(m*s + b); %funzione di trasferimento open-loop

rlocus(W);
axis([-0.5, 0.11, -0.5, 0.5]);
sgrid(0.6, 0.36);

rlocus(W);
```

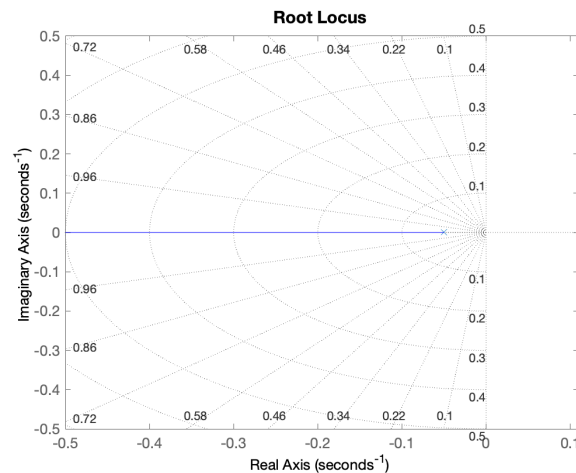


Figura 3.1: Luogo delle radici

Come si evidenzia dal grafico, l'unico polo appartenente al sistema ha valore $-0,05$ ed è rappresentato nel semipiano a sinistra dell'asse immaginario da una \times . Si noti anche che la

funzione di trasferimento $G(s)$ non presenta nessun polo, si conseguenza il ramo del luogo termina a $-\infty$.

Mediante il comando *sgrid* in *Matlab*, fornendogli come parametri il coefficiente di smorzamento ξ e la pulsazione naturale ω_n , vengono disegnati tratteggiati nel luogo:

- una **semiellisse** che individua tutti i punti dove la pulsazione naturale ha valore costante $\omega_n = 0,36$ rad/s
- due **segmenti** lungo i quali il coefficiente di smorzamento vale $\xi = 0,6$

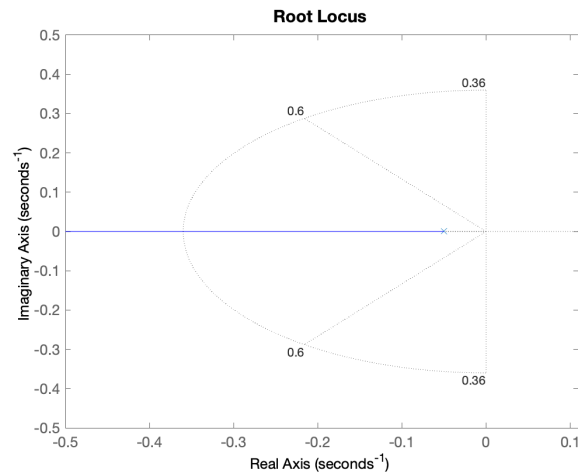


Figura 3.2: Luogo delle radici con comando *sgrid*

Sapendo che e tutto il luogo delle radici si trova nel semipiano sinistro del piano complesso, si può affermare con certezza che il sistema risulta stabile per tutti i valori che può assumere il guadagno K_p .

Servendosi del comando *rlocfind* é possibile scegliere un punto del luogo delle radici e conoscere automaticamente il valore di K_p necessario per ottenere quel risultato:

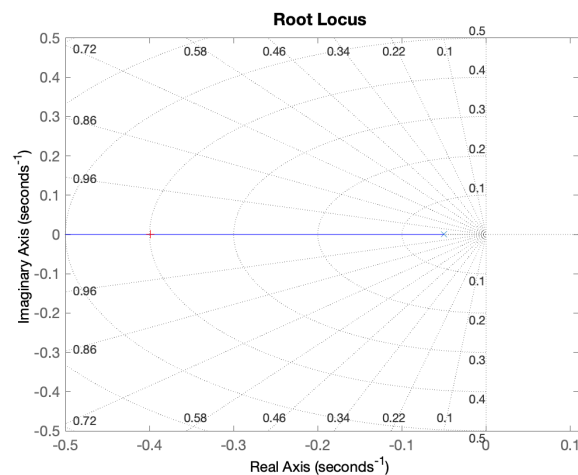


Figura 3.3: Luogo delle radici con comando *rlocfind*

Il software *Matlab* inoltre stamperá il seguente output:

```
[Kp,poles] = rlocfind(W);
```

```
selected_point = -0.4006 + 0.0031i
Kp = 350.6357
poles = -0.4006
```

Il valore ricavato si puó direttamente sostituire nella funzione di trasferimento del sistema in catena chiusa $H(s)$ per visualizzarne la risposta al gradino:

```
Kp = 350.6357;
H = feedback(Kp*W,1);
t = 0:0.1:20;
figure
step(r*H,t)
```

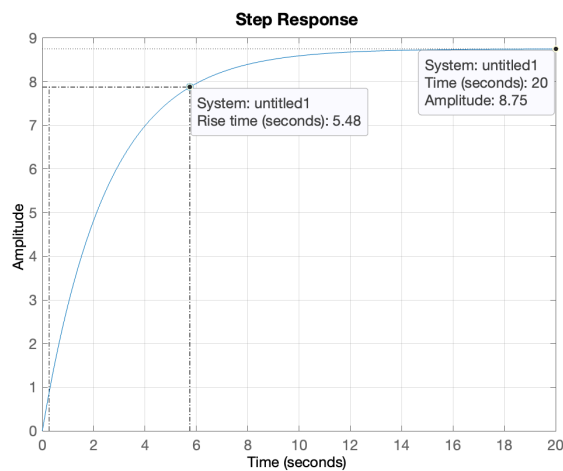


Figura 3.4: Risposta al gradino con $K_p = 350,6357$

Di conseguenza il compensatore ricavato mediante luogo delle radici risulta analogo ad un compensatore \mathbf{P} e conserva gli stessi problemi sull'errore a regime e sull'effettiva realizzazione pratica legata al tempo di salita e al costo economico.

