Dipartimento di Ingegneria Industriale

Corso di Laurea in
Ingegneria Industriale orientamento Meccatronica

Elaborato finale

Analisi di una logica di controllo per sospensioni semi-attive magnetoreologiche tramite ottimizzazione mediante esplorazione esaustiva

Relatore
Prof. Ing. Panzani Giulio
Correlatore
Prof. Ing. Zaccarian Luca

Laureando
Serafini Daniele

Anno accademico 2020/2021

Ringraziamenti

...Ringrazio me stesso, per aver capito di non essere un automa...

Indice

Sommario

In questo elaborato si vuole andare ad approfondire, attraverso la raccolta dati effettuata tramite simulazioni, l'efficacia della tecnica di controllo per sospensioni semi-attive proposta da Ruben Begnis, Giulio Panzani, Mirko Brentari e Luca Zaccarian nel paper intitolato "An LMI-based approach for the control of semi-active magnetorheological suspensions" [2].

Il metodo di controllo mira ad ottimizzare il comfort di guida provato dai passeggeri di un SUV generico tramite l'analisi di sospensioni semi-attive basate su fluidi magnetoreologici, che presentano la caratteristica di modificare la loro viscosità al variare di un campo magnetico applicato su di essi: la variazione di viscosità permette un'applicazione variabile delle forze di ammortizzazione necessarie per attutire le dinamiche verticali del veicolo, aumentando così il comfort di guida percepito dai passeggeri.

Preso in analisi un modello quarter-car e considerato un profilo generalizzato di superficie stradale, si vuole andare ad analizzare i parametri interni del modello di controllo che ottimizzano il comfort di guida ottenuti nel paper sopracitato, confrontandoli con i dati raccolti da un numero elevato di simulazioni effettuate in modalità passiva, successivamente in modalità semi-attiva ed infine imponendo ulteriori regole sulle simulazioni semi-attive: si dimostra così che i parametri che ottimizzano il problema trovati precedentemente da Ruben Begnis, Giulio Panzani, Mirko Brentari e Luca Zaccarian sono effettivamente parametri ottimali, cioè al variare di essi il comfort di guida non migliora, bensì converge al comfort ottenuto dalla loro analisi.

Inizialmente viene affrontata un'introduzione preliminare, cioè una sezione di elaborato che andrà ad analizzare il profilo stradale, il modello quarter-car, in particolare sospensioni magnetoreologiche, il modello matematico ed il modello di controllo utilizzato nelle analisi e successivamente vengono introdotte le definizioni delle cifre di costo e delle funzioni di trasferimento. Nei capitoli successivi viene presentata l'analisi dei dati raccolti, inizialmente considerando le simulazioni passive e le simulazioni semi-attive utilizzando i parametri indicati nel paper sopra indicato [2], successivamente considerando le altre due tipologie di simulazioni semi-attive.

Lo sviluppo dei dati necessari è avvenuto in ambiente *Matlab*® prima con la creazione di un circuito di controllo in ambiente *Simulink*®, successivamente con la scrittura di codice congruo all'analisi e alla rappresentazione dei dati raccolti. Questo mi ha permesso di conoscere meglio alcune delle tecniche usate per implementare la teoria del controllo a livello computazionale e soprattutto utilizzare alcune delle funzioni interne a *Matlab*® che permettono di effettuare un numero elevato di simulazioni in modo efficace e conciso.

L'analisi dati che supporta questa tesi è stata effettuata tramite lo sviluppo di funzioni di trasferimento per andare effettivamente a controllare le componenti di frequenza prodotte dai sistemi dinamici e successivamente dall'analisi dei parametri interni al modello di controllo, notando che nonostante ci sia ancora margine di variazione per i parametri che ottimizzano il problema, il comfort di guida ottimo non varia, permettendo la deduzione che il modello di controllo raggiunge una soglia ottima.

1 Preliminari

In questo capