

Tesina di Complementi di Controlli Automatici

2-DOF HELICOPTER CONTROLLING

Ciro Arena P38/53

Vito Giura P38/56

6 febbraio 2022

Indice

| | | |
|----------|--|-----------|
| 1 | Introduzione | 3 |
| 2 | Modello matematico | 5 |
| 2.1 | Descrizione del modello non lineare | 5 |
| 2.2 | Linearizzazione | 8 |
| 2.3 | Raggiungibilità e Osservabilità | 11 |
| 3 | Controllo con assegnamento degli autovalori | 12 |
| 3.1 | Matrice di guadagno in feed-forward | 13 |
| 3.2 | Aggiunta dell'azione integrale | 15 |
| 3.3 | Aggiunta dell'osservatore | 18 |
| 4 | Controllo ottimo LQ | 22 |
| 4.1 | LQ con matrice di guadagno in feed-forward | 22 |
| 4.2 | LQ con azione integrale | 24 |
| 4.3 | Controllo LQG | 25 |
| 5 | Controllo H_∞ | 29 |

Capitolo 1

Introduzione

Il sistema MIMO oggetto di studio di questa tesina è un elicottero a 2 gradi di libertà (*2-DOF Helicopter*) a cui sono state applicate le diverse tecniche di controllo studiate nel corso di Complementi di Controlli Automatici.

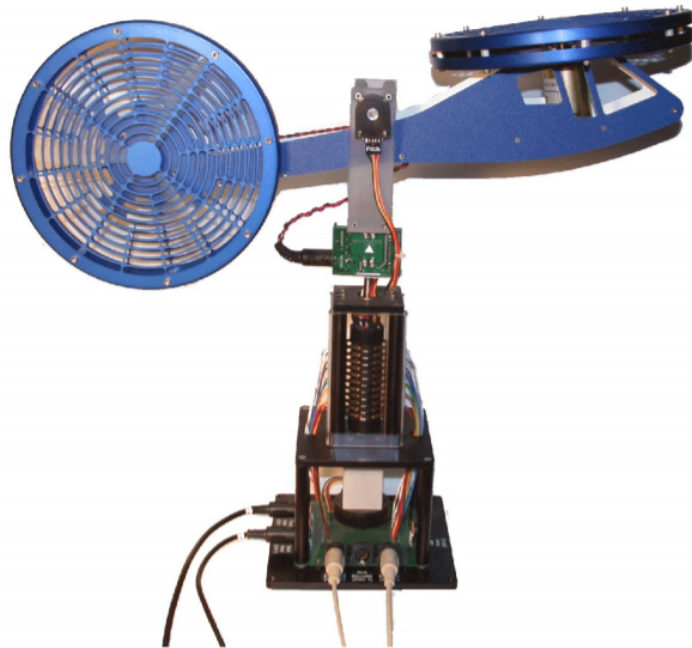


Figura 1.1: Esempio di 2-DOF Helicopter

Come mostrato in figura 1.1, il nostro sistema consiste nel modello di un elicottero montato su una base fissa, con due eliche che sono alimentate da motori DC (*Direct Current*, a corrente continua): l'elica anteriore controlla l'elevazione del naso dell'elicottero attorno all'asse di *pitch* (uscende dal foglio) mentre l'elica posteriore controlla i movimenti laterali dell'elicottero attorno all'asse di *yaw* (parallelo al foglio). L'obiettivo del controllo di questo modello consisterà nella stabilizzazione della componente orizzontale in un arbitrario orientamento desiderato.

Il modello semplificato dell'elicottero con due gradi di libertà rappresenta un caso di studio estremamente interessante, perché costituisce un sistema la cui modellazione non è troppo complicata ma che, al contempo, consente comunque lo studio di tecniche di controllo avanzate per i sistemi multivariabili tali da poter essere applicate in situazioni reali relativamente al controllo del volo.

Tale elaborato come al solito comincerà col ricavare le equazioni del moto facendo ricorso metodo di Eulero-Lagrange dando come risultato un sistema di equazioni non lineari. Dato che si dovranno applicare tecniche di controllo lineare, sarà necessario linearizzare il set di equazioni ottenuto al passo precedente attorno ad un punto di equilibrio ricavando così le matrici del sistema linearizzato. A questo punto si potrà procedere con l'illustrazione delle diverse tecniche di controllo adottate partendo con lo State Feedback Design a cui andremo ad aggiungere gradualmente osservatori ed azione integrale e concludendo infine con il controllo H_∞ , passando ovviamente attraverso il controllo LQ ed LQG. Per ognuna di queste tecniche per le quali si è progettata un'apposita legge di controllo si andrà a considerare il modello non lineare per vedere se nell'intorno del punto di equilibrio scelto, il sistema non lineare e il suo corrispondente linearizzato forniscono una risposta simile nel momento in cui vengono sottoposti all'azione della legge di controllo progettata.

Capitolo 2

Modello matematico

2.1 Descrizione del modello non lineare

Come già detto in precedenza il primo step consiste nel ricavare le equazioni del moto applicando l'equazione del moto della meccanica Lagrangiana, cioè

$$\frac{d}{dt} \frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} - \frac{\partial \mathcal{L}}{\partial \mathbf{q}} = \mathbf{Q} \quad (2.1)$$

dove

- \mathbf{q} è il vettore di coordinate lagrangiane;
- $\mathcal{L} = \mathcal{L}(\dot{\mathbf{q}}, \mathbf{q})$ è la cosiddetta funzione *Lagrangiana* ed è data dalla differenza tra l'energia cinetica e l'energia potenziale del sistema, cioè

$$\mathcal{L}(\dot{\mathbf{q}}, \mathbf{q}) = T(\dot{\mathbf{q}}, \mathbf{q}) - U(\mathbf{q})$$

con T energia cinetica e U energia potenziale;

- $\mathbf{Q} = \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix}$ è il vettore delle forze generalizzate date dai contributi forniti dalle due eliche.

e ci sarà un'equazione del moto per ognuno dei motori.

Prima però di specificare meglio quali sono le espressioni che caratterizzano questi termini, bisogna prima andare ad evidenziare nel dettaglio quali sono i parametri salienti di questo sistema. Come già detto nell'introduzione, il nostro modello lo possiamo schematizzare come una coppia di eliche localizzate alle estremità una sbarra vincolata al proprio centro su una base fissa in modo tale che l'intera struttura possa ruotare quasi liberamente intorno ad un asse verticale ed uno orizzontale, a tal proposito si veda la figura 2.1.

Verranno fatte le seguenti convenzioni per la modellazione:

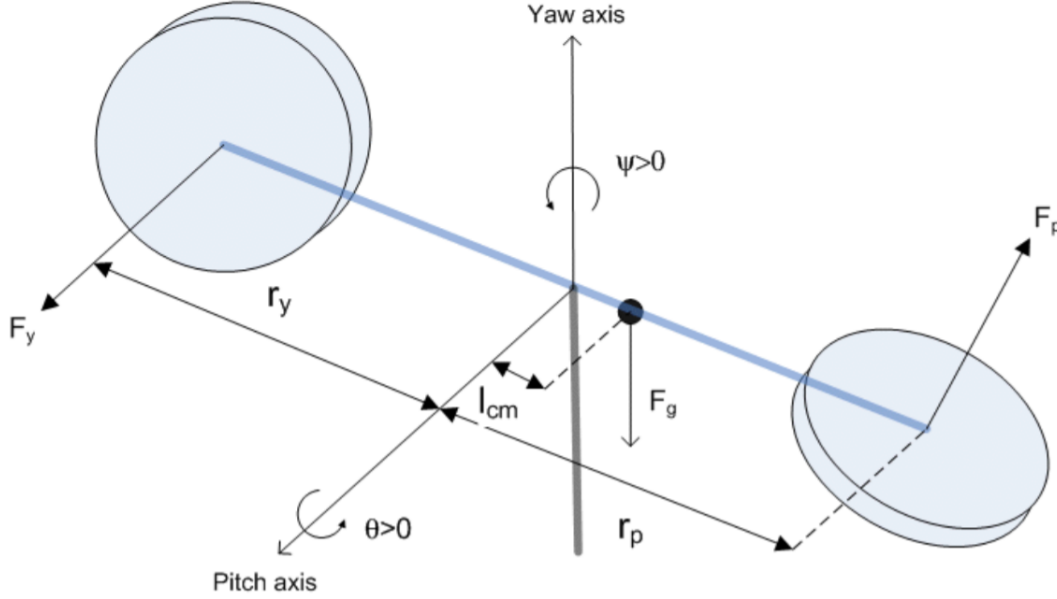


Figura 2.1: Schematizzazione del 2-DOF Helicopter

- La rotazione intorno all'asse orizzontale, chiamato asse di *pitch*, avverrà con un angolo θ che sarà considerato positivo quando la parte anteriore dell'elicottero si muove verso l'alto, stesso identico discorso per la sua velocità angolare $\dot{\theta}$;
- La rotazione intorno all'asse verticale, chiamato asse di *yaw*, avverrà con un angolo ψ che sarà considerato positivo se la rotazione attorno a tale asse avviene in senso orario, stesso identico discorso per la sua velocità angolare $\dot{\psi}$;
- Considereremo $\theta = 0$ quando l'elicottero è in posizione orizzontale;
- Non ci sono né movimenti assiali e né movimenti di rollio.

Di conseguenza già possiamo renderci conto che il vettore delle coordinate lagrangiane sarà

$$\mathbf{q} = \begin{bmatrix} q_1 & q_2 & q_3 & q_4 \end{bmatrix}^T = \begin{bmatrix} \theta & \psi & \dot{\theta} & \dot{\psi} \end{bmatrix}^T \quad (2.2)$$

Dopodiché indicheremo con m_{heli} la massa complessiva della struttura (sbarra, motori etc.) e con l_{cm} la distanza misurata lungo il corpo dell'elicottero tra il baricentro e l'asse di pitch. Inoltre indicheremo con $J_{eq,p}$ il momento di inerzia equivalente attorno all'asse di pitch e con $J_{eq,y}$ il momento di inerzia equivalente attorno all'asse di yaw. Detto ciò a questo punto possiamo andare a scriverci le espressioni della funzione Lagrangiana, in cui l'energia potenziale è dovuta alla forza di gravità e sarà

$$U = m_{heli} g l_{cm} \sin(\theta) \quad (2.3)$$

mentre l'energia cinetica totale la scriveremo così

$$T = T_{r,p} + T_{r,y} + T_t \quad (2.4)$$

ossia come la somma delle energie cinetiche rotazionali dovute alle rotazioni intorno all'asse di pitch e di yaw, $T_{r,p}$ e $T_{r,y}$, e dell'energia cinetica traslativa dovuta al movimento del centro di massa.