3.3 Introduzione rete ritardatrice

Sia nella progettazione con un compensatore \mathbf{P} , sia con il luogo delle radici, é fondamentale correggere l'errore a regime e_r . Per adempiere questo compito bisogna aggiungere al sistema una rete ritardatrice, ossia l'utilizzo di una *Phase Lag* con funzione di trasferimento:

$$C(s) = \frac{s + z_0}{s + p_0} \quad (3.6)$$

Una rete ritardatrice é formata da una coppia zero-polo, dove il polo agisce prima dello zero generando uno sfasamento in ritardo fino ad un massimo di $\pi/2$. Questa componente consente di diminuire e_r e ridurre la pulsazione di attraversamento ω_a , per poter ricavare una risposta meno veloce ma piú precisa.

Si ipotizzi di collocare uno zero $z_0 = 0,3$ e un polo $p_0 = 0,03$, di conseguenza la risposta subisce una diminuzione dell'errore a regime pari a $z_0/p_0 = 10$.

Attraverso i comandi *Matlab* é possibile tracciare il luogo delle radici e la risposta al gradino:

```
z0 = 0.3;
p0 = 0.03;
R_r = (s+z0)/(s+p0); % rete ritardatrice
rlocus(R_r*W);
axis([-0.6, 0.11, -0.5, 0.5]);
sgrid(0.6, 0.36);
[Kp,poles] = rlocfind(R_r*W)
```

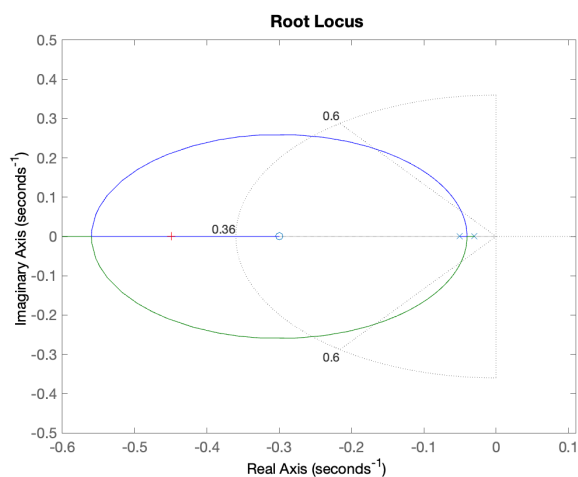


Figura 3.5: Luogo delle radici con *Phase Lag*

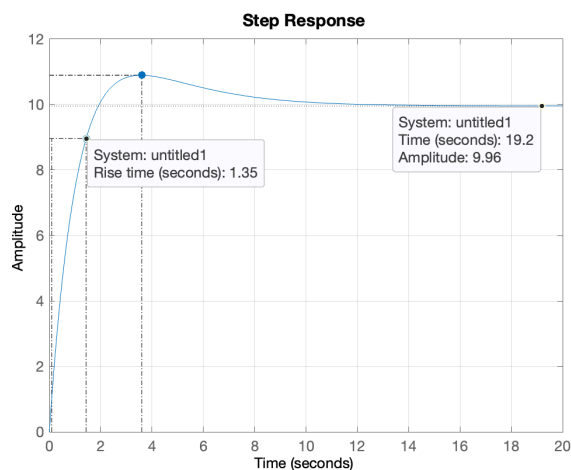


Figura 3.6: Risposta al gradino con *Phase Lag*

3.4 Considerazioni conclusive

Analizzando il modello con il metodo del luogo delle radici si presentano gli stessi inconvenienti del controllore \mathbf{P} , ovvero un errore a regime e_r costante, nel caso in cui il guadagno non sia sufficientemente alto, e un costo eccessivo o addirittura irrealizzabile a livello pratico, nel caso di un guadagno troppo elevato, nonostante diminuisca notevolmente il tempo di salita t_r .

