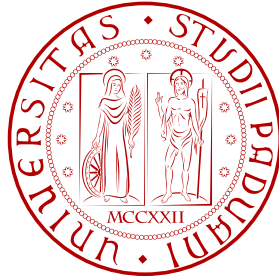


UNIVERSITÀ DEGLI STUDI DI PADOVA



Facoltà di Ingegneria  
Corso di Laurea Magistrale in Ingegneria dell'Automazione

Tesi di laurea

**Motion Cueing per simulatori dinamici di veicolo:  
applicazione delle funzioni di Laguerre a un  
controllore MPC e tuning**

Relatore: **Prof. Ing. Alessandro Beghi**

Correlatore: **Ing. Mattia Bruschetta**

Laureando: **Andrea Vezzano**

ANNO ACCADEMICO 2010 – 2011



# Indice

<b>Introduzione</b>	<b>V</b>
<b>1 Piattaforma di simulazione e Model Predictive Control</b>	<b>1</b>
1.1 Piattaforma VI-grade . . . . .	1
1.2 Model Predictive Control . . . . .	3
1.2.1 Vantaggi e svantaggi . . . . .	4
1.2.2 Elementi costitutivi . . . . .	5
1.3 Il problema MPC applicato al <i>motion cueing</i> . . . . .	6
1.3.1 Modellizzazione del sistema . . . . .	7
1.3.2 Formulazione del problema MPC . . . . .	16
1.3.3 Caso non vincolato . . . . .	19
1.3.4 Vincoli in ingresso ed uscita . . . . .	20
1.4 Analisi delle condizioni di simulazione . . . . .	22
1.4.1 Scelta del riferimento . . . . .	22
1.4.2 Strategie di predizione . . . . .	23
<b>2 Applicazione delle funzioni di Laguerre a un controllore MPC</b>	<b>27</b>
2.1 Funzioni di Laguerre . . . . .	27
2.2 Uso delle funzioni di Laguerre nella descrizione del sistema . . . . .	29
2.3 Applicazione al modello MPC . . . . .	31
2.3.1 Caso non vincolato . . . . .	34
2.3.2 Vincoli in ingresso e uscita . . . . .	35
2.4 Simulazioni . . . . .	37
<b>3 <i>Tuning</i> del controllore MPC</b>	<b>47</b>
3.1 <i>Tuning</i> del controllore MPC in riferimento al <i>motion cueing</i> classico	48
3.1.1 Parametri di taratura . . . . .	48
3.1.2 Risultati, osservazioni e limiti del <i>tuning</i> . . . . .	51
3.2 <i>Tuning</i> con architettura di controllo <i>feed-forward</i> . . . . .	58
3.2.1 Schema di controllo con <i>feed-forward</i> . . . . .	58
3.2.2 Risultati delle simulazioni . . . . .	61
3.2.3 Riduzione di <i>tilt coordination</i> . . . . .	67
3.3 <i>Tuning</i> e funzioni di Laguerre . . . . .	67

<b>Conclusioni e sviluppi futuri</b>	<b>71</b>
<b>Bibliografia</b>	<b>73</b>

# Introduzione

Il presente lavoro di tesi di laurea, si pone come la naturale prosecuzione del lavoro iniziato da Daniele D’Ambrosio[2] e proseguito dalle tesi “gemelle” di Mauro Baseggio [3] e Mauro Pozzi [4]. Il primo, per conto della ditta VI-grade [28], nel suo lavoro di tesi implementa una propria versione del controllo predittivo (MPC, Model Predictive Control) per realizzare algoritmi di *motion cueing* per il simulatore di guida automobilistica realizzato da VI-grade stessa. Il lavoro in questione, essendo il primo passo di un progetto complessivo ed essendo stato svolto a piattaforma simulativa non ancora assemblata, è un lavoro abbastanza teorico ed implementa un controllore lungo il solo asse  $x$ .

Con il lavoro di Pozzi e Baseggio tale modello viene esteso ai tre assi cartesiani e prevede, come primo passo, l’introduzione del modello percettivo umano, ovvero dell’apparato vestibolare.

Lo scopo di VI-grade infatti è quello di realizzare un simulatore di guida innovativo sotto molteplici punti di vista: innanzitutto la struttura meccanica, che non è quella di un esapodo tradizionale, in secondo luogo le dimensioni ridotte e infine, ma questo è l’aspetto più caratteristico e ambizioso del progetto, l’intenzione di riuscire a replicare, nonostante le dimensioni e dunque le capacità di moto contenute, delle accelerazioni il più possibile simili a quelle effettivamente riscontrabili in una situazione di guida anche estrema, come appunto la guida al limite su pista. Strumento principe per ottenere tale risultato è, come già accennato, l’MPC che, grazie alla sua peculiarità di andare in predizione e di calcolare l’ingresso ottimo ad ogni iterazione, si vuole comprendere e tarare nel miglior modo possibile, e in maniera indipendente (ovvero senza ricorrere ai toolbox preinstallati di MATLAB) in vista di una traduzione del codice nel linguaggio adoperato dal controllore della piattaforma.

Per riuscire a trasmettere quanto più possibile accelerazioni “veritiere” al pilota, si sfrutta la *tilt coordination*, ovvero l’accoppiamento tra moto longitudinale e angolo di *pitch* (per il moto lungo  $x$ ) e tra moto trasversale e angolo di *roll* (per il moto lungo  $y$ ). Il tutto grazie al fatto che il sistema vestibolare umano è facilmente ingannabile e dunque si sfrutta la componente gravitazionale che risulta dall’inclinazione della piattaforma per andare a riprodurre accelerazioni altrimenti impossibili da ottenere a causa dei limiti di spazio e di attuazione della piattaforma meccanica.

Il contributo che la presente tesi vuole fornire è duplice, per ambiti e obiettivi.

Nella prima parte si è guardato alla fase *on-line* dell'algoritmo di controllo con l'intento di velocizzare i tempi di calcolo in vista dell'utilizzo in *real-time* al quale è destinato. Allo scopo è stato utilizzato un approccio matematico basato sull'uso di particolari funzioni ortonormali, le funzioni di Laguerre.

Nella seconda parte ci si è occupati di una problematica fondamentale per l'impostazione del controllore, quella del *tuning*, che consiste nello studio del comportamento del controllore stesso in relazione alle variazioni dei parametri di regolazione disponibili ed alla conseguente elaborazione di strategie di azione sui parametri stessi per ottenere dinamiche di controllo desiderate.

Lo scopo principe del *tuning* è di ottenere la validazione del comportamento dell'algoritmo MPC tramite il confronto con il comportamento del *motion cueing* classico e, una volta ottenuta, elaborare una struttura di controllo che sia in grado, con gli sviluppi successivi, di fornire ad un utente non esperto una procedura di semplificata (e trasparente all'architettura di controllo utilizzata) per le operazioni di taratura della piattaforma.

# Capitolo 1

## Piattaforma di simulazione e Model Predictive Control

Il lavoro di tesi che ci si appresta a descrivere fa parte di un progetto più esteso, sviluppato per la ditta VI-grade, il cui scopo finale è la realizzazione di una piattaforma di simulazione di guida innovativa che coniughi la compattezza delle dimensioni fisiche alla fedeltà nella riproduzione degli stimoli sensoriali cui è sottoposto un pilota in condizioni di guida estrema.

Più nello specifico ci si vuole occupare dell'aspetto del controllo dei movimenti della piattaforma ovvero di quello che viene chiamato *motion cueing system* e che ha il compito di tradurre le sensazioni che si vogliono trasmettere al pilota in movimenti meccanici della piattaforma di simulazione.

Si introdurrà inanzitutto il sistema per il quale si è sviluppato l'algoritmo di controllo, ovvero la piattaforma di simulazione stessa.

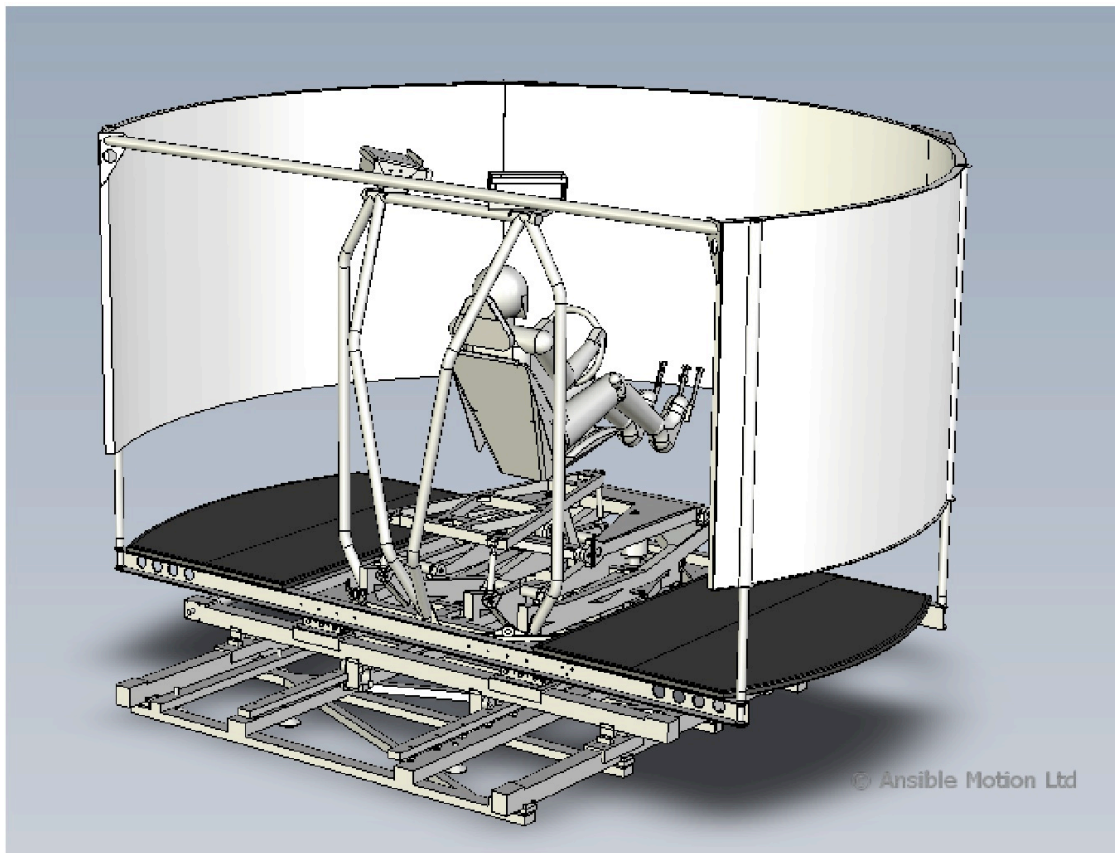
### 1.1 Piattaforma VI-grade

VI-grade è un'azienda fondata nel 2005, che offre e sviluppa software per simulazioni per tutte le compagnie che vogliono aggiungere un collegamento agli studi ingegneristici teorici e la prova nella realtà di quanto si è studiato. VI-DriveSim è una nuova linea di prodotti creata da VI-Grade la quale cerca di dare alle compagnie automobilistiche un mezzo di integrazione tra sviluppo di nuovi prototipi o sistemi di controllo e test su una piattaforma che rispecchia l'ambiente reale. VI-DriveSim si differenzia in due possibili configurazioni: statica e dinamica. Quest'ultima è composta dai seguenti componenti hardware e software:

- piattaforma di moto rivoluzionaria;
- modelli e dati dei veicoli allo stato dell'arte basati sul software già esistente VI-CarRealTime;
- algoritmo di motion cueing;

- provvista di grafica e suoni;
- attuatori che riescono a dare un feedback al controllo pilota.

La piattaforma mobile è basata su una nuova architettura a 6 DOF (Degree of freedom) la quale lavora con delle escursioni relativamente ridotte rispetto ai sistemi esapodali descritti precedentemente. In figura 1.1 è rappresentata la piattaforma mobile di VI-DriveSim, inoltre in tabella 1.1 sono riportati i valori delle performance del simulatore. I movimenti longitudinale  $x$  e  $y$  insieme alla rotazione dello  $yaw$



**Figura 1.1:** Piattaforma VI-Grade

sono disaccoppiati mentre *roll*, *pitch* e movimenti lungo l'asse  $z$  sono accoppiati tra di loro. VI-DriveSim vuole dunque realizzare un nuovo tipo di approccio per lo sviluppo di una piattaforma di guida, la quale oltre allo scopo principale di test per nuovi sistemi elettronici e training di guida, visto le dimensioni ridotte potrà essere utilizzata per studiare la percezione multisensoriale umana, oppure presso scuole di guida per un primo e sicuro approccio alla guida sulle strade o semplicemente per divertimento.

A causa del poco spazio a disposizione per gli spostamenti, un adeguato motion



DOF	Escursione	Velocità	Accelerazione
X	1m	1.3 m/s	3.3 m/s <sup>2</sup>
Y	11 m	1.3 m/s	3.6 m/s <sup>2</sup>
Z	0.31 m	0.9 m/s	4.9 m/s <sup>2</sup>
Roll	40 deg	112 deg /s	600 deg /s <sup>2</sup>
Pitch	24 deg	61 deg /s	600 deg /s <sup>2</sup>
Yaw	40 deg	61 deg /s	240 deg /s <sup>2</sup>

**Tabella 1.1:** Performance Piattaforma VI-Grade

cueing sarà fondamentale per riuscir a riprodurre al meglio le accelerazioni; guardando le escursioni massime elencate nella tabella 1.1, per accelerazioni ad esempio, costanti e a lunga durata, l'utilizzo esclusivo dell'asse  $x$  per riprodurre tali valori porterà la cabina in pochi attimi verso i suoi limiti di funzionamento. Per questo motivo si vuole sviluppare un algoritmo che, guardando in avanti di qualche secondo (conoscendo dunque la planimetria della pista), sarà capace di prevedere che non riuscirà a riprodurre fedelmente l'accelerazione utilizzando solamente un asse ( $x$ ) ma potrà, ingannando la percezione umana, anche andare ad agire sul *pitch* della piattaforma per la riproduzione di segnali a bassa frequenza lasciando che i movimenti di breve durata e improvvisi vengano riprodotti attraverso spostamenti longitudinali.

## 1.2 Model Predictive Control

Il *motion cueing* attualmente operativo sulla piattaforma VI-grade si basa su un approccio di tipo classico che prevede fondamentalmente su tecniche di filtraggio dei segnali di accelerazione da cui vengono tratti gli ingressi agli attuatori (si veda APPENDICE MCS). L'obiettivo che questo e i precedenti lavori di tesi si prefiggono è quello di sostituire tale approccio con una tecnica di controllo più efficiente e performante, il *Model Predictive Control* (da qui in avanti MPC), che si procede ora ad introdurre illustrandone le caratteristiche fondamentali. Per una descrizione più dettagliata si rimanda a (Tesi M.B. e Macje).

Il controllo predittivo è una strategia nata in ambito industriale sul finire degli anni '70, esso si è subito affermato nell'industria petrolchimica e successivamente è stato studiato attentamente in ambito accademico ed ha trovato applicazione in

molteplici settori, grazie alla sua peculiarità di trattare vincoli sul problema da controllare. L'MPC infatti è una tecnica di controllo che fornisce gli ingressi ottimi per un sistema, secondo la logica di minimizzazione di una funzione costo opportuna. Scendendo un pò più nel dettaglio, l'idea che ne sta alla base è quella di calcolare l'azione di controllo (ovvero l'ingresso) risolvendo un problema di controllo ottimo ad anello aperto con orizzonte temporale finito, garantendo il rispetto dei limiti. Tale calcolo viene svolto per ogni iterazione all'interno della finestra temporale fissata. Della sequenza di ingressi ottimi calcolata si applica all'impianto da controllare solo il primo elemento (tecnica *receding horizon*, si veda nel seguito) e si reimposta il problema per il passo successivo, utilizzando il nuovo stato dell'impianto come condizione iniziale.

### 1.2.1 Vantaggi e svantaggi

Il motivo della scelta dell'MPC è sicuramente legato ai numerosi vantaggi che il suo utilizzo comporta, specialmente in ambito industriale per problemi molto complessi, e che possono essere brevemente riassunti come segue.

- **Gestione dei vincoli del problema.**

La necessità di applicare degli ingressi di controllo che rispettino necessariamente dei vincoli imposti dai limiti fisici della piattaforma rende il controllo MPC particolarmente adatto alla risoluzione del problema grazie alla naturale capacità di gestire i vincoli sulle variabili del sistema. Ciò ha importanti ripercussioni anche sull'efficienza del sistema poiché permette, in certi casi, di operare vicino ai vincoli in sicurezza, sfruttando così maggiormente le potenzialità dell'impianto.

- **Versatilità**

La tecnica MPC si adatta a svariati tipi di sistema, da SISO a MIMO anche molto complessi, a problemi a fase non minima o instabili. Strutturalmente è un controllo ad anello aperto ma in certi casi può essere pensato come componente di retroazione, o anche in ottica di feedforward per compensare i disturbi in ingresso al sistema. Anche la scelta della funzione costo è totalmente arbitraria a patto che essa sia di tipo quadratico.

Per contro il controllo predittivo presenta anche alcuni svantaggi:

- **Onere computazionale**

L'MPC risolve ad ogni passo di campionamento un problema di ottimo vincolato. La presenza di molte variabili, vincoli, in aggiunta ad una frequenza di campionamento elevata possono condurre a problemi computazionalmente molto onerosi. Va d'altro canto sottolineato che lo sviluppo della tecnologia e di strumenti di calcolo sempre più performanti stanno rendendo tale inconveniente sempre meno rilevante. Tuttavia, come si vedrà in seguito, per

lo specifico problema del *motion cueing* qui trattato, esso si rivela particolarmente presente e una parte consistente del presente lavoro sarà dedicato proprio allo studio di una tecnica in grado di ridimensionare il problema.

- **Stabilità**

La stabilità, in termini di un problema con vincoli, è di difficile accertamento e di conseguenza difficilmente formalizzabile. Nella pratica finché è possibile il controllore esercita la sua funzione e dunque si comporta in maniera stabile, tuttavia una brusca variazione dei valori in gioco potrebbe portare a comportamenti indesiderati.

- **Dipendenza dal modello**

L'MPC fa uso di un modello interno dell'impianto da controllare, necessario per attuare la predizione. È chiaro che tale modello può non essere accurato o subire delle variazioni e di conseguenza producendo una predizione non coerente con l'evoluzione effettivamente seguita dall'impianto.

### 1.2.2 Elementi costitutivi

Vengono ora brevemente descritti in modo qualitativo gli elementi e i procedimenti fondamentali che concorrono alla costruzione dell'algoritmo MPC.

#### Modello e predizione

Innanzitutto bisogna stabilire che modello dell'impianto adottare. Inizialmente il controllo predittivo si basava su modelli FIR o di risposta al gradino, che tuttavia consentono di descrivere solamente impianti stabili e spesso risultano di ordine elevato. Quest'ultimo si può ridurre ricorrendo a descrizioni tramite funzioni di trasferimento, che dunque possono essere applicate anche ad impianti instabili, tuttavia sono di difficile gestione nel caso di problemi multivariabili. Questi difetti di tali soluzioni hanno portato negli anni ad affermare l'uso di modelli in spazio di stato, che consentono di sfruttare la teoria dei sistemi lineari e di gestire in maniera efficace sistemi multivariabili. Essi permettono inoltre di inserire in modo semplice la modellizzazione di disturbi e rumori e di sfruttare la teoria del filtraggio statistico (ovvero un osservatore di stato) nel caso di sistemi con stato non accessibile. Scelto il modello dell'impianto del problema, si fissano gli orizzonti di predizione  $N_p$  e di controllo  $N_c$  e si calcola l'evoluzione del sistema in tale finestra temporale, scrivendo le future uscite  $y(t+k|t)$ ,  $k=1, \dots, N_p$  in funzione degli ingressi futuri  $u(t+k|t)$ ,  $k=0, \dots, N_c-1$ .

### Funzionale di costo e calcolo della legge di controllo

L'ingresso ottimo del sistema viene calcolato andando a minimizzare un funzionale di costo, o funzione obiettivo, scelto come detto in precedenza sulla base delle necessità contestuali al problema, con l'obiettivo di fondo di seguire una traiettoria di riferimento, il tutto garantendo il rispetto dei vincoli. Un possibile funzionale di costo è il seguente:

$$J = \sum_{j=1}^{N_p} \delta(j) [\hat{y}(t+j|t) - r(t+j)]^2 + \sum_{j=1}^{N_c} \lambda(j) [\Delta u(t+j-1)]^2$$

Esso minimizza l'errore  $\hat{y}(t+j|t) - r(t+j)$  o la spesa in termini di variazione dell'ingresso  $\Delta u(t+j-1)$  a seconda dei valori assunti dai pesi  $\delta(j)$  e  $\lambda(j)$ . Si sottolinea ancora una volta che la scelta del funzionale non è univoca ma arbitraria, l'importante è che sia di tipo quadratico. Data la presenza di vincoli la minimizzazione si traduce in un problema di programmazione quadratica, nel caso eventuale di assenza di vincoli la soluzione si ricava in maniera analitica. Spesso la soluzione del problema di programmazione quadratica non è così immediata, data la complessità dello stesso; è dunque prassi consolidata imporre  $N_c < N_p$  e assumere che il segnale d'ingresso non subisca più variazioni dopo  $N_c$  passi:  $\Delta u(t+j-1) = 0$ ,  $j \in [N_c, N_p]$ . In tal modo la dimensione del problema si riduce con immediata ripercussione sulla sua complessità computazionale.

### Applicazione dell'ingresso e tecnica "receding horizon"

Dato che il modello dell'impianto di cui si fa uso potrebbe non essere accurato o che potrebbe esserci presenza di disturbi non misurabili, della sequenza del segnale di controllo calcolata al paragrafo precedente si applica in realtà solo il primo elemento,  $u(t|t)$ , mentre il resto è scartato. All'istante iterativo seguente si disporrà del nuovo valore delle uscite e dello stato del sistema sulla base del quale si ripeterà tutto il procedimento fin qui descritto, ottenendo una nuova sequenza di ingressi della quale si applicherà solo il primo elemento,  $u(t+1|t+1)$ , che in generale sarà diverso da quello calcolato e scartato all'iterazione precedente,  $u(t+1|t)$ . Tale tecnica prende il nome di *receding horizon*, dal momento che l'orizzonte di predizione è sempre della stessa durata ma viene "spostato in avanti" di un passo ad ogni iterazione. Si noti che il fatto di avere a disposizione la nuova uscita per effettuare l'ottimizzazione, significa supporre che il modello del sistema in analisi sia strettamente proprio, ovvero che l'uscita  $y(k)$  dipenda unicamente dagli ingressi passati e non da quello attuale  $u(k)$ .

## 1.3 Il problema MPC applicato al *motion cueing*

In questa sezione si esporrà nello specifico il problema MPC oggetto di studio, descrivendo brevemente i modelli adottati, la funzione costo, le tecniche adottate

e sperimentate e quant'altro si ritenga indispensabile per una piena comprensione del lavoro originale svolto in questa tesi.

Quanto si andrà a descrivere è basato sul lavoro svolto e i risultati raggiunti da Mauro Baseggio e Mauro Pozzi, ai quali si farà costante riferimento e a cui si rimanda per una più ampia e completa visione degli argomenti trattati.

### 1.3.1 Modellizzazione del sistema

Lo scopo di una piattaforma di simulazione è quello di riprodurre le sensazioni prodotte dalla guida su pista tramite comandi imposti agli attuatori. È dunque di fondamentale importanza conoscere in che modo e in quale misura un dato ingresso di comando produca sul pilota una determinata sensazione, in altre parole è necessario fornire la descrizione di un modello del sistema piattaforma. Esso si può individuare come la giustapposizione di due sottosistemi (si veda la Figura 1.2): il sistema meccanico, che descrive come gli ingressi di controllo vengano trasmessi all'interno dell'abitacolo (in realtà nel caso della piattaforma a livello dello *chassis*) e il sistema percettivo umano, individuato nel sistema vestibolare che traduce le accelerazioni e le velocità applicate in accelerazioni e velocità percepite dal cervello del pilota.

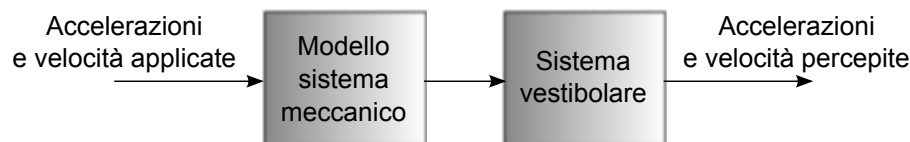


Figura 1.2: Modello serie del sistema piattaforma

#### Modello del sistema vestibolare

Ci si limita qui a fornire una sintetica descrizione dei modelli scelti per la descrizione dei due sottosistemi in questione. Per una trattazione esaustiva degli studi effettuati sul sistema vestibolare si rimanda a [4].

Il sistema vestibolare svolge la funzione principale di controllare il movimento e l'equilibrio, fornendo informazioni relative ad accelerazioni trasversali ed angolari (rotazioni) applicate al corpo umano. Esso è localizzato nell'orecchio interno ed è costituito da due sottosistemi: i canali semicircolari e le macule.

- **Canali semicircolari**

I canali semicircolari (*semicircular channels, SCC*) sono gli organi preposti a percepire le rotazioni applicate al corpo (possono essere considerati accelerometri angolari fortemente smorzati) e, per ogni orecchio, sono in numero pari a tre: orizzontale, anteriore (o superiore) e posteriore, ortogonali l'uno all'altro.

Il modello scelto è frutto del lavoro di Telban e Cardullo [10], basato a sua volta da quanto ottenuto da Chalmers [5]. La funzione di trasferimento che lo descrive è la seguente

$$\frac{\hat{\omega}_i(s)}{\omega_i(s)} = 5.73 \frac{80s^2}{(1 + 80s)(1 + 5.73s)} = \frac{s^2}{s^2 + s \left( \frac{1}{T_a} + \frac{1}{T_L} \right) + \frac{1}{T_a T_L}} \quad (1.1)$$

dove  $\hat{\omega}(s)$  e  $\omega(s)$  indicano rispettivamente le velocità angolari percepite e applicata attorno all' $i$ -esimo asse cartesiano,  $i = x, y, z$ .

- **Macule**

Le macule (*otoliths*) rilevano il moto lineare della testa, inviando al sistema nervoso informazioni riguardo alla forza specifica percepita dal corpo umano. Con forza specifica si intende una combinazione lineare di accelerazione traslazionale e forza gravitazionale, tale fatto verrà illustrato in dettaglio nel seguito introducendo il concetto di *tilt coordination*.

La funzione di trasferimento che descrive le macule è fornita ancora da Telban e Cardullo ed è la seguente

$$\frac{\hat{f}_i(s)}{f_i(s)} = K_{OTO} \frac{1 + s\tau_L}{(1 + s\tau_1)(1 + s\tau_2)} = \frac{K\tau_a}{\tau_l\tau_s} \frac{s + \frac{1}{\tau_a}}{s^2 + s \left( \frac{1}{\tau_s} + \frac{1}{\tau_L} \right) + \frac{1}{\tau_s\tau_L}} \quad (1.2)$$

con  $K_{OTO} = 0.4$ ,  $\tau_1 = 5$ ,  $\tau_2 = 0.016$  e  $\tau_L = 10$  e dove  $\hat{f}(s)$  e  $f(s)$  indicano le accelerazioni lineari, percepite e applicata rispettivamente, lungo l' $i$ -esimo asse cartesiano,  $i = x, y, z$ .

La costruzione dell'algoritmo MPC prevede una realizzazione del modello in spazio di stato. Tramite Matlab si è giunti alla seguente realizzazione per i canali semicircolari

$$A_{s,i} = \begin{bmatrix} -\frac{1}{T_a} - \frac{1}{T_L} & 1 \\ -\frac{1}{T_a T_L} & 0 \end{bmatrix} \quad B_{s,i} = \begin{bmatrix} -\frac{1}{T_a} - \frac{1}{T_L} \\ \frac{1}{T_a T_L} \end{bmatrix} \quad C_{s,i} = [1 \quad 0] \quad D_{s,i} = [1] \quad (1.3)$$

mentre per le macule si ha

$$A_{o,i} = \begin{bmatrix} -\frac{1}{\tau_s} - \frac{1}{\tau_L} & 1 \\ -\frac{1}{\tau_s\tau_L} & 0 \end{bmatrix} \quad B_{o,i} = \begin{bmatrix} \frac{K\tau_a}{\tau_l\tau_s} \\ \frac{K}{\tau_l\tau_s} \end{bmatrix} \quad C_{o,i} = [1 \quad 0] \quad D_{o,i} = [0] \quad (1.4)$$

Come detto precedentente, i modelli 1.3 e 1.4 riportati si riferiscono ad un singolo asse, ma il problema in analisi si sviluppa in tutte e tre le dimensioni. È dunque necessario procedere all'accoppiamento degli assi per poter descrivere il

modello complessivo del sistema vestibolare.