Per quanto riguarda le eliche, abbiamo già detto che queste sono alimentate da motori DC e quindi la forza aerodinamica dell'intera struttura è controllata variando la velocità dei rotori. Pertanto le variabili di controllo del nostro sistema sono le tensioni di alimentazione dei due motori DC, dove le tensioni di alimentazione del motore che provoca il movimento di pitch e di yaw le indicheremo rispettivamente con $V_{m,p}$ e $V_{m,y}$. La particolarità di questo elicottero però consiste nel fatto che è un sistema MIMO con risposta accoppiata, cioè il motore di pitch controlla direttamente l'angolo di pitch usando la forza di spinta generata dalla propria elica creando al contempo una coppia sull'asse di yaw come effetto della resistenza dell'aria, e allo stesso identico modo cioè il motore di yaw controlla direttamente l'angolo di yaw usando la forza di spinta generata dalla propria elica creando al contempo una coppia sull'asse di pitch sempre per effetto della resistenza dell'aria. In definitiva potremo pervenire alle seguenti espressioni per le componenti delle forze generalizzate

$$\begin{cases} Q_1 = \tau_p(V_{m,p}, V_{m,y}) - B_p \dot{\theta} \\ Q_2 = \tau_y(V_{m,p}, V_{m,y}) - B_y \dot{\psi} \end{cases} \quad (2.5)$$

dove, come possiamo vedere, tali equazioni includono anche i coefficienti di smorzamento rotatorio viscoso che agiscono sull'asse di pitch e di yaw e che indicheremo rispettivamente con B_p e B_y . Inoltre come già menzionato le coppie complessive che agiscono sull'asse di pitch e di yaw, τ_p e τ_y , ognuna di queste dipende da ambo le tensioni di alimentazione, nello specifico valgono le seguenti relazioni

$$\begin{cases} \tau_p(V_{m,p}, V_{m,y}) = K_{pp}V_{m,p} + K_{py}V_{m,y} \\ \tau_y(V_{m,p}, V_{m,y}) = K_{yp}V_{m,p} + K_{yy}V_{m,y} \end{cases} \quad (2.6)$$

dove K_{pp} , K_{py} , K_{yp} e K_{yy} sono delle opportune costanti di coppia dei motori.

A tal proposito è estremamente importante, ai fini del controllo, tenere conto anche del range di tensione che possono fornire questi due motori DC: in particolare i motori di pitch e di yaw che andremo ad utilizzare, possono fornire una tensione compresa nel range di $\pm 24V$ e di $\pm 15V$ rispettivamente.

Mettendo insieme tutti i risultati appena ottenuti e sostituendoli nella (2.1) giungeremo finalmente alla coppia di equazioni non lineari che descrivono il moto

del nostro sistema

$$(J_{eq,p} + m_{heli} l_{cm}^2) \ddot{\theta} = K_{pp} V_{m,p} + K_{py} V_{m,y} - B_p \dot{\theta} - m_{heli} l_{cm}^2 \sin \theta \cos \theta \dot{\psi}^2 - m_{heli} g l_{cm} \cos \theta \quad (2.7)$$

$$(J_{eq,y} + m_{heli} l_{cm}^2 \cos^2 \theta) \ddot{\psi} = K_{yp} V_{m,p} + K_{yy} V_{m,y} - B_y \dot{\psi} + 2m_{heli} l_{cm}^2 \sin \theta \cos \theta \dot{\psi} \dot{\theta} \quad (2.8)$$

Ovviamente i valori di tutti i parametri del sistema sono noti e sono riassunti nella seguente tabella

Tabella 2.1: Parametri del modello

| Simbolo | Valore | Unità di misura |
|------------|--------|-----------------|
| m_{heli} | 1.3872 | Kg |
| l_{cm} | 0.186 | m |
| $J_{eq,p}$ | 0.0384 | $Kg \cdot m^2$ |
| $J_{eq,y}$ | 0.0432 | $Kg \cdot m^2$ |
| B_p | 0.800 | N/V |
| B_y | 0.318 | N/V |
| K_{pp} | 0.204 | Nm/V |
| K_{yy} | 0.072 | Nm/V |
| K_{py} | 0.0068 | Nm/V |
| K_{yp} | 0.0219 | Nm/V |

2.2 Linearizzazione

Ricapitolando abbiamo visto che le funzioni Lagrangiane dipendono sostanzialmente da 4 coordinate lagrangiane: gli angoli θ e ψ , e le loro derivate temporali $\dot{\theta}$ e $\dot{\psi}$. Abbiamo quindi ben 4 variabili di stato per il nostro sistema. Inoltre abbiamo pure detto che l'obiettivo del controllo è quello di portare l'elicottero in un orientamento desiderato dello spazio, quindi se ne deduce facilmente che le variabili di uscita del modello saranno le prime due variabili di stato, ossia θ e ψ . Infine un'altra cosa già menzionata, è che il controllo verrà realizzato grazie ai due attuatori del sistema che sono i motori DC delle due eliche, ciascuno alimentato da una tensione di alimentazione, e quindi di fatto si hanno due ingressi di controllo. Possiamo dunque concludere che il nostro è un sistema di ordine 4 e *quadrato*, poiché abbiamo 2 ingressi e 2 uscite.

Fatta questa premessa possiamo subito ad effettuare la linearizzazione delle equazioni (2.7) e (2.8). La prima cosa da decidere, è intorno a quale punto di equilibrio vanno linearizzate le nostre equazioni, per questo motivo andiamo a vedere quali

sono i punti di equilibrio del sistema risolvendo la classica equazione

$$0 = f(\bar{x}, \bar{u})$$

Ovvero andiamo a sostituire agli angoli θ e ψ dei generici valori costanti, in modo tale da trasformare le equazioni (2.7) e (2.8) nel seguente set di equazioni

$$\begin{cases} K_{pp}V_{m,p} + K_{py}V_{m,y} = m_{heli}gl_{cm} \cos \theta \\ K_{yp}V_{m,p} + K_{yy}V_{m,y} = 0 \\ \dot{\theta} = 0 \\ \dot{\psi} = 0 \end{cases} \quad (2.9)$$

Ci rendiamo perciò conto che il nostro sistema ha un'infinità numerabile di punti di equilibrio, infatti assegnando valori costanti a θ e ψ , possiamo risolvere il sistema di equazioni precedentemente scritto nelle incognite $V_{m,p}$ e $V_{m,y}$ che corrisponderanno ai nostri ingressi di controllo. A questo punto decidiamo arbitrariamente che il nostro punto di lavoro sia lo zero, pertanto gli ingressi di controllo del nostro modello li potremo calcolare risolvendo il sistema di equazioni (2.9) e che faremo con l'ausilio di MATLAB

```
x1=0;
x0=[x1;0;0;0]; %stato di equilibrio
a=[Kpp Kpy; Kyp Kyy];
b=[Mheli*g*Lcm*cos(x1);0];
u0=a\b; %ingresso di equilibrio
```

I valori calcolati saranno $\bar{V}_{m,p} = 12.5348$ e $\bar{V}_{m,y} = -3.8127$.

Ora dato che abbiamo deciso di linearizzare intorno allo zero, cioè andiamo ad effettuare una linearizzazione intorno a piccoli angoli, il nostro modello non lineare sarà caratterizzato dalla seguente coppia di equazioni lineari

$$\begin{cases} (J_{eq,p} + m_{heli}l_{cm}^2)\ddot{\theta} = K_{pp}V_{m,p} + K_{py}V_{m,y} - B_p\dot{\theta} - m_{heli}gl_{cm} \\ (J_{eq,y} + m_{heli}l_{cm}^2)\ddot{\psi} = K_{yp}V_{m,p} + K_{yy}V_{m,y} - B_y\dot{\psi} \end{cases} \quad (2.10)$$

Infine per poterci ricavare le matrici del sistema linearizzato, una cosa che possiamo fare è crearci un modello Simulink relativo al sistema (2.10) (mostrato in figura 2.2) e applicare il comando "linmod" di MATLAB.

In conclusione queste saranno le espressioni che assumeranno le matrici del sistema

linearizzato intorno lo stato $x_0 = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}^T$:

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{B_p}{J_{tp}} & 0 \\ 0 & 0 & 0 & -\frac{B_y}{J_{ty}} \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \frac{K_{pp}}{J_{tp}} & \frac{K_{py}}{J_{tp}} \\ \frac{K_{yp}}{J_{ty}} & \frac{K_{yy}}{J_{ty}} \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \quad D = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

Avendo posto

$$J_{tp} = J_{eq,p} + m_{heli} l_{cm}^2 \quad J_{ty} = J_{eq,y} + m_{heli} l_{cm}^2$$

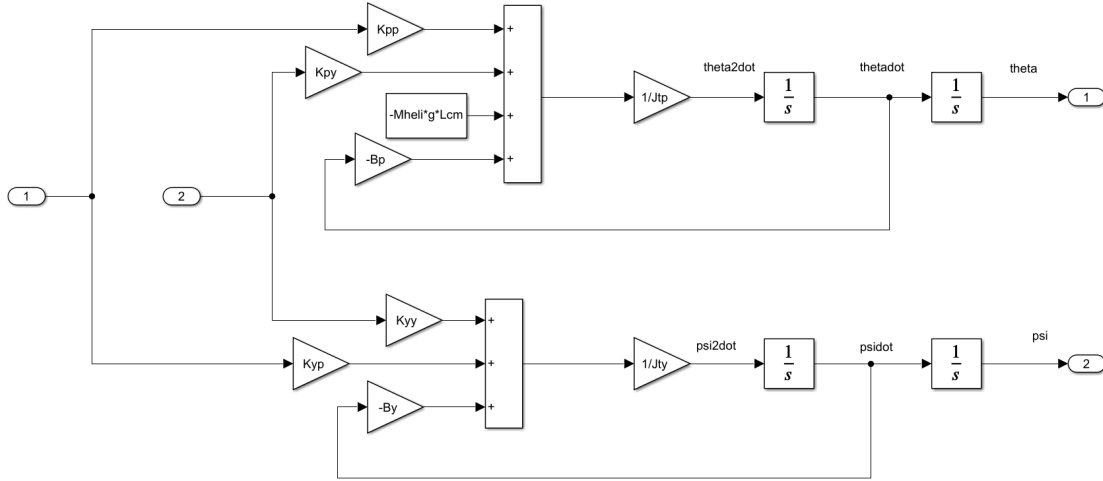


Figura 2.2: Modello simulink del sistema linearizzato

A conclusione di questa parte poi si noti una cosa che può trarre facilmente in inganno. Sappiamo bene dalla teoria di modellistica e simulazione che se $u = \bar{u} + \delta u$ è l'ingresso del nostro sistema non lineare, con \bar{u} ingresso di equilibrio, il sistema linearizzato avrà come ingresso δu , per cui se consideriamo per esempio il motore di pitch, questa δu non potrà banalmente assumere valori compresi tra $\pm 24V$, ma dato che $\bar{u} \approx 12,5$ allora l'ingresso del sistema linearizzato dovrà assumere valori compresi tra $24 - 12,5 \approx 11,5V$ e $-24 - 12,5 \approx -36,5V$. Ragionando allo stesso identico modo per il motore di yaw, avremo che i valori di ingresso al sistema lineare dovranno oscillare tra circa $-20,2V$ e $27,8V$.