Per ovviare a questo problema si introduce una coppia zero-polo (*rete ritardatrice*) che diminuisce contemporaneamente il tempo di salita e l'errore a regime, e al tempo stesso rende il sistema meccanicamente realizzabile.

Per ultimo si riscontra che l'inserimento di uno zero, appartenente alla rete ritardatrice, genera una leggera sovralongazione M_p in funzione della sua posizione, tuttavia risulta irrilevante per il caso in esame, in quanto il controllore riesce comunque a mantenere le specifiche progettuali imposte all'inizio della trattazione.

Capitolo 4

Analisi in frequenza

4.1 Considerazioni preliminari

Fino ad ora é stato focalizzato lo studio sulla risposta al gradino del sistema nel dominio del tempo, in modo da rispettare i requisiti progettuali: rise time t_r e overshoot M_p .

Nello studio della funzione di trasferimento é preferibile analizzare il comportamento del sistema nel dominio della frequenza, in quanto lo studio nel dominio del tempo richiederebbe il calcolo in quattro dimensioni.

Definendo la variabile di Laplace $s = j\omega$, dove ω viene detta *pulsazione* (o *frequenza angolare*), dalla funzione di trasferimento si ottiene la risposta in frequenza (o risposta armonica) del sistema $W(j\omega)$ definita come:

$$W(j\omega) : \mathbb{C} \rightarrow \mathbb{C} \quad \text{con} \quad 0 \leq \omega \leq \infty \quad (4.1)$$

dove si valuta la funzione di trasferimento in corrispondenza di valori immaginari puri.

Per valutare la risposta in frequenza del modello si definiscono alcuni parametri:

- B_p = banda passante (tipicamente a 3 dB);
- ω_r = pulsazione di risonanza;
- M_{rel} = picco di risonanza relativo;

dove la banda passante B_p individua l'intervallo di pulsazioni $[0, B_p]$ in cui si ha:

$$|W(j\omega)|_{dB} = |W(0)|_{dB} - 3dB \quad (4.2)$$

mentre la pulsazione di risonanza ω_r ($\omega_r > 0$, unica se esiste) é la frequenza per la quale si ha il valore massimo di $|W(j\omega)|$, ed é definita come:

$$|W(j\omega_r)| = \max_{(\omega > 0)} |W(j\omega)|_{dB} \quad (4.3)$$

e infine il picco di risonanza relativo M_{rel} rappresenta il rapporto tra il modulo della risposta in frequenza alla pulsazione di risonanza ω_r e $|W(0)|$ calcolato in dB:

$$M_{rel} = \left[\frac{|W(j\omega_r)|}{|W(0)|} \right]_{dB} \quad (4.4)$$

Lo studio della risposta in frequenza risulta molto efficace perché permette di dedurre la stabilità del sistema, sia per la catena aperta che per la catena chiusa. L'analisi si svolge mediante due rappresentazioni grafiche:

- il **diagramma di Bode** che rappresenta l'ampiezza e la fase di $|W(j\omega)|$;
- il **diagramma di Nyquist** che grafica $W(j\omega)$ nel piano complesso in funzione del parametro reale ω ;

Nell'analisi della risposta in frequenza è necessario definire alcuni nuovi parametri:

- **pulsazione di attraversamento** ω_a (unica se esiste), definita come la frequenza per la quale il diagramma di Bode interseca l'asse orizzontale (delle pulsazioni), ossia:

$$|W(j\omega_a)|_{dB} = 20 \log |W(j\omega_a)| = 0 \quad (4.5)$$

- **fase di attraversamento** φ_a , definita come la fase assunta dalla risposta in frequenza in corrispondenza della pulsazione di attraversamento:

$$\varphi_a = \arg(W(j\omega_a)) \quad (4.6)$$

- **margin di fase** m_φ , una grandezza definita dalla relazione:

$$m_\varphi = \pi + \varphi_a \quad (4.7)$$

4.2 Considerazioni catena aperta e catena chiusa

Analogamente ai capitoli precedenti, si prende in considerazione un compensatore proporzionale e si svolge l'analisi del sistema ottenendo:

$$W(s) = C(s)G(s) = \frac{K_p}{ms + b} \quad (4.8)$$

dove K_p è unitario, il diagramma di Bode è rappresentato in Figura (1.6) e la risposta al gradino in Figura (1.4).