Per i canali semicircolari si ha:

$$A_s = \begin{bmatrix} A_{s,x} & 0_{2 \times 2} & 0_{2 \times 2} \\ 0_{2 \times 2} & A_{s,y} & 0_{2 \times 2} \\ 0_{2 \times 2} & 0_{2 \times 2} & A_{s,z} \end{bmatrix} \quad B_s = \begin{bmatrix} B_{s,x} & 0_{2 \times 1} & 0_{2 \times 1} \\ 0_{2 \times 1} & B_{s,y} & 0_{2 \times 1} \\ 0_{2 \times 1} & 0_{2 \times 1} & B_{s,z} \end{bmatrix} \quad C_s = \begin{bmatrix} C_{s,x} & 0_{1 \times 2} & 0_{1 \times 2} \\ 0_{1 \times 2} & C_{s,y} & 0_{1 \times 2} \\ 0_{1 \times 2} & 0_{1 \times 2} & C_{s,z} \end{bmatrix}$$

e

$$D_s = \begin{bmatrix} D_{s,x} & 0 & 0 \\ 0 & D_{s,y} & 0 \\ 0 & 0 & D_{s,z} \end{bmatrix} \quad x_s = \begin{bmatrix} x_{s,x} \\ x_{s,y} \\ x_{s,z} \end{bmatrix} \quad y_s = \begin{bmatrix} \hat{\omega}_x \\ \hat{\omega}_y \\ \hat{\omega}_z \end{bmatrix} \quad u_s = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \quad (1.5)$$

Seguendo l'approccio proposto da Chalmers [5] è possibile approssimare le velocità angolari  $\dot{\beta}_i$  con le velocità  $\hat{\beta}_i$ :

$$\hat{\beta}_i = \dot{\omega}_i \quad (1.6)$$

dove con  $\beta_i$  si intendono gli angoli di eulero che mappano il sistema di riferimento solidale alla piattaforma rispetto a quello assoluto inerziale. Tale approssimazione risulta adeguata nel problema in questione, dove le variazioni angolari permesse alla piattaforma non superano gli 0.25rad. In questo modo il modello complessivo finale per i canali semicircolari diventa:

$$\begin{cases} x_s(t+1) = A_s x_s(t) + B_s \beta(t) \\ \hat{w} = C_s x_s(t) + D_s \beta(t) \end{cases} \quad (1.7)$$

Per quanto riguarda le macule, con procedimento analogo si ottiene che il modello complessivo è dato da:

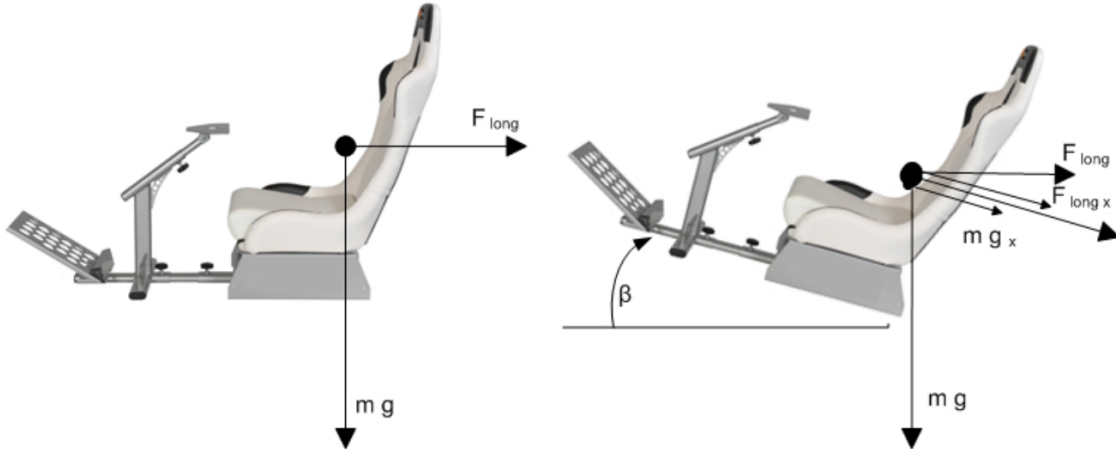
$$A_{of} = \begin{bmatrix} A_{o,x} & 0_{2 \times 2} & 0_{2 \times 2} \\ 0_{2 \times 2} & A_{o,y} & 0_{2 \times 2} \\ 0_{2 \times 2} & 0_{2 \times 2} & A_{o,z} \end{bmatrix} \quad B_{of} = \begin{bmatrix} B_{o,x} & 0_{2 \times 1} & 0_{2 \times 1} \\ 0_{2 \times 1} & B_{o,y} & 0_{2 \times 1} \\ 0_{2 \times 1} & 0_{2 \times 1} & B_{o,z} \end{bmatrix} \quad C_{of} = \begin{bmatrix} C_{o,x} & 0_{1 \times 2} & 0_{1 \times 2} \\ 0_{1 \times 2} & C_{o,y} & 0_{1 \times 2} \\ 0_{1 \times 2} & 0_{1 \times 2} & C_{o,z} \end{bmatrix}$$

e

$$D_{of} = \begin{bmatrix} D_{o,x} & 0 & 0 \\ 0 & D_{o,y} & 0 \\ 0 & 0 & D_{o,z} \end{bmatrix} \quad x_{of} = \begin{bmatrix} x_{o,x} \\ x_{o,y} \\ x_{o,z} \end{bmatrix} \quad y_o = \begin{bmatrix} \hat{f}_x \\ \hat{f}_y \\ \hat{f}_z \end{bmatrix} \quad u_o = \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix} \quad (1.8)$$

Anche in questo caso si rende necessaria una modifica all'ingresso, dal momento che non si dispone delle accelerazioni risultanti  $f_i$  ma solo di quelle longitudinali  $a_i$  applicate, tuttavia prima di procedere alla modifica delle matrici 1.8 è necessario introdurre il concetto della tilt coordination, elemento fondamentale negli algoritmi di motion cuing per simulatori dinamici di veicolo.

**Tilt coordination** Come è facile intuire e come Daniele d’Ambrosio ha sperimentato nel suo lavoro di tesi [2], la semplice traslazione lungo gli assi  $x - y$  della piattaforma del simulatore non è in grado di riprodurre le accelerazioni in gioco in un problema vero. Tuttavia, grazie al fatto che il corpo umano è incapace di distinguere le accelerazioni traslazionali da quelle gravitazionali con le sole macule (infatti per tale scopo è necessaria un’ulteriore informazione sensoriale, in genere fornita dalla vista) si può ingannare il sistema sensoriale umano andando ad inclinare la piattaforma di un angolo opportuno, dando al pilota la percezione di un’accelerazione che andrà a sommarsi a quella fornita dal moto longitudinale. Tale tecnica è detta tilt coordination.



**Figura 1.3:** Tilt coordination per l’asse  $x$  [2].

Come si vede da figura 1.3, inclinando il sedile della piattaforma di un angolo di pitch  $\beta$ , le forze longitudinali e gravitazionali si scompongono e la forza risultante percepita,  $F_x$ , risulta essere la somma delle componenti  $F_{long_x}$  e  $mg_x$ . Più nel dettaglio, assumendo un’inclinazione di un angolo di pitch  $\theta$  e di uno di roll  $\phi$  (l’angolo di yaw è indifferente per la tilt coordination), si ha che il vettore di accelerazione gravitazionale  $g$ , riportato nella terna solidale alla piattaforma del simulatore tramite opportune matrici di rotazione, vale

$$g_s = R_x R_y g_i = R_x(\beta_x) R_y(\beta_y) \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} = \begin{bmatrix} -g \sin \theta \\ g \cos \theta \sin \phi \\ g \cos \theta \cos \phi \end{bmatrix} \quad (1.9)$$

dove si sono indicati coi pedici  $s$  e  $i$  i vettori relativi alla terna del simulatore e inerziale rispettivamente. L’accelerazione complessivamente percepita sarà quindi  $f_s = a_s - g_s$ :

$$f_s = \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix} = \begin{bmatrix} a_x + g \sin \theta \\ a_y - g \cos \theta \sin \phi \\ a_z - g \cos \theta \cos \phi \end{bmatrix} \approx \begin{bmatrix} a_x + g\theta \\ a_y - g\phi \\ a_z - g \end{bmatrix} \quad (1.10)$$



L'ultima approssimazione è stata fatta linearizzando nell'intorno dello 0 (sotto l'ipotesi quindi di angoli piccoli). La tecnica della *tilt coordination*, oltre a permettere di realizzare accelerazioni maggiori di quelle simulabili con la sola traslazione, consente se lo si desidera di spezzare il problema in due per risolverlo in maniera più semplificata: si va ad inseguire le alte frequenze del riferimento traslando la piattaforma e le basse inclinandola. Tale soluzione al problema del *motion cueing* è piuttosto semplice ed elementare e non sfrutta appieno le potenzialità della piattaforma, tuttavia è il principio su cui si basano la maggior parte degli algoritmi di *motion cueing* presenti in commercio (i cosiddetti *washout filters* VEDERE APPENDICE).

Inserendo ora nel modello la *tilt coordination* (utilizzando cioè l'espressione 1.10), è possibile esprimere l'accelerazione complessiva  $f_s$  in funzione delle accelerazioni longitudinali applicate alla piattaforma  $a_s$  e degli angoli  $\beta_S$  di inclinazione della stessa rispetto al riferimento assoluto:

$$f_s = H \begin{bmatrix} a \\ \beta \end{bmatrix} \quad H = \begin{bmatrix} 1 & 0 & 0 & 0 & g & 0 \\ 0 & 1 & 0 & -g & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \quad (1.11)$$

Se gli ingressi desiderati fossero  $[a \ \beta]^T$ , basterebbe a questo punto moltiplicare la matrice  $B_{o_f}$  1.3.1 per la matrice  $H$  appena scritta, tuttavia come si vuole utilizzare come ingresso la variazione degli angoli di Eulero perchè rappresentano anche l'ingresso per il modello dei canalisemicircolari, ovvero si vuole avere  $u = [a \ \dot{\beta}]^T$ . Tale obiettivo si ottiene aggiungendo allo stato  $x_{o_f}$  del sistema tre ulteriori stati, ovvero gli angoli di Eulero  $\beta_x, \beta_y, \beta_z$ , e andando ad apportare una leggera modifica alle matrici del sistema. Si partiziona infatti la nuova matrice  $B_{o_{a,\beta}}$  nel seguente modo:

$$B_{o_{a,\beta}} = B_{o_f} H = [ B_1 \quad B_2 ] \quad (1.12)$$

A questo punto è sufficiente includere  $B_2$  nella nuova matrice  $A_o$  per ottenere il modello delle macule con in ingresso le variazioni degli angoli di Eulero  $\dot{\beta}$ :

$$A_o = \left[ \begin{array}{ccc|c} A_x & 0_{2 \times 2} & 0_{2 \times 2} & B_2 \\ 0_{2 \times 2} & A_y & 0_{2 \times 2} & \\ 0_{2 \times 2} & 0_{2 \times 2} & A_z & \\ \hline & 0_{3 \times 6} & & 0_{3 \times 3} \end{array} \right] \quad x_o = \begin{bmatrix} x_{o_x} \\ x_{o_y} \\ x_{o_z} \\ \beta_x \\ \beta_y \\ \beta_z \end{bmatrix} \quad u_o = \begin{bmatrix} a_x \\ a_y \\ a_z \\ \dot{\beta}_x \\ \dot{\beta}_y \\ \dot{\beta}_z \end{bmatrix} \quad (1.13)$$

e

$$B_o = \begin{bmatrix} B_1 & 0_{6 \times 3} \\ 0_{3 \times 3} & I_{3 \times 3} \end{bmatrix} \quad C_o = [C_{o_f} \quad 0_{3 \times 3}] \quad D_o = [D_{o_f} \quad 0_{3 \times 3}] \quad y_o = \begin{bmatrix} \hat{f}_x \\ \hat{f}_y \\ \hat{f}_z \end{bmatrix} \quad (1.14)$$

Si può ora procedere alla scrittura del modello vestibolare complessivo, comprendente quello dei canali semicircolari 1.3.1,1.5 e quello delle macule con tilt coordination 1.13, 1.14. Basterebbe includere in un unico modello le matrici ricavate, tuttavia si aggiungono anche sei variabili di stato (posizione e velocità del centro di massa della piattaforma lungo i tre assi  $x, y, z$ ) per poter in seguito aggiungere i vincoli necessari all'MPC. Queste aggiunte sono identificate come "stati integrali" vista la loro evoluzione di stato:

$$\begin{bmatrix} \dot{p}_x \\ v_x \\ p_y \\ v_y \\ p_z \\ v_z \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} p_x \\ v_x \\ p_y \\ v_y \\ p_z \\ v_z \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} a_x \\ a_y \\ a_z \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = A_e \begin{bmatrix} p_x \\ v_x \\ p_y \\ v_y \\ p_z \\ v_z \end{bmatrix} + B_e u \quad (1.15)$$

Si aggiunge in uscita anche  $\dot{\beta}$ , per poterla vincolare nel problema MPC, semplicemente aggiungendo una riga nelle matrici di evoluzione dell'uscita. L'evoluzione di stato del sistema vestibolare è dunque così descrivibile:

$$\begin{cases} \dot{x}_{vest} = A_{vest} x_{vest} + B_{vest} u \\ y_{vest} = C_{vest} x_{vest} + D_{vest} u \end{cases} \quad (1.16)$$

con

$$A_{vest} = \begin{bmatrix} A_{scc} & 0_{6 \times 9} & 0_{6 \times 6} \\ 0_{9 \times 6} & A_{oto} & 0_{9 \times 6} \\ 0_{6 \times 6} & 0_{6 \times 9} & A_e \end{bmatrix} \quad B_{vest} = \begin{bmatrix} 0_{6 \times 3} & B_{scc} \\ B_{oto} \\ B_e \end{bmatrix} \quad (1.17)$$

e

$$C_{vest} = \begin{bmatrix} C_{scc} & 0_{3 \times 9} & 0_{3 \times 6} \\ 0_{3 \times 6} & C_{oto} & 0_{3 \times 6} \\ 0_{9 \times 6} & 0_{9 \times 6} & I_{9 \times 9} \\ & 0_{3 \times 24} & \end{bmatrix} \quad D_{vest} = \begin{bmatrix} 0_{3 \times 3} & D_{scc} \\ D_{oto} \\ 0_{9 \times 6} \\ 0_{3 \times 3} & I_{3 \times 3} \end{bmatrix} \quad x_{vest} = \begin{bmatrix} x_{scc} \\ x_{oto} \\ p_x \\ v_x \\ p_y \\ v_y \\ p_z \\ v_z \end{bmatrix} \quad (1.18)$$

Le variabili d'ingresso sono  $u = [a \ \dot{\beta}]^T$ , mentre l'uscita è il vettore seguente:

$$y_{vest} = [y_{scc} \ y_{oto} \ \phi \ \theta \ \psi \ p_x \ v_x \ p_y \ v_y \ p_z \ v_z \ \dot{\phi} \ \dot{\theta} \ \dot{\psi}]^T \quad (1.19)$$

### Modello del sistema meccanico

Non disponendo di un modello adeguato della piattaforma ed avendo a che fare con un problema di dimensioni notevoli si è deciso di modellare il sistema meccanico

come un semplice ritardo (discreto) di attuazione pari ad un passo di campionamento. Le matrici con cui si modella effettivamente il sistema meccanico per ogni singolo grado di libertà sono dunque le seguenti:

$$A_{a,\omega_i} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \quad B_{a,\omega_i} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad C_{a,\omega_i} = [0 \quad 1] \quad D_{a,\omega_i} = 0 \quad (1.20)$$

Non vengono più apportate distinzioni per gli ingressi angolari e dunque questi ultimi saranno in velocità e non più in accelerazione. Come sviluppo futuro non è da escludere una modellizzazione accurata del sistema meccanico vero e la sua introduzione nel problema per affinare l'algoritmo di cueing, ma in questa fase, piuttosto complessa, dell'introduzione del modello vestibolare e della comprensione dei meccanismi dell'MPC si è come detto preferito farne a meno. Il modello complessivo a sei gradi di libertà del sistema meccanico, visto come semplice ritardo, sarà quindi:

$$\Sigma_m = (A_m, B_m, C_m, D_m) \quad (1.21)$$

con:

$$A_m = \begin{bmatrix} A_{a_x} & 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} \\ 0_{2 \times 2} & A_{a_y} & 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} \\ 0_{2 \times 2} & 0_{2 \times 2} & A_{a_z} & 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} \\ 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} & A_{\omega_x} & 0_{2 \times 2} & 0_{2 \times 2} \\ 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} & A_{\omega_y} & 0_{2 \times 2} \\ 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} & A_{\omega_z} \end{bmatrix} \quad B_m = \begin{bmatrix} B_{a_x} & 0_{2 \times 1} & 0_{2 \times 1} & 0_{2 \times 1} & 0_{2 \times 1} & 0_{2 \times 1} \\ 0_{2 \times 1} & B_{a_y} & 0_{2 \times 1} & 0_{2 \times 1} & 0_{2 \times 1} & 0_{2 \times 1} \\ 0_{2 \times 1} & 0_{2 \times 1} & B_{a_z} & 0_{2 \times 1} & 0_{2 \times 1} & 0_{2 \times 1} \\ 0_{2 \times 1} & 0_{2 \times 1} & 0_{2 \times 1} & B_{\omega_x} & 0_{2 \times 1} & 0_{2 \times 1} \\ 0_{2 \times 1} & 0_{2 \times 1} & 0_{2 \times 1} & 0_{2 \times 1} & B_{\omega_y} & 0_{2 \times 1} \\ 0_{2 \times 1} & 0_{2 \times 1} & 0_{2 \times 1} & 0_{2 \times 1} & 0_{2 \times 1} & B_{\omega_z} \end{bmatrix} \quad (1.22)$$

$$C_m = \begin{bmatrix} C_{a_x} & 0_{1 \times 2} & 0_{1 \times 2} & 0_{1 \times 2} & 0_{1 \times 2} & 0_{1 \times 2} \\ 0_{1 \times 2} & C_{a_y} & 0_{1 \times 2} & 0_{1 \times 2} & 0_{1 \times 2} & 0_{1 \times 2} \\ 0_{1 \times 2} & 0_{1 \times 2} & C_{a_z} & 0_{1 \times 2} & 0_{1 \times 2} & 0_{1 \times 2} \\ 0_{1 \times 2} & 0_{1 \times 2} & 0_{1 \times 2} & C_{\omega_x} & 0_{1 \times 2} & 0_{1 \times 2} \\ 0_{1 \times 2} & 0_{1 \times 2} & 0_{1 \times 2} & 0_{1 \times 2} & C_{\omega_y} & 0_{1 \times 2} \\ 0_{1 \times 2} & 0_{1 \times 2} & 0_{1 \times 2} & 0_{1 \times 2} & 0_{1 \times 2} & C_{\omega_z} \end{bmatrix} \quad D_m = 0_{6 \times 6} \quad (1.23)$$

con  $U_m^T = [a_x \ a_y \ a_z \ \omega_x \ \omega_y \ \omega_z]$  che rappresenta gli ingressi applicati e  $Y_m^T = [a_x \ a_y \ a_z \ \omega_x \ \omega_y \ \omega_z]$  che rappresenta le uscite, ritardate di un passo rispetto agli ingressi.

### Sistema serie

Ricavati i modelli in spazio di stato relativi al sistema meccanico (1.22 e 1.23) ed al sistema vestibolare (1.17 e 1.18), discretizzato quest'ultimo si può procedere alla sintesi del sistema dato dalla serie dei due. Le matrici risultanti sono le seguenti (per semplicità non si distinguono nella notazione le matrici del sistema vestibolare continue da quelle discretizzate):

$$A_{serie} = \begin{bmatrix} A_m & 0 \\ B_{vest} & C_m \end{bmatrix} \quad B_{serie} = \begin{bmatrix} B_m \\ B_{vest} & D_m \end{bmatrix} \quad (1.24)$$

$$C_{serie} = [D_{vest} \ C_m \ C_{vest}] \quad D_{serie} = [D_{vest} \ D_m] \quad (1.25)$$

Da cui si ricava il modello su cui eseguire l'MPC:

$$\Sigma_{serie} = (A_{serie}, B_{serie}, C_{serie}, D_{serie}) \quad (1.26)$$

### Modello disaccoppiato

Il modello complessivo a sei gradi di libertà 1.26 ottenuto nel paragrafo precedente è di dimensione notevole e quindi può risultare problematico applicare l'MPC direttamente su di esso, data la mole dei tempi di calcolo richiesti dal controllo predittivo. Tuttavia osservando attentamente la struttura delle matrici relative al sistema meccanico ed al sistema vestibolare (1.22 e 1.23, 1.17 e 1.18) è immediato accorgersi che per quanto riguarda il sistema meccanico i sei gradi di libertà sono completamente disaccoppiati (le matrici sono diagonali) mentre per quanto concerne il sistema vestibolare è possibile suddividerlo in quattro sottosistemi l'uno indipendente dall'altro:

1.  $\Sigma_1$ , relativo a  $\psi$ ;
2.  $\Sigma_2$ , relativo ad  $a_z$ ;
3.  $\Sigma_3$ , relativo alla coppia  $(a_x, \theta)$ ;
4.  $\Sigma_4$ , relativo alla coppia  $(a_y, \phi)$ .

Ciò è possibile grazie al fatto che la linearizzazione introdotta nella *tilt coordination* (1.10) ha ridotto il sistema complessivo ai due accoppiamenti indipendenti accelerazione longitudinale lungo  $x$ , posizione angolare di *pitch*  $(a_x, \theta)$  e accelerazione laterale lungo  $y$ , posizione angolare di *roll*  $(a_y, \phi)$ , oltre ad aver separato la dinamica dell'accelerazione in  $z$  da quella delle rotazioni attorno agli altri due assi.

I quattro sottosistemi individuati possono essere messi ciascuno in serie al corrispondente sistema meccanico producendo quattro differenti modelli complessivi su cui eseguire l'MPC, disaccoppiati l'uno dall'altro, ciascuno dei quali può essere implementato su uno specifico calcolatore per un totale di quattro controllori diversi che operano in parallelo, rendendo più efficiente lo svolgimento del problema e riducendo i tempi di calcolo.

Nell'analisi che segue e nello sviluppo successivo della tesi ci si concentrerà unicamente sul sistema  $\Sigma_3$  che costituisce più degli altri il fulcro dell'algoritmo di *motion cueing*. Il sistema  $\Sigma_4$  infatti potrà essere controllato in maniera del tutto duale, mentre il grado di libertà  $a_z$  (come facilmente intuibile) non è particolarmente sollecitato. Infine l'angolo di *yaw*  $\psi$  (legato quindi a  $\omega_z$ ), importante per l'immersione del pilota nell'ambiente simulato, non presenta particolari difficoltà nell'inseguire un riferimento in velocità che si mantenga entro i limiti di mobilità della piattaforma.

Da qui in avanti il sistema  $\Sigma_3$  sarà indicato con la notazione  $\Sigma_{xy}$ .

L'accoppiamento longitudinale-*pitch* viene modellato in spazio di stato in analogia a quanto visto fin'ora. Il sottosistema meccanico avrà forma

$$\Sigma_{m_{xy}} = (A_{m_{xy}}, B_{m_{xy}}, C_{m_{xy}}, D_{m_{xy}}) \quad (1.27)$$

con

$$A_{m_{xy}} = \begin{bmatrix} A_{a_x} & 0_{2 \times 2} \\ 0_{2 \times 2} & A_{\dot{\omega}_y} \end{bmatrix} \quad B_{m_{xy}} = \begin{bmatrix} B_{a_x} & 0_{2 \times 1} \\ 0_{2 \times 1} & B_{\dot{\omega}_y} \end{bmatrix} \quad C_{m_{xy}} = \begin{bmatrix} C_{a_x} & 0_{1 \times 2} \\ 0_{1 \times 2} & C_{\dot{\omega}_y} \end{bmatrix} \quad D_{a_{xy}} = 0_{2 \times 2}$$

Analogamente si procede per il sistema vestibolare. Per i canali semicircolari, dal momento che ci si riduce ad un solo asse, si ha  $y_{scc_y} = \hat{\omega}_y$ ,  $u_{scc_y} = \omega_y = \dot{\theta}$  e dunque le matrici risultano:

$$A_{scc_y} = A_{s_y} \quad B_{scc_y} = B_{s_y} \quad C_{scc_y} = C_{s_y} \quad D_{scc_y} = D_{s_y} \quad (1.28)$$

Le matrici cui si fa riferimento sono le 1.3.

Per le macule è necessario prestare attenzione all'inserimento dell'azione di *tilt coordination*. La 1.11 per il solo asse  $x$  si riduce alla

$$f_x = H \begin{bmatrix} \beta_y \\ a_x \end{bmatrix} = \begin{bmatrix} g & 1 \end{bmatrix} \begin{bmatrix} \theta \\ a_x \end{bmatrix} \quad (1.29)$$

Procedendo come nel caso del sistema complessivo si ottengono infine le

$$A_{oto_x} = \left[ \begin{array}{c|c} A_{o_x} & B_2 \\ \hline 0_{1 \times 2} & 0 \end{array} \right] \quad B_{oto_x} = \begin{bmatrix} B_1 & 0_{2 \times 1} \\ 0 & 1 \end{bmatrix} \quad x_{oto_x} = \begin{bmatrix} x_x \\ \theta \end{bmatrix} \quad u_{oto_x} = \begin{bmatrix} a_x \\ \dot{\theta} \end{bmatrix} \quad (1.30)$$

e

$$C_{oto_x} = [C_{o_x} \quad 0] \quad D_{oto_x} = [D_{o_x} \quad 0] \quad (1.31)$$

che forniscono  $y_{oto_x} = \hat{f}_x$ .

Si può poi procedere alla costruzione del sistema vestibolare complessivo, mediante introduzione degli stati integrali la cui dinamica complessiva è stata descritta in 1.15 e che risulta di semplificazione immediata. Il modello risulta:

$$\Sigma_{vest_{xy}} = (A_{vest_{xy}}, B_{vest_{xy}}, C_{vest_{xy}}, D_{vest_{xy}}) \quad (1.32)$$

con

$$A_{vest_{xy}} = \begin{bmatrix} A_{scc_y} & 0_{2 \times 3} & 0_{2 \times 2} \\ 0_{3 \times 3} & A_{oto_x} & 0_{3 \times 2} \\ 0_{2 \times 3} & 0_{2 \times 3} & A_{e_{xy}} \end{bmatrix} \quad B_{vest_{xy}} = \begin{bmatrix} 0_{2 \times 1} & B_{scc_y} \\ B_{oto_x} \\ B_{e_{xy}} \end{bmatrix} \quad (1.33)$$

e

$$C_{vest_{xy}} = \begin{bmatrix} C_{scc_y} & 0_{1 \times 3} & 0_{1 \times 2} \\ 0_{1 \times 2} & C_{oto_x} & 0_{1 \times 2} \\ 0_{3 \times 2} & 0_{3 \times 2} & I_{3 \times 3} \\ & 0_{1 \times 7} & \end{bmatrix} \quad D_{vest_{xy}} = \begin{bmatrix} 0 & D_{scc_y} \\ D_{oto_x} \\ 0_{3 \times 2} \\ 0 & 1 \end{bmatrix} \quad x_{vest_{xy}} = \begin{bmatrix} x_{scc_y} \\ x_{oto_x} \\ p_x \\ v_x \end{bmatrix} \quad (1.34)$$

avendo utilizzato le matrici

$$A_{e_{xy}} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \quad B_{e_{xy}} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \quad (1.35)$$

Le variabili d'ingresso sono  $u = [a_x \ \dot{\theta}]^T$ , mentre l'uscita è il vettore seguente, composto da sei elementi:

$$y_{vest_{xy}} = [\hat{\omega}_y \ \hat{f}_x \ \theta \ p_x \ v_x \ \dot{\theta}]^T \quad (1.36)$$

Il sistema serie complessivo relativo all'accoppiamento  $(a_x, \theta)$  su cui si esegue l'MPC in questo lavoro di tesi sarà quindi dato da:

$$\Sigma_{serie_{xy}} = (A_{serie_{xy}}, B_{serie_{xy}}, C_{serie_{xy}}, D_{serie_{xy}}) \quad (1.37)$$

con:

$$A_{serie_{xy}} = \begin{bmatrix} A_{m_{xy}} & 0_4 \\ B_{vest_{xy}} C_{m_{xy}} & A_{vest_{xy}} \end{bmatrix} \quad B_{serie_{xy}} = \begin{bmatrix} B_{m_{xy}} \\ B_{vest_{xy}} D_{m_{xy}} \end{bmatrix} \quad (1.38)$$

e

$$C_{serie_{xy}} = [D_{vest_{xy}} C_{m_{xy}} \ C_{vest_{xy}}] \quad D_{serie_{xy}} = [D_{vest_{xy}} \ D_{m_{xy}}] \quad (1.39)$$

In realtà nel modello 1.37 non va inserito il modello vestibolare 1.32, 1.33, 1.34 ma la sua discretizzazione, dal momento che la serie va fatta tra modelli o entrambi discreti o entrambi continui e quello meccanico è discreto per costruzione. Le 1.38 e 1.39 presentano al loro interno la notazione continua del modello vestibolare unicamente per non appesantire la notazione introducendo ulteriori pedici, ma con queste righe si è sottolineato a dovere il fatto di dover procedere alla discretizzazione del sistema 1.32 prima di procedere alla costruzione della serie 1.37.