2.3 Raggiungibilità e Osservabilità

Le prime proprietà strutturali che dobbiamo prendere in considerazione sono la raggiungibilità e l'osservabilità. Il fatto che il nostro sistema sia completamente raggiungibile e/o completamente osservabile, non costituisce un prerequisito fondamentale, ma certamente può fare molto comodo per la progettazione di sistemi di controllo e osservatori annessi. Per verificare ciò facendo stampare a video se il sistema è completamente raggiungibile/osservabile o meno, abbiamo scritto il seguente codice:

```
%% Controllabilità e Osservabilità

% verifica della completa raggiungibilità
n=length(A); %dimensione del sistema
Mr=ctrb(A,B); %matrice di controllabilità
nr=rank(Mr); %dimensione del sottospazio di raggiungibilità
nnr=n-nr; % dimensione del sottospazio di NON raggiungibilità

if nnr==0
    disp('Il sistema è completamente raggiungibile')
else
    disp('Il sistema NON è completamente raggiungibile')
end

Mo=obsv(A,C); %matrice di osservabilità
no=rank(Mo); %dimensione del sottospazio di osservabilità
nno=n-no; %dimensione del sottospazio di NON osservabilità
if nno==0
    disp('Il sistema è completamente osservabile')
else
    disp('Il sistema NON è completamente osservabile')
end
```

Il risultato è che il nostro sistema è completamente raggiungibile e completamente osservabile

Capitolo 3

Controllo con assegnamento degli autovalori

Adesso passiamo finalmente al controllo del nostro modello linearizzato con la prima tecnica studiata nel corso di Complementi di Controlli e che è l' *assegnamento degli autovalori*.

Le specifiche, in generale, che desideriamo soddisfare sono

- Tempo di assestamento di 3.5 s
- Sovraelongazione percentuale dello 0 %
- Sforzo di controllo che deve essere contenuto nel range di funzionamento dei motori descritti nel datasheet

Per soddisfare tutte queste specifiche la scelta dei poli è stata effettuata con una procedura di *trial and error* che alla fine ha portato ad approssimare il nostro sistema del quarto ordine, ad un sistema del secondo ordine con una coppia di poli reali dominanti vicini tra loro

%assegnamento poli

```
p1=-0.9;  
p2=p1-0.1;  
p3=10*p1;  
p4=10*p2;  
p=[p1 p2 p3 p4];
```

%matrice di guadagno K

```
K=place(A,B,p);
```

3.1 Matrice di guadagno in feed-forward

La prima tecnica di controllo con l'assegnamento degli autovalori che andremo ad adottare, prevederà una semplice matrice di guadagno in feed-forward. Essendo questo un sistema quadrato, perché ha 2 ingressi e 2 uscite, la matrice di preconditionamento del riferimento G la calcoleremo con la seguente formula

$$G = [C(-A + BK)^{-1}B]^{-1} \quad (3.1)$$

cioè ricorrendo all'inversa, e non alla pseudoinversa. E questo sarà il codice MATLAB per calcolarla

```
Gbar=C*((-A+B*K)\B);
I=eye(2);
G=Gbar\I;
```

Per effettuare tutte le simulazioni a riguardo abbiamo progettato il seguente schema Simulink in cui vengono messi a confronto il controllore lineare il modello non lineare

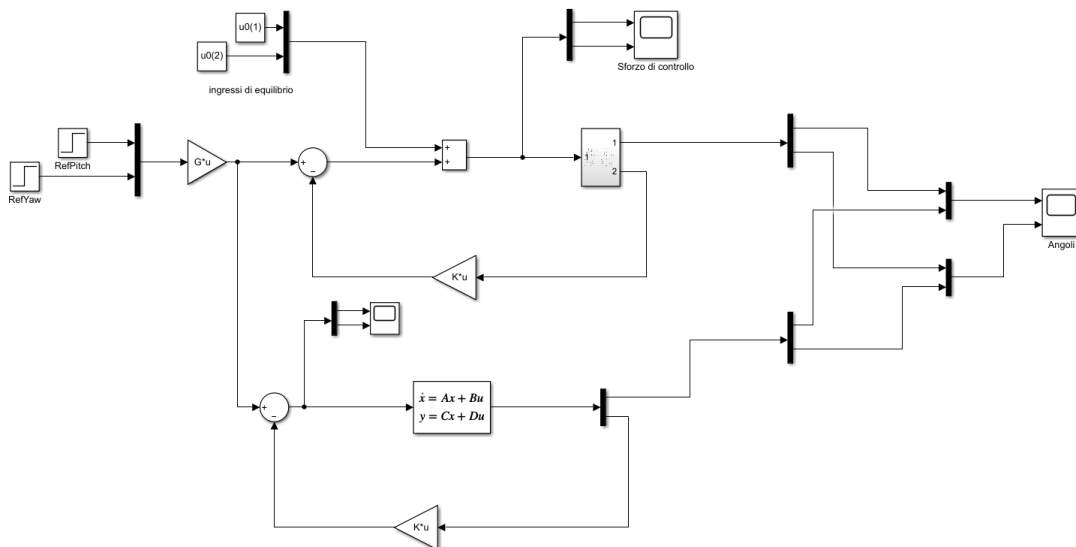


Figura 3.1: Schema Simulink della retroazione di stato con guadagno

Dando un riferimento di 0,3 radianti (che corrispondono a circa 17 gradi) avremo i seguenti grafici per l'andamento dell'angolo di pitch e dell'angolo di yaw, per ognuno dei quali è stato fatto il confronto tra non lineare (in blu) e lineare (in rosso)

Come si può notare, il controllore lineare così progettato viene seguito dal modello non lineare, a meno di un tempo di salita un po' più elevato per l'angolo di

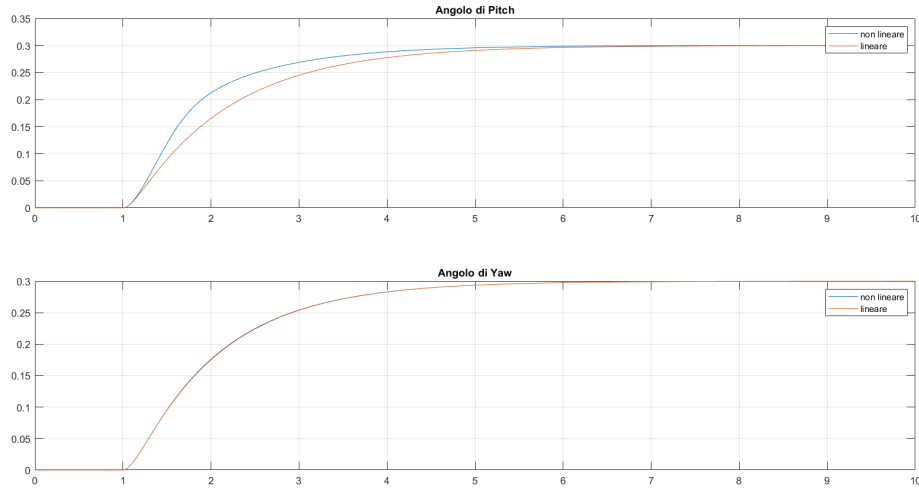


Figura 3.2: Andamenti dell'angolo di Pitch e di Yaw

pitch. Per quanto riguarda invece lo sforzo di controllo questo risulterà essere più che accettabile

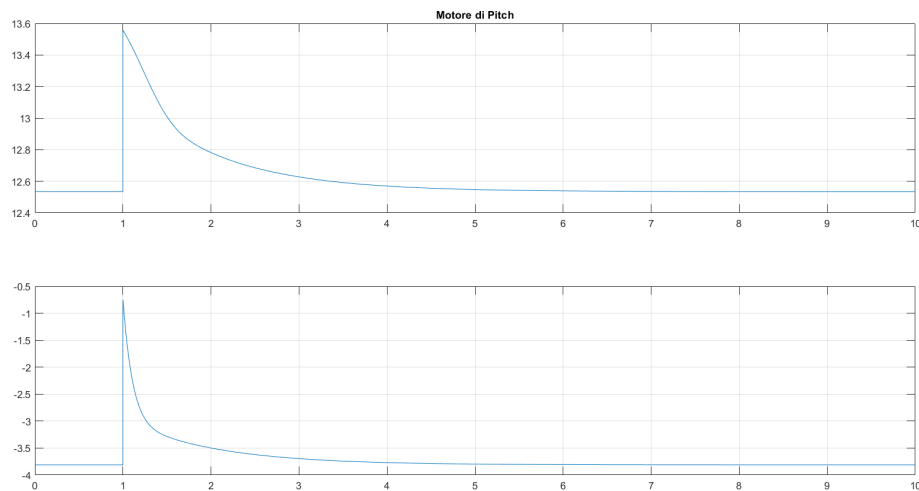


Figura 3.3: Andamenti del motore di Pitch e di Yaw

Nonostante i grafici prodotti siano in linea con le specifiche richieste, questa soluzione risulterà essere poco robusta nel momento in cui dovessero agire dei disturbi. Ad essere precisi, il sistema lineare in questione dato che presenta un doppio polo nell'origine, non risentirà in alcun modo di eventuali incertezze sui parametri del sistema, non vale però lo stesso discorso però per l'azione di eventuali disturbi, infatti qualora agisse un disturbo, per esempio, in ingresso al sistema, avremmo i risultati prodotti in figura 3.4.