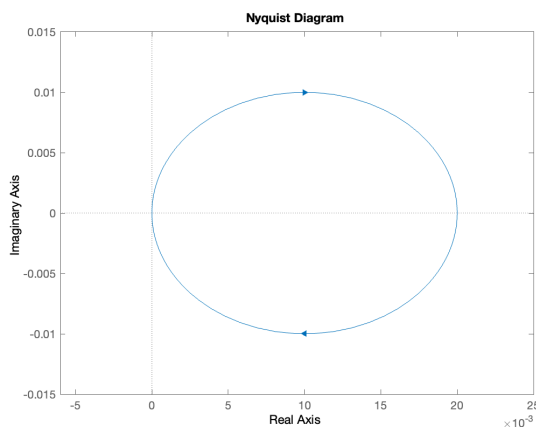


Figura 4.1: Diagramma di Nyquist

Si é tracciato il diagramma di Nyquist usando il comando *nyquist* in *Matlab*.

Nel Capitolo (2) se é già discussa la stabilità del sistema in catena aperta, in questa sezione, grazie al diagramma di Nyquist, ci si concentra sulla stabilità del sistema in catena chiusa, mediante il criterio ridotto di Nyquist:

$$N = n_{W+} - n_{H+} \quad (4.9)$$

dove N rappresenta il numero di giri che il grafico compie attorno al punto critico $-1 + j0$, mentre n_{W+} e n_{H+} rappresentano rispettivamente il numero di poli con parte reale positiva della funzione di trasferimento del sistema in catena aperta e il numero di poli a parte reale positiva della funzione di trasferimento del sistema in catena chiusa. Da Figura (4.1) si nota che $N = 0$, in quanto il diagramma non circonda mai il punto critico, e dunque il sistema retroazionato risulta BIBO-stabile.

4.3 Rete ritardatrice

Una rete attenuatrice, chiamata anche ritardatrice o *lag controller*, come già discusso nella sezione (3.3), permette di ridurre l'errore a regime e nel contempo di diminuire la pulsazione di attraversamento, ottenendo una risposta meno rapida ma più precisa.

L'equazione rimane quella presentata nella precedente introduzione (3.6), si pone $z_0 = 0,1$ e $p_0 = 0,02$, ottenendo una riduzione dell'errore a regime di un fattore pari a $z_0/p_0 = 5$.

Si ricavano i seguenti grafici della risposta al gradino e del diagramma di Bode:

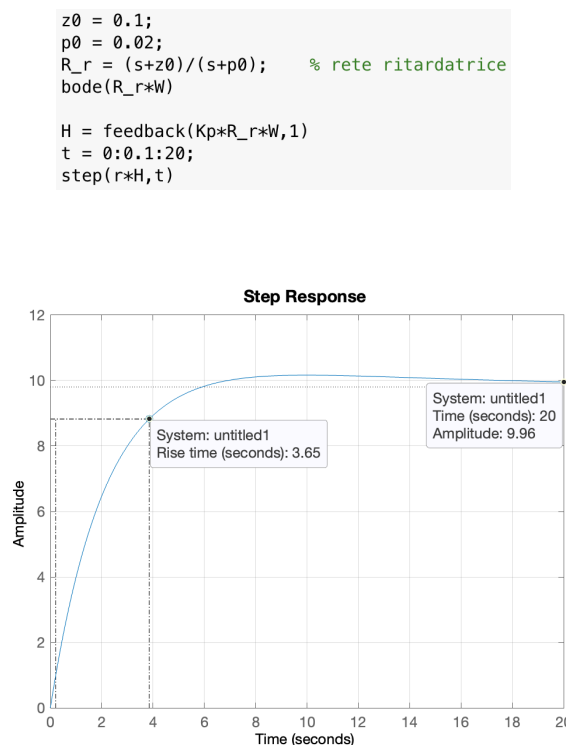


Figura 4.2: Risposta al gradino con rete ritardatrice

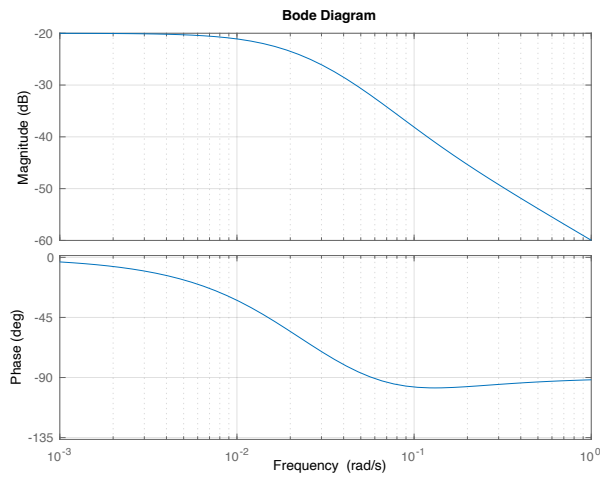


Figura 4.3: Diagramma di Bode con rete ritardatrice

dove si evidenzia che le richieste progettuali sono state rispettate, ottenendo un errore a regime $e_r = 1,6\%$ e un tempo di salita $t_r = 3,65$ s.