### 1.3.2 Formulazione del problema MPC

Si procede ora all'implementazione formale e rigorosa del controllo MPC per un modello in spazio di stato, partendo da quanto proposto da Wang [1], realizzato da Daniele d'Ambrosio [2] ed in seguito sviluppato e ampliato da Mauro Baseggio e Mauro Pozzi [3][4].

Sia dunque dato un sistema MIMO discreto del tipo

$$\begin{cases} x_m(k+1) = A_m x(k) + B_m u(k) \\ y(k) = C_m x(k) \end{cases} \quad (1.40)$$

dove il vettore degli ingressi  $u$  ha dimensione  $n_{in}$ , quello delle uscite  $y$  ha dimensione  $n_{out}$  e  $x_m$  è la variabile di stato assunta di dimensione  $n$ ; si è implicitamente assunto che l'ingresso non influenzi direttamente le variabili di uscita. Come spiegato da [6], è utile al fine dell'implementazione avere in ingresso la variazione di stato  $\Delta u(k)$ , a tal scopo si adotta la seconda tecnica proposta sempre in [6] per riscrivere il modello 1.40 in forma alternativa. Si prende dunque la differenza dello stato tra istanti successivi

$$\begin{aligned} \Delta x_m(k+1) &= x_m(k+1) - x_m(k) \\ &= A_m (x_m(k) - x_m(k-1)) + B_m (u(k) - u(k-1)) \end{aligned} \quad (1.41)$$

e definendo la variazione d'ingresso

$$\Delta u(k) = u(k) - u(k-1)$$

si riscrive la 1.41 come

$$\Delta x_m(k+1) = A_m \Delta x_m(k) + B_m \Delta u(k)$$

A questo punto il modello non ha più in ingresso  $u(k)$  ma  $\Delta u(k)$ ; il prossimo passo è quello di collegare  $\Delta x_m(k)$  all'uscita  $y(k)$ ; si definisce dunque la nuova variabile di stato

$$x(k) = \begin{bmatrix} \Delta x_m(k) \\ y(k) \end{bmatrix}$$

Inoltre considerando

$$\begin{aligned} y(k+1) - y(k) &= C_m (x_m(k+1) - x_m(k)) = C_m \Delta x_m(k+1) \\ &= C_m A_m \Delta x_m(k) + C_m B_m \Delta u(k) \end{aligned}$$

si ottiene il seguente modello aumentato:

$$\begin{aligned} \begin{bmatrix} \Delta x_m(k+1) \\ y(k+1) \end{bmatrix} &= \overbrace{\begin{bmatrix} A_m & 0_m^T \\ C_m A_m & I \end{bmatrix}}^A \begin{bmatrix} \Delta x_m(k) \\ y(k) \end{bmatrix} + \overbrace{\begin{bmatrix} B_m \\ C_m B_m \end{bmatrix}}^B \Delta u(k) \\ y(k) &= \underbrace{\begin{bmatrix} 0_m & I \end{bmatrix}}_C \begin{bmatrix} \Delta x_m(k) \\ y(k) \end{bmatrix} \end{aligned} \quad (1.42)$$

con  $0_m \in \mathbb{R}^{n_{in} \times n}$  e  $I$  matrice identità di dimensione pari a quella dello stato.

A questo punto, detto  $N_p$  l'orizzonte di predizione e  $N_c$  quello di controllo, con  $N_c \leq N_p$ , si va in predizione calcolando le uscite future in funzione della traiettoria di controllo  $\Delta u(k)$ ,  $\Delta u(k+1)$ ,  $\dots$ ,  $\Delta u(k+N_c-1)$ , dove  $k$  rappresenta l'istante attuale di campionamento. In primo luogo si ha che l'evoluzione di stato del sistema 1.42 è data da:

$$\begin{aligned} x(k+1|k) &= A x(k) + B \Delta u(k) \\ x(k+2|k) &= A x(k+1|k) + B \Delta u(k+1) = \\ &= A^2 x(k) + AB \Delta u(k) + B \Delta u(k+1) \\ &\vdots \\ x(k+N_p|k) &= A^{N_p} x(k) + A^{N_p-1} B \Delta u(k) + A^{N_p-2} B \Delta u(k+1) + \dots + \\ &\quad + A^{N_p-N_c} B \Delta u(k+N_c-1) \end{aligned} \quad (1.43)$$

di conseguenza le future variabili d'uscita sono date da:

$$\begin{aligned}
y(k+1|k) &= CAx(k) + CB\Delta u(k) \\
y(k+2|k) &= CAx(k+1|k) + CB\Delta u(k+1) = \\
&= CA^2x(k) + CAB\Delta u(k) + CB\Delta u(k+1) \\
&\vdots \\
y(k+N_p|k) &= CA^{N_p}x(k) + CA^{N_p-1}B\Delta u(k) + CA^{N_p-2}B\Delta u(k+1) + \dots + \\
&\quad + CA^{N_p-N_c}B\Delta u(k+N_c-1)
\end{aligned} \tag{1.44}$$

Si noti che tutte le uscite future sono scritte in funzione dello stato attuale e dei futuri ingressi  $\Delta u(k+i)$ ,  $i = 0, \dots, N_c - 1$ . Esse possono essere scritte in forma vettoriale come

$$Y = Fx(k) + \Phi\Delta U \tag{1.45}$$

dove

$$F = \begin{bmatrix} CA \\ CA^2 \\ CA^3 \\ \vdots \\ CA^{N_p} \end{bmatrix}, \Phi = \begin{bmatrix} CB & 0 & 0 & \dots & 0 \\ CAB & CB & 0 & \dots & 0 \\ CA^2B & CAB & CB & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ CA^{N_p-1}B & CA^{N_p-2}B & CA^{N_p-3}B & \dots & CA^{N_p-N_c}B \end{bmatrix} \tag{1.46}$$

avendo definito i vettori

$$Y = \begin{bmatrix} y(k+1|k) \\ y(k+2|k) \\ \vdots \\ y(k+N_p|k) \end{bmatrix} \in \mathbb{R}^{(N_p \cdot n_{out}) \times 1} \quad \Delta U = \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \vdots \\ \Delta u(k+N_c-1) \end{bmatrix} \in \mathbb{R}^{(N_c \cdot n_{in}) \times 1} \tag{1.47}$$

Definito il modello in spazio di stato e dato un riferimento  $r(k)$ , l'obiettivo del controllo predittivo è portare il sistema, ovvero l'uscita, il più possibile vicino al riferimento assegnato, andando a minimizzare un funzionale di costo dipendente dalla sequenza di ingressi incogniti  $\Delta U$ . Il funzionale adottato è il seguente:

$$J(\Delta U) = (R_s - Y)^T Q (R_s - Y) + \Delta U^T R \Delta U + U^T S U \tag{1.48}$$

Il primo termine serve a minimizzare l'errore tra le uscite predette ed il riferimento, mentre i rimanenti pesano gli ingressi e le loro variazioni. Le matrici  $Q$ ,  $R$  e  $S$  sono rispettivamente le matrici contenenti i pesi sull'errore, sulla variazione dell'ingresso e sull'ingresso complessivo, esse hanno dimensioni opportune dipendenti dal numero di ingressi ed uscite e dai passi di predizione e controllo, possono essere "statiche", ovvero composte da pesi sempre uguali, oppure "dinamiche", nel senso che i pesi variano con gli istanti di predizione. La matrice  $R_s$  è un vettore di dimensioni pari a quelle di  $Y$  e contiene il riferimento; se non si dispone di informazioni future su



di esso essa contiene solamente il valore attuale  $r(k)$  supposto costante per tutta la durata della predizione, se invece si fa utilizzo di una strategia look-ahead allora essa riporta l'evoluzione del riferimento  $r(k+1), r(k+2), \dots, r(k+N_p)$ .

Per risolvere il problema innanzitutto è necessario riscrivere il funzionale di costo 1.48 in modo che dipenda unicamente da  $\Delta U$ . Il vettore degli ingressi  $U$  può essere riscritto nella seguente forma:

$$\begin{bmatrix} u(k) \\ u(k+1) \\ u(k+2) \\ \vdots \\ u(k+N_c-1) \end{bmatrix} = \begin{bmatrix} I & 0 & 0 & \dots & 0 \\ I & I & 0 & \dots & 0 \\ I & I & I & \dots & 0 \\ \vdots & & & \ddots & \vdots \\ I & I & I & \dots & I \end{bmatrix} \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \Delta u(k+2) \\ \vdots \\ \Delta u(k+N_c-1) \end{bmatrix} + \begin{bmatrix} u(k-1) \\ u(k-1) \\ u(k-1) \\ \vdots \\ u(k-1) \end{bmatrix} \quad (1.49)$$

o in forma vettoriale come

$$U = T\Delta U + U_i \quad (1.50)$$

Sostituendo la 1.50 e la 1.45 nella 1.48, si riscrive il funzionale nella maniera desiderata:

$$J = (R_s - Fx - \Phi\Delta U)^T Q (R_s - Fx - \Phi\Delta U) + \Delta U^T R \Delta U + (T\Delta U + U_i)^T S (T\Delta U + U_i) \quad (1.51)$$

Dopo un pò di passaggi algebrici, la 1.51 si riduce nella seguente forma, nella quale sono stati trascurati i termini non dipendenti da  $\Delta U$ , dato che i funzionali possono essere definiti a meno di una costante:

$$J = \Delta U^T (\Phi^T Q \Phi + R + T^T S T) \Delta U + 2\Delta U^T (\Phi^T Q (R_s - Fx(k)) + T^T S U_i) \quad (1.52)$$

Il funzionale è ora espresso nella forma classica per un problema di programmazione quadratica (QP):

$$J = \frac{1}{2} \Delta U^T H \Delta U + \Delta U^T F \quad (1.53)$$

con

$$H = 2 (\Phi^T Q \Phi + R + T^T S T) \quad F = 2 (\Phi^T Q (R_s - Fx(k)) + T^T S U_i) \quad (1.54)$$

### 1.3.3 Caso non vincolato

Si risolve ora il problema in questione nel caso, estremamente semplice, di assenza di vincoli. Sotto tale ipotesi non è necessario procedere alla risoluzione di un problema QP dal momento che la soluzione si ricava in maniera analitica, infatti il valore che minimizza il funzionale 2.3 si ottiene derivando il funzionale stesso rispetto a  $\Delta U$  e ponendo la derivata pari a zero:

$$\frac{\partial J}{\partial \Delta U} = 0 \quad \Rightarrow \quad \Delta U_{ott} = H^{-1} F \quad (1.55)$$

Si è fatta l'assunzione implicita che l'inversa della matrice  $H$  esista. La soluzione trovata in 1.55 contiene tutta la legge ottima di controllo, ma in base al principio del *receding horizon* ad ogni singolo passo di campionamento si calcola un set d'ingresso  $\Delta U_{ott}$  per i successivi  $N_c$  passi, del quale si utilizza soltanto il primo elemento:

$$\Delta u_{ott}(k) = [I_{n_{in}} \ 0_{n_{in}} \ \dots \ 0_{n_{in}}] \Delta U_{ott}$$

In seguito il minimo viene ricalcolato considerando i cambiamenti avuti nel sistema al passo successivo, tutto ciò deve avvenire in tempo reale. Per interpretare il risultato 1.55 in termini di controllo a retroazione, aspetto che ai fini della presente tesi non è molto rilevante, si veda [1].

### 1.3.4 Vincoli in ingresso ed uscita

La situazione più interessante del controllo predittivo è ovviamente quella, più complessa, di presenza di limiti su ingressi ed uscite. In tal caso al problema QP 2.3 si aggiunge una disuguaglianza del tipo

$$A \Delta U \leq b \quad (1.56)$$

con  $A$  e  $b$  matrici opportune contenenti tutte le informazioni sui vincoli. La 1.56 è giustamente scritta rispetto alla variabile  $\Delta U$ , su cui si opera la minimizzazione del funzionale 2.3. È dunque necessario procedere alla traduzione di tutti i vincoli sul sistema per riferirli alla variazione d'ingresso, prima di dare in pasto il problema QP ad un ottimizzatore.

#### Vincoli sulla variazione di ingresso

È questo il caso più semplice che si presenta, in quanto non necessita di “conversione” dei vincoli visto che riguardano già  $\Delta U$ . Si suppone, qui e nei casi seguenti, che i vincoli agiscano in termini di doppia disuguaglianza del tipo  $\Delta u_{min} \leq \Delta u(\cdot) \leq \Delta u_{max}$  (con  $\Delta u_{min}$  e  $\Delta u_{max}$  vettori di dimensione pari a quella dell'ingresso); la strategia che si adotta è quella di spezzare quest'ultima in due disuguaglianze nello stesso verso, del tipo 1.56, in modo tale da ricondursi poi a quest'ultima scrittura. In questo caso si pone dunque

$$\begin{bmatrix} I & 0 & \dots & 0 \\ 0 & I & \dots & 0 \\ \vdots & & & \vdots \\ 0 & 0 & \dots & I \\ -I & 0 & \dots & 0 \\ 0 & -I & \dots & 0 \\ \vdots & & & \vdots \\ 0 & 0 & \dots & -I \end{bmatrix} \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \vdots \\ \Delta u(k+N_c-1) \end{bmatrix} \leq \begin{bmatrix} \Delta u^{max} \\ \Delta u^{max} \\ \vdots \\ \Delta u^{max} \\ -\Delta u^{min} \\ -\Delta u^{min} \\ \vdots \\ -\Delta u^{min} \end{bmatrix} \quad (1.57)$$

La matrice identità che compare ha dimensioni opportune; la disuguaglianza appena scritta viene indicata in forma vettoriale come  $M_1 \Delta U \leq N_1$ .

### Vincoli sull'ingresso

Per quanto riguarda i vincoli in ingresso, anch'essi nella forma di doppia disuguaglianza, ovvero  $L_1 \leq u(\cdot) \leq L_2$ , si ricorre alla scrittura 1.49, già usata in precedenza, per tradurre il vincolo sulla variazione d'ingresso. Procedendo come nel caso precedente si ottiene

$$\begin{bmatrix} I & 0 & \dots & 0 \\ I & I & \dots & 0 \\ \vdots & & & \vdots \\ I & I & \dots & I \\ -I & 0 & \dots & 0 \\ -I & -I & \dots & 0 \\ \vdots & & & \vdots \\ -I & -I & \dots & -I \end{bmatrix} \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \vdots \\ \Delta u(k+N_c-1) \end{bmatrix} \leq \begin{bmatrix} L_2 - u(k-1) \\ L_2 - u(k-1) \\ \vdots \\ L_2 - u(k-1) \\ -L_1 + u(k-1) \\ -L_1 + u(k-1) \\ \vdots \\ -L_1 + u(k-1) \end{bmatrix} \quad (1.58)$$

In analogia col caso precedente, si battezzano le due matrici qui introdotte come  $M_2$  e  $N_2$ . Si noti che  $N_2$ , a differenza delle altre fin qui introdotte, dipende dalla condizione iniziale  $u(k-1)$ , con  $k$  istante attuale di calcolo. Dunque tale matrice non può essere definita *off-line* come le altre sin qui introdotte ma deve essere ridefinita ad ogni passo, dal momento che l'ingresso evolve diventando ad ogni iterazione la condizione iniziale dell'istante successivo.

### Vincoli sull'uscita

In questo caso, volendo sfruttare la relazione 1.45, si adotta una notazione leggermente diversa rispetto ai due casi precedenti. Supponiamo infatti che il vettore delle uscite abbia dei limiti inferiori e superiori  $Y_{min}$  e  $Y_{max}$  rispettivamente, ovvero che valga la relazione  $Y_{min} \leq Y \leq Y_{max}$ , con i vettori dei limiti non più di dimensione pari a quella dell'uscita ma a quella del prodotto di quest'ultima per l'orizzonte di predizione (come indicato nella 1.47). A questo punto si può sfruttare la 1.45 e riscrivere in modo immediato i limiti in funzione della variazione d'ingresso:

$$\begin{bmatrix} \Phi \\ -\Phi \end{bmatrix} \Delta U \leq \begin{bmatrix} Y_{max} - Fx(k) \\ -Y_{min} + Fx(k) \end{bmatrix} \quad (1.59)$$

Le matrici introdotte vengono identificate come  $M_3$  e  $N_3$ , per  $N_3$  valgono i discorsi fatti nel caso precedente in merito ad  $N_2$ , con la differenza che ora è lo stato  $x(k)$  che varia ad ogni iterazione.

### Programmazione Quadratica

Una volta tradotti i vincoli come richiesto, si può passare alla risoluzione del problema tramite la programmazione quadratica. Per chiarezza si riportano qui le

formule già scritte in precedenza: si vuole trovare l'ingresso ottimo che minimizzi il funzionale

$$J = \frac{1}{2} \Delta U^T H \Delta U + \Delta U^T F \quad (1.60)$$

soggetto al vincolo di disuguaglianza

$$A \Delta U \leq b \quad \text{con} \quad A = \begin{bmatrix} M_1 \\ M_2 \\ M_3 \end{bmatrix} \quad b = \begin{bmatrix} N_1 \\ N_2 \\ N_3 \end{bmatrix} \quad (1.61)$$

nel quale sono state costruite le matrici  $A$  e  $b$  imponendo i vincoli del problema in questione. A questo punto il problema di programmazione quadratica viene risolto da un ottimizzatore e dunque in questo lavoro di tesi non si entra nei dettagli specifici; qualora il lettore desideri farlo può rifarsi a [2] o a [6].

Nell'implementazione eseguita durante il lavoro di tesi, ci si è avvalsi dell'ottimizzatore qpOASES, software *open-source* sviluppato in  $C++$  da OPTEC [29], dal momento che i tradizionali ottimizzatori di MATLAB risultano più lenti ed il problema del calcolo in tempo reale costituisce un punto fondamentale, data la necessità di implementare il controllore sul simulatore VI-grade.

## 1.4 Analisi delle condizioni di simulazione

Nel seguito del presente lavoro di tesi sono riportati i risultati di diverse simulazioni in ambiente MATLAB, volte a verificare i comportamenti del controllore MPC in diversi ambiti di utilizzo e in diverse configurazioni. Si rende necessario dunque specificare cosa si intende per simulazione, su quali dati esse sono basate e sotto quali condizioni sono state portate a compimento.

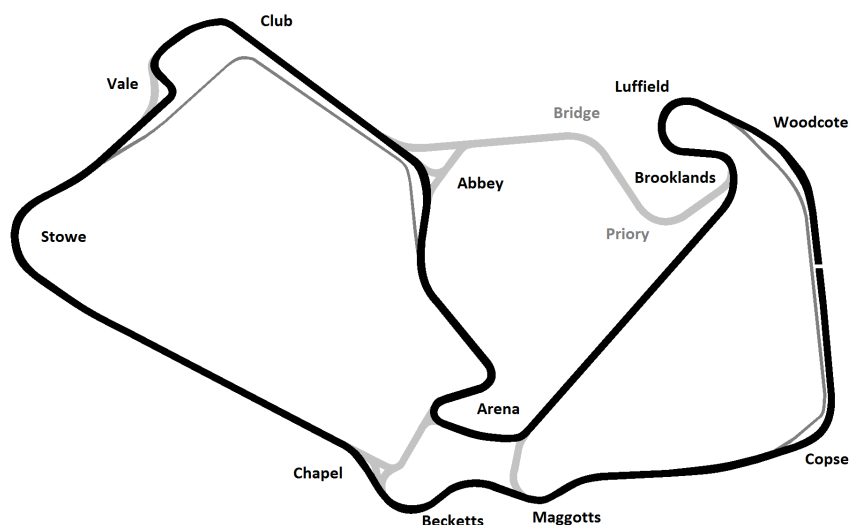
### 1.4.1 Scelta del riferimento

Innanzitutto è bene chiarire cosa si è usato come segnale di riferimento.

Grazie alla collaborazione con VI-grade è stato possibile disporre di dati telemetrici completi relativi ad alcune sessioni di guida, da parte di un pilota mediamente esperto, sui circuiti di Silverstone (GB, versione Grand Prix modificata nel 2010, figura (1.4), [33]) e Barcellona (E).

Si è deciso di utilizzare i dati provenienti dal circuito inglese, per continuità con le tesi precedenti di Daniele d'Ambrosio, Mauro Baseggio e Mauro Pozzi.

I dati di telemetria non possono costituire direttamente il segnale di riferimento per due motivi fondamentali. In primo luogo le uscite che si vogliono inseguire, cioè che si vogliono riprodurre il più fedelmente possibile, sono costituite da accelerazioni (e velocità angolari) *percepite*, mentre i dati da telemetria provengono da misurazioni effettuate sullo *chassis* di un veicolo reale, e forniscono pertanto grandezze *applicate*. In secondo luogo le accelerazioni sviluppate da un'auto da corsa in



**Figura 1.4:** Planimetria del circuito di Silverstone [34].

pista sono, per ampiezza e gamma di frequenze presenti nel segnale, fisicamente ingestibili da parte della piattaforma di simulazione che, a causa delle sue dimensioni ristrette, è sottoposta a dei limiti di escursione molto stringenti. Per questo motivo le accelerazioni longitudinali e le velocità angolari ricavate dai dati di telemetria verranno, in ogni ambito di utilizzo filtrati da un filtro passa-basso (si può fare riferimento allo schema Figura 3.1 che verrà introdotto nel capitolo 3), scalati secondo una qualche procedura di scalatura automatica (per le simulazioni realizzate per la presente tesi si è utilizzata una scalatura di tipo lineare) e infine “tradotte” in termini di grandezze percepite tramite il passaggio per il modello vestibolare. Solo a questo punto esse vengono assunte come riferimento per il controllore MPC.

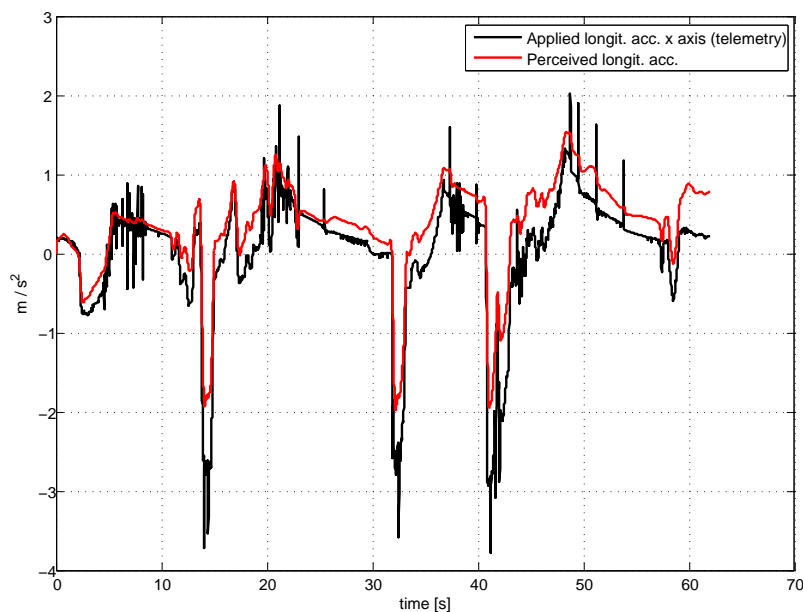
La Figura 1.5 mette a confronto il segnale di accelerazione “grezzo” e la sua versione adattata a riferimento per il *motion cueing* in una finestra temporale di 62 s.

### 1.4.2 Strategie di predizione

Il punto saliente del controllo MPC, e che lo rende particolarmente efficiente, è la possibilità di fare predizione dell’evoluzione futura del sistema da controllare. Nell’ambito del *motion cueing* le modalità con cui viene utilizzata la predizione individuano due strategie fondamentali:

- ***Look-ahead***

Nella strategia *look-ahead* si suppone di conoscere perfettamente a priori l’andamento dell’accelerazione lungo l’intera finestra temporale di simulazione, ovvero di disporre del futuro del giro di pista che il pilota sta compiendo. Tale strategia sfrutta appieno le potenzialità dell’MPC ma si pone in una



**Figura 1.5:** Confronto tra accelerazione da telemetria e segnale di riferimento.

situazione ideale, e come tale poco realistica, giacchè si assume che il pilota replichi esattamente le medesime condizioni di guida in ogni giro di pista per il determinato circuito, il che è ovviamente impossibile, anche se nel caso di piloti esperti e con una buona conoscenza del circuito stesso le differenze tra le diverse tornate possono essere marginali.

- **Riferimento costante**

Al'estremo opposto del *look-ahead* si colloca la strategia che assume che il riferimento sia costante lungo tutta la finestra di predizione. È questo di fatto il *worst-case*, fondamentale da analizzare per capire quanto l'MPC sia in grado di fronteggiare la situazione più svantaggiosa, ovvero quella della completa assenza di informazioni sul tracciato e sulle modalità di guida del pilota. L'assunzione di riferimento costante nella finestra di predizione ben si adatta alle situazioni di guida su strada e con piloti poco esperti. D'altro canto essa non sfrutta a fondo le potenzialità dell'MPC risultando in un controllo meno efficiente.

Vi è anche una terza possibile strategia, che si pone concettualmente a metà strada fra le prime due, quella basata sul concetto di *matching*. In pratica tale strategia assume che il tracciato percorso sia noto ed utilizza come riferimento per la predizione il segnale di accelerazione ricavato dalle telemetrie precedenti, opportunamente adattato attraverso semplici operazioni di scalatura e traslazione. Per una

trattazione più approfondita del *matching* si rimanda comunque alla tesi di Mauro Baseggio [3].

Nel prosieguo della presente tesi si andranno ad affrontare due problematiche legate all'MPC che per loro natura richiederanno l'adozione di due differenti strategie di predizione. Nel capitolo 2 l'esigenza di adottare finestre di predizione relativamente ampie al fine di sfruttare al massimo la capacità predittiva dell'MPC porterà all'utilizzo di una strategia *look-ahead*; nel capitolo 3 invece sarà richiesto di porsi in condizioni di *worst case*, e di utilizzare pertanto un riferimento di predizione costante.





# Capitolo 2

## Applicazione delle funzioni di Laguerre a un controllore MPC

Nel controllo MPC, come si è visto nel capitolo precedente, si utilizza la predizione per modellare il futuro ingresso di controllo. Nel caso di segnali con frequenza di campionamento molto elevata, dinamiche di processo complesse o richieste di elevate prestazioni nel controllo, un'approssimazione soddisfacente dell'andamento futuro del sistema possono richiedere un orizzonte di predizione, e di conseguenza il calcolo di una sequenza di ingressi di controllo, molto elevati con conseguente aumento dell'onere computazionale.

Tenendo conto di questo fatto, sulla base del lavoro di Wang [1], nel presente lavoro di tesi si è utilizzato un approccio fondato sulla parametrizzazione del segnale di controllo tramite funzioni polinomiali ortonormali discrete. Tale parametrizzazione consente di ottenere una descrizione del segnale di controllo desiderato molto compatta e di conseguenza di ridurre i parametri da calcolare necessari per modellare la traiettoria di controllo.

### 2.1 Funzioni di Laguerre

Le funzioni ortogonali in questione sono le funzioni di Laguerre discrete, ottenute dalla discretizzazione delle funzioni di Laguerre continue (si veda Wang [1]). La descrizione della generica funzione di Laguerre di ordine  $N$  è data tramite la sua  $z$ -trasformata:

$$\Gamma_N(z) = \frac{\sqrt{1-a^2}}{1-az^{-1}} \left( \frac{z^{-1}-a}{1-az^{-1}} \right)^{N-1} \quad (2.1)$$

dove  $a$  rappresenta il polo della funzione di Laguerre discreta, necessariamente compreso tra 0 e 1 per la stabilità della rete stessa. Tale parametro può essere scelto liberamente dall'utente ed è anche chiamato *scaling factor*; la sua scelta permette di determinare la rapidità con la quale la famiglia di funzioni decresce a zero.