Riassumendo quindi, gli svantaggi del controllore impiegato sono

1. Si deve presupporre la conoscenza della variabili di stato;

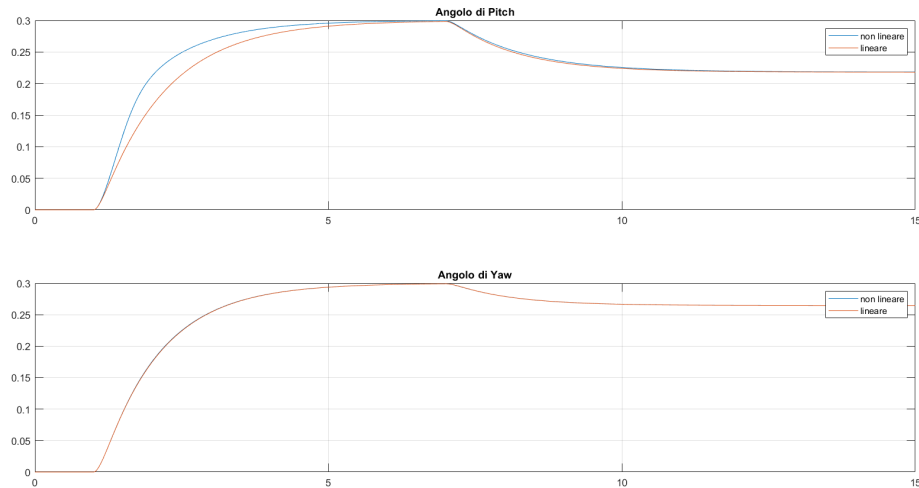


Figura 3.4: Andamenti dell'angolo di Pitch e di Yaw in presenza di disturbi

2. non c'è astatismo rispetto ai disturbi a gradino
3. l'andamento dell'angolo di pitch del lineare e del non lineare non coincidono alla perfezione come visto in figura 4.2

3.2 Aggiunta dell'azione integrale

Per poter risolvere la problematica legata alla mancanza di astatismo rispetto ai disturbi evidenziata nella progettazione del controllore precedente, una soluzione molto comune che viene adottata è l'aggiunta di un'azione integrale. Quando si aggiunge un'azione integrale però, bisogna andare a considerare il sistema di ordine aumentato, dove l'ordine con cui aumenta è pari al numero di integratori aggiunti, e il numero di integratori aggiunti è pari al numero di uscite del sistema. Nel nostro caso, avendo di base un sistema di ordine 4 e con 2 uscite, il sistema complessivo sarà di ordine 6, con le seguenti matrici così fatte:

```
%% Progettazione in retroazione di stato con azione integrale
[p, n]=size(C); %restituisce il numero di uscite
%matrici del sistema aumentato di ordine n+p
Ag=[A zeros(n,p); -C zeros(p,p)];
Bg=[B; zeros(p,p)];
Cg=[C zeros(p,p)];
Dg=D;
```

Dopodiché per poter vedere se è possibile o meno assegnare gli autovalori del sistema a ciclo chiuso ci dobbiamo accertare che la coppia (A_g, B_g) del sistema aumentato

sia completamente controllabile. Un semplice test per verificare ciò, dalla teoria, è possibile effettuarlo controllando che

$$\text{rank}\left(\begin{bmatrix} A & B \\ -C & 0 \end{bmatrix}\right) = n + p$$

dove n è l'ordine del sistema, e p il numero di uscite. Nel nostro caso tale test è verificato.

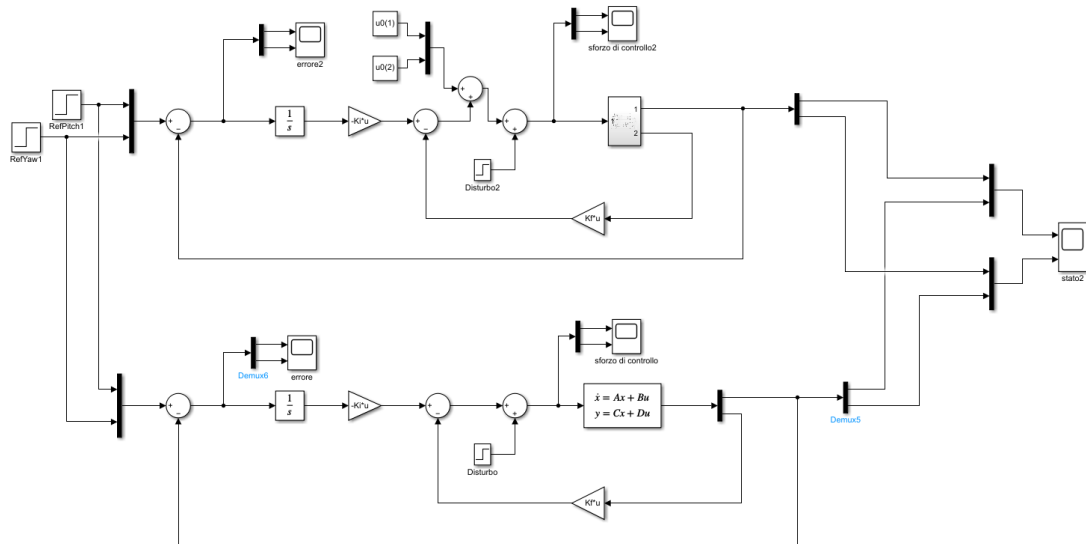


Figura 3.5: Schema Simulink con azione integrale

Dato che adesso c'è un sistema aumentato, il numero di poli da assegnare dovrà essere maggiore, precisamente dovremo assegnare 2 poli in più e questo poli li assegneremo anch'essi distanti di una decade dai due poli reali dominanti per non compromettere l'approssimazione del sistema ad uno del secondo ordine

```
p1=-0.9;
p2=p1-0.1;
p3=10*p1;
p4=10*p2;
p5=p3-2;
p6=p4-2;
p=[p1 p2 p3 p4 p5 p6];
```

```
%matrice di guadagno K
K=place(Ag,Bg,p);
Kf=K(:,1:4);
Ki=K(:,5:end);
```

E di conseguenza anche la matrice di guadagno K avrà dimensioni aumentate rispetto a prima, dove la parte aggiuntiva sarà relativa alle variabili aggiunte del banco di integratori. Lo schema Simulink che si è andato ad implementare è mostrato in figura 3.10.

I risultati delle simulazioni sono i seguenti per l'angolo di pitch e di yaw. Come

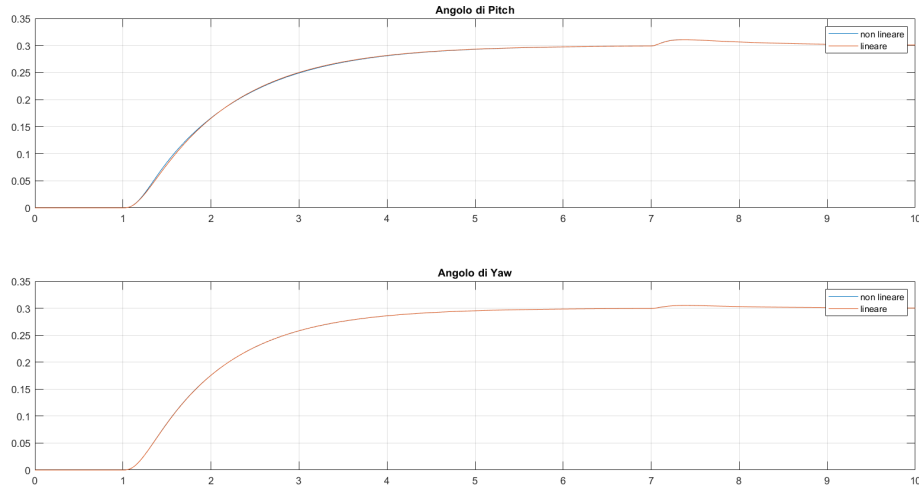


Figura 3.6: Andamento dell'angolo di pitch e di yaw

possiamo notare stavolta con l'aggiunta dell'azione integrale, c'è perfetta coincidenza tra i risultati prodotti dal modello non lineare e lineare, e inoltre è garantito pure astatismo rispetto a disturbi costanti, infatti all'istante $t=7$ s agisce un disturbo, ma alla fine il sistema si riporterà nell'orientamento specificato dal riferimento. Per quanto riguarda invece lo sforzo di controllo si avrà un leggero aumento rispetto a prima ma che tutto sommato anche questo si può ritenere accettabile.

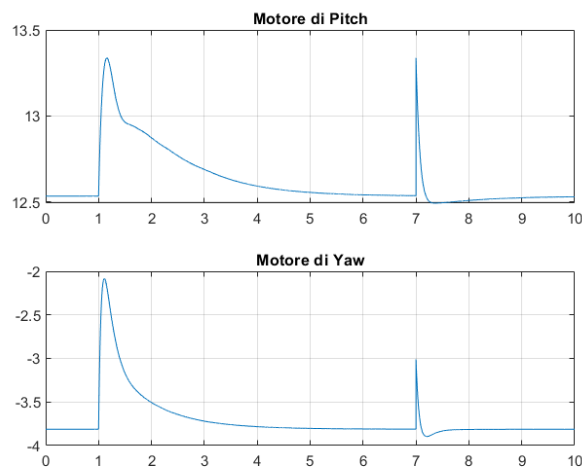


Figura 3.7: Andamenti del motore di Pitch e di Yaw

Notiamo dunque che 2 degli svantaggi che avevamo evidenziato per il controllore precedente, adesso non ci sono più, pena però un lieve aumento dello sforzo di controllo. Per di più resta il problema che si devono conoscere le variabili di stato.

3.3 Aggiunta dell'osservatore

Come già accennato, le tecniche precedentemente adottate presentano il grosso difetto che si debba conoscere le variabili di stato del sistema. Per questo motivo adesso rispetto alla retroazione di stato con azione integrale, si va ad aggiungere un osservatore a cui viene affidato tale compito. Sappiamo che l'equazione dell'osservatore risulta essere

$$\dot{\hat{x}} = A\hat{x} + Bu - L(y - \hat{y}) \quad (3.2)$$

per cui per progettare l'osservatore bisognerà progettare la matrice L , e tale matrice L non viene scelta a caso, ma secondo un opportuno criterio, che è quello di far tendere asintoticamente a zero l'errore di stima $e(t) = x(t) - \hat{x}(t)$. Dato che l'errore di stima dalla teoria sappiamo che è in evoluzione libera, cioè presenta la seguente espressione

$$e(t) = e^{(A-LC)t}e(0) \quad (3.3)$$

Se ne deduce che affinché l'errore di stima tenda asintoticamente a zero, con un errore iniziale $e(0)$ non nullo, la matrice $A - LC$ deve essere asintoticamente stabile, quindi questo è il criterio in base al quale dovrà essere progettata la matrice L . È possibile dimostrare che l'assegnamento arbitrario degli autovalori della matrice $A - LC$ è un problema duale rispetto a quello di assegnamento di autovalori della matrice $A - BK$. Pertanto in sostanza quello che possiamo fare è usare gli stessi comandi di prima per progettare la matrice L ma mettendo un opportuno segno di trasposizione.