4.4 Considerazioni conclusive

Come discusso nei precedenti due capitoli, il controllore \mathbf{P} soddisfa i requisiti progettuali solo nell'analisi teorica, ma passando sul piano reale non é fisicamente realizzabile. Per correggere questo inconveniente si aggiunge una rete ritardatrice al sistema.

Questa soluzione permette di ottenere ottimi risultati, sia per l'errore a regime e_r , che diminuisce e permette il perfetto tracciamento del gradino, sia per il tempo di salita t_r , che viene incrementato in modo da migliorare il comportamento dinamico del sistema.

Capitolo 5

Analisi in Simulink

5.1 Considerazioni preliminari

Per spiegare meglio il funzionamento del controllo di velocità tramite Cruise Control, nel seguente capitolo viene esposta un'analisi del modello mediante l'ambiente *Simulink* presente in *Matlab*.

Il programma *Simulink* viene utilizzato per progettare, analizzare e simulare i sistemi dinamici descritti da equazioni differenziali, utilizzando le funzioni di trasferimento o gli spazi di stato. Le specifiche di progettazione e i parametri sistema utilizzati nell'analisi fanno riferimento a quelli definiti all'inizio della trattazione nel Capitolo (1).

Per maggiore chiarezza si riporta di seguito la Tabella (1.1) dei parametri iniziali:

m	massa del veicolo	1000 [kg]
b	coefficiente di attrito	50 [Ns/m]
u	forza elettromotrice	500 [N]
r	velocità di riferimento	10 [m/s]

5.2 Creazione modello Simulink

Per creare uno schema a blocchi del sistema si comincia da un blocco integratore, che prende in ingresso l'accelerazione a e genera in uscita la velocità v :

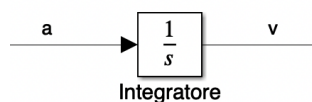


Figura 5.1: Blocco integratore

Tramite l'equazione che regola il moto del veicolo (1.1) è possibile ottenere mediante integrazione la velocità v , ossia la variabile di interesse per il modello:

$$v = \int \frac{u(t) - bv}{m} = \int \frac{dv}{dt} dt \quad (5.1)$$

Per ricavare l'accelerazione \mathbf{a} si applica la seconda legge di Newton, ossia calcolando la differenza della forza elettromotrice \mathbf{u} e della forza d'attrito $F_a = bv$ e rapportandola con la massa \mathbf{m} del veicolo:

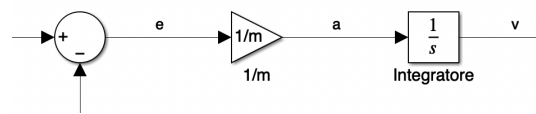


Figura 5.2: Schema a blocchi del processo $P(s)$

La forza di attrito bv si calcola mediante una retroazione negativa della velocità in uscita, che viene moltiplicata tramite il guadagno b :

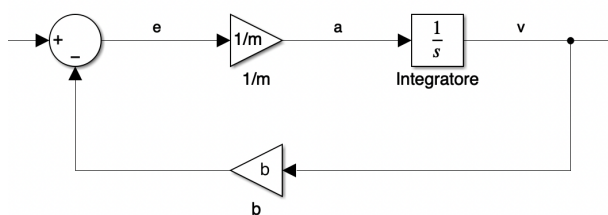


Figura 5.3: Schema a blocchi del processo $P(s)$

La forza elettromotrice \mathbf{u} viene inserita come sorgente di tipo gradino di ampiezza u :

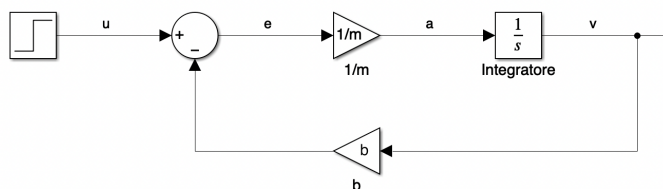


Figura 5.4: Schema a blocchi del processo $P(s)$

Infine per visualizzare la velocità si collega l'uscita a un blocco visualizzatore (*oscilloscopio*):

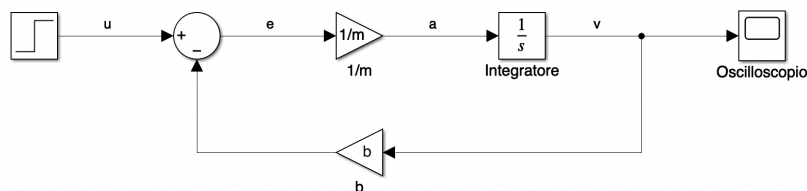


Figura 5.5: Schema a blocchi del processo $P(s)$

Dopo aver modellato lo schema a blocchi e aver inserito tutti i parametri necessari, si può procedere con l'analisi in catena aperta.