Come già anticipato le reti di Laguerre godono della proprietà di ortonormalità, che nel dominio della frequenza si esprime tramite le relazioni

$$\frac{1}{2\pi} \int_{-\pi}^{\pi} \Gamma_m(e^{j\omega}) \Gamma_m(e^{j\omega})^* d\omega = 1 \quad (2.2)$$

$$\frac{1}{2\pi} \int_{-\pi}^{\pi} \Gamma_m(e^{j\omega}) \Gamma_n(e^{j\omega})^* d\omega = 0 \quad (2.3)$$

dove  $m \neq n$  e  $m = 1, 2, \dots$ . Si evidenzia che nel progetto del controllore MPC verranno utilizzate esplicitamente le funzioni di Laguerre nel dominio del tempo, ottenibili mediante  $z$ -trasformata inversa dalle corrispondenti funzioni nel dominio della frequenza. Tuttavia applicando la  $z$ -trasformata inversa alla rete di Laguerre non è possibile ottenere delle espressioni compatte per le funzioni di Laguerre nel dominio temporale. Risulta molto più conveniente esprimere le funzioni cercate basandosi su una rappresentazione in spazio di stato della rete.

Si noti infatti che vale la relazione

$$\Gamma_k(z) = \Gamma_{k-1}(z) \frac{z^{-1} - a}{1 - az^{-1}} \quad (2.4)$$

con  $\Gamma_1(z) = \frac{\sqrt{1-a^2}}{1-az^{-1}}$ .

Si indichino ora con  $l_1(k)$  la  $z$ -trasformata inversa di  $\Gamma_1(z, a)$ ,  $l_2(k)$  la  $z$ -trasformata inversa di  $\Gamma_2(z, a)$  e similmente fino a  $l_N(k)$  anti  $z$ -trasformata di  $\Gamma_N(z, a)$ . Tale insieme di funzioni di Laguerre possono essere raggruppate nel vettore

$$L(k) = [l_1(k) \ l_2(k) \ \dots \ l_N(k)]^T \quad (2.5)$$

Sfruttando la relazione (2.4) è possibile giungere alla conclusione che l'insieme delle funzioni di Laguerre (2.5) soddisfa l'equazione alle differenze

$$L(k+1) = A_l L(k) \quad (2.6)$$

dove la matrice  $A_l$  ha dimensioni  $N \times N$  ed è funzione dei parametri  $a$  e  $\beta = 1 - a^2$ , e la condizione iniziale è data da

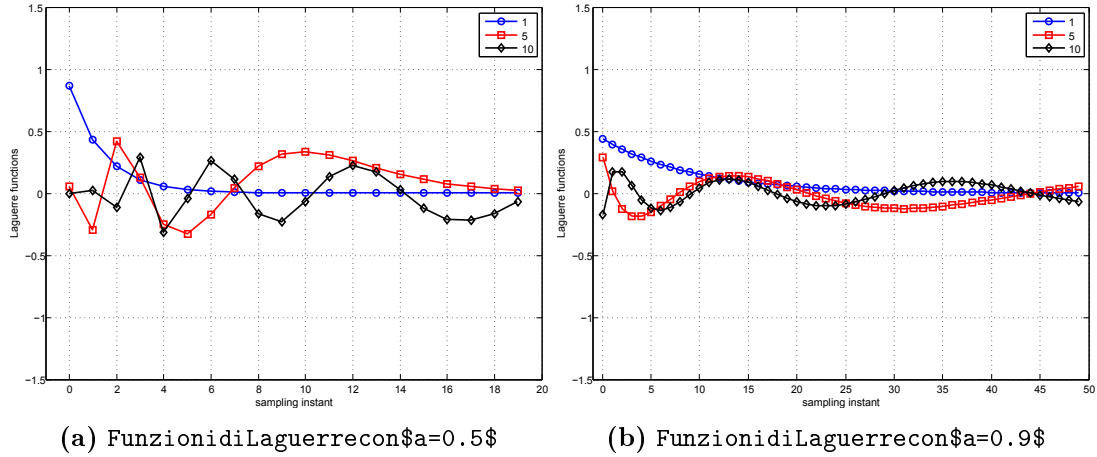
$$L(0)^T = \sqrt{\beta} [1 \ -a \ a^2 \ -a^3 \ \dots \ (-1)^{N-1} a^{N-1}] \quad (2.7)$$

L'ortonormalità espressa in (2.2) e (2.3) si riflette nel dominio del tempo nelle

$$\sum_{k=0}^{\infty} l_i(k) l_j(k) = 0 \quad i \neq j \quad (2.8)$$

$$\sum_{k=0}^{\infty} l_i(k) l_j(k) = 1 \quad i = j \quad (2.9)$$

La Figura 2.1 mostra l'andamento delle funzioni  $l_1$ ,  $l_5$  ed  $l_{10}$  per due diversi valori del parametro  $a$ . Si noti come all'aumentare dello *scaling factor* le funzioni convergono a zero più lentamente.



**Figura 2.1:** Funzioni di Laguerre per due valori di  $a$ . Nota:  $1 = l_1$ ,  $5 = l_5$ ,  $10 = l_{10}$ .

## 2.2 Uso delle funzioni di Laguerre nella descrizione del sistema

L'obiettivo dell'uso delle funzioni di Laguerre è quello di esprimere il segnale di controllo futuro utilizzando un numero ristretto di parametri. Si supponga che la risposta impulsiva di un sistema stabile sia  $H(k)$ , allora con un dato numero  $N$  di termini  $H(k)$  è scritta come

$$H(k) = c_1 l_1(k) + c_2 l_2(k) + \dots + c_N l_N(k) \quad (2.10)$$

dove  $c_1, c_2, \dots, c_N$  sono i coefficienti da determinare in base ai dati. Data l'ortonormalità delle funzioni di Laguerre i coefficienti sopra descritti sono definiti dalla relazione

$$c_i = \sum_{k=0}^{\infty} H(k) l_i(k) \quad (2.11)$$

dove  $i = 1, 2, \dots, N$ . In realtà, grazie alla proprietà di ortonormalità i coefficienti minimizzano anche la somma degli errori quadratici

$$J_{SE} = \sum_{k=1}^{\infty} \left( H(k) - \sum_{i=1}^N c_i l_i(k) \right)^2 \quad (2.12)$$

con un'approssimazione che migliora all'aumentare del numero  $N$  di funzioni di Laguerre utilizzate.

Nel presente lavoro, si guarda alla traiettoria  $\Delta u(k_i), \Delta u(k_i+1), \Delta u(k_i+2), \dots, \Delta u(k_i+k), \dots$ , a partire dall'istante  $k_i$ , come la risposta impulsiva di un sistema stabile. pertanto, nello spirito di quanto visto fin'ora, si è espressa tale dinamica tramite un insieme di funzioni di Laguerre  $l_1(k), l_2(k), \dots, l_N(k)$ , combinate linearmente tramite opportuni coefficienti che saranno ricavati tramite l'ottimizzazione dell'ingresso. Più precisamente, ad un futuro istante di campionamento arbitrario  $k$  si avrà

$$\Delta u(k_i + k) = \sum_{j=1}^N c_j(k_i) l_j(k) \quad (2.13)$$

dove  $k_i$  rappresenta l'istante iniziale dell'orizzonte in cui si fa predizione e  $k$  l'istante di campionamento futuro;  $N$  è il numero di termini utilizzati nella combinazione lineare e i  $c_j, j = 1, 2, \dots, N$  sono i coefficienti, funzioni dell'istante iniziale  $k_i$ . Si noti che in questo modo l'orizzonte di controllo  $N_c$  scompare dai parametri necessari all'ottimizzazione, lasciando il posto al parametro  $N$ , numero di termini (e quindi di funzioni) di Laguerre, cui viene lasciato il compito di descrivere la complessità della traiettoria di ingresso insieme al parametro  $a$ . Si potrà ad esempio, ricordando il significato del parametro, scegliere un  $a$  dal valore più elevato per ottenere un orizzonte di controllo lungo con un numero  $N$  di parametri inferiore e dunque ottenere un risparmio computazionale.

L'equazione (2.13) può essere riscritta in forma vettoriale:

$$\Delta u(k_i + k) = L(k)^T \eta \quad (2.14)$$

dove il vettore parametrico  $\eta$  racchiude gli  $N$  coefficienti di Laguerre:

$$\eta = [c_1, c_2, \dots, c_N]^T, \quad (2.15)$$

e  $L(k)^T$  è il vettore trasposto delle funzioni di Laguerre come già precedentemente definito nella (2.5). Usando in tal modo le funzioni di Laguerre, dato un modello in spazio di stato  $\Sigma = (A, B, C)$  con  $\Delta u(\cdot)$  come segnale d'ingresso con l'informazione iniziale sullo stato  $x(k_i)$ , la predizione dello stato futuro  $x(k_i + m|k_i)$  all'istante di campionamento  $m$  diventa

$$x(k_i + m|k_i) = A^m x(k_i) + \sum_{i=0}^{m-1} A^{m-i-1} B L(i)^T \eta \quad (2.16)$$

dove la funzione  $\Delta u(k_i + i)$  è stata rimpiazzata da  $L(i)^T \eta$  secondo la (2.14). Allo stesso modo è possibile scrivere l'uscita del sistema ad un futuro istante  $m$  come

$$y(k_i + m|k_i) = C A^m x(k_i) + \sum_{i=0}^{m-1} C A^{m-i-1} B L(i)^T \eta \quad (2.17)$$

Con questa formulazione, sia la predizione della variabile di stato che della variabile di uscita sono espresse in funzione del vettore dei coefficienti  $\eta$  della rete di Laguerre e non più in funzione del vettore dei valori di ingresso  $\Delta U$  come nell'approccio classico. In questo modo sarà  $\eta$  (di dimensione  $N$ ) ad essere ottimizzato in fase di computazione e non  $\Delta U$  (di dimensione  $N_c$ , solitamente molto elevata in caso di campionamento fitto per ottenere buoni risultati).

## 2.3 Applicazione al modello MPC

Si richiama qui quanto esposto in sezione 1.3.2, a partire dal quale si andrà ad intervenire su alcuni aspetti di modo da integrare le funzioni di Laguerre nell'algoritmo di controllo.

Si parta dunque da un sistema MIMO del tipo descritto dalla (1.40) e si proceda come spiegato alla costruzione del modello aumentato (1.42), definendo di conseguenza le matrici  $F$  e  $\phi$  come da (1.46) e modificando il funzionale di costo (1.48) fino ad ottenere la forma espressa dalla (2.3), che si riporta qui per chiarezza espositiva:

$$J = \frac{1}{2} \Delta U^T H \Delta U + \Delta U^T F$$

con

$$H = 2 (\Phi^T Q \Phi + R + T^T S T) \quad F = 2 (\Phi^T Q (R_s - Fx(k)) + T^T S U_i)$$

A questo punto, ricordando la (2.14) si vanno a modificare le matrici caratteristiche delle relazioni sopra espresse. Tenendo a mente la (2.17) si nota come la predizione richieda necessariamente il calcolo della somma di convoluzione

$$S_c(m) = \sum_{i=0}^{m-1} A^{m-i-1} BL(i)^T \quad (2.18)$$

A questo proposito si noti che

$$\begin{aligned} S_c(1) &= BL(0)^T \\ S_c(2) &= ABL(0)^T + BL(1)^T = ABL(0)^T + BL(0)^T A_l^T \\ &= AS_c(1) + S_c(1)A_l^T \\ S_c(3) &= A^2BL(0)^T + ABL(1)^T + BL(2)^T \\ &= AS_c(2) + S_c(1)(A_l^2)^T \end{aligned} \quad (2.19)$$

dove è stato fatto uso della (2.6). In generale la (2.19) evidenzia la relazione ricorsiva

$$S_c(m) = AS_c(m-1) + S_c(1)(A_l^{m-1})^T \quad (2.20)$$

con  $m = 2, 3, \dots, N_p$ .

Questa osservazione si rivela fondamentale per la successiva implementazione dell'algoritmo che utilizza le funzioni di Laguerre. Infatti la matrice  $\Phi$  originale viene sostanzialmente modificata per far sì che i nuovi ingressi sino i vettori contenenti i coefficienti di Laguerre. A tale scopo si estende ora la trattazione dell'applicazione delle funzioni di Laguerre al problema MPC vista in sezione (2.2) al caso MIMO. Si assumerà che ad ogni ingresso corrisponda uno specifico polo di Laguerre, di modo da poter gestire separatamente la velocità di decrescenza a zero di ciascuno di essi. Si indichi dunque il vettore degli ingressi all'istante  $k$  con

$$\Delta u(k) = [\Delta u_1(k), \Delta u_2(k), \dots, \Delta u_m(k)]^T \quad (2.21)$$

e si partizioni la matrice  $B$  del sistema secondo la

$$B = [B_1, B_2, \dots, B_m] \quad (2.22)$$

dove  $m$  rappresenta il numero di ingressi e  $B_i$  la colonna  $i$ -esima della matrice  $B$ . Per ogni ingresso si scelgono poi i parametri  $a$  ed  $N$  di modo che, richiamando la (2.14), si può scrivere

$$\Delta u_i(k) = L_i(k)^T \eta_i \quad (2.23)$$

dove  $\eta_i$  e  $L_i(k)$  rappresentano la descrizione tramite rete di Laguerre dell'ingresso  $i$ -esimo, più specificamente

$$L_i(k)^T = [l_1^i(k), l_2^i(k), \dots, l_{N_i}^i(k)]. \quad (2.24)$$

Sulla base della partizione della matrice  $B$  suesposta e della conoscenza del vettore di stato  $x(k_i)$ , la predizione dell'uscita futura all'istante  $k$  si scrive

$$\begin{aligned} y(k_i + r | k_i) &= CA_m x(k_i) + \sum_{j=0}^{r-1} CA^{r-j-1} [B_1 L_1(j)^T, B_2 L_2(j)^T, \dots, B_m L_m(j)^T] \eta \\ &= CA^m x(k_i) + \phi(r)^T \eta \end{aligned} \quad (2.25)$$

dove il vettore dei parametri  $\eta$  ha struttura

$$\eta^T = [\eta_1^T, \eta_2^T, \dots, \eta_m^T] \quad (2.26)$$

e la matrice  $\phi(k)^T$  vale

$$\phi(r)^T = \sum_{j=0}^{r-1} CA^{r-j-1} [B_1 L_1(j)^T, B_2 L_2(j)^T, \dots, B_m L_m(j)^T]. \quad (2.27)$$

L'osservazione fondamentale è che il  $k$ -esimo blocco matriciale

$$\phi_k(r)^T = \sum_{j=0}^{r-1} C A^{r-j-1} B_k L_k(j)^T \quad (2.28)$$

ha la medesima struttura della  $S_c(m)$  definita nella (2.18), pertanto può essere agevolmente calcolata tramite la relazione (2.20).

In questo modo vi sono tutti gli elementi per riscrivere il funzionale di costo (1.52) proposto nell'approccio MPC originale di modo da poter utilizzare i coefficienti di Laguerre alla stregua di ingressi da ottimizzare. Nello specifico si avrà

$$J_{Lag} = \frac{1}{2} \eta^T H_{Lag} \eta + \eta^T F_{Lag} \quad (2.29)$$

con rispettivamente

$$\begin{aligned} H_{Lag} &= 2 (\Phi_{Lag}^T Q \Phi_{Lag} + R_{Lag} + T_{Lag}^T S T_{Lag}) \\ F_{Lag} &= 2 (\Phi_{Lag}^T Q (R_s - Fx(k)) + T_{Lag}^T S U_i) \end{aligned} \quad (2.30)$$

dove

$$\Phi_{Lag} = \begin{bmatrix} \phi(0) \\ \phi(1) \\ \vdots \\ \phi(N_p - 1) \end{bmatrix} \quad (2.31)$$

e dove

$$T_{Lag} = T L^T \quad (2.32)$$

con la matrice  $L$  costruita come

$$L^T = \begin{bmatrix} L(0)^T \\ L(2)^T \\ \vdots \\ L(N_c - 1)^T \end{bmatrix} \quad (2.33)$$

con

$$L(k)^T = \begin{bmatrix} L_1(k)^T & 0_2^T & \dots & 0_m^T \\ 0_1^T & L_2(k)^T & \dots & 0_m^T \\ \vdots & \vdots & \ddots & \vdots \\ 0_1^T & 0_2^T & \dots & L_m(k)^T \end{bmatrix} \quad (2.34)$$

e dove gli  $L_i(k)$ ,  $i = 1, 2, \dots, m$  sono definiti secondo la (2.24),  $m$  è il numero di ingressi,  $N_c$  l'orizzonte di controllo e  $0_i^T$  sono vettori riga nulli di dimensione opportuna. Infine la matrice diagonale  $R_{Lag}$  ha struttura

$$R_{Lag} = \begin{bmatrix} R_1 & 0_2^T & \dots & 0_m^T \\ 0_1^T & R_2 & \dots & 0_m^T \\ \vdots & \vdots & \ddots & \vdots \\ 0_1^T & 0_2^T & \dots & R_m \end{bmatrix} \quad (2.35)$$

con i blocchi diagonali  $R_i$ ,  $i = 1, 2, \dots, m$ , di forma

$$R_i = \begin{bmatrix} r_w^i & 0 & \dots & 0 \\ 0 & r_w^i & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & r_w^i \end{bmatrix} \quad (2.36)$$

dove gli  $r_w^i$  rappresentano i pesi d'ingresso di ciascun input memorizzati nella matrice originale dei pesi di ingresso  $R$  presentata in (1.48). Una volta risolto il problema di ottimizzazione QP in funzione dei coefficienti  $\eta$  il set di ingressi ottimi  $\Delta U = [\Delta u(k_i) \ \Delta u(k_i + 1) \ \dots \ \Delta u(k_i + N_c - 1)]$  viene recuperato utilizzando la semplice relazione

$$\Delta U = L^T \eta. \quad (2.37)$$

Di nuovo si vuole evidenziare che in questo modo la parte dell'algoritmo computazionalmente più rilevante, ovvero l'ottimizzazione, viene eseguita per calcolare  $N_{tot} = \sum_{k=1}^{n_{in}} N(k)$  parametri, quantità solitamente molto inferiore agli  $N_c$  ingressi futuri della versione classica dell'algoritmo MPC.

### 2.3.1 Caso non vincolato

In sezione 1.3.3 si giungeva per il problema di ottimo in assenza di vincoli la soluzione:

$$\frac{\partial J}{\partial \Delta U} = 0 \quad \Rightarrow \quad \Delta U_{ott} = H^{-1} F$$

Nella formulazione che utilizza le funzioni di Laguerre la soluzione è del tutto analoga alla (1.55) una volta sostituite opportunamente le matrici:

$$\frac{\partial J_{Lag}}{\partial \eta} = 0 \quad \Rightarrow \quad \eta_{ott} = H_{Lag}^{-1} F_{Lag} \quad (2.38)$$

Di nuovo, in analogia col caso precedente, si assumerà implicitamente che l'inversa delle matrici  $H_{Lag}$  esista. Le soluzioni trovate in (2.38) contengono tutta la legge ottima di controllo, ma in base al principio del *receding horizon* ad ogni singolo



passo di campionamento si calcola un set d'ingresso  $\Delta U_{ott}$ , ricavabile dalla (2.38) utilizzando la (2.37), del quale si utilizza soltanto il primo elemento:

$$\Delta u_{ott}(k) = [I_{n_{in}} \ 0_{n_{in}} \ \dots \ 0_{n_{in}}] \Delta U_{ott}$$

### 2.3.2 Vincoli in ingresso e uscita

Nel caso di problema di ottimo vincolato le matrici dei vincoli ricavate in sezione 1.3.4 vengono modificate, per poter essere applicate al vettore dei coefficienti  $\eta$ , come di seguito esposto.

#### Vincoli sulla variazione di ingresso

In questo caso la disuguaglianza matriciale  $M_1 \Delta U \leq N_1$ , espressa esplicitamente dalla 1.57 e qui riproposta

$$\begin{bmatrix} I & 0 & \dots & 0 \\ 0 & I & \dots & 0 \\ \vdots & & & \vdots \\ 0 & 0 & \dots & I \\ -I & 0 & \dots & 0 \\ 0 & -I & \dots & 0 \\ \vdots & & & \vdots \\ 0 & 0 & \dots & -I \end{bmatrix} \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \vdots \\ \Delta u(k+N_c-1) \end{bmatrix} \leq \begin{bmatrix} \Delta u^{max} \\ \Delta u^{max} \\ \vdots \\ \Delta u^{max} \\ -\Delta u^{min} \\ -\Delta u^{min} \\ \vdots \\ -\Delta u^{min} \end{bmatrix}$$

viene immediatamente riscritta, richiamando la (2.37), come

$$M_1^{Lag} \eta \leq N_1 \quad (2.39)$$

dove

$$M_1^{Lag} = M_1 L^T. \quad (2.40)$$

#### Vincoli sull'ingresso

In maniera analoga per quanto riguarda i vincoli in ingresso, la

$$\begin{bmatrix} I & 0 & \dots & 0 \\ I & I & \dots & 0 \\ \vdots & & & \vdots \\ I & I & \dots & I \\ -I & 0 & \dots & 0 \\ -I & -I & \dots & 0 \\ \vdots & & & \vdots \\ -I & -I & \dots & -I \end{bmatrix} \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \vdots \\ \Delta u(k+N_c-1) \end{bmatrix} \leq \begin{bmatrix} L_2 - u(k-1) \\ L_2 - u(k-1) \\ \vdots \\ L_2 - u(k-1) \\ -L_1 + u(k-1) \\ -L_1 + u(k-1) \\ \vdots \\ -L_1 + u(k-1) \end{bmatrix}$$

riscrivibile in forma compatta come  $M_2 \Delta U \leq N_2$  viene sostituita dalla

$$M_2^{Lag} \eta \leq N_2 \quad (2.41)$$

con

$$M_2^{Lag} = M_2 L^T. \quad (2.42)$$

### Vincoli sull'uscita

In questo caso la disuguaglianza

$$\begin{bmatrix} \Phi \\ -\Phi \end{bmatrix} \Delta U \leq \begin{bmatrix} Y^{max} - Fx(k) \\ -Y^{min} + Fx(k) \end{bmatrix}$$

in forma compatta  $M_2 \Delta U \leq N_2$ , viene modificata confrontando (2.3) con (2.30) giungendo alla riscrittura

$$\begin{bmatrix} \Phi_{Lag} \\ -\Phi_{Lag} \end{bmatrix} \eta \leq \begin{bmatrix} Y^{max} - Fx(k) \\ -Y^{min} + Fx(k) \end{bmatrix} \quad (2.43)$$

e definendo di conseguenza

$$M_3^{Lag} = \begin{bmatrix} \Phi_{Lag} \\ -\Phi_{Lag} \end{bmatrix} \quad (2.44)$$

### Programmazione Quadratica

Una volta modificate le matrici dei vincoli come descritto è dunque possibile riformulare il problema di programmazione quadratica nel caso di utilizzo delle funzioni di Laguerre: si vuole minimizzare l'indice di costo

$$J_{Lag} = \frac{1}{2} \eta^T H_{Lag} \eta + \eta^T F_{Lag} \quad (2.45)$$

nel rispetto del vincolo di disuguaglianza

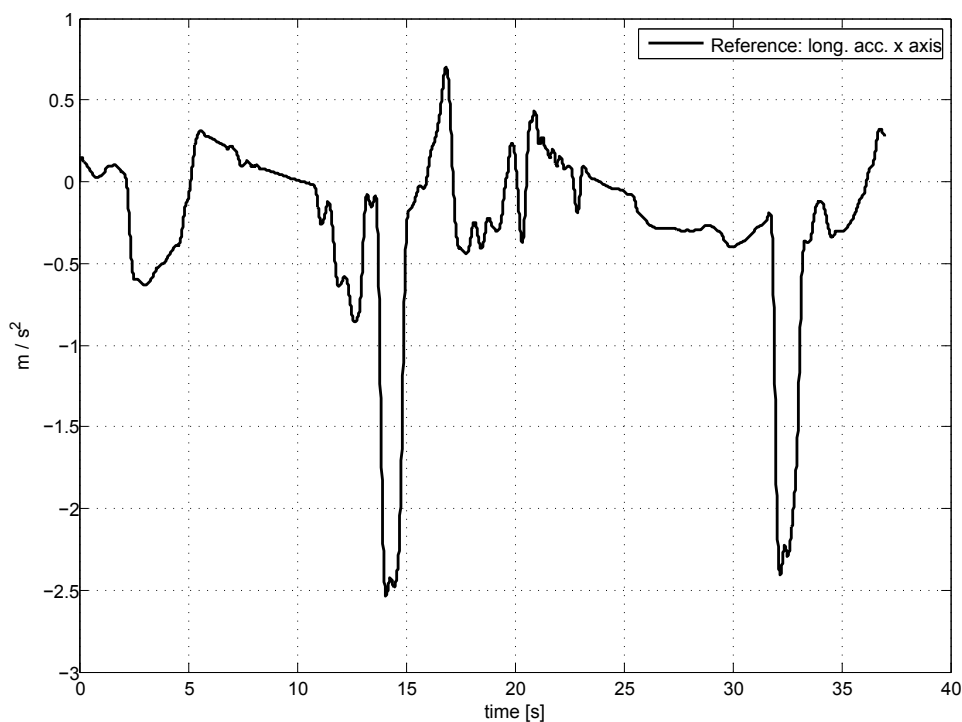
$$A \eta \leq b \quad \text{con} \quad A = \begin{bmatrix} M_1^{Lag} \\ M_2^{Lag} \\ M_3^{Lag} \end{bmatrix} \quad b = \begin{bmatrix} N_1 \\ N_2 \\ N_3 \end{bmatrix} \quad (2.46)$$

In questa forma il problema puo essere passato all'ottimizzatore qpOASES, che restituirà in uscita il vettore ottimo dei coefficienti di Laguerre  $\eta$ .

## 2.4 Simulazioni

In questa sezione si riportano le simulazioni effettuate per verificare il comportamento dell'algorithmo MPC modificato con le funzioni di. In considerazione dello scopo delle prove effettuate, la riduzione dei tempi di calcolo, orizzonti di predizione relativamente ampi. Ciò ha portato, data la necessità di confrontare i risultati degli algoritmi MPC originale e modificato con le funzioni di Laguerre, alla scelta di usufruire soltanto di parte dei dati telemetrici relativi a un giro di pista del circuito illustrato in Figura 1.4, in particolare dei primi 38 s.

Il segnale di riferimento adottato è illustrato in Figura 2.2 e come si può notare presenta caratteristiche abbastanza variegata (frenate impegnative e curve in appoggio, oltre a lunghi rettilinei), pertanto ben si presta ad essere utilizzato come banco di prova per il *motion cueing* sviluppato.