In generale per poter progettare un controllore in retroazione di stato che fa uso anche dell'osservatore, vale il principio di separazione in base al quale le matrici di guadagno K e L vengono progettate in maniera indipendente l'una dall'altra, con l'unico vincolo però che gli autovalori della matrice L siano messi ancora più distanti dall'asse immaginario nel semipiano sinistro del piano complesso perché è importante che la dinamica con cui l'errore va a zero sia più veloce della dinamica del sistema a ciclo chiuso, perché l'osservatore deve essere pronto a riportare a zero l'errore di stima. Infatti in genere tali autovalori vengono messi ad una decade di distanza rispetto a quelli usati per progettare la matrice K . Questa però non è la regola perché quello che può succedere è che si vada ad amplificare errori di misura, o ad aumentare lo sforzo di controllo, proprio come nel nostro caso. Infatti a questo punto si decide di "avvicinare" i poli dell'osservatore a discapito di una dinamica

più lenta per far mantenere contenuti gli sforzi di controllo. Nel nostro caso dati gli sforzi di controllo molto elevati visionati nelle simulazioni, non solo abbiamo dovuto abbassare i poli dell'osservatore, ma abbiamo ritenuto opportuno modificare anche i poli del sistema a ciclo chiuso, come si può vedere dal codice qui di seguito riportato

```

p1=-0.9;
p2=p1-0.1;
p3=7.5*p1;
p4=7.6*p1;
p5=p3-0.5;
p6=p4-0.5;
p=[p1 p2 p3 p4 p5 p6];

%matrice di guadagno K
K=place(Ag,Bg,p);

Kf=K(:,1:4);
Ki=K(:,5:end);

% Progettazione osservatore
po=[6.85*p4, 6.85*p4+0.1, 6.85*p4+0.2, 6.85*p4+0.3];
L=place(A',C',po)';
Aoss=A-L*C;
Boss=[B-L*D L];
Coss=eye(4);
Doss=zeros(4,4);

```

Questo è dovuto principalmente al fatto che si è voluto considerare il caso ancora peggiore in cui le condizioni iniziali dell'impianto non coincidessero con quelle dell'osservatore, precisamente si è ipotizzato per l'impianto una condizione iniziale di $x_0 = [0.05; 0.05; 0; 0]$, mentre le condizioni iniziali dell'osservatore sono state settate a zero, in modo tale che l'errore di stima partisse da un valore iniziale non nullo e convergesse a zero secondo la dinamica dettata dai poli assegnati. In base alle simulazioni effettuate, questo costituisce il motivo principale per cui si è dovuto fare numerose prove (cambiando opportunamente i poli) in maniera tale l'osservatore funzionasse anche sul non lineare senza fondere i motori dati e l'andamento dell'angolo di pitch e di yaw fosse quanto meno accettabile in base alle specifiche richieste. Mentre invece se avessimo settato le condizioni iniziali dell'impianto anche'esso a zero, o per lo meno a valori più bassi di 0.05 per gli angoli di pitch e di yaw avremmo avuto sicuramente meno problemi.

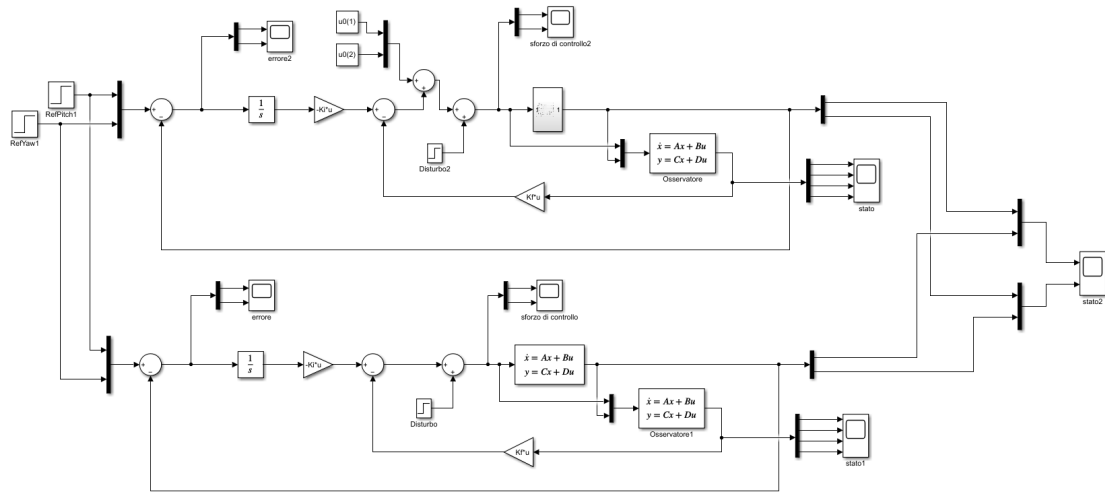


Figura 3.8: Schema Simulink con osservatore

Lo schema simulink progettato per le nostre simulazioni è quello in figura 4.5. Lanciando questo programma, avremo i seguenti risultati relativamente allo sforzo di controllo

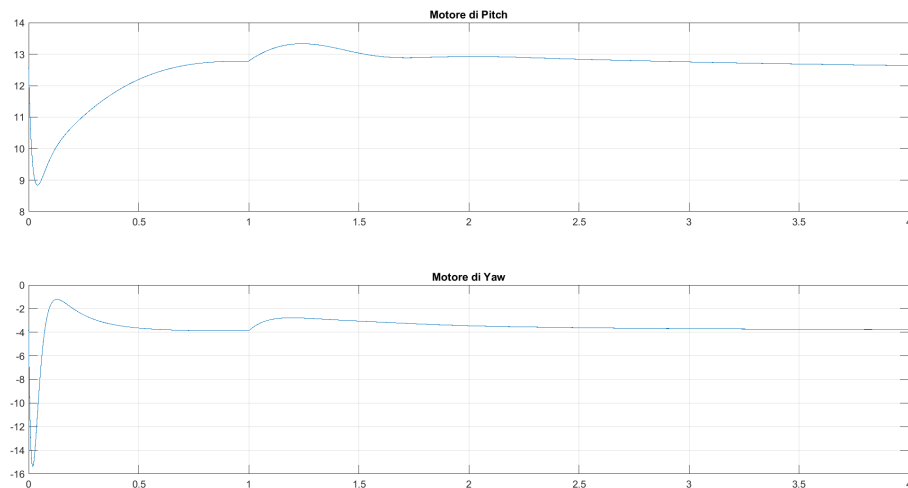


Figura 3.9: Andamenti del motore di Pitch e di Yaw

Come possiamo notare lo sforzo di controllo aumenta, e per il motore di yaw abbiamo cercato di spingere lo sforzo di controllo ai limiti massimi consentiti, e non oltre. Per fare ciò abbiamo dovuto pagare qualcosa in termini di prestazioni per l'angolo di pitch e di yaw (figura.). Come possiamo osservare infatti, per preservare lo sforzo di controllo abbiamo dovuto abbassare i poli dell'impianto a ciclo chiuso e quelli dell'osservatore, pena però la presenza di una sottoelongazione per il modello non lineare relativamente all'angolo di pitch e pena in generare un aumento di 0.3 s del tempo di assestamento. Quindi ricapitolando per poter avere una stima delle variabili di stato tramite l'introduzione dell'osservatore, lo sforzo di controllo

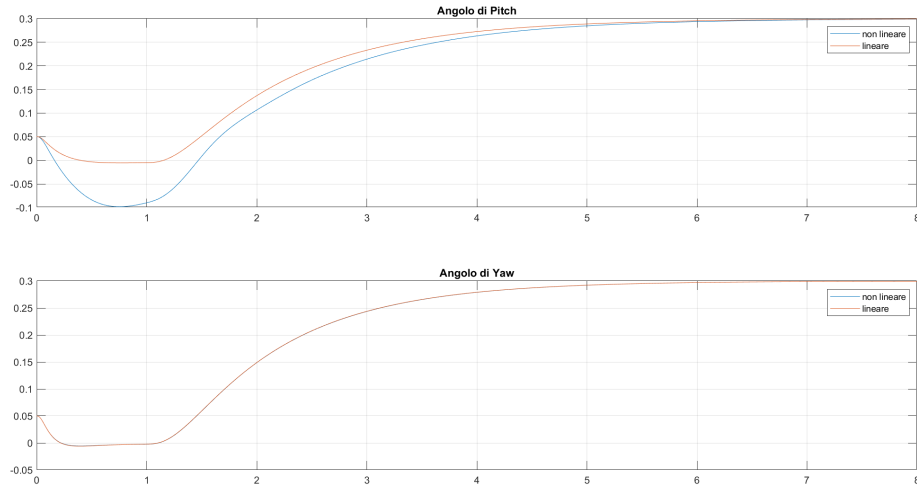


Figura 3.10: Andamento dell'angolo di pitch e di yaw

è aumentato oltre misura, e per contenerlo entro il range di funzionamento dei motori abbiamo dovuto rilassare la specifica sul tempo di assestamento. Purtroppo la coperta è corta e le decisioni da prendere dipendono da quello che si vuole fare:

- se si vuole a tutti i costi rispettare la specifica sul setting time, si dovranno usare motori più potenti
- Se invece si è obbligati ad usare questi motori si accetta un rilassamento delle specifiche
- Se invece si desidera rispettare le specifiche sia sul setting time che sullo sforzo di controllo, ci si dovrà rassegnare a non usare un osservatore

Capitolo 4

Controllo ottimo LQ

Il controllo ottimo LQ (*Linear Quadratic*) è un'altra tecnica di sintesi di controllori multivariabili basata sulla retroazione di stato. In questo caso la legge di controllo è progettata con lo scopo di minimizzare un certo funzionale di costo quadratico, in cui vengono pesati opportunamente lo stato e l'ingresso attraverso delle matrici Q ed R rispettivamente. Il controllo LQ che noi andiamo a considerare è su orizzonte temporale infinito, e per ottenere la legge di controllo ottimo che risolve questo problema, bisogna trovare l'unica soluzione definita positiva dell'equazione algebrica di Riccati

$$A^T \bar{P} + \bar{P} A + Q - \bar{P} B R^{-1} B^T \bar{P} = 0 \quad (4.1)$$

e sostituirla nella seguente espressione per ottenere la trice di guadagno K

$$K = -R^{-1} B^T \bar{P} \quad (4.2)$$

Dato che anche in questo caso parliamo di una tecnica che fa uso del concetto di retroazione di stato, e che si differenzia solo per il modo con cui viene progettata la matrice di guadagno K , anche qui procederemo vedendo prima una tecnica di controllo che fa uso della sola matrice di guadagno in feed-forward, poi andremo a fare uso di un azione integrale per garantire astatismo rispetto ai disturbi e infine dato che, come sappiamo, il grosso difetto della retroazione di stato è che si devono conoscere le variabili di stato, anche qui abbiamo implementato un osservatore che stavolta dovrà anch'esso rispettare un criterio di ottimalità: il filtro di kalman.