5.3 Simulazione catena aperta

In questo paragrafo si esegue la simulazione dello schema a blocchi precedentemente implementato.

In primo luogo si definisce il tempo di esecuzione della simulazione (nella Figura (5.6) é stato impostato un tempo di esecuzione pari a 120 s) e successivamente si stabilisce il tipo di passo che si intende usare (*Variable-step/Fixed-step*) e il tipo di *Solver*.

Dopo aver ultimato le configurazioni si inizia la simulazione del modello premendo il tasto *Start Simulation* presente nell'interfaccia di *Simulink*, o direttamente dal visualizzatore, ottenendo la seguente risposta al gradino:

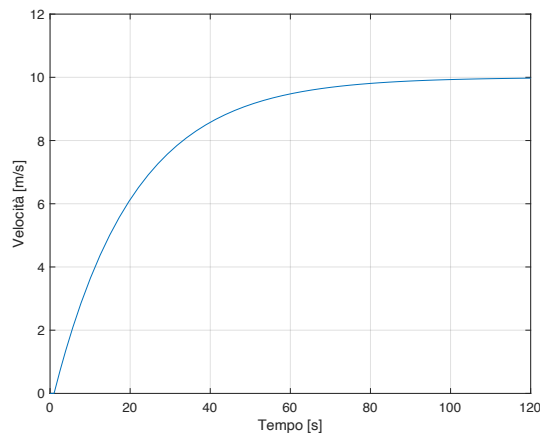


Figura 5.6: Simulazione risposta al gradino in catena aperta

Dalla Figura (5.6) é possibile osservare che il comportamento del sistema é come quello calcolato precedentemente. Si puó notare che il grafico risulta coerente alla risposta ad anello aperto in Figura (1.4). Le specifiche, infatti, non sono rispettate, per cui risulta necessario migliorare il tempo di salita t_r utilizzando i controllori progettati in *Matlab* nei precedenti capitoli: **PI**, **PID** e rete ritardatrice.

5.4 Simulazione catena chiusa

Si procede inserendo i controllori progettati precedentemente (**PI**, **PID**, rete ritardatrice) prima del sistema in catena aperta, mantenendo una retroazione negativa.

Per una maggiore semplicitá grafica, il sistema in catena aperta di Figura (5.5) si inserisce tramite un blocco sottosistema (*subsystem*), e si considera come una "black box".

5.4.1 Controllore PI

Per l'analisi in catena chiusa si utilizza un controllore proporzionale-integrativo (**PI**), e si inseriscono i guadagni ricavati nel Capitolo (2) ($K_p = 1000$, $K_i = 50$). Come riferimento per il sistema, si utilizza un segnale di velocitá con forma a gradino di ampiezza v e si visualizza l'uscita con un oscilloscopio.