**Figura 2.2:** Riferimento di accelerazione percepita adottato.

In Figura 2.3 si riporta innanzitutto l'inseguimento di accelerazione percepita ottenuto utilizzando l'algorithmo MPC originale. La simulazione è stata effettuata utilizzando 1 secondo di predizione, settando i pesi in modo da ottenere un buon inseguimento e limitare leggermente l'utilizzo del grado di libertà angolare di *pitch* a favore di un maggior utilizzo dell'escursione longitudinale. Per un maggiore dettaglio del suddetto settaggio ci si rifaccia alla Tabella 2.1

Variabile	Peso
$y_1$	0.1
$y_2$	0.8
$y_3$	2
$y_4$	0.3
$y_5$	0.1
$y_6$	0.1
$u_1$	$10^{-5}$
$u_2$	$10^{-5}$
$\Delta u_1$	$10^{-5}$
$\Delta u_2$	$10^{-5}$

Tabella 2.1: Configurazione dei pesi per la simulazione

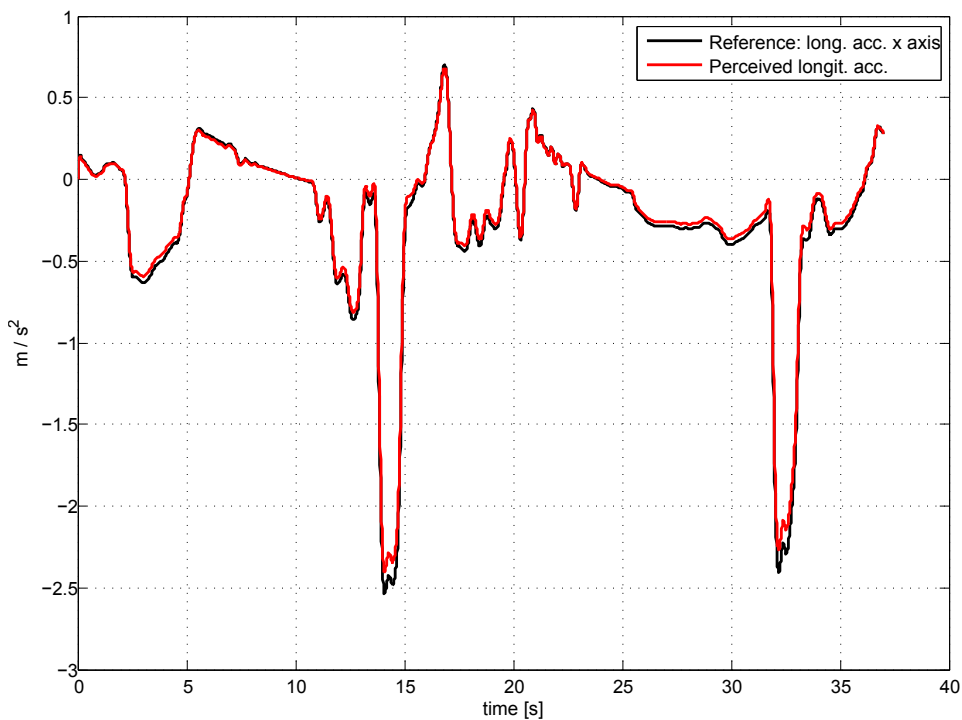
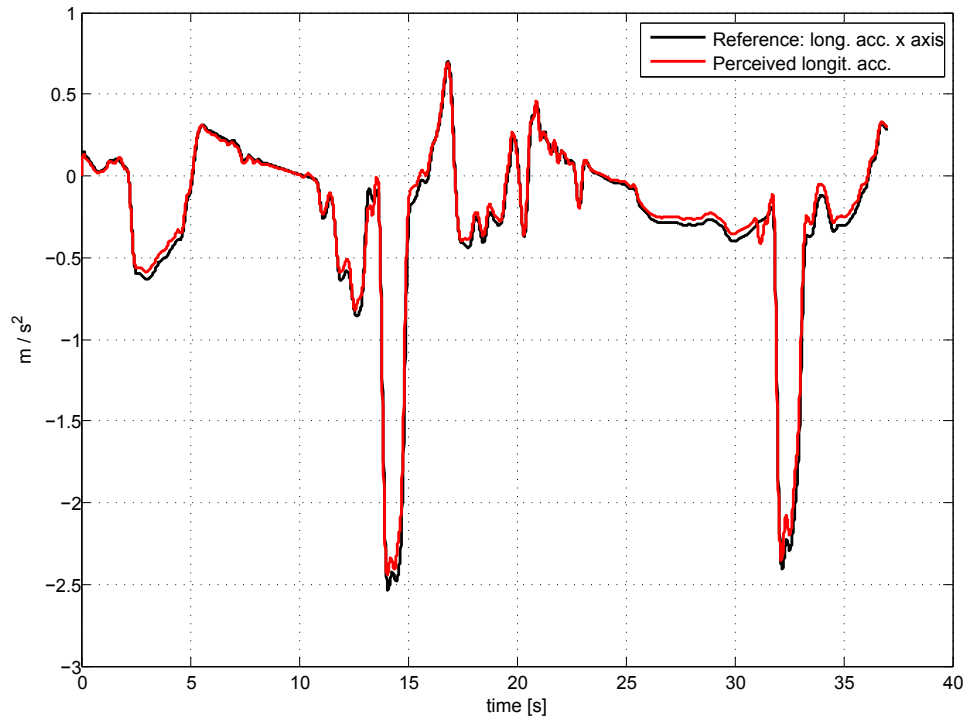


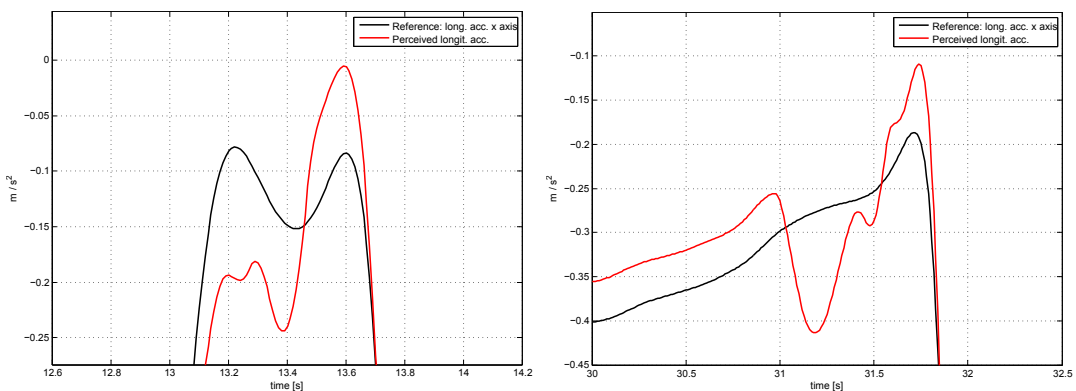
Figura 2.3: Inseguimento di accelerazione percepita.

In Figura 2.4 è rappresentata l'inseguimento di accelerazione percepita ottenuto utilizzando l'algoritmo modificato con le funzioni di Laguerre impostando i medesimi settaggi di Tabella 2.1 e con i parametri di Laguerre  $a = 0.99$ ,  $N = 30$ . Si noti come si riesca ad ottenere un ottimo inseguimento generale, rilevando però degli andamenti anomali in corrispondenza delle due brusche frenate relative alle curve Becketts e Stowe rispettivamente. Un maggiore dettaglio dei suddetti fenomeni si può osservare in Figura 2.5, dove appare chiaro che poco prima delle frenate in



**Figura 2.4:** Inseguimento di accelerazione percepita con funzioni di Laguerre.

questione il controllo imprima delle alternanze di decelerazioni e accelerazioni non presenti nel riferimento originale e che potrebbero causare delle sensazioni discordanti al pilota. La causa delle anomalie va ricercata nell'andamento intrinseco delle



(a) Accelerazione percepita con funzioni di Laguerre, ingrandimento 1

(b) Accelerazione percepita con funzioni di Laguerre, ingrandimento 2

**Figura 2.5:** Anomalie nell'inseguimento di accelerazione utilizzando le funzioni di Laguerre.

funzioni di Laguerre, più nello specifico nella limitata variabilità che può avere un

segnale costruito come combinazione lineare delle stesse a causa della loro rapida decrescenza a zero.

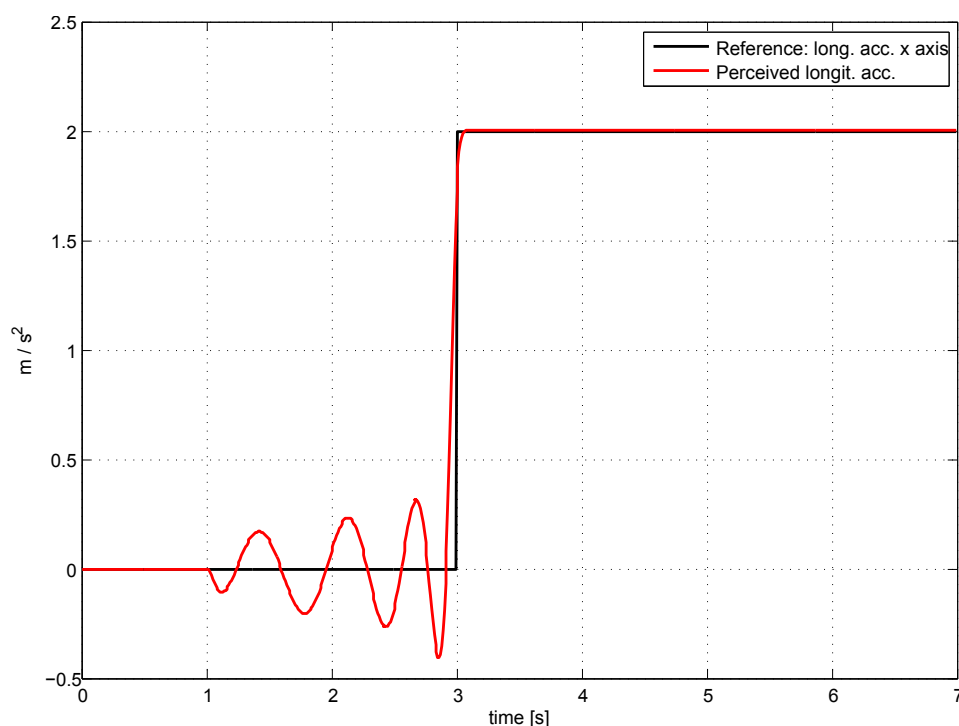
Per verificare empiricamente tale affermazione è stato effettuato un test che ora ci si accinge a descrivere. Innanzitutto la Figura 2.6 riporta l'inseguimento di un gradino di ampiezza 2 con fronte di salita a 3 s. Il controllore è stato settato in maniera tale da non usufruire del *pitch* per il controllo di accelerazione longitudinale, separando i due gradi di libertà (si è posto  $g = 0$  nell'algoritmo) e non richiedendo nient'altro che l'inseguimento del riferimento di accelerazione a gradino (pesi nulli ovunque in uscita tranne che in accelerazione percepita, si veda la Tabella 2.2); l'orizzonte di controllo è stato posto pari a 2 s. Si è inoltre avuto l'accorgimento di eliminare i limiti di piattaforma, sia in uscita che in ingresso, riducendosi a un semplice problema di ottimo non vincolato.

Variabile	Peso
$y_1$	0
$y_2$	0.8
$y_3$	0
$y_4$	0
$y_5$	0
$y_6$	0
$u_1$	$10^{-5}$
$u_2$	$10^{-5}$
$\Delta u_1$	$10^{-5}$
$\Delta u_2$	$10^{-5}$

**Tabella 2.2:** Configurazione dei pesi per il tuning.

Dal grafico si nota chiaramente come a partire da un secondo di simulazione, cioè non appena la finestra di predizione è in grado di rilevare il fronte di salita, il segnale in uscita dal controllore inizia ad oscillare intorno allo zero, mostrando un comportamento del tutto analogo a quando osservato in Figura 2.5. Per rendersi conto di quanto accade si è proceduto all'analisi prelevando il vettore dei coefficienti  $\eta$  ottimo direttamente in uscita dall'ottimizzatore al centesimo passo, convertendoli poi nella sequenza di ingressi di controllo corrispondente. In altre parole si è ricavata la sequenza di ingressi che all'istante 1 s il controllore applicherebbe al sistema per inseguire nel miglior modo possibile il riferimento a gradino. La Figura 2.5 mostra l'uscita in accelerazione corrispondente a tale sequenza.

Si vede come la sequenza di controllo applicata, pur essendo la migliore possibile in quanto soluzione ottima, non sia affatto adeguata per l'inseguimento di un riferimento a gradino. Appare chiaro quindi che il limite sta nella poca variabilità di segnale di controllo che le funzioni di Laguerre consentono di ottenere in situazioni di questo tipo rendendole poco adatte all'utilizzo con un algoritmo che faccia uso del *look-ahead*.

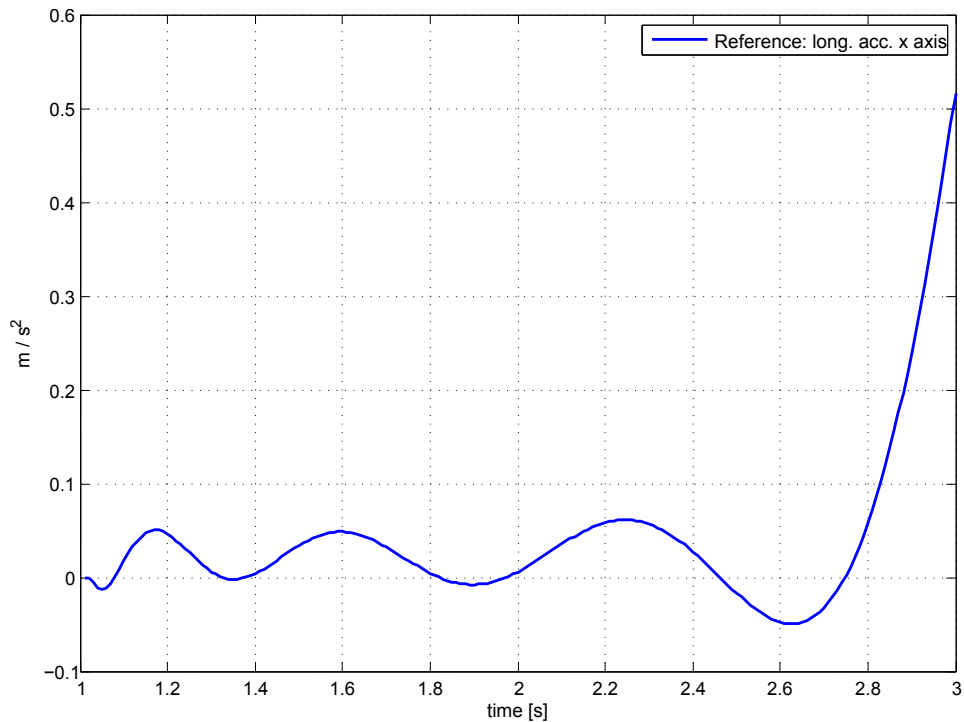


**Figura 2.6:** Inseguimento di un gradino di riferimento.

Per completezza le Figure 2.8-2.9 estendono il confronto tra i due algoritmi riportando la posizione angolare di *pitch*, la posizione longitudinale, l'accelerazione applicata e il jerk, grandezza definita come la derivata dell'accelerazione e utile parametro di confronto, anch'esso applicato e percepito. Si noti come in corrispondenza degli istanti critici la distorsione del segnale di accelerazione percepita si ripercuota sui segnali di posizione di *pitch* e longitudinale (Figure 3.15(b)-3.15(c)).

Lo scopo principe dell'uso delle funzioni di Laguerre nell'algoritmo MPC è la riduzione dei tempi di calcolo degli ingressi ottimi, al fine di poter realizzare il controllo in tempo reale anche con tempi di predizione elevati. È indubbio infatti che la strategia di controllo predittiva tragga i maggiori benefici allorché l'orizzonte di controllo sia di maggiore estensione, di modo da poter gestire in maniera ottimale gli ingressi. Tuttavia la scelta di un tempo di campionamento pari a  $T_s = 0.01$  s. comporta, anche solo per una frazione di tracciato come qui considerato, un onere computazionale che rende il tempo di calcolo degli ingressi molto al di sopra di quanto si vorrebbe. L'algoritmo che utilizza le funzioni di Laguerre come espresso in sezione 2.2 consente di spostare il dimensionamento del problema dal parametro  $N_c$  orizzonte di controllo (cioè il numero di ingressi futuri calcolati ad ogni passo di campionamento pari a  $\frac{T_p}{T_s}$  con  $T_p$  finestra di predizione in secondi) al parametro  $N$ , numero di coefficienti di Laguerre utilizzati per ciascun ingresso.

La Tabella 2.3 riporta i tempi di calcolo degli ingressi da parte dei due algoritmi per orizzonti di predizione differenti. I tempi considerati si riferiscono alla sola pro-



**Figura 2.7:** Accelerazione in uscita in corrispondenza del controllo applicato al tempo 1 s, finestra di predizione di 2 s.

cedura di ottimizzazione, escludendo dunque eventuali l'inizializzazione e i calcoli *off-line*, e i dati relativi all'algoritmo modificato sono stati ottenuti utilizzando i parametri caratteristici  $a = 0.99$ ,  $N = 30$ . Come si puo notare L'algoritmo modifi-

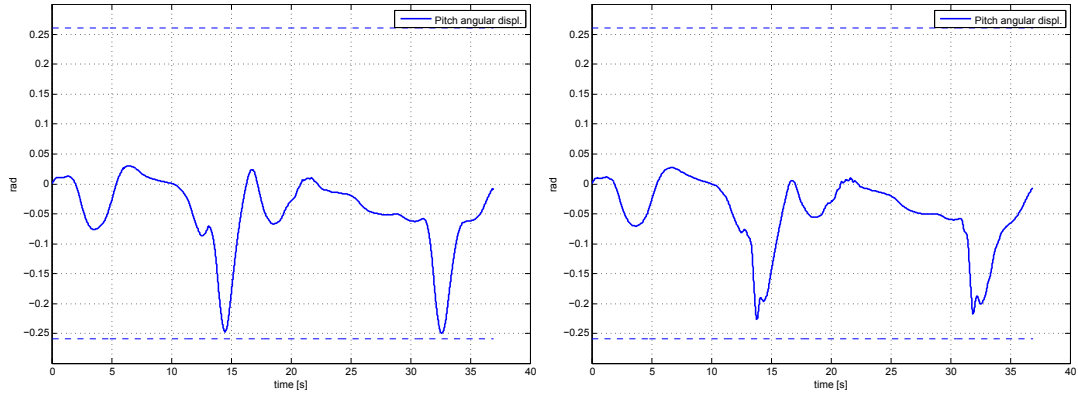
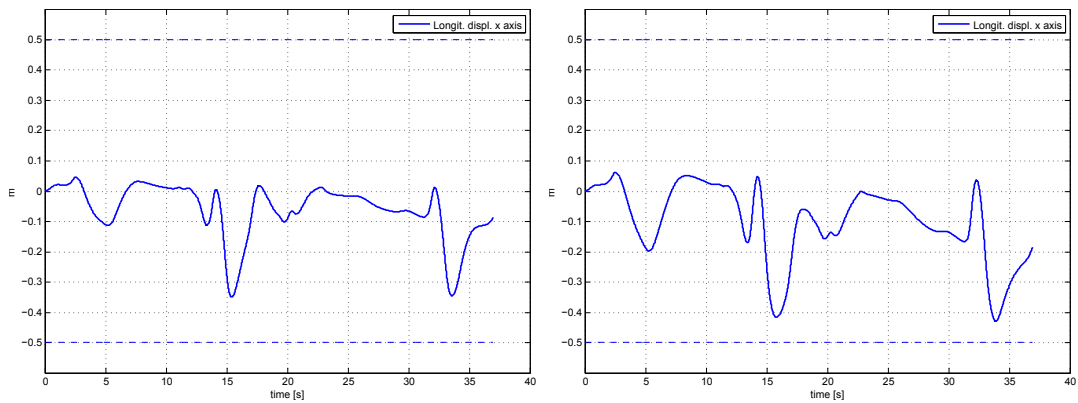
	$N_c = 0.5 \text{ s}$	$N_c = 1 \text{ s}$	$N_c = 2 \text{ s}$	$N_c = 3 \text{ s}$
<b>MPC originale</b>	31.08 s	103.79 s	575.25 s	2466.13 s
<b>MPC Laguerre</b>	2.75 s	4.41 s	20.21 s	264.15 s

**Tabella 2.3:** Tempi di ottimizzazione dei due algoritmi per diversi orizzonti di controllo

cato con le funzioni di Laguerre offre corposi vantaggi, risultando mediamente più veloce nel calcolo degli ingressi di un fattore 20-25. Nella tabella sono evidenziati in rosso i tempi per cui l'algoritmo corrispondente non sarebbe in grado di fornire un controllo *real-time* della piattaforma poichè il tempo di elaborazione dei dati supererebbe i 38 s di durata della finestra di simulazione. I dati sono ovviamente da riportare alla specifica macchina utilizzata per le simulazioni, ma servono a dare un'idea qualitativa della problematica legata ai tempi di calcolo.

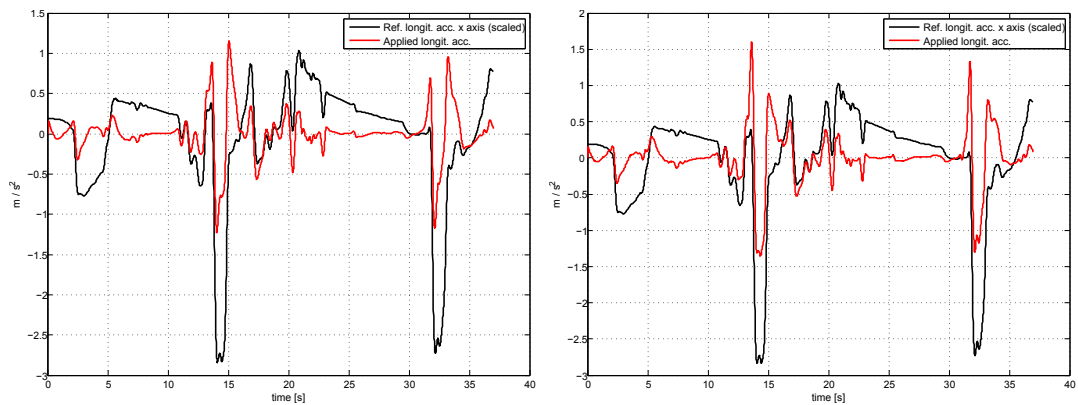
Si rifletta ora come possa apparire insolito che il tempo di calcolo relativo all'algoritmo con funzioni di Laguerre aumenti a parità di  $N$ . In realtà per calcolare gli



(a) Posizione di *pitch*, algoritmo originale(b) Posizione di *pitch*, algoritmo con funzioni di Laguerre

(c) Posizione longitudinale, algoritmo originale

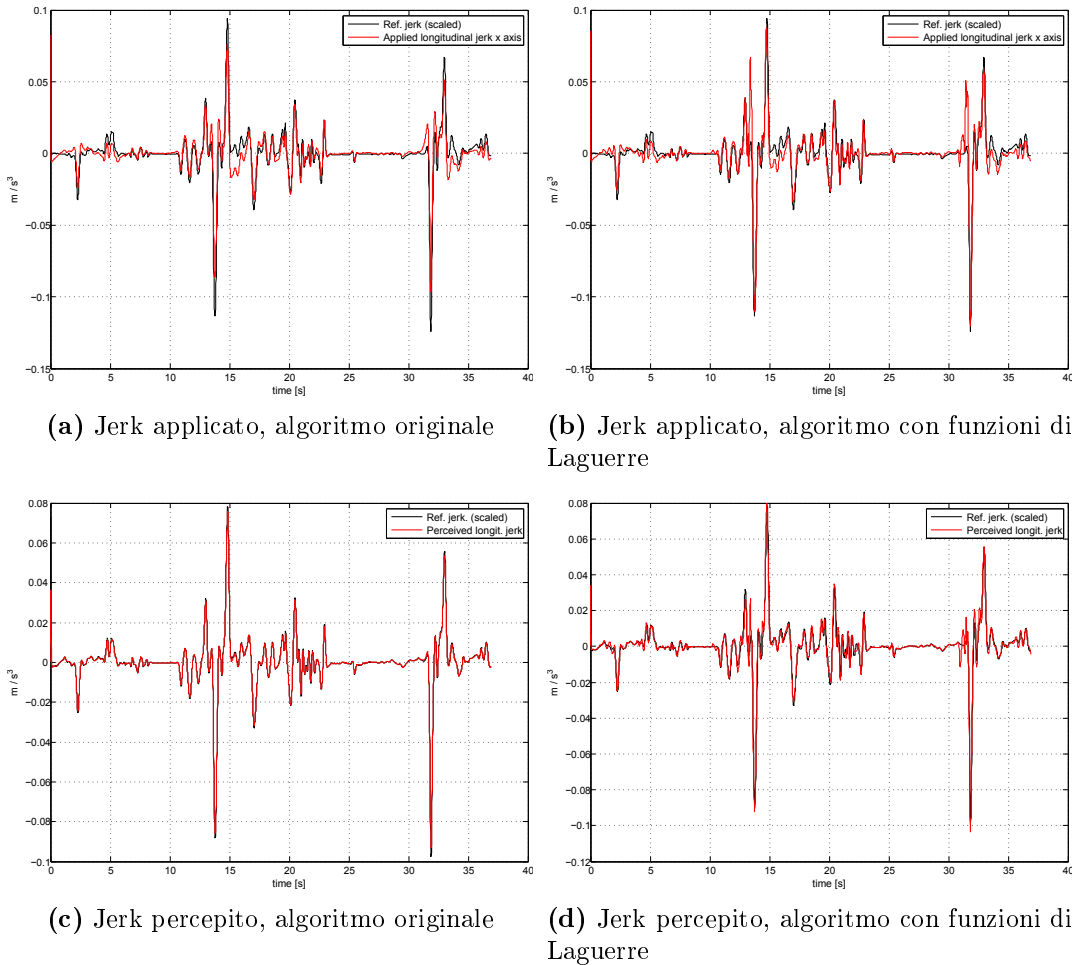
(d) Posizione longitudinale, algoritmo con funzioni di Laguerre



(e) Accelerazione applicata, algoritmo originale

(f) Accelerazione applicata, algoritmo con funzioni di Laguerre

**Figura 2.8:** Confronto dei risultati ottenuti con l'algoritmo MPC originale e utilizzando le funzioni di Laguerre.



**Figura 2.9:** Confronto dei risultati ottenuti con l'algoritmo MPC originale e utilizzando le funzioni di Laguerre.

effettivi ingressi di controllo a partire dai coefficienti di Laguerre ricavati dall'ottimizzatore è necessario effettuare delle moltiplicazioni tra matrici le cui dimensioni sono proporzionali all'orizzonte di controllo  $N_c$ . In particolare la matrice dei vincoli  $A$  descritta nella (2.46):

$$A = \begin{bmatrix} M_1^{Lag} \\ M_2^{Lag} \\ M_3^{Lag} \end{bmatrix}$$

ha un numero di righe direttamente proporzionale a  $N_c$  ed essa viene moltiplicata più volte all'intero della procedura di ottimizzazione, rallentando pesantemente il calcolo. Di fatto per valori di  $N_c$  ancora superiori (sopra i 2 s) l'algoritmo rallenta vistosamente proprio a causa dell'aumento dimensionale di tali matrici, vanificando i vantaggi introdotti dall'approssimazione di Laguerre ai fini del controllo in tempo reale. Tuttavia, a differenza degli andamenti oscillatori in prossimità di brusche va-

riazioni del riferimento, tale inconveniente non è intrinseco alle funzioni di Laguerre ma è legato piuttosto all'algoritmo di ottimizzazione ed in particolare alla struttura delle matrici dei vincoli; una costruzione più efficiente di tali matrici potrebbe costituire un interessante sviluppo futuro.

In conclusione appare chiaro come l'integrazione delle funzioni di Laguerre nell'algoritmo di controllo MPC sviluppato porti innegabili vantaggi in termini di riduzione dell'onere computazionale, ma presenti delle lacune strutturali che ne limitano fortemente l'applicazione ad un controllore MPC che implementi una strategia di controllo *look-ahead*.

Nel prossimo capitolo si valuterà le funzioni di Laguerre in applicazioni che richiedano l'analisi del *worst case*, per il quale si guarderà ad un controllore che utilizzi per la predizione un riferimento costante.



## Capitolo 3

### *Tuning* del controllore MPC

Il controllo MPC consente un uso ottimale della piattaforma di simulazione, consentendo alla stessa di muoversi fino ai limiti imposti dalle caratteristiche fisiche della struttura che la costituisce. È questo uno dei principali vantaggi degli algoritmi di controllo predittivo rispetto al *motion cueing* classico, che opera di fatto il filtraggio dei segnali di riferimento rimanendo molto più cauto riguardo ai limiti di piattaforma, il che penalizza molto spesso le prestazioni. I *feedbacks* rilasciati dai piloti hanno però portato alla luce altri aspetti legati allo sfruttamento della piattaforma.

Uno degli aspetti più interessanti legati all'utilizzo dell'algoritmo MPC nel simulatore è la *tilt coordination*, che come già introdotto nella sezione 1.3.1 permette di utilizzare i gradi di libertà rotazionali della piattaforma, per riprodurre, a livello percettivo, accelerazioni lineari. Le prove effettuate hanno però evidenziato alcuni limiti pratici di tale utilizzo. In particolare la *tilt coordination* deve essere utilizzata con parsimonia, evitando che nel segnale d'ingresso di posizione angolare compaiano componenti di alta frequenza che potrebbero generare il cosiddetto *motion sickness*. Anche l'ampiezza del segnale di posizione angolare non dovrebbe essere troppo elevata, per evitare che inversioni di segno dell'accelerazione percepita comportino un recupero troppo lento e di conseguenza uno scompenso sensoriale tra ciò che il pilota si aspetta e ciò a cui è in realtà sottoposto dalla piattaforma.

In maniera analoga il pieno sfruttamento della piattaforma nelle direzioni longitudinali e laterali potrebbe non essere sempre la miglior soluzione in quanto in situazioni nelle quali si operi in prossimità dei limiti di escursione l'algoritmo predittivo potrebbe generare dei richiami (e quindi delle inversioni di direzione) in contrasto con l'accelerazione richiesta, che nella migliore delle ipotesi andrebbero compensati usando la posizione angolare e quindi ancora abusando della *tilt coordination*.

Per questa serie di motivi si vorrebbe fornire all'utente la possibilità di regolare l'intensità e la forma dei segnali in gioco in maniera semplice, di modo da poter settare in ogni situazione di utilizzo il comportamento della piattaforma affinché l'esperienza d'uso sia la più verosimile possibile.

Un altro motivo che giustifica la ricerca di una regolazione (*tuning*) dei parametri della piattaforma è l'esigenza di un confronto delle dinamiche espresse dal controllore MPC con quelle del *motion cueing* classico. La piattaforma VI-Grade utilizza infatti un algoritmo di controllo collaudato e i cui comportamenti sono ben noti, costituendo di fatto un metro di paragone cui fare riferimento e su cui basarsi come punto di partenza per il settaggio dei parametri del controllore MPC. In altre parole riprodurre il comportamento in ingresso del *motion cueing* classico costituisce un solido punto di inizio da cui partire per poter in futuro migliorare le prestazioni del simulatore.