4.1 LQ con matrice di guadagno in feed-forward

Per questo argomento saremo molto più sintetici rispetto a prima, perché come già detto per le diverse tecniche di controllo LQ ricalcano quelle con assegnamento degli autovalori dove l'unica differenza è il comando MATLAB con il quale si va a

progettare la matrice di retroazione K , e che è "lqr". Infatti il codice che implementeremo sarà lo stesso identico codice già visto nel paragrafo 3.1, con la matrice di preconditionamento del riferimento che non cambierà, ma con la matrice K così calcolata

```
sysHeli=ss(A,B,C,D);
Q=diag([8 8 10 10]);
R=diag([0.03 0.1]);
K=lqr(sysHeli,Q,R);
```

Anche lo schema simulink in questo caso sarà lo stesso di quello già presentato in figura 3.1. I risultati delle simulazioni invece sono i seguenti per l'angolo di pitch e di yaw

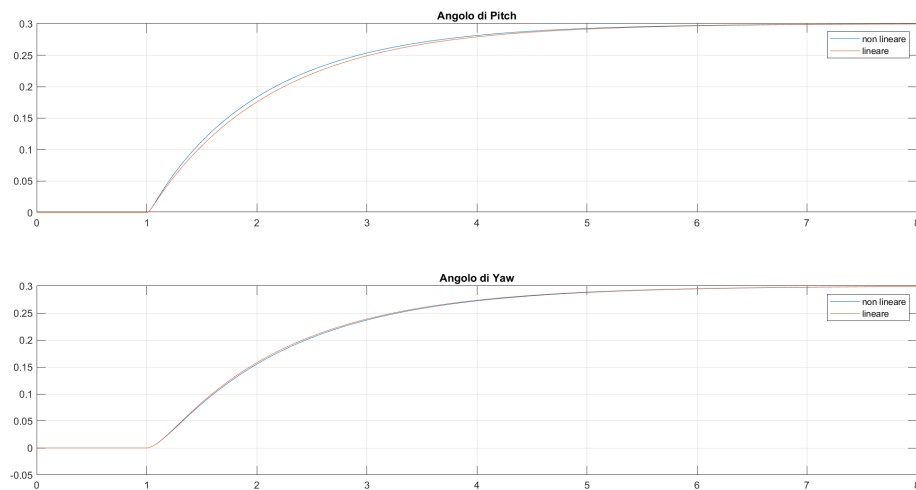


Figura 4.1: Andamenti dell'angolo di Pitch e di Yaw

e questi altri per lo sforzo di controllo

Già da qui possiamo riscontrare la maggiore efficienza del controllo LQ rispetto all'assegnamento degli autovalori. Infatti se facciamo un confronto con i grafici relativi all'angolo di pitch tra il controllore con matrice di guadagno G progettato con assegnamento degli autovalori e questo qui progettato con controllo ottimo LQ, notiamo che stavolta il modello non lineare segue meglio il modello lineare, rispetto al caso precedente in cui il modello non lineare aveva un tempo di salita più piccolo. Questo perché stavolta abbiamo la possibilità di regolare singolarmente le varie variabili di stato, nello specifico aumentando i pesi della matrice Q relativi alle velocità di pitch e di yaw abbiamo avuto la possibilità di abbassare le velocità facendo in modo che diminuisca il tempo di salita. Questo però ovviamente ha portato ad un rallentamento del tempo di assestamento che siamo però riusciti a

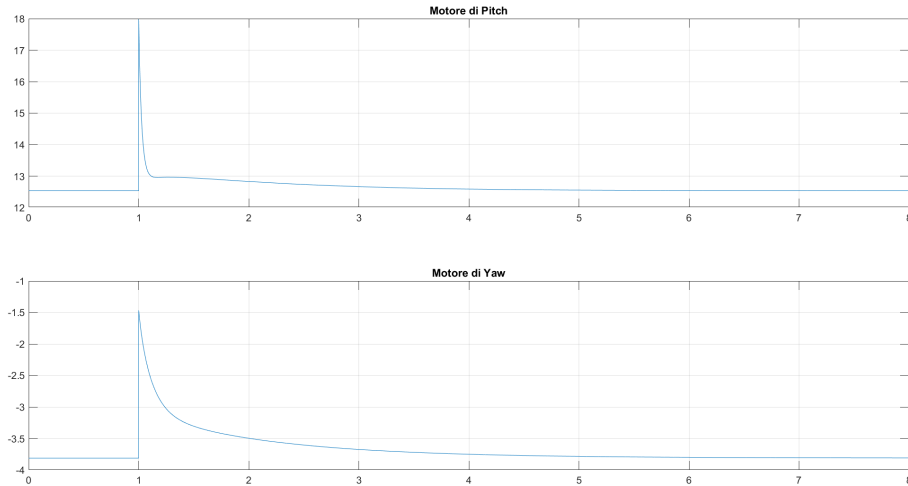


Figura 4.2: Andamenti del motore di Pitch e di Yaw

risolvere diminuendo i pesi della matrice R relativa allo sforzo di controllo. Dunque noi ci saremmo potuti limitare ad aumentare come fatto solo i pesi della matrice Q relativi all'angolo di pitch e di yaw, ma avremmo ottenuto gli stessi identici grafici già ottenuti utilizzando l'assegnamento degli autovalori, invece procedendo come descritto sopra siamo riusciti ad ottenere risultati addirittura migliori.

4.2 LQ con azione integrale

Procedendo sulla stessa scia già vista in precedenza, per poter garantire astatismo rispetto ai disturbi andiamo ad impiegare un'azione integrale al posto della matrice di guadagno in feed-forward. Anche qui lo schema simulink è lo stesso di figura 3.10 e il codice MATLAB è lo stesso già visto nel paragrafo 3.2, dove l'unica differenza è che nella progettazione della matrice Q avremo dimensioni maggiori, cioè se prima avevamo 4 variabili di stato e la matrice Q l'abbiamo scelta diagonale 4×4 , adesso noi poiché stiamo utilizzando un'azione integrale, l'ordine del sistema aumenterà di un numero pari al numero di azioni integrali che sappiamo nel nostro caso essere 2. Quindi stavolta semplicemente la matrice Q dovrà essere una matrice 6×6 :

```
Q=diag([10 10 10 10 14 14]);
R=diag([0.1 0.01]);
K=lqr(Ag,Bg,Q,R);
```

Qui si dovrà appunto regolare dei pesi anche per le variabili di stato aggiuntive e relative agli integratori. Lanciando il nostro schema Simulink avremo i seguente risultati per l'angolo di pitch e di yaw

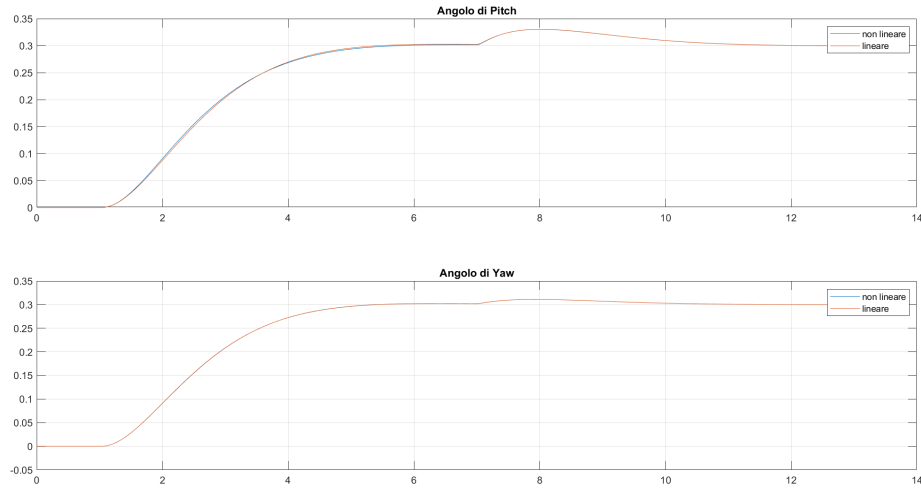


Figura 4.3: Andamento dell'angolo di pitch e di yaw

Mentre per lo sforzo di controllo si avrà che

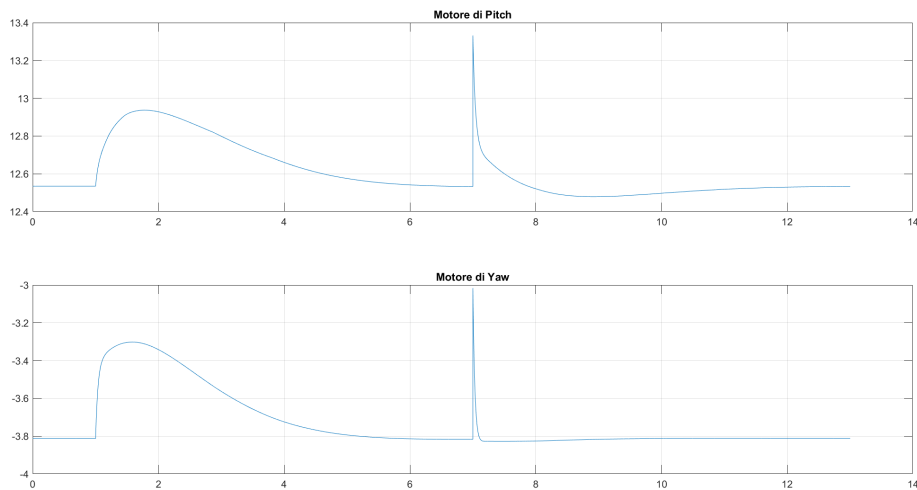


Figura 4.4: Andamenti del motore di Pitch e di Yaw

4.3 Controllo LQG

Il controllo *Linear Quadratic Gaussian*, o LQG, è ottenuto dalla combinazione della soluzione LQ di un problema deterministico con stato misurabile, e un filtro di Kalman usato per la stima dello stato quando rumori stocastici affliggono l'impianto. Quindi il controllo LQG è un problema lineare quadratico risolto in ambito stocastico. Infatti noi sappiamo che la progettazione di un controllore con la tecnica di controllo ottimo LQ è una tecnica fondamentalmente di retroazione di stato, per cui anche in questo caso si avrà il problema e l'esigenza di dover misurare le variabili

di stato. Il problema è che in questo caso non possiamo fare ricorso all'osservatore di Luemberger, perché nel stimare lo stato, poiché in genere si parte da un errore iniziale non nullo, l'evoluzione dello stato può avere andamenti non previsti dalla legge di controllo ottimo, cioè in sostanza è che si perde l'ottimalità. Per cui nasce l'esigenza di progettare uno stimatore ottimo che è il filtro di Kalman.