Di seguito viene rappresentato lo schema sopra descritto, implimentato in *Simulink*:

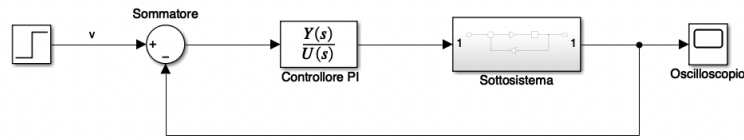


Figura 5.7: Schema a blocchi del sistema retroazionato

Viene riportato il codice *Matlab* usato per definire i parametri da inserire nella simulazione:

```

m = 1000; % massa del veicolo
b = 50; % coefficiente di attrico
u = 500; % forza elettromotrice
r = 10; % velocità di riferimento
Kp = 1000; % guadagno proporzionale
Ki = 50; % guadagno integrativo

```

Si comincia la simulazione con il comando *Start Simulation* e si ottiene la seguente risposta al gradino:

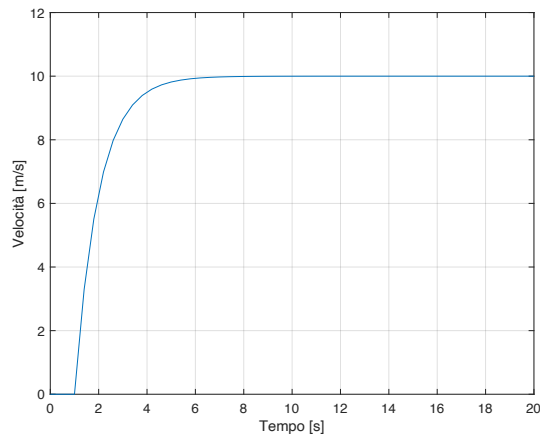


Figura 5.8: Simulazione risposta al gradino in catena chiusa

É possibile evidenziare da Figura (5.8) che le limitazioni progettuali vengono pienamente soddisfatti, in quanto non si presentano né sovraelongazioni né errori a regime permanente, inoltre anche il tempo di salita t_r risulta minore di 4 s. Si nota infine che il t_r non é eccessivamente basso, quindi risulta realizzabile a livello fisico.

Come già introdotto nel Capitolo (2), anche mediante un controllore **PID** si riescono a soddisfare le specifiche imposte, per il fatto che il controllore derivativo reale rende il sistema più stabile, diminuendo le sovraelongazioni. Tuttavia il **PID** é più complesso a livello realizzativo, poiché é necessario implementare anche il guadagno derivativo K_d , che risulta molto più costoso. In conclusione, per il problema del Cruise Control in esame, si preferisce l'approccio con il compensatore **PI**.

5.4.2 Rete ritardatrice

In ultima analisi si prende in considerazione una rete ritardatrice con funzione di trasferimento già definita dall'espressione (3.6). Si modifica lo schema in *Simulink*, andando a sostituire la funzione di trasferimento del **PI** con quella della rete e inserendo un guadagno proporzionale K_p :

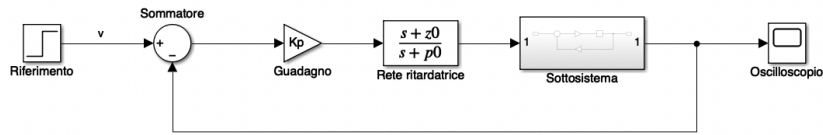


Figura 5.9: Schema a blocchi con *Phase Lag*

In questo caso si inseriscono i valori determinati nel Capitolo (4) ($z_0 = 0, 1$, $p_0 = 0, 02$) e si utilizza come costante proporzionale $K_p = 500$.

```

m = 1000; % massa del veicolo
b = 50; % coefficiente di attrico
u = 500; % forza elettromotrice
r = 10; % velocità di riferimento
Kp = 500; % guadagno proporzionale
z0 = 0.1; % zero
p0 = 0.02; % polo

```

Tramite la simulazione del sistema si ottiene la seguente risposta al gradino:

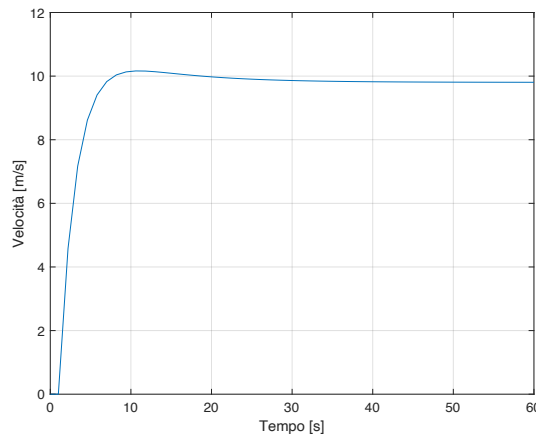


Figura 5.10: Simulazione risposta al gradino in catena chiusa con *Phase Lag*

Analizzando la Figura (5.10) si nota una leggera sovravelongazione M_p , che risulta accettabile poiché rispetta i limiti imposti ($M_p < 10\%$). Inoltre si ottiene un tempo di salita t_r minore di 5 s, mentre l'errore a regime e_r può essere trascurato ($e_r \sim 0$).