Nel seguito si dedicherà ampio spazio al problema del *tuning* del controllore MPC, principalmente in riferimento all'algoritmo di *motion cueing* classico utilizzato da VI-grade, evidenziandone le problematiche legate alla struttura intrinseca dell'algoritmo e proponendo un approccio alternativo che mira a risolvere alcune di esse.

### 3.1 *Tuning* del controllore MPC in riferimento al *motion cueing* classico

Come già anticipato il comportamento del *motion cueing* classico costituisce un punto di partenza per la taratura dei parametri del controllore MPC. Per questo motivo nel presente lavoro si è studiato il problema di riprodurre, attraverso una opportuna selezione del valore dei parametri disponibili, i segnali di posizione angolare e lineare rilevati da Vi-Grade utilizzando lo schema classico di *motion cueing*. Nello specifico ci si concentrerà sul sotto-sistema longitudinale-pitch, senz'altro il più rappresentativo dell'accoppiamento tra gradi di libertà traslazionali e rotazionali.

#### 3.1.1 Parametri di taratura

In figura (3.1) è rappresentato lo schema di controllo naturale utilizzato per le operazioni di *tuning*, mettendo in evidenza come ingressi, oltre al riferimento, i parametri su cui è possibile andare ad agire per raggiungere lo scopo.

Tali parametri sono molteplici, ma fondamentalmente divisi in tre tipologie:

- **Le frequenze di taglio dei filtri:** L'algoritmo MPC prevede una fase preparatoria, in cui il segnale di riferimento viene trattato per poterlo rendere tracciabile, nella pratica, dalla piattaforma. I segnali originali infatti provengono dalle telemetrie fornite da VI-Grade, e presentano caratteristiche non riproducibili dal simulatore; si pensi a titolo di esempio il segnale di accelerazione che presenta picchi di intensità troppo elevati, nonché componenti ad alta frequenza troppo marcate per poter essere inseguite correttamente.

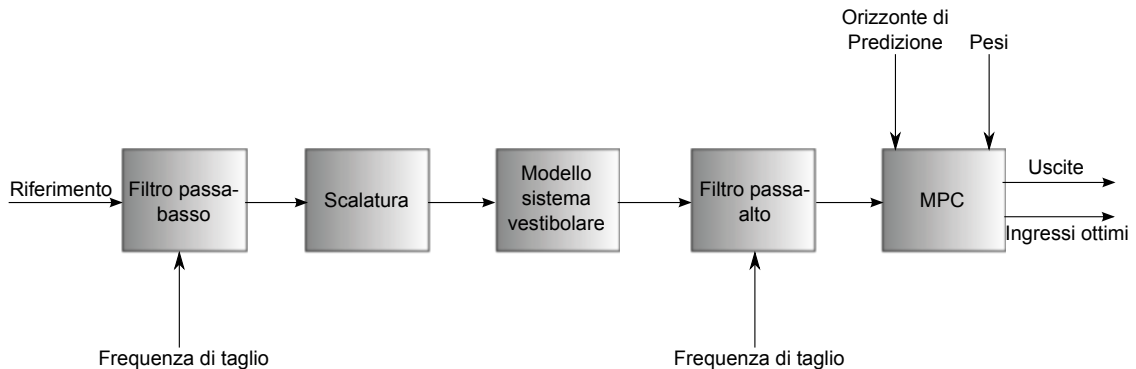


Figura 3.1: Schema di controllo per il *tuning*

Si opera dunque un doppio filtraggio, prima un passa-basso che elimina le componenti a frequenza troppo elevata restituendo un segnale più dolce, poi il segnale viene scalato per poter essere trattato entro i limiti fisici di piattaforma, ed infine viene effettuato un filtraggio passa-alto volto ad eliminare gli andamenti più persistenti evitando ancora una volta di raggiungere i limiti imposti. È dunque evidente che la regolazione delle frequenze di taglio dei filtri, in particolare del filtro passa-alto modifica sostanzialmente il segnale di riferimento, rendendolo intrinsecamente più o meno facile da inseguire e con differente apporto di componenti di rotazione di pitch e traslazione longitudinale.

- **I pesi delle matrici della funzione costo:** La funzione costo presentata in sezione J nel capitolo 1, presenta delle matrici quadrate diagonali che hanno la funzione di pesare il contributo di ciascun termine (nel caso errore) all'indice di costo stesso. Maggiore sarà il peso, minore sarà il discostamento dal segnale di riferimento tollerato. Nei casi in cui non vi fosse un segnale specifico di riferimento da inseguire (si pensi alle stesse posizioni angolare e longitudinale) il tracking è stato eseguito rispetto al segnale nullo, mirando in tal modo a stabilizzare le evoluzioni della piattaforma attorno al punto d'origine. È chiaro dunque che la variazione dei suddetti pesi influenza la dinamica dei segnali in gioco, sia in termini di forma che di ampiezza, e consente in generale di poter modellare, intervenendo opportunamente, la precisione dell'inseguimento del riferimento desiderato e in che modo gli ingressi contribuiscono al raggiungimento di tale obiettivo.
- **L'orizzonte di predizione:** Componente fondamentale della strategia di controllo MPC è la predizione dell'andamento futuro dell'uscita. Quest'ultima viene utilizzata per minimizzare l'errore di inseguimento e può essere calcolata su di un orizzonte temporale più o meno lungo. Il controllore MPC

è intrinsecamente in grado di calcolare l'uscita futura sulla base del reale andamento del riferimento, ma per lo specifico *task* del tuning il riferimento viene preso costante lungo tutta la durata dell'orizzonte di predizione, di modo da ricondursi ad una condizione di *worst case* che si avvicini maggiormente a quella in cui opera il *motion cueing* classico (che non attua alcuna predizione). La conseguenza diretta, intuitiva e sperimentalmente verificata, è che utilizzando un orizzonte di predizione più lungo il controllore tenderà ad essere più conservativo nel muovere la piattaforma, in conseguenza del fatto che prevederà che un intervento troppo deciso, in termini di ampiezza del segnale, potrebbe portare il controllo troppo vicino ai limiti di escursione. Viceversa un orizzonte di predizione breve comporterà segnali di controllo tendenzialmente più consistenti, volti ad eliminare immediatamente gli errori di inseguimento in mancanza di ulteriori (seppur approssimate) informazioni sull'evoluzione futura dell'errore oltre l'orizzonte suddetto. Si tiene a sottolineare che l'utilizzo di un orizzonte di predizione più lungo in questo frangente non rappresenta necessariamente la scelta migliore in termini di precisione di inseguimento del riferimento, in quanto il riferimento tenuto costante per tutta la durata di tale orizzonte può divergere eccessivamente dal riferimento reale, portando il controllore ad operare correzioni nella direzione sbagliata, o comunque di non corretta entità; per lo specifico *task* del *tuning* la variazione di questo parametro è solamente strumentale ad un *matching* più fedele possibile dei comportamenti dei controllori MPC e MC classico.

Le figure 3.2-3.3-3.4 riportano i risultati della simulazione ottenuta tarando i parametri così come riportato in Tabella 3.1.

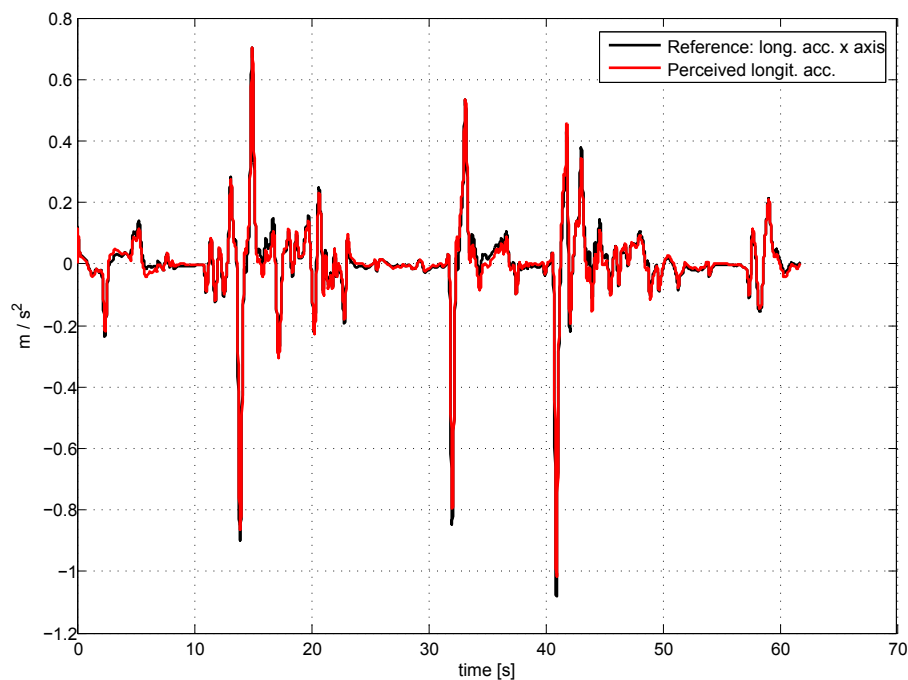
Parametro	Peso
$y_1$	80
$y_2$	80
$y_3$	650
$y_4$	460
$y_5$	60
$y_6$	1
$u_1$	1
$u_2$	$10^{-3}$
$\Delta u_1$	$10^{-5}$
$\Delta u_2$	$10^{-2}$

**Tabella 3.1:** Configurazione dei pesi per il test di inseguimento del gradino.

Come già enunciato in precedenza l'operazione di *tuning* risulta necessaria per verificare le reazioni del sistema controllato alle modifiche dei parametri che interessano il funzionamento dell'algoritmo di controllo, allo scopo di poter desumere



una serie di procedure che consentano poi ad un utente esterno di poter regolare il comportamento del controllore a seconda delle proprie esigenze. La sperimentazione ha evidenziato come il problema non risulti di immediata risoluzione a causa dell'elevato grado di accoppiamento delle grandezze in gioco. Nel seguito si andranno dunque ad esporre alcune delle problematiche e delle limitazioni legate a questo aspetto.



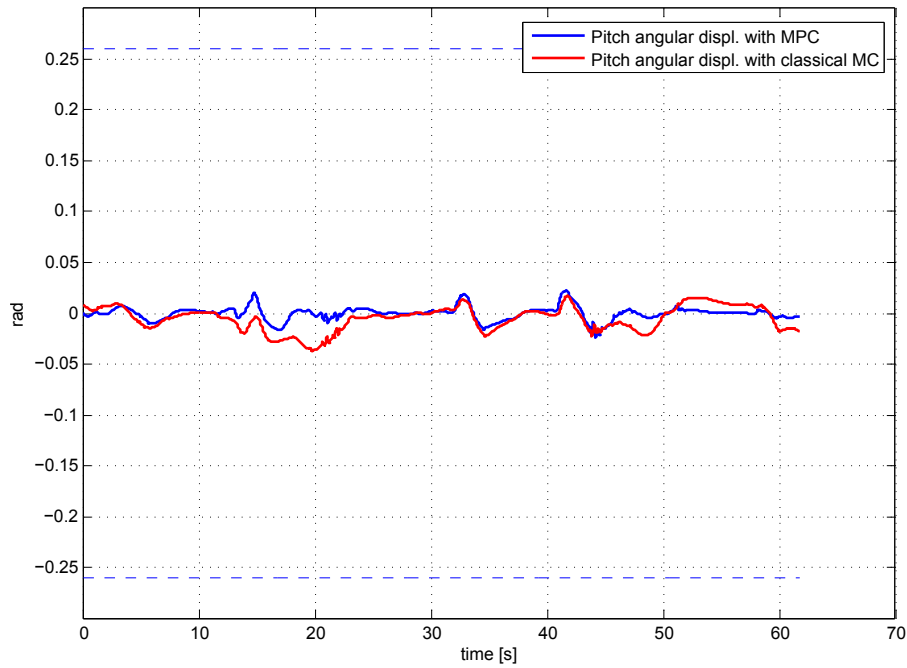
**Figura 3.2:** Tuning del controllore rispetto al MC classico: inseguimento di accelerazione longitudinale percepita

### 3.1.2 Risultati, osservazioni e limiti del *tuning*

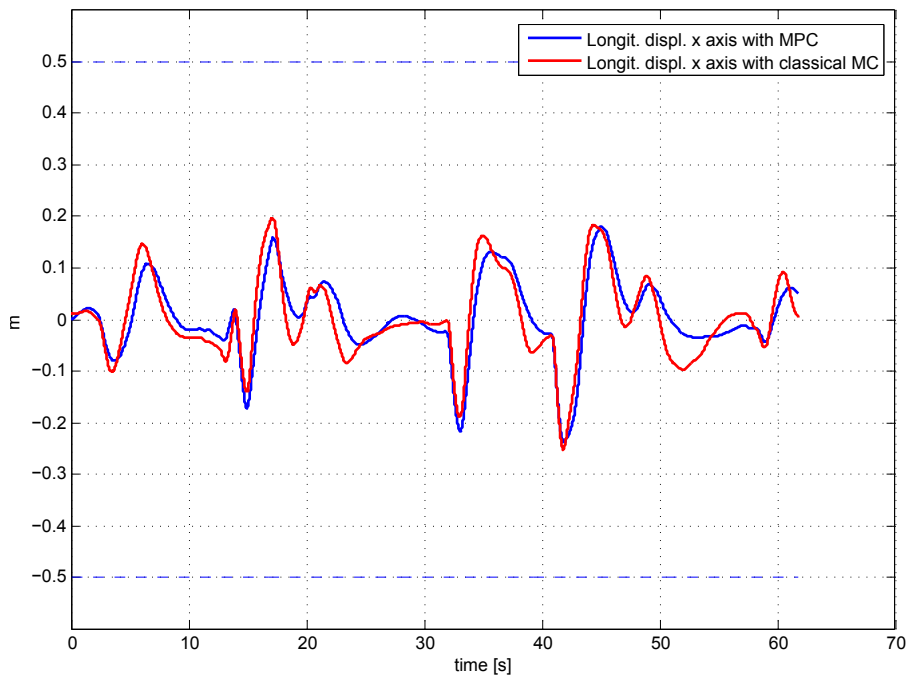
Le variazioni dei parametri caratteristici dell'algoritmo in generale portano a comportamenti del sistema relativamente intuitivi per quanto riguarda le grandezze direttamente interessate (si pensi, a titolo di esempio all'effetto della variazione del peso in posizione longitudinale sulla posizione longitudinale stessa), ma come già anticipato esse implicano anche una serie di effetti collaterali su altre grandezze. Si riprendono ora brevemente le diverse categorie di parametri presentate nel paragrafo precedente, esemplificando alcuni degli effetti delle variazioni dei parametri significativi sulle grandezze interessate.

#### Frequenze di taglio dei filtri

Tralasciando il filtro passa-basso, che effettua di fatto un'operazione preparatoria del segnale grezzo, ed in particolare sul segnale applicato e non percepito (non



**Figura 3.3:** Tuning del controllore rispetto al MC classico: Posizione angolare di *pitch*



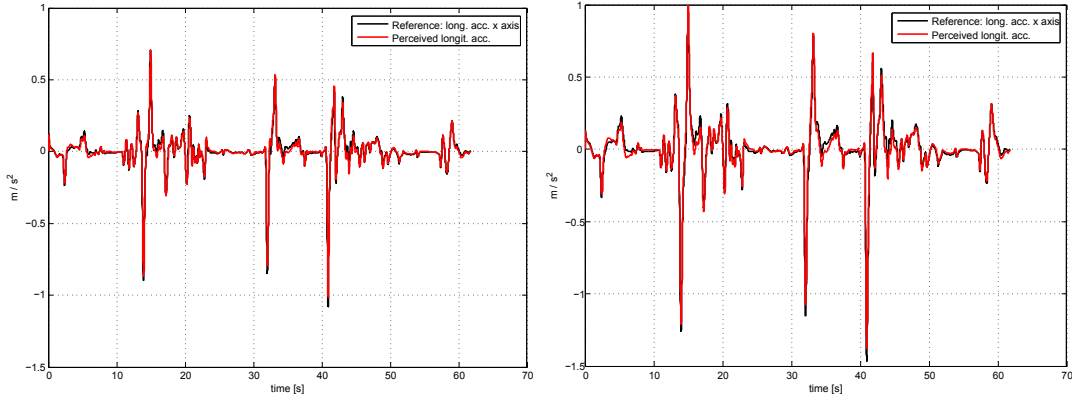
**Figura 3.4:** Tuning del controllore rispetto al MC classico: Posizione longitudinale

filtrato cioè dal modello del sistema vestibolare, si faccia riferimento alla figura 3.1), ci si concentra sull'effetto della variazione della frequenza di taglio del filtro passa-alto. Le conseguenze di tale variazione sono globali, nel senso che l'azione è effettuata sul riferimento che viene passato all'algoritmo di controllo e quindi prima del calcolo degli ingressi ottimi. In Figura 3.5 si illustra questo concetto. Il grafico confronta l'inseguimento del riferimento di accelerazione percepita, la posizione angolare di *pitch* e longitudinale prima con i parametri settati come nel paragrafo precedente (colonna di sinistra), e successivamente con la sola modifica, nello specifico la riduzione, del parametro frequenza di taglio del filtro passa-alto (colonna di destra). Come si può notare il riferimento di accelerazione in figura 3.5(b) risulta sostanzialmente modificato, sia in ampiezza (si presti attenzione all'intervallo di valori dell'asse  $y$ ) sia nella forma, notando in particolare come nella 3.5(b) siano meno rilevanti le componenti ad alta frequenza. Questa situazione è ovvia conseguenza dell'aver allargato, tramite la riduzione della frequenza di taglio del filtro, la banda passante del filtro stesso, passando in questo modo al controllore un riferimento intrinsecamente più difficile da seguire perchè più ricco di componenti su una gamma più ampia di frequenze. Ciò si ripercuote nel movimento della piattaforma come si evince chiaramente dalle figure 3.5(d)-3.5(f). In entrambi i casi si nota un incremento di ampiezza dei segnali di controllo di posizione dovuti alla maggiore complessità del segnale di riferimento. In particolare la presenza di andamenti persistenti più ampi porta il segnale di posizione di *pitch* ad ampiezze maggiori, lasciando la gestione delle alte frequenze (ora però in un *range* più ampio) all'escursione longitudinale della piattaforma che deve dunque anch'essa far fronte a variazioni di accelerazione più consistenti.

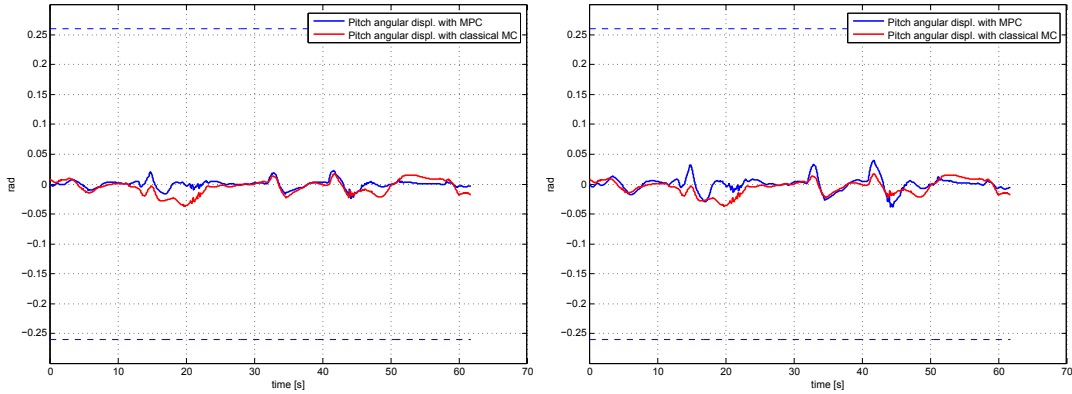
Si capisce dunque come il parametro in questione, agendo a monte del sistema di controllo, colpisca indistintamente sia l'ingresso longitudinale che quello angolare nella medesima direzione, risultando pertanto adatto ad un primo *step* di dimensionamento delle azioni di controllo ma non ad una taratura settoriale che miri a regolare un particolare ingresso o andamento.

### **Pesi delle matrici della funzione costo**

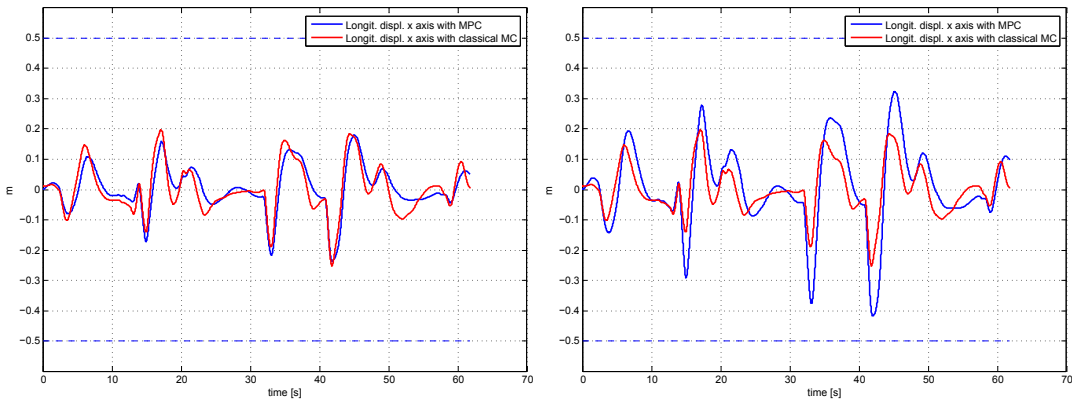
A differenza di quanto visto per le frequenze di taglio dei filtri i pesi delle matrici associate alla funzione costo permettono di agire specificamente sulla tolleranza dell'errore di inseguimento di una specifica grandezza; nel caso essa non abbia un segnale di riferimento da inseguire, come già anticipato, questo viene passato nullo, così che la tolleranza all'errore si traduce in tolleranza al discostamento dalla posizione di riposo. Una tipica richiesta, che verrà richiamata più avanti perchè centrale nel presente lavoro di tesi, è quella di poter diminuire l'apporto di *tilt coordination* al *tracking* dell'accelerazione, al fine di limitare le sensazioni sgradevoli sul pilota cui si accennava ad inizio capitolo. Si vorrebbe che tale intervento lasciasse relativamente inalterato il contributo di posizione longitudinale, di modo da non costringerlo a spingersi troppo verso i limiti di piattaforma. In questo caso risulta



(a) Accelerazione percepita, tuning originale (b) Accelerazione percepita, frequenza di taglio modificata



(c) Posizione di *pitch*, tuning originale (d) Posizione di *pitch*, frequenza di taglio modificata



(e) Posizione longitudinale, tuning originale (f) Posizione longitudinale, frequenza di taglio modificata

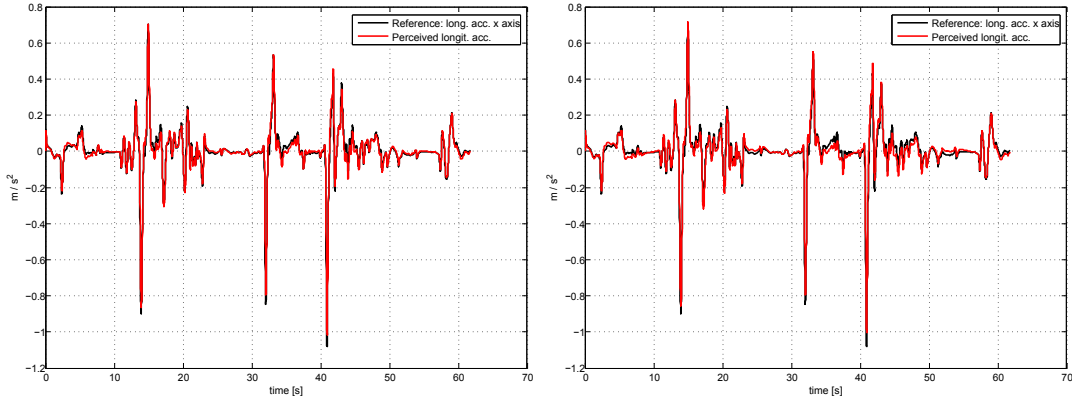
**Figura 3.5:** Effetti della riduzione della frequenza di taglio del filtro passa-alto

naturale agire, aumentandolo, sul peso in posizione angolare, di modo da limitare l'escursione dell'angolo di *pitch* e riducendo quindi l'uso della *tilt coordination*. Tut-

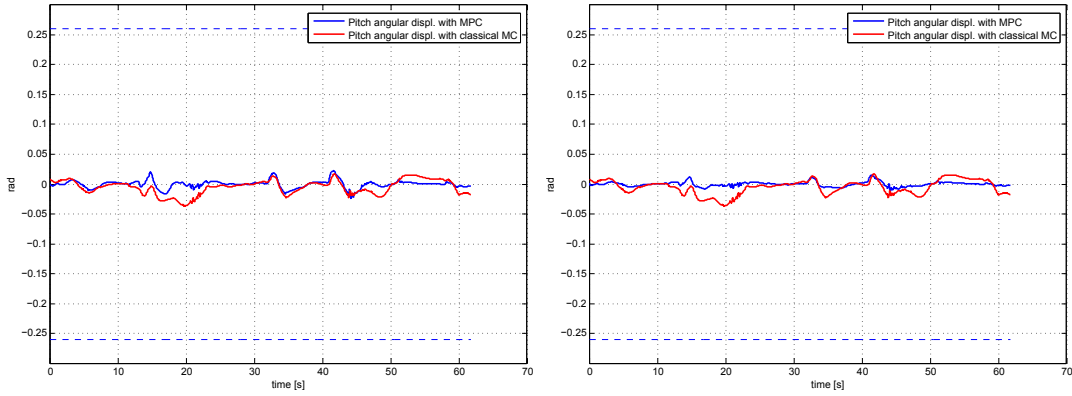
tavia l'accoppiamento tra i gradi di libertà longitudinale e angolare di *pitch* porta a conseguenze che si estendono anche all'ingresso longitudinale. La Figura 3.6 illustra la situazione. Come si può vedere l'entità dell'azione di *pitch* viene ridotta, ma il mancato apporto di *tilt coordination* viene scaricato sul movimento longitudinale di piattaforma stravolgendo il relativo segnale di posizione come si può vedere da Figura 3.6(f). Si può pensare di ricorrere a delle modifiche combinate di più parametri, quali l'aumento, in contemporanea col peso in posizione angolare, del peso in posizione longitudinale, come appare logico per limitare l'escursione longitudinale della piattaforma. Tale accorgimento però non produce i benefici sperati a causa della relatività dei pesi nella funzione costo. Di fatto l'aumento contemporaneo dei due pesi non apporta alcuna modifica sostanziale alla dimensione dei segnali interessati poichè le due modifiche si annullano, restituendo l'equilibrio originale, a discapito tuttavia della qualità dell'inseguimento dell'accelerazione longitudinale percepita che peggiora inevitabilmente. L'esempio chiarifica un aspetto generale dell'azione di regolazione tramite i pesi, ovvero l'interdipendenza dei gradi di libertà angolare e longitudinale rispetto alle modifiche di parametri relativi ad uno solo di essi.

### Orizzonte di predizione

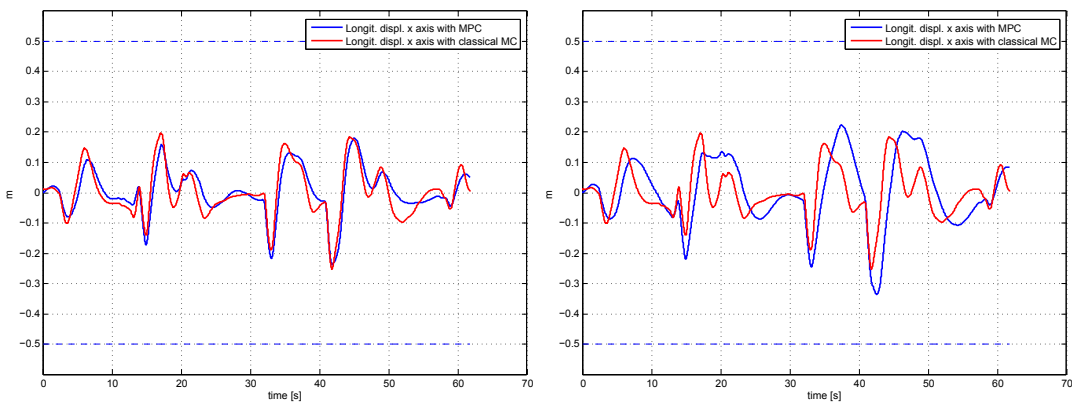
L'orizzonte di predizione è un altro parametro la cui modifica ha effetti globali, ma a differenza delle frequenze di taglio dei filtri agisce a livello del calcolo degli ingressi ottimi e non sul riferimento da inseguire che rimane dunque invariato. Le conseguenze della modifica di tale parametro sono quelli descritti in sezione 3.1.1 e in Figura 3.13 se ne dà esempio nel caso in cui esso venga aumentato rispetto al settaggio delle Figure 3.2-3.3-3.4. Si nota come l'azione del longitudinale sia notevolmente ridimensionata pur mantenendo il profilo originale, mentre la posizione di *pitch*, pur mantenendosi mediamente più nei dintorni della posizione di riposo, accentua le azioni correttive in corrispondenza dei picchi di intensità (si faccia riferimento ad esempio a quanto accade intorno ai 15, 32-33 e 41-42 secondi). una possibile spiegazione del suddetto comportamento è che il controllore, sulla base di un insieme di informazioni future più ampio, redistribuisca l'azione di correzione dal movimento longitudinale a quello angolare, valutando che, per come si evolverà la traiettoria nel futuro, l'apporto della variazione di *pitch* sia sufficiente. Vale la pena di ribadire nuovamente che in questo frangente il controllore non opera necessariamente le scelte migliori in corrispondenza di un orizzonte di predizione più lungo, dato che l'utilizzo di un riferimento costante lungo tutta la durata del tale orizzonte, può rendere le informazioni aggiuntive fuorvianti.