È possibile dimostrare che la progettazione del filtro di Kalman a regime, è un problema duale rispetto a quello del problema di controllo ottimo LQ su orizzonte infinito, infatti anche qui si dovrà risolvere un'equazione algebrica di Riccati

$$0 = A\tilde{P} + \tilde{P}A^T + \tilde{Q} - \tilde{P}C^T\tilde{R}^{-1}C\tilde{P} \quad (4.3)$$

la cui unica soluzione definita positiva mi permette di determinare la matrice di guadagno

$$L = PC^T\tilde{R}^{-1} \quad (4.4)$$

se ne deduce che la determinazione di tale matrice si può effettuare risolvendo un problema di controllo ottimo LQ sulla coppia (A^T, C^T) .

Quindi per la progettazione del filtro di Kalman in MATLAB useremo comandi simili a quelli visti in precedenza

```
T=(10/3)*1e-3;
v_x=8e-4;
v_y=8e-8;
Qtilde=v_x*diag([1 1 1 1]);
Rtilde=v_y*diag([1 1]);
L=lqr(A',C',Qtilde,Rtilde)';

Afk=A-L*C;
Bfk=[B-L*D L];
Cfk=eye(4);
Dfk=zeros(4,4);
```

Dove stavolta va specificata opportunamente anche la presenza di eventuali rumori di processo e di misura di cui assumeremo note le varianze. Come sono state scelte le matrici \tilde{Q} e \tilde{R} ? Tipicamente quello che si fa è partire da un dato iniziale che è un'informazione su quanto sono precise le misure, e quindi il valore della \tilde{R} , dopodiché si va opportunamente a tunare \tilde{Q} . Dato che nel nostro caso l'obiettivo era fare in modo che il tutto funzionasse sul non lineare ovviamente, cioè che l'osservatore fornisse stime veloci, abbiamo dovuto scegliere una \tilde{Q} che è più grande rispetto a \tilde{R} , in questa maniera l'osservatore si fiderà più delle misure che delle equazioni di

stato, quindi sceglierà una L molto grande per correggere e l'osservatore diventa più veloce. Questo ovviamente penalizzerà lo sforzo di controllo.

Si può verificare poi che la progettazione del filtro di kalman può essere tranquillamente affiancata a quella di un controllore in retroazione di stato progettata con controllo LQ in maniera indipendente tra di loro, ovvero sussiste un principio di separazione tra le due cose, ed è in questo consiste il controllo LQG.

Il controllore K progettato con LQ è il seguente

```
[p, n]=size(C); %restituisce il numero di uscite

%matrici del sistema aumentato di ordine n+p
Ag=[A zeros(n,p); -C zeros(p,p)];
Bg=[B; zeros(p,p)];
Cg=[C zeros(p,p)];
Dg=D;

Q=diag([8 8 0.0006 0.1 14 14]);
R=diag([0.0003 0.0003]);
K=lqr(Ag,Bg,Q,R);

Kf=K(:,1:4);
Ki=K(:,5:end);
```

Lo Schema Simulink usato per le nostre simulazioni sarà questo qui

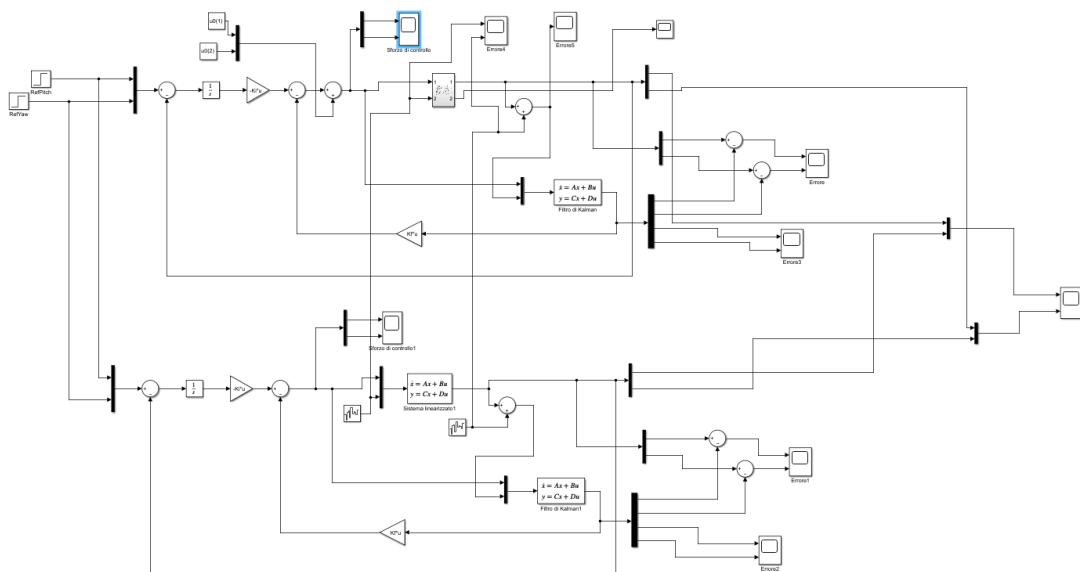


Figura 4.5: Schema Simulink con filtro di Kalman

dove per poter simulare i rumori di processo e di misura abbiamo usato dei blocchi "white noise" settato con un sample time pari al valore T visto nel codice di prima, e che stato scelto come l'inversa della frequenza a due decadi di distanza da quella di attraversamento dell'impianto.

I risultati delle simulazioni sono riportati qui di seguito per l'angolo di pitch e di yaw

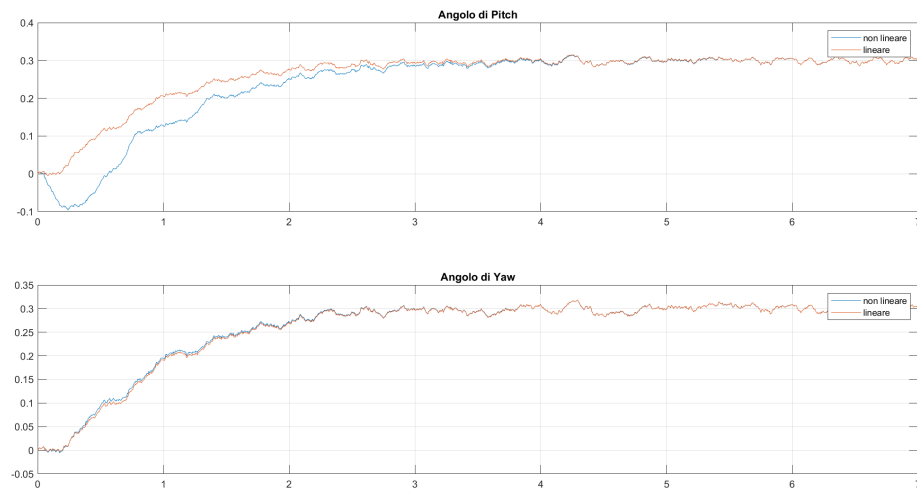


Figura 4.6: Andamento dell'angolo di pitch e di yaw

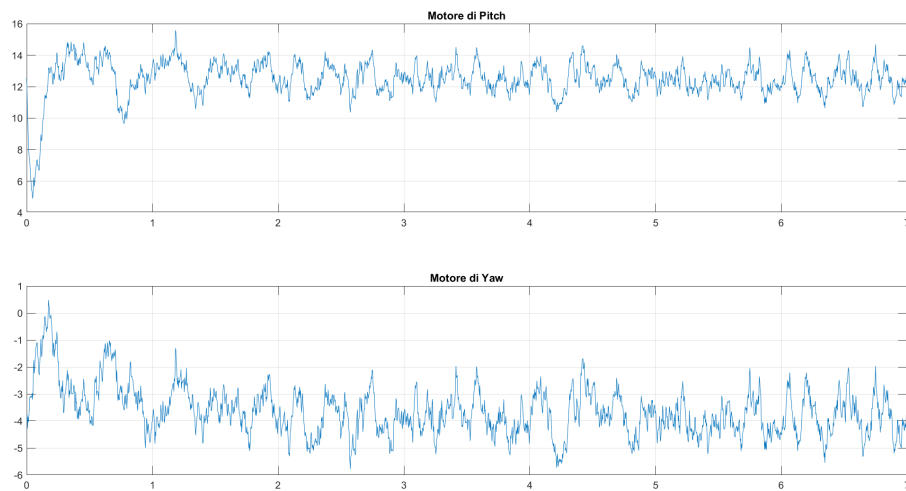


Figura 4.7: Andamenti del motore di Pitch e di Yaw

Capitolo 5

Controllo H_∞

Il controllo H_∞ è una tecnica di sintesi di controllori per sistemi multivariabili sviluppata a partire dagli anni 80 e che si distacca da quelle trattate fino ad ora. Infatti mentre il controllo ottimo fa ricorso ad una rappresentazione implicita i-s-u dell'impianto, il controllo H_∞ sostanzialmente costituisce un approccio basato sul dominio della frequenza, proprio come le tecniche di sintesi sviluppate nel controllo classico per la progettazione della risposta armonica di regolatori monovariabili. Considerando il classico schema a blocchi in anello chiuso visto a Controlli Automatici, nel caso multivariabile valgono le seguenti relazioni nel dominio della frequenza tra le variabili:

$$\begin{cases} Y = T(Y^0 - N) + SD_y + SGD_u \\ U = RS(Y^0 - D_y - N) \\ E = S(Y^0 - D_y - N) \end{cases} \quad (5.1)$$

In base a queste formule possiamo tradurre il soddisfacimento di specifiche sulle performance statiche e dinamiche attraverso delle condizioni sulle funzioni di sensitività e sensitività complementare:

1. $T \cong I$ alle basse frequenze, in maniera tale da avere un buon tracking del riferimento, inoltre T deve essere piccolo dove sono presenti le componenti frequenziali del rumore, ovvero alle alte frequenze.
2. Per quanto riguarda la S , questa è tale per cui $S + T = I$ quindi dovrà essere piccole alle basse frequenze per minimizzare l'effetto del rumore N sull'uscita.
3. Inoltre dobbiamo fare in modo che RS sia piccola per minimizzare lo sforzo di controllo.