Si evidenzia che l'accelerazione iniziale del sistema non risulta troppo elevata, di conseguenza non viene interdetta la realizzabilità fisica del modello. La rete ritardatrice rappresenta un'ottima soluzione alternativa a quelle viste in precedenza.

5.5 Considerazioni conclusive

L'ambiente *Simulink* è uno strumento perfetto per analizzare, progettare e soprattutto simulare la risposta del sistema, sia in catena aperta che in catena chiusa. Questo permette al programmatore di vedere in maniera concreta le conseguenze delle modifiche apportate al sistema.

Non sono state eseguite le simulazioni con i controllori **P** e **PID** per evitare un'eccessiva ridondanza dei risultati, poiché il controllore **PI** soddisfa pienamente i nostri requisiti. Inoltre con gli altri tipi di compensatore avremmo ottenuto risultati pressoché analoghi a quello ottenuto tramite il **PI**

Esattamente come ci si aspettava dalle analisi eseguite nei rispettivi capitoli, sia il controllore **PI** che la rete ritardatrice si sono dimostrate valide risposte per l'implementazione di una soluzione per il problema in esame.

Capitolo 6

Conclusioni

Con questa tesi si è tentato di costruire, analizzare e simulare un modello di sistema Cruise Control, prima in catena aperta e successivamente in catena chiusa. Tramite un modello matematico è stato possibile condurre un'analisi dinamica del sistema, in modo da progettare vari tipi di controllori, mettendo in pratica diverse tipologie di metodi realizzativi. È stato riscontrato che una soluzione teorica, nonostante rispetti i limiti progettuali, non sempre può essere applicata a livello pratico.

Lo studio del modello è stato condotto usando come approcci:

- i compensatori **P**, **PI** e **PID**;
- il metodo dell'analisi del luogo delle radici, con l'introduzione di una rete ritardatrice;
- l'analisi in frequenza, con l'aggiunta di una rete ritardatrice;

Mediante il software *Matlab* e l'ambiente *Simulink* è stato possibile facilitare notevolmente la modellizzazione e l'analisi del sistema, come anche la progettazione dei controllori, potendo visualizzare le rispettive risposte al gradino di riferimento.

Per semplificare la progettazione, requisito non trascurabile per la realizzazione di un controllore, lo studio si è focalizzato principalmente sui controllori di tipo **P**, **PI**, **PID**.

Il primo modello permette di rispettare i vincoli realizzativi solamente a livello teorico, poiché tramite un guadagno eccessivamente alto (che comporta già un costo elevato) si ricava un tempo di salita t_r molto basso che causerebbe sollecitazioni troppo elevate, e quindi distruttive, per l'autoveicolo.

Le altre due tipologie di controllori, **PI** e **PID**, permettono di ottenere una risposta meno aggressiva da parte del sistema, diminuendo il tempo di salita. Anche in questi casi le specifiche imposte sono pienamente rispettate senza causare problemi, che invece si presentano nel caso del singolo controllore proporzionale.

L'unico requisito che risulta fondamentale tenere sotto controllo è la sovraelongazione M_p , introdotta dal guadagno integrativo.

L'analisi mediante luogo delle radici e l'analisi in frequenza sono delle alternative che consentono di ottenere buoni risultati. Tuttavia, per soddisfare i limiti realizzativi, richiedono entrambe l'utilizzo di una rete ritardatrice. Quest'ultima gode di un'ottima semplicità di progetto, come i controllori già elencati, in quanto comporta semplicemente l'aggiunta di un

polo e di uno zero alla funzione di trasferimento in esame. Purtroppo questo approccio richiede qualche osservazione in più, a differenza dei compensatori precedentemente discussi.

In conclusione, per la progettazione del Cruise Control, la soluzione più adeguata è quella di un controllore di tipo **PI**, che rispetta le specifiche e che al tempo stesso non presenta un costo di realizzazione eccessivo.

Bibliografia

- [1] M. BISIACCO, M.E. VALCHER. *Controlli automatici*. Libreria Progetto Padova, Padova, 2018.
- [2] M. BISIACCO, G. PILLONETTO. *Sistemi e modelli*. Società Editrice Esculapio, 2020.
- [3] M. BISIACCO, S. BRAGHETTO. *Teoria dei sistemi dinamici*. Esculapio Collana Progetto Leonardo, Bologna, 2015.
- [4] P. BOLZERN, R. SCATTOLINI, N. SCHIAVONI. *Fondamenti di controlli automatici*. McGraw-Hill Education, 2015.
- [5] CRUISE CONTROL,
<https://ctms.engin.umich.edu/CTMS/index.php?example=CruiseControlsection=SystemModeling>