(a) Accelerazione percepita, tuning originale (b) Accelerazione percepita, peso in posizione di *pitch* modificato



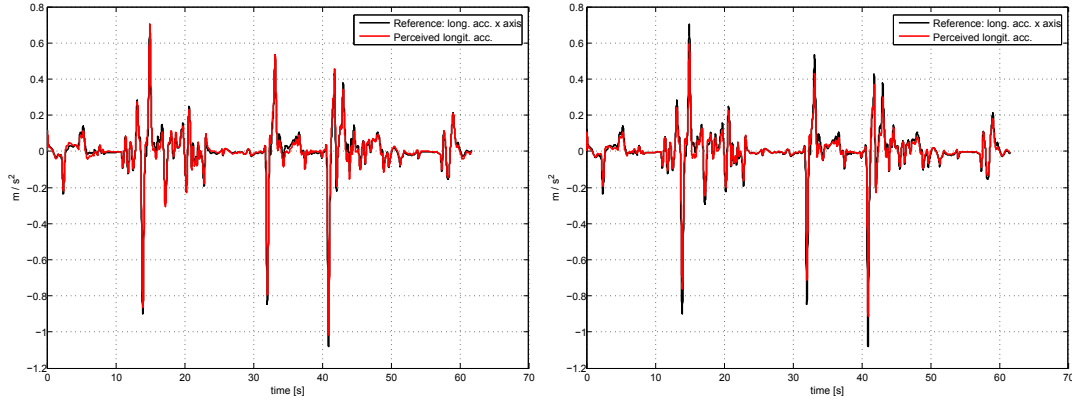
(c) Posizione di *pitch*, tuning originale (d) Posizione di *pitch*, peso in posizione di *pitch* modificato



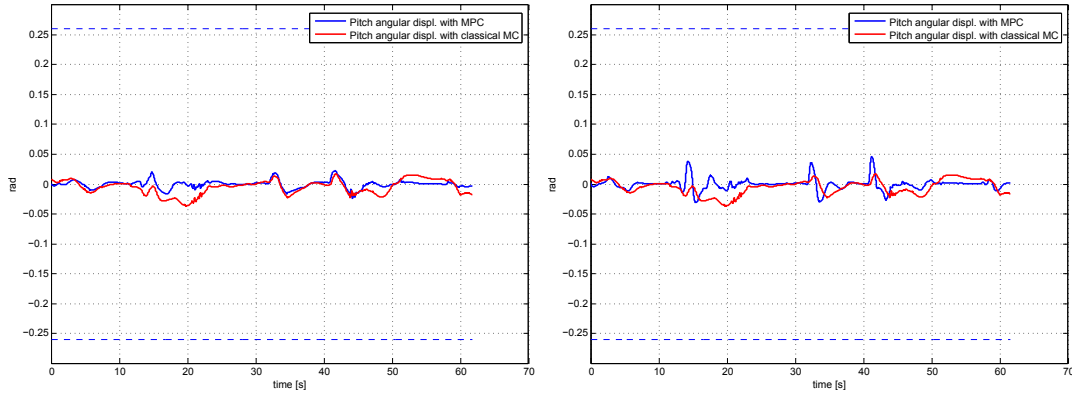
(e) Posizione longitudinale, tuning originale (f) Posizione longitudinale, peso in posizione di *pitch* modificato

**Figura 3.6:** Effetti dell'aumento del peso in posizione angolare di *pitch*

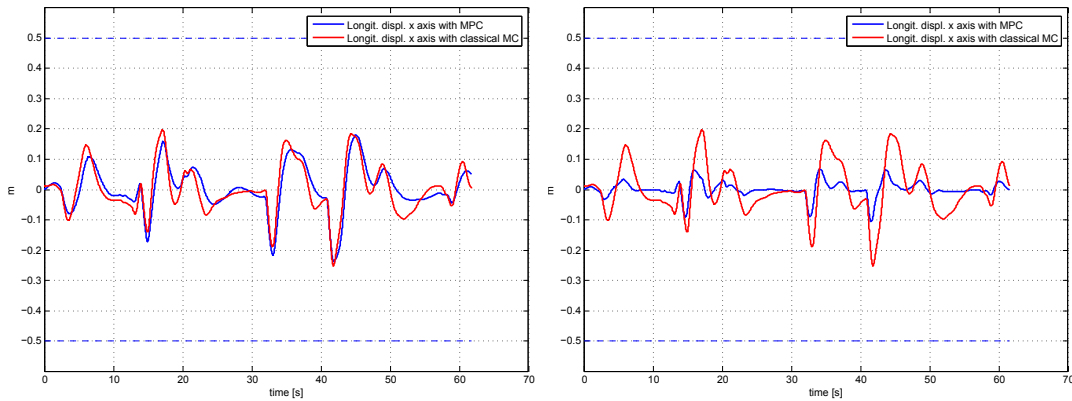
Riassumendo l'esperienza di *tuning* con la struttura di controllo MPC di Figuro



(a) Accelerazione percepita, tuning originale (b) Accelerazione percepita, orizzonte di predizione modificato



(c) Posizione di *pitch*, tuning originale (d) Posizione di *pitch*, orizzonte di predizione modificato



(e) Posizione longitudinale, tuning originale (f) Posizione longitudinale, orizzonte di predizione modificato

**Figura 3.7:** Effetti dell'aumento dell'orizzonte di predizione

ra 3.1 ha evidenziato come la variazione dei parametri sia abbastanza intuitiva (si pensi agli stessi pesi legati a singole grandezze specifiche) ma che al contempo essa

comporti una serie di effetti concatenati su diverse altre parti del sistema, rendendo di fatto limitato il potenziale di regolazione dei comportamenti della piattaforma. Il punto focale di tali limitazioni risiede nell'accoppiamento dei gradi di libertà longitudinale e angolare di *pitch*, che impedisce, di fatto, di poter regolare a piacimento e con precisione le dinamiche di un singolo grado di libertà senza influire negativamente sull'altro.

Per superare questa problematica nel presente lavoro di tesi si è sviluppato uno schema di controllo alternativo che mira ad eliminare, o quantomeno limitare, l'interdipendenza dei due gradi di libertà a livello di configurazione, pur mantenendola a livello di azione di controllo, dato che è su tale accoppiamento che si basa la fondamentale idea di *tilt coordination*.

## 3.2 *Tuning* con architettura di controllo *feed-forward*

Nella sezione precedente si è visto come l'accoppiamento delle componenti longitudinale e di *pitch* limitino i possibili comportamenti degli ingressi relativi, portando ad un utilizzo non ideale della piattaforma e rendendo a volte difficile il *tuning* dei parametri basato sul comportamento del *motion cueing* classico. Inoltre ciò rende virtualmente impossibile la configurazione indipendente di una particolare grandezza, come ad esempio la posizione angolare di *pitch* nel già discusso e centrale problema della riduzione della *tilt coordination*. Si è cercato dunque, in questo lavoro di tesi, di elaborare uno schema di controllo alternativo a quello naturale (perché basato sulla semplice architettura di contorno all'MPC) della sezione 3.1 che consentisse di superare queste problematiche.

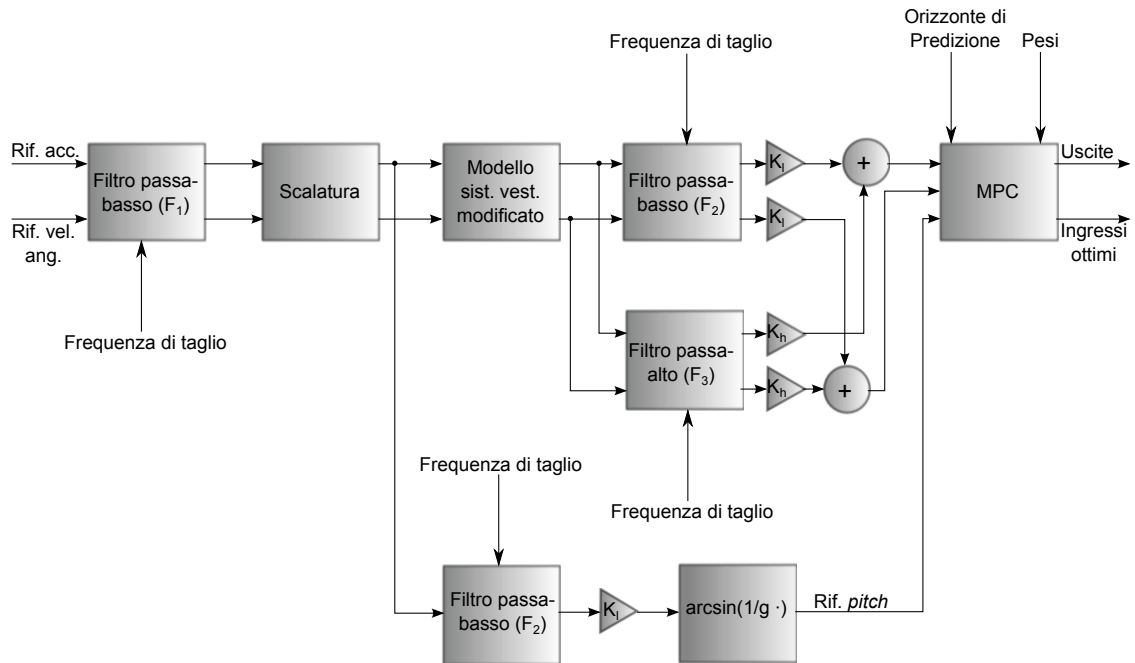
### 3.2.1 Schema di controllo con *feed-forward*

L'idea centrale dello schema alternativo che ci si appresta a descrivere è quello di forzare, attraverso un anello di *feed-forward*, il segnale di posizione angolare di *pitch* a seguire un riferimento non più nullo, come accadeva in precedenza, ma costruito appositamente per ottenere in uscita un soddisfacente inseguimento dell'accelerazione percepita, limitatamente ad alcune delle sue componenti frequenziali. Per la costruzione di un simile riferimento ci si è rifatti all'idea originale su cui si basava l'architettura di *motion cueing* classico [2]. Quest'ultima consisteva nel filtraggio del segnale di riferimento con filtri di caratteristiche opposte, uno passa-basso e uno passa-alto, di modo da separare in due sezioni lo spettro di componenti frequenziali del segnale. Lo scopo è quello di destinare il *tracking* delle componenti a bassa frequenza dell'accelerazione longitudinale al movimento angolare (*tilt coordination*) e lasciare quello delle frequenze più elevate alle traslazioni longitudinali.

La Figura 3.8 riporta lo schema completo dell'architettura di controllo adottata.

I segnali di riferimento di accelerazione longitudinale percepita e velocità angolare di *pitch* vengono sottoposti alla medesima fase preparatoria descritta in sezione





**Figura 3.8:** Schema di controllo con *feed-forward* per il *tuning*

3.1.2, prima attraverso un filtraggio passa-basso tramite il filtro  $F_1$  e successivamente tramite un'operazione di scalatura automatica. All'uscita dei primi due blocchi i segnali vengono rielaborati dal modello del sistema vestibolare (in versione modificata) di modo da ottenere in uscita le relative versioni percepite. A questo punto avviene la separazione delle componenti di bassa frequenza da quelle di alta frequenza tramite i filtri passa-basso  $F_2$  e passa-alto  $F_3$ , ricombinate successivamente dopo essere state opportunamente pesate tramite i guadagni di bassa frequenza  $K_l$  e di alta frequenza  $K_h$ . Infine ai riferimenti così ricostruiti viene aggiunto un riferimento per la posizione angolare di *pitch* ottenuto a partire dal segnale di riferimento di accelerazione longitudinale prelevato prima del blocco che modella il sistema percettivo.

Si analizza ora nel dettaglio lo schema evidenziando in particolare le nuove modalità di variazione dei parametri a disposizione.

### Modello del sistema vestibolare modificato

Il blocco ha il compito di convertire i segnali di riferimento provenienti da telemetria (previa opportuni filtraggio e scalatura) nei corrispondenti segnali percepiti da un pilota sottoposto a tale riferimento.

Il modello del sistema vestibolare adottato è strutturalmente lo stesso utilizzato nell'algoritmo MPC ma in questa versione subisce una modifica sottile ma fondamentale, ovvero la rimozione dell'accoppiamento dei due gradi di libertà, ottenuta ponendo a zero l'accelerazione di gravità  $g$ . Questo accorgimento si rivela

indispensabile per far sì che i segnali percepiti in uscita dal blocco che modella il sistema vestibolare siano del tutto indipendenti e possano essere così filtrati al passo successivo con la stessa filosofia progettuale che veniva adottata nel *motion cueing* classico. Quel che si sta facendo a questo livello infatti è una semplice elaborazione dei segnali e sarebbe errato introdurre qui un accoppiamento tipico invece della fase di controllo puro che viene svolta invece all'interno dell'algoritmo MPC.

### Filtri $F_2$ e $F_3$

All'atto di separare le componenti frequenziali vengono utilizzati due filtri di tipo butterworth, tendenzialmente aventi la medesima frequenza di taglio in modo da non avere sovrapposizioni o perdita di frequenze in fase di ricomposizione del segnale. Tale ricomposizione avviene semplicemente sommando i due differenti contributi tramite i due sommatori presenti (uno per il segnale di accelerazione longitudinale, l'altro per quello di velocità angolare).

I parametri regolabili a questo livello sono costituiti dalla frequenza di taglio dei filtri, che suddivide lo spettro di frequenze, e dai guadagni  $K_l$  e  $K_h$  che permettono di limitare o esaltare l'uno o l'altro insieme di frequenze venutisi a creare a seguito del filtraggio. Ad esempio impostando una frequenza di taglio molto bassa e scegliendo  $K_l < K_h$  si andranno a penalizzare le frequenze più basse del segnale di riferimento, spesso causa di quegli andamenti persistenti che portano alla saturazione degli attuatori, ottenendo una sorta di quell'effetto di *wash-out* già visto nel *motion cueing* classico, ma senza perdere del tutto le componenti a bassa frequenza come avveniva con tale approccio di controllo, dato che queste ultime vengono solo scalate di un fattore  $K_l$ .

### Anello di *feed-forward*

Il punto cruciale dell'architettura di controllo in esame è costituito dal segnale di riferimento di posizione di *pitch* che viene aggiunto ai classici riferimenti di accelerazione e velocità angolare. Esso si costruisce prelevando il segnale di riferimento di accelerazione longitudinale applicata (e non percepita) e filtrandolo con il filtro  $F_2$ , lo stesso già visto in precedenza, in modo da ricavarne le componenti a bassa frequenza. A questo punto si richiama la relazione

$$\begin{aligned} a_{tot} &= a_{trasl} + a_{ang} \\ &= a_{trasl} + g \sin(\theta) \end{aligned} \quad (3.1)$$

che quantifica l'azione di *tilt coordination* esprimendo l'accelerazione totale percepita dal pilota come somma della componente dovuta al movimento longitudinale della piattaforma ( $a_{trasl}$ ) e di quella fornita dalla rotazione dell'angolo  $\theta$  di *pitch*. Dalla (3.1), tralasciando la componente  $a_{trasl}$  dato che si sta ora trattando la componente a bassa frequenza del segnale di accelerazione e che si vuole che esso venga

inseguito esclusivamente tramite *tilt coordination*, si ricava facilmente il segnale corrispondente alla posizione di *pitch* come

$$\theta = \arcsin\left(\frac{1}{g} a_{low}\right) \quad (3.2)$$

dove si è ribattezzato con  $a_{low}$  il segnale di accelerazione longitudinale filtrato dal filtro  $F_2$ .

Si ponga attenzione al fatto che il riferimento in posizione di *pitch* così ottenuto è sostanzialmente diverso dai riferimenti di accelerazione e velocità angolare. Questi ultimi costituiscono le uscite che si vogliono ottenere attraverso l'applicazione di un ingresso adeguato, mentre il riferimento in posizione angolare costituisce già un segnale di ingresso che si vuole replicare il meglio possibile al fine di ottenere un inseguimento di accelerazione soddisfacente. Ne consegue che la posizione angolare viene resa molto più indipendente dalle dinamiche interne al controllore MPC in quanto il suo andamento viene stabilito sulla base di un ingresso già calcolato e riportato in avanti come riferimento al controllore (*feed-forward*). Lo stesso parametro di peso in posizione angolare già incontrato nella sezione precedente detterà ora la bontà del *matching* del segnale di posizione angolare di *pitch* reale col riferimento fornitogli dall'anello di *feed-forward*, e non più la predisposizione dello stesso a rimanere nell'intorno della posizione di riposo. È chiaro che a quel punto una volta determinato il segnale di riferimento di *feed-forward* tramite la scelta della frequenza di taglio del filtro  $F_2$ , la regolazione della dinamica del grado di libertà longitudinale potrà essere effettuata in maniera molto più indipendente facendo uso dei parametri già visti, in special modo i pesi delle matrici della funzione costo.

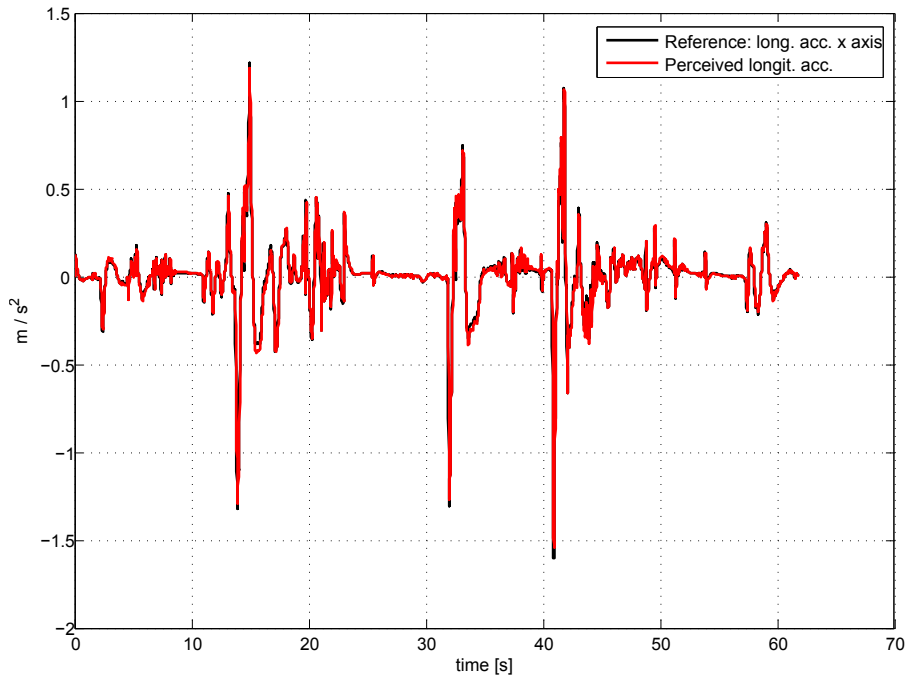
Si vuol mettere in evidenza come il fatto che il riferimento di posizione angolare costituisca concettualmente un ingresso di comando comporti che il segnale di riferimento di accelerazione longitudinale da cui esso viene ricavato venga prelevato a livello degli attuatori (accelerazione applicata alla piattaforma) e non a livello del sistema percettivo umano (accelerazione percepita); tale fatto è ben visibile nello schema di Figura 3.8, dove il segnale in questione viene prelevato prima del blocco che modella il sistema vestibolare.

### 3.2.2 Risultati delle simulazioni

Si riportano in questa sezione i grafici delle simulazioni effettuate con la struttura di controllo *feed-forward*

I primi tre grafici (Figura 3.9-3.10-3.11) illustrano i risultati del *tuning* ancora con riferimento al *motion cueing* classico. Si può notare come il segnale di accelerazione di Figura 3.9 sia molto diverso da quelli incontrati nelle sezioni precedenti, da cui peraltro deriva, a causa della separazione e successiva ricombinazione pesata delle frequenze secondo quanto descritto in sezione 3.2.1. Si noti poi come il segnale di posizione angolare di *pitch* segua discretamente l'andamento a bassa

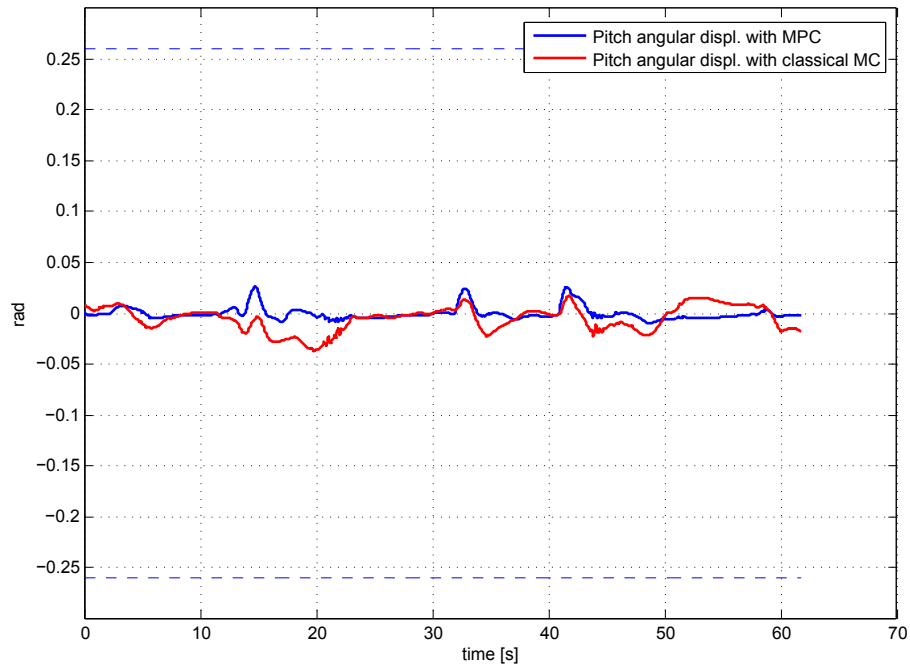
frequenza del corrispondente segnale relativo al *motion cueing* classico, pur non tralasciando le manifestazioni di alta frequenza del segnale stesso, come si può notare dall'ingrandimento in Figura 3.12.



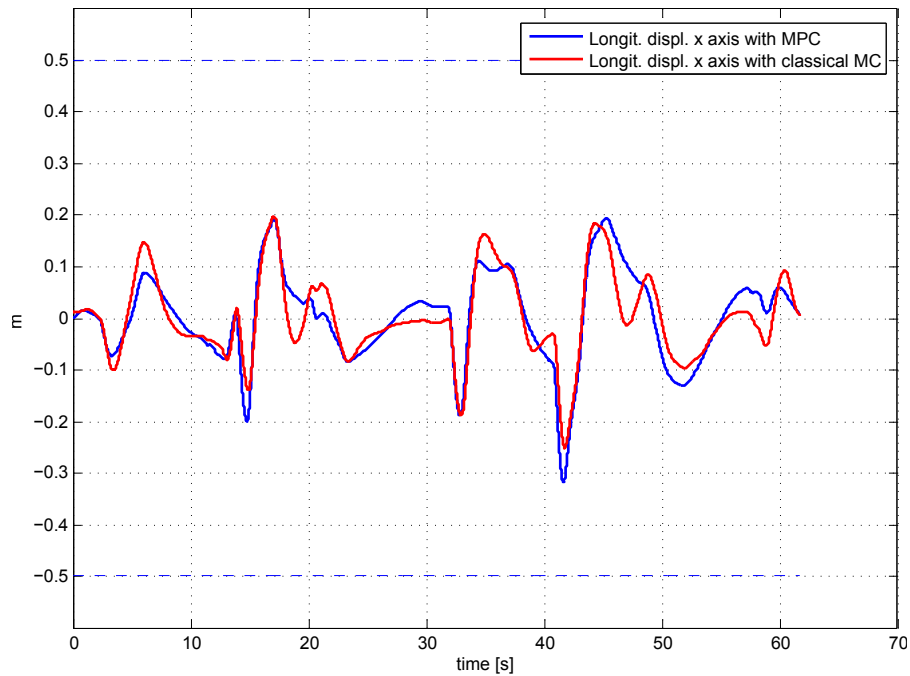
**Figura 3.9:** Tuning del controllore con struttura *feed-forward* rispetto al MC classico: inseguimento di accelerazione longitudinale percepita

In Figura 3.13(a) è rappresentato di nuovo un ingrandimento dei segnali di posizione di *pitch* di Figura 3.10 includendo anche il segnale di *feed-forward*. Si nota come il *pitch* in uscita dall'MPC segua l'andamento a bassa frequenza del riferimento angolare, aggiungendovi comportamenti a frequenze più elevate (di cui tale riferimento è sprovvisto, derivando dal filtraggio passa-basso del segnale di accelerazione applicata) che ricalcano quanto eseguito dal *motion cueing* classico. È importante evidenziare come il nuovo schema di controllo consenta di regolare l'accuratezza dell'inseguimento del riferimento di *feed-forward* semplicemente attraverso l'uso del parametro peso in posizione angolare di *pitch*. In Figura 3.13(b) si illustra il caso in cui tale peso venga consistentemente aumentato; come si vede il segnale di posizione di *pitch* si mantiene molto più aderente al riferimento riducendo di conseguenza anche l'ampiezza delle oscillazioni ad alta frequenza. In aggiunta la modifica parametrica non influisce sostanzialmente sull'andamento del segnale di posizione longitudinale (Figura 3.13(c)), se non marginalmente; ciò evidenzia la maggiore robustezza dei segnali relativi ad un grado di libertà rispetto a modifiche parametriche relative all'altro.

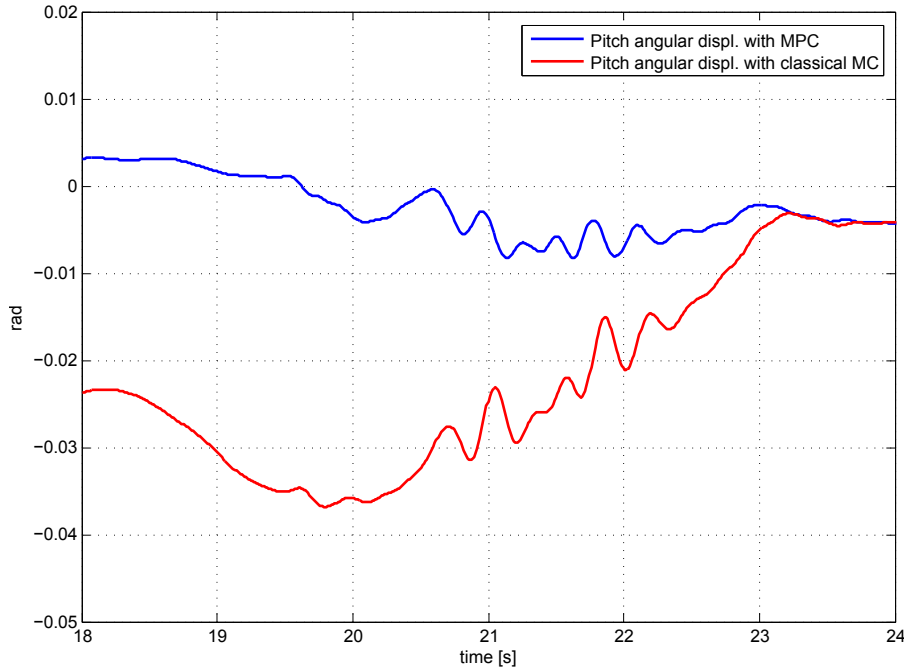
In buona sostanza col nuovo schema di controllo si è delineata la possibilità di operare il *tuning* del controllore in riferimento al *motion cueing* classico tramite la



**Figura 3.10:** Tuning del controllore con struttura *feed-forward* rispetto al MC classico: posizione angolare di *pitch*



**Figura 3.11:** Tuning del controllore con struttura *feed-forward* rispetto al MC classico: posizione longitudinale



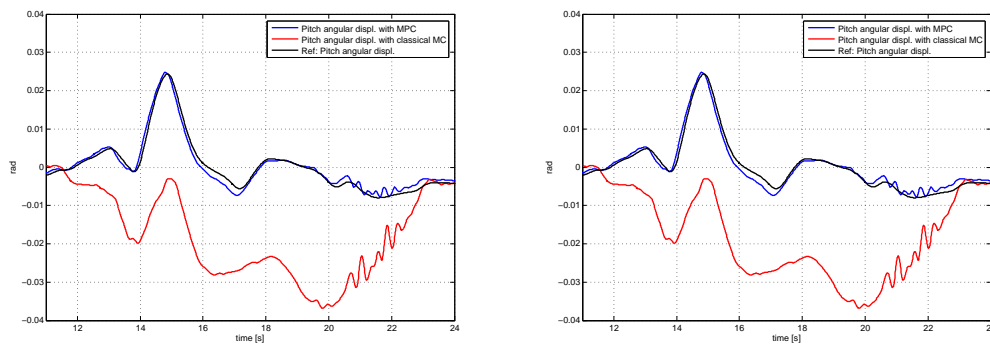
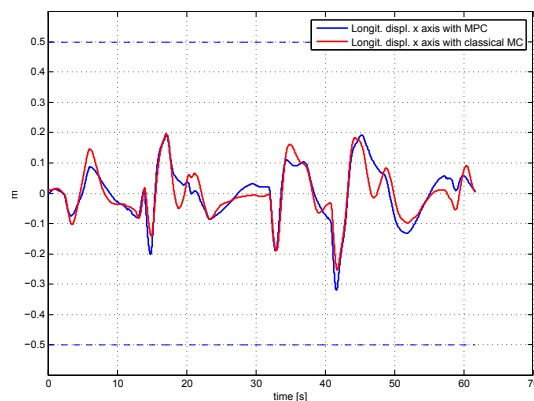
**Figura 3.12:** Tuning del controllore con struttura *feed-forward* rispetto al MC classico: posizione angolare di *pitch*, ingrandimento.