Il soddisfacimento di tutte queste specifiche può essere riassunto attraverso un'unica condizione che è la seguente

$$\|G_{zw}\|_\infty < \gamma \quad (5.2)$$

In particolare obbiettivo di tale controllo è progettare un regolatore chiuso in retroazione attraverso la matrice $P(s)$ tale da minimizzare questa norma in H_∞ .

Il primo step è quello di stampare massimo e minimo valori singolari del sistema per avere un'idea del condition number

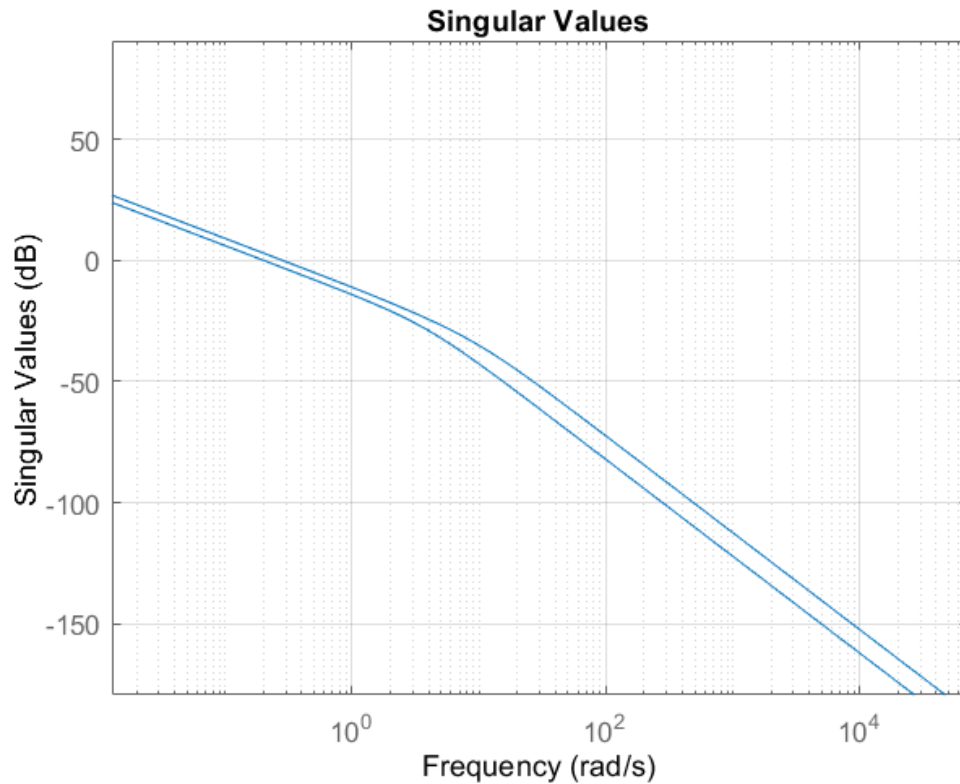


Figura 5.1: Numero di condizionamento

Dopodiché andiamo a settare le nostre matrici di peso W_s , W_k e W_t opportunamente, dove la matrice W_t nello specifico la setteremo nulla poiché risulta essere complementare alla W_s e nel nostro caso non c'è alcun vincolo sulle incertezze moltiplicative da dover soddisfare. Quindi

```
W1 = 5000/((180*s+1));
Ws=append(4*W1,4*W1);
Wk = diag([48 45]);
Wt = [];
```

```
%creazione della matrice di P(s)
P = augw(G,Ws,Wk,Wt);
```

e si potrà poi procedere alla creazione della matrice $P(s)$ attraverso il comando "augw". In seguito si può procedere alla creazione del controllore K attraverso il comando *hinfsyn* che va utilizzato come di seguito

```
%creazione del controllore
```

```
[K,CL,GAM] = hinfsyn(P);
```

```
L = G*K; %funzione d'anello
```

```
T = feedback(L,eye(2)); %funzione di sensitività complementare
```

```
S = eye(2)-T; %funzione di sensitività
```

qui si può notare che abbiamo definito pure la funzione d'anello $L(s)$, la funzione di sensitività $S(s)$ e la funzione di sensitività complementare $T(s)$ i sigma plot ci sono stati molto utili nella progettazione delle matrici di peso viste prima.

Adesso per visualizzare i nostri risultati abbiamo progettato il seguente schema simulink Come possiamo notare abbiamo inserito un segnale sinusoidale all'uscita

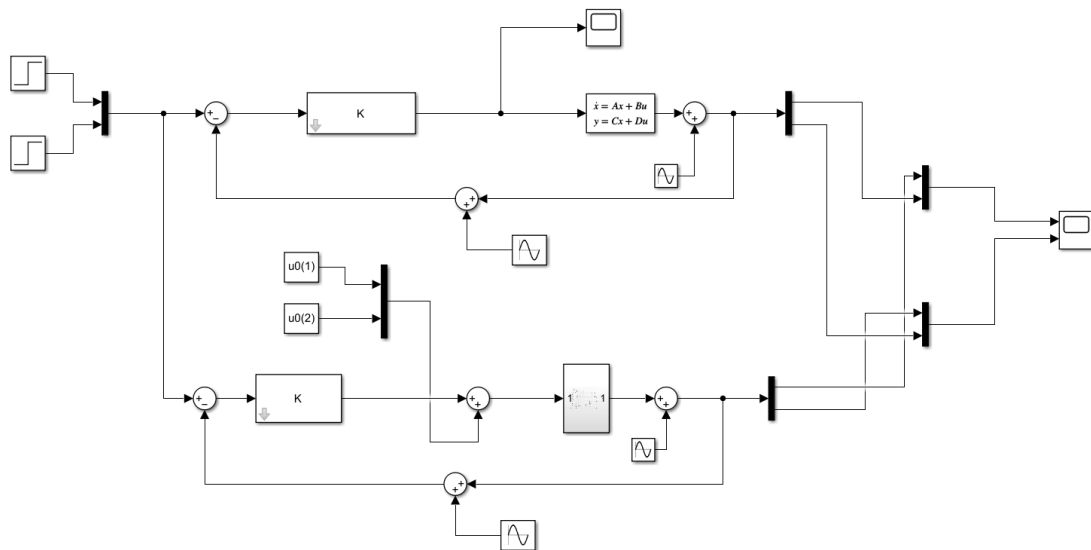


Figura 5.2: Schema Simulink del controllo Hinf

dell'impianto, e un altro sulla linea di retroazione di stato. Questi due stanno a rappresentare un disturbo sull'uscita e un rumore di misura rispettivamente. Qui la nostra intenzione è stata far vedere che inserendo un segnale a frequenza molto bassa in uscita all'impianto, e un segnale a frequenza elevata sulla linea di retroazione, riusciremo ad ottenere comunque i nostri risultati, perché abbiamo scelto le matrici di peso in modo tale che la $T(s)$ sia molto bassa alle alte frequenze, e la $S(s)$ fosse molto bassa invece alle basse frequenze.

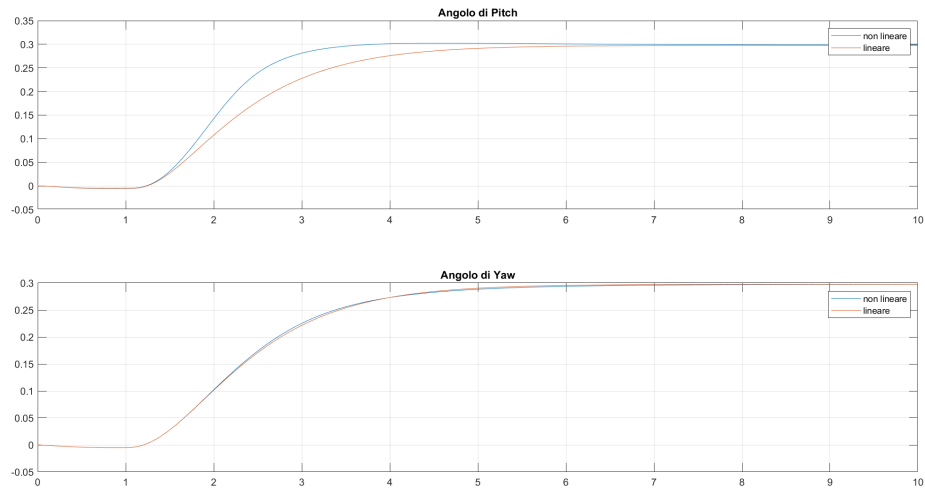


Figura 5.3: Andamenti dell'angolo di Pitch e di Yaw

Questi saranno i risultati delle nostre simulazioni per l'angolo di pitch e di yaw mentre per lo sforzo di controllo in assenza di disturbi e rumori sarà questo

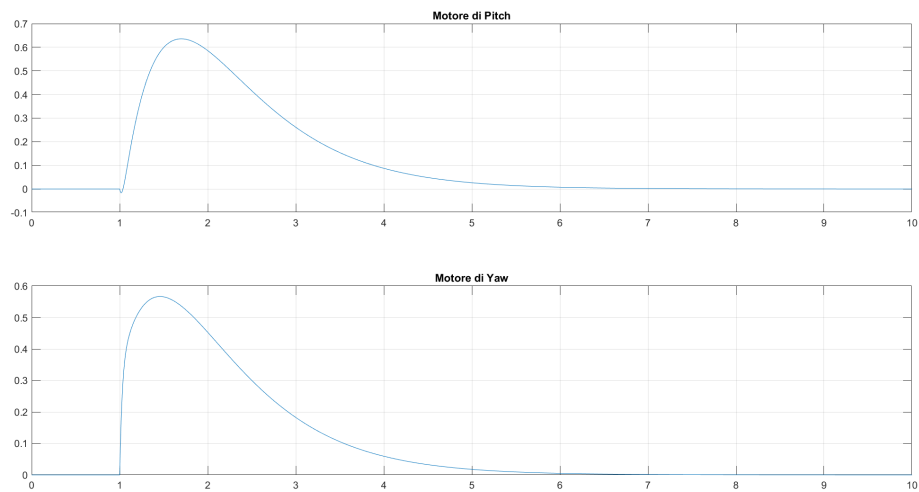


Figura 5.4: Andamenti del motore di Pitch e di Yaw