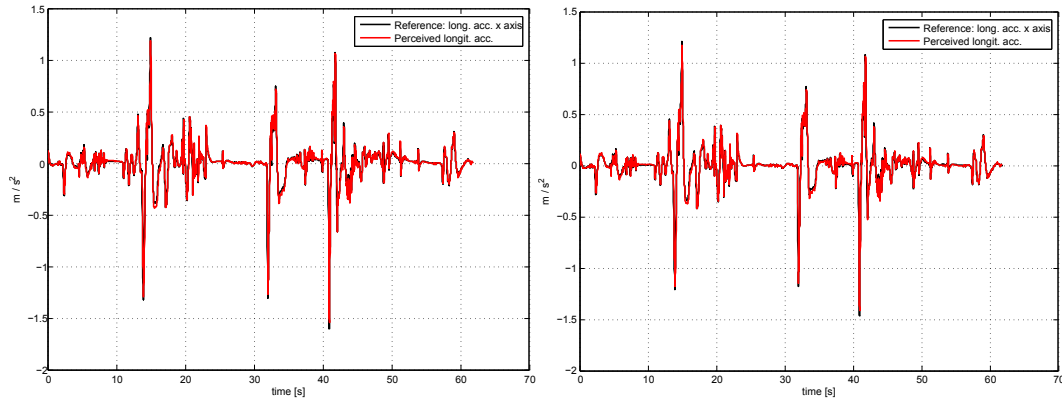
seguente procedura:

1. Si regola a grandi linee la dimensione dei segnali di posizione utilizzando i guadagni  $K_l$  e  $K_h$ ; in questa fase preliminare può risultare conveniente introdurre un ulteriore fattore moltiplicativo  $K_g$  comune ai due filtri  $F_2$  ed  $F_3$ , di modo da effettuare una semplice operazione di scalatura dei due segnali in uscita dai filtri stessi.
2. Si imposta la frequenza di taglio (comune) dei filtri  $F_2$  ed  $F_3$ , in modo da affidare alla posizione di *pitch* il grado di *tilt coordination* voluta in termini di gestione delle frequenze. Diminuendo la frequenza di taglio si rende più selettivo il filtro  $F_2$  consentendo alla posizione angolare di occuparsi solo delle frequenze più basse (e quindi più difficili da controllare col movimento longitudinale della piattaforma); viceversa le frequenze escluse verranno inglobate nel segnale in uscita dal filtro  $F_3$  ( $F_3$  diventa meno selettivo) diventando di competenza del movimento longitudinale. Agendo sul guadagno  $K_l$  separatamente è ora possibile regolare l'ampiezza del segnale di *feed-forward* e di conseguenza del segnale di posizione angolare di *pitch* (in misura dettata dal parametro di peso associato). Anche il guadagno  $K_h$  può risultare utile allo scopo, poichè esso rende più o meno significative le componenti ad alta frequenza, e di conseguenza facendo sovvenire la possibilità che il longitudinale

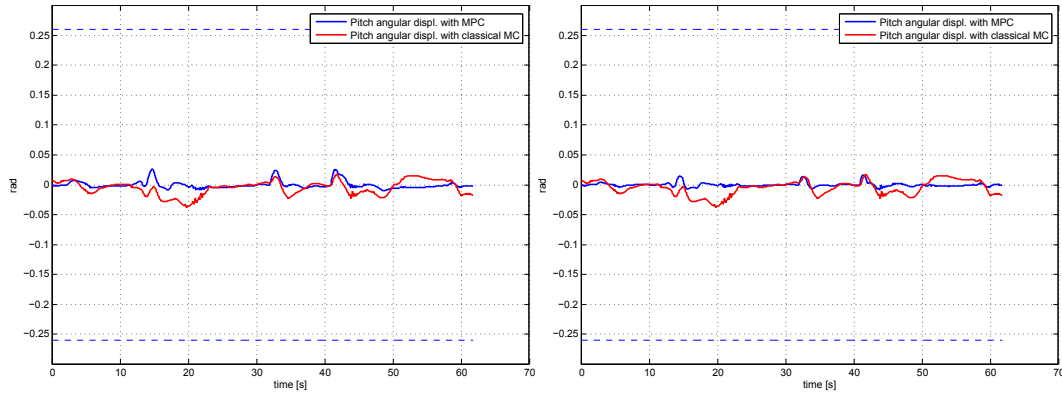
debba essere aiutato dalla *tilt coordination* nella gestione di tali componenti. Va detto però che tale effetto è molto limitato, confermando ancora una volta la robustezza dei segnali alle variazioni non desiderate.

- Una volta tarato il segnale di *pitch* si completa il *tuning* regolando il segnale di posizione longitudinale. I principali parametri su cui agire sono il già citato guadagno  $K_h$  (che agisce prevalentemente sull'ampiezza del segnale effettuando di fatto una scalatura del riferimento) e il peso in posizione longitudinale la cui variazione, rispetto al *tuning* effettuato in sezione 3.1, risulta praticamente ininfluenza sul segnale di posizione di *pitch*, soprattutto in corrispondenza di un peso in posizione angolare elevato, mentre consente di modificare pesantemente sia l'ampiezza che la forma del segnale di posizione longitudinale.

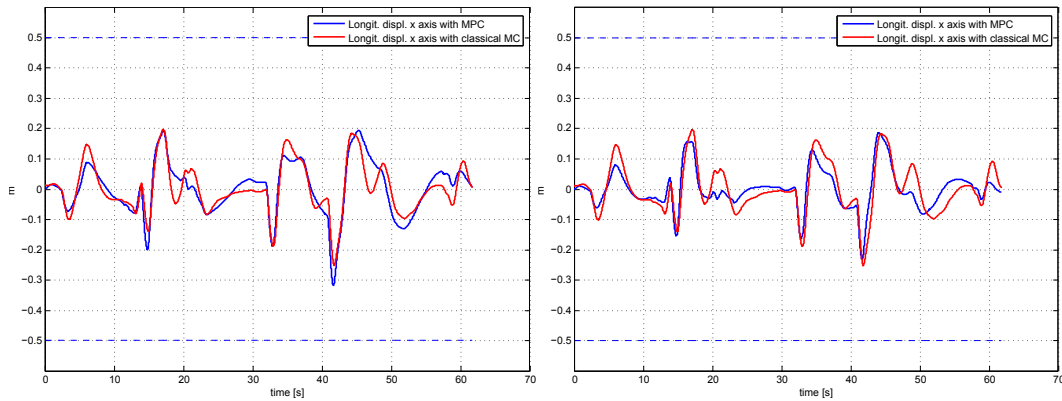
(a) Posizione di *Pitch* con riferimento(b) Aumento del peso in posizione di *pitch*(c) Posizione longitudinale, aumento del peso in posizione di *pitch***Figura 3.13:** Effetti dell'aumento del peso in posizione angolare di *pitch*



(a) Accelerazione percepita, tuning *feed-forward* (b) Accelerazione percepita, riduzione di *tilt coordination*



(c) Posizione di *pitch*, tuning *feed-forward* (d) Posizione di *pitch*, riduzione di *tilt coordination*



(e) Posizione longitudinale, tuning *feed-forward* (f) Posizione longitudinale, riduzione di *tilt coordination*

Figura 3.14: Riduzione di *tilt coordination*



### 3.2.3 Riduzione di *tilt coordination*

Uno dei tipici problemi introdotti in questo capitolo consiste nel ridurre l'apporto di *tilt coordination* senza modificare troppo l'andamento del segnale in posizione longitudinale della piattaforma.

Nella sezione 3.1.2 si era visto come tale richiesta fosse difficile, se non impossibile da assolvere a causa della stretta correlazione tra i due gradi di libertà considerati. Il nuovo schema invece consente l'azione separata sui singoli gradi di libertà, in particolare per la riduzione della *tilt coordination* è necessario diminuire l'intensità del segnale di posizione angolare. Ciò può essere fatto agendo semplicemente sul guadagno statico  $K_l$  del filtro passa-basso  $F_2$ , di modo da ridurre l'ampiezza del segnale di accelerazione in bassa frequenza, senza praticamente intervenire sul segnale di posizione longitudinale.

In Figura 3.14 sono esposti gli effetti di una riduzione consistente del parametro in questione. Come si può notare il segnale di posizione di *pitch* di Figura 3.14(d) appare attenuato, mentre non si riscontra praticamente alcun effetto sul segnale di posizione longitudinale (Figura 3.14(f)).

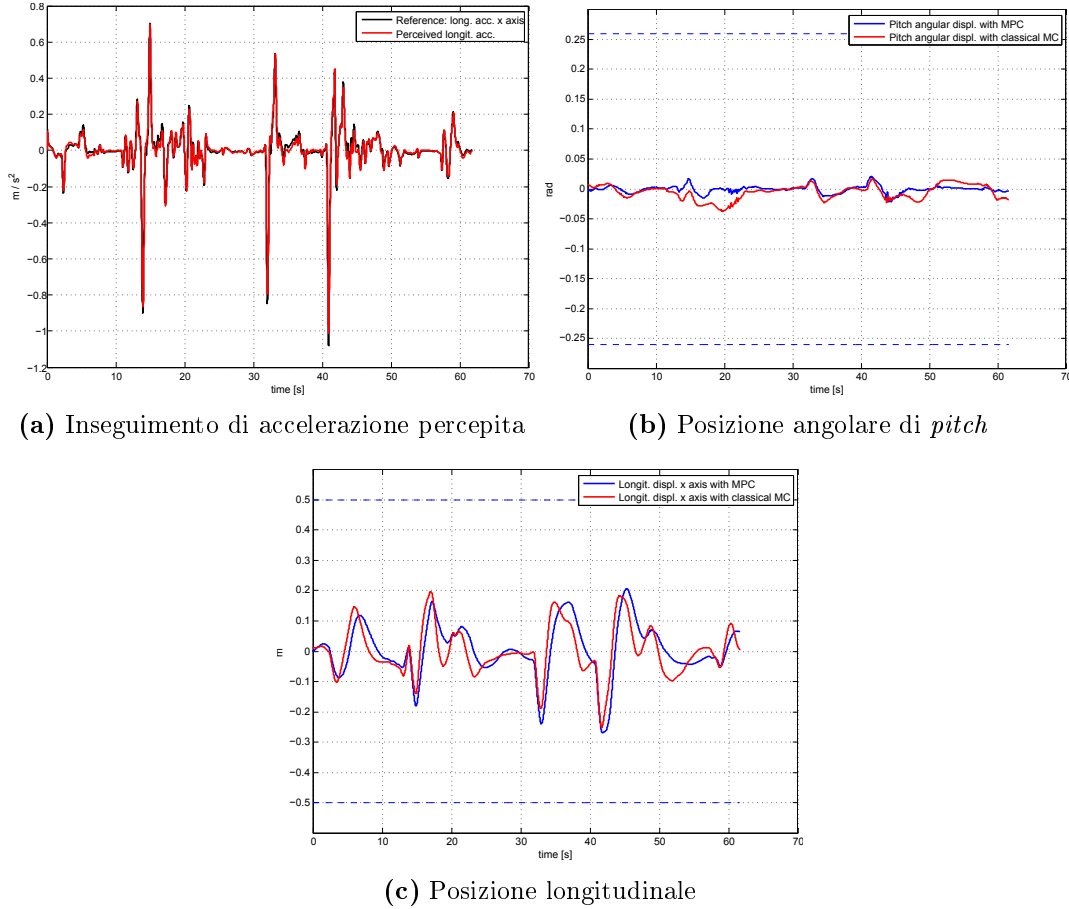
## 3.3 *Tuning* e funzioni di Laguerre

Nel capitolo precedente si erano analizzate le funzioni di Laguerre in applicazioni che prevedevano un approccio di controllo con strategia *look-ahead* rilevando dei difetti strutturali strettamente legati all'uso di tale strategia e che ne impedivano di fatto l'utilizzo. Si vuole ora testare il comportamento del controllore MPC con funzioni di Laguerre in applicazioni di *tuning*, che prevedono quindi l'assunzione di un riferimento costante in fase di predizione, evidenziandone eventuali vantaggi e svantaggi.

In Figura 3.15 sono riportati i consueti grafici di accelerazione percepita, posizione di *pitch* e longitudinale ottenuti integrando le funzioni di Laguerre nell'algoritmo MPC secondo le modalità descritte nel capitolo precedente.

Si può notare come il risultato raggiunto sia in linea con quanto ottenuto dal *tuning* originale.

Ciò che però realmente interessa è il vantaggio in termini di onere computazionale che l'uso delle funzioni di Laguerre può fornire. A questo proposito ricordiamo che come già spiegato in sezione 3.1.1, il tempo di predizione è un parametro che influisce naturalmente sulle dimensioni dei segnali di controllo e dunque rientra nel novero dei parametri di taratura necessari al *tuning*. Ora per raggiungere i risultati ottenuti dal *tuning* originale è stato necessario adottare un orizzonte di predizione pari a 0.2 s. Si riscontra sperimentalmente che per tali parametrizzazioni il vantaggio fornito dall'algoritmo con funzioni di Laguerre è di fatto inesistente. Per ottenere delle buone risposte dal sistema controllato dall'MPC modificato è necessario utilizzare un numero  $N$  di funzioni (e di conseguenza di coefficienti) pari ad almeno 20. Supponendo di utilizzare un orizzonte di predizione (coincidente con



**Figura 3.15:** *Tuning* del controllore con funzioni di Laguerre

quello di controllo)  $T_{pred} = 0.2$  s e ricordando che il periodo di campionamento adottato è  $T_s = 0.01$  s, nell’algoritmo originale l’ottimizzatore effettua il calcolo di  $N_{tot}$  valori con

$$N_{tot} = n_{in} \cdot N_c = 2 \cdot 20 = 40$$

dove si è indicato con  $n_{in}$  il numero di ingressi di controllo. Nel caso dell’algoritmo modificato con funzioni di Laguerre si ha

$$N_{tot} = n_{in} \cdot N = 2 \cdot 20 = 40$$

cioè la complessità computazionale a livello di ottimizzazione risulta essere esattamente la stessa. È chiaro che a questo punto la convenienza dell’utilizzo delle funzioni di Laguerre per l’implementazione del controllo MPC svanisce.

La tabella 3.2 certifica quanto affermato in via teorica riportando i tempi di esecuzione delle due simulazioni.

---

	Tempo di ottimizzazione
MPC originale, $T_{pred} = 0.2$	1.17 s
MPC Laguerre, $N = 20$	1.05 s

**Tabella 3.2:** Tempi di ottimizzazione dei due algoritmi per il *tuning*

Si noti come addirittura nel caso limite  $N_c = N$  l'uso delle funzioni di Laguerre risulti svantaggioso in termini di velocità di esecuzione, rendendolo di fatto sconsigliato.

In conclusione per la specifica applicazione del *tuning* del controllore MPC le funzioni di Laguerre non apportano i vantaggi computazionali sperati, rendendo il loro utilizzo superfluo. Tali vantaggi si rendono apprezzabili per orizzonti di predizione pari o superiori a 0.5 s (ci si rifaccia alla tabella 2.3) che non sono utilizzabili per il task che ci si è preposti.



# Conclusioni e sviluppi futuri

Ereditando l'impianto strutturale dell'algoritmo MPC, prima impostato da Daniele D'Ambrosio e successivamente completato da Mauro Baseggio e Mauro Pozzi, è stato possibile proseguire oltre l'ambito di sintesi del controllore dedicandosi ad aspetti quali il miglioramento della sua efficienza computazionale e la taratura dei parametri, due problematiche senza dubbio di elevato interesse per la prosecuzione dello sviluppo del simulatore VI-grade.

Si è sperimentato l'uso delle funzioni di Laguerre per la costruzione di una variante dell'algoritmo MPC in grado di svolgere il controllo in maniera computazionalmente meno dispendiosa, valutandone l'efficacia in situazioni in cui venivano adottate diverse strategie nel fare predizione.

Nell'ambito della strategia di controllo *look-ahead*, all'atteso vantaggio in termini computazionali ottenuto in alcuni casi ha fatto seguito la rilevazione di due fondamentali limitazioni all'utilizzo delle funzioni di Laguerre per l'MPC adottato.

Il primo limite, intrinseco nella struttura stessa delle funzioni di Laguerre, riguarda la formazione di fenomeni anomali nell'inseguimento del riferimento in presenza di brusche variazioni dello stesso. I comportamenti rilevati in queste situazioni non sono ritenuti accettabili ai fini della realizzazione di un buon *motion cueing*, ed essendo legati fondamentalmente all'andamento temporale delle funzioni di Laguerre, e più nello specifico alla loro convergenza a zero entro una finestra temporale piuttosto limitata, ne rendono di fatto impossibilitato l'utilizzo, specialmente in un ambito come quello del *look-ahead* in cui la lunghezza dell'orizzonte di predizione si rivela un parametro fondamentale per l'ottenimento di prestazioni elevate.

Il secondo limite incontrato è di natura completamente differente e consta nel rapido degrado delle prestazioni computazionali dell'algoritmo per orizzonti di predizione che superino i  $2\div 3$  s. Si è potuto verificare infatti che il problema, non preventivabile se si pensa che il problema di ottimizzazione è, per l'algoritmo che utilizza le funzioni di Laguerre, dimensionato dal numero di funzioni stesse usate per costruire gli ingressi di controllo, è legato alle dimensioni delle matrici dei vincoli associate al problema QP, indipendentemente dall'uso o meno delle funzioni di Laguerre. Diversamente dal problema delle anomalie, tale limite non è legato direttamente alla natura delle funzioni di Laguerre e non ne pregiudica dunque a priori l'utilità.

D'altro canto anche l'applicazione delle funzioni in oggetto in un ambito di tipo

*worst case* come quello dell'adozione di riferimento costante in predizione non ha apportato i benefici sperati. Se è vero che in presenza di riferimento costante in predizione si elimina il problema dei comportamenti anomali (in quanto per definizione non vi possono essere repentine variazioni del segnale di riferimento nella finestra di predizione), ci si è dovuti rendere conto che le particolari condizioni in cui ci si è dovuti porre per sperimentare il comportamento dell'algoritmo con funzioni di Laguerre hanno reso tale vantaggio di fatto superfluo. Come si è visto infatti per la pratica del *tuning*, che era il punto focale dell'uso dell'MPC in ambito *worst case*, si è fatto largo utilizzo di pratiche di *wash-out*, che constando in sostanza di un filtraggio passa-alto finisce per ridimensionare pesantemente il segnale di riferimento, eliminando di fatto le variazioni troppo marcate che compromettono il funzionamento dell'algoritmo con funzioni di Laguerre. Inoltre lo stesso *tuning* basato sul confronto col *motion cueing* classico ha richiesto l'adozione di finestre di predizione molto limitate, per le quali i vantaggi in termini di onere computazionale dell'algoritmo in questione diventano irrilevanti o addirittura inesistenti.

Alla luce di queste considerazioni si può concludere che le funzioni di Laguerre si sono rivelate poco adatte per il fine di apportare dei reali vantaggi in termini di miglioramento dei tempi di elaborazione, tuttavia l'idea base su cui si poggia il loro utilizzo, ovvero la riscrittura degli ingressi di controllo come loro combinazione lineare sui quali effettuare l'ottimizzazione, rimane valida, e non si esclude che nel futuro la loro sostituzione con un'altra famiglia di funzioni ortonormali, unitamente ad una costruzione più intelligente delle matrici dei vincoli, possano riaprire la strada con esse intrapresa.

Il lavoro svolto sul *tuning* ha invece chiarito alcuni aspetti comportamentali del controllore MPC, spesso intuitivi e preventivabili ma che necessitavano di una conferma sperimentale. Si è visto come già solo l'affrontare la taratura di un sistema ridotto come quello che modella l'accoppiamento longitudinale-*pitch* comporti non poche difficoltà a causa della stretta interdipendenza tra parametri relativi anche a gradi di libertà diversi.

Si è cercato di superare alcune di queste difficoltà proponendo un architettura di controllo alternativa a quella classica basata sul *wash-out*, che ha previsto l'uso di un segnale di *feed-forward* relativo alla posizione di *pitch* con l'obiettivo dichiarato di separare i due gradi di libertà a livello di settaggio dei parametri, mantenendone le positive interdipendenze al momento del *motion cueing*. Lo strumento si è rivelato egregio per il raggiungimento degli obiettivi prefissati, specialmente per la parziale risoluzione di un problema sempre molto presente come la riduzione dell'apporto di *tilt coordination*, e si candida pertanto a buon punto di partenza per l'auspicabile sviluppo di architetture più evolute che consentano di realizzare il fine ultimo di una pratica interfaccia che consenta ad utenti non esperti di poter regolare, con semplici azioni, ogni aspetto del comportamento della piattaforma commerciale VI-grade.

# Riferimenti bibliografici

- [1] Wang L., *Model predictive control system design and implementation using MATLAB*, Springer, 2009
- [2] d'Ambrosio D., *Algoritmi di motion cueing per simulatori di veicolo*, Tesi di Laurea Magistrale in Ingegneria dell'Automazione, Università di Padova, 2010
- [3] Baseggio M., *Generazione dei riferimenti per il Motion Cueing basato su MPC in simulatori dinamici di veicoli*, Tesi di Laurea Magistrale in Ingegneria dell'Automazione, Università di Padova, 2011
- [4] Pozzi M., *Motion Cueing per simulatori dinamici di veicolo: algoritmi MPC basati su modelli percettivi*, Tesi di Laurea Magistrale in Ingegneria dell'Automazione, Università di Padova, 2011
- [5] Augusto B. D. C. , Loureiro R. J. L., *Motion Cueing in the Chalmers Driving Simulator: A Model Predictive Control Approach*, Master of Science Thesis, Chalmers University of Technology, Göteborg, Sweden, 2009
- [6] Maciejowski J. M., *Predictive control with constraints*, Prentice Hall, 2002
- [7] Dagdelen M., Reymond G., Kemeny A., Bordier M., Maizi N., *Model - based predictive motion cueing strategy for vehicle driving simulators*, Control Engineering Practice 17 (2009) 995-1003
- [8] Bemporad A., Morari M., Ricker N. L., *Model Predictive Control Toolbox 3 User's Guide*, The MathWorks, Inc. 2009
- [9] Morari M., Ricker N. L., *Model Predictive Control Toolbox for use with MATLAB - User's guide*, Version 1, The MathWorks, Inc. 1998
- [10] Telban R. J., Cardullo F. M., *Motion cueing algorithm development: Initial investigation and redesign of the algorithms*, Technical report CR-2000-209863, Nasa, Marzo 2000
- [11] Telban R. J., Cardullo F. M., *Motion cueing algorithm development: Human-centered linear and nonlinear approaches*, Technical report CR-2005-213747, Nasa, Maggio 2005

- 
- [12] Zacharias, G.L., *Motion Cue Models for Pilot-Vehicle Analysis*. 1978, AMRL-TR-78-2, Department of Defense, Alexandria, VA.
- [13] Steinhausen W., *Observations of the Cupula in the Ampullae of the Semicircular Canals of the labyrinth of a Living Pike*, Pflueger's Arc. ges. Physiol., Vol. 232, pp. 500-512, 1933, NASA Technical Translation NASA TT F-13,665, May 1971.
- [14] Van Egmond A. A. J., Groen J. J. and Jongkees L. B. W., *The Mechanics of the Semicircular Canal*, J. Physiol., Vol. 110, pp. 1-17, 1949.
- [15] Melville Jones G., Barry W. and Kowalsky N., *Dynamics of the Semicircular Canals Compared in Yaw, Pitch and Roll*, Aerospace Med., Vol. 35, pp. 984-989, 1964.
- [16] Young L. R. and Oman C. M., *Model for Vestibular Adaptation to Horizontal Rotation*, Aerospace Med., Vol. 40, pp. 1076-1080, 1969.
- [17] Ormsby C. C., *Model of Human Dynamic Orientation*, Ph.D. Thesis, MIT, Cambridge, Mass., January 1974.
- [18] Peters R. A., *Dynamics of the Vestibular System and their Relation to Motion Perception, Spatial Disorientation, and Illusions*, NASA CR-1309, April 1969.
- [19] Steer R. W. Jr., *The Influence of Angular and Linear Acceleration and Thermal Stimulation on the Human Semicircular Canal*, Sc.D. Thesis, MIT, Cambridge, Mass., 1967.
- [20] Meiry J. L., *The Vestibular System and Human Dynamic Space Orientation*, Sc.D. Thesis, MIT, Cambridge, Mass., 1965.
- [21] Young L. R. and Meiry J. L., *A Revised Dynamic Otolith Model*, Aerospace Med., Vol. 39, pp. 606-608, 1968.
- [22] Reid L. D. and Nahon M.A., *Flight simulation motion-base drive algorithms: Part 1 - developing and testing the equations*. Technical Report ,UTIAS No.296, University of Toronto, 1985
- [23] Goldberg J.M., Fernandez C., *Physiology of Peripheral Neurons Innervating Semicircular Canals of the Squirrel Monkey. II. Response to Sinusoidal Stimulation and Dynamics of Peripheral Vestibular System*. Journal of Neurophysiology, 1971. 34(4): p. 661-675.
- [24] Fernandez C., Goldberg J. M., *Physiology of Peripheral Neurons Innervating Otolith Organs of the Squirrel Units, III. Response Dynamics*. Journal of Neurophysiology, 1976. 39(5): p. 996-1008.



- 
- [25] Hosman R.J., Van der Vaart J. C., *Vestibular Models and Thresholds of Motion Perception. Results of Tests in a Flight Simulator*. 1978, LR-265, Delft University of Technology, Department of Aerospace Engineering.
- [26] Hosman R.J., *Pilot's Perception and Control of Aircraft Motions*, 1996, Ph.D. Thesis, Delft University of Technology, Delft, The Netherlands
- [27] Sivan R., Ish-Shalom J., Huang J-K., *An optimal control approach to the design of moving flight simulators*. IEEE Transactions on Systems, Man and Cybernetics, Vol. SMC-12, No. 6, Nov./Dec. 1982, pp. 818-827
- [28] <http://www.vi-grade.com/>
- [29] <http://www.kuleuven.be/optec/software/qpOASES>
- [30] <http://www.unmc.edu/physiology/Mann/mann9.html>
- [31] <http://bestpractice.bmj.com/best-practice/monograph/73/basics/pathophysiology.html>
- [32] <http://www.medicine.mcgill.ca/physio/cullenlab/Notes209.htm>
- [33] <http://www.silverstone.co.uk>
- [34] <http://www.nogripracing.com/forum/showthread.php?t=194569>
- [35] <http://www.telegraph.co.uk/sport/motorsport/formulaone/michael-schumacher/8282461/Michael-Schumacher-suffers-from-motion-sickness.html>
- [36] <http://f1grandprix.motorionline.com/mercedes-conferma-schumacher-soffre-di-nausea-quando-e-al-simulatore/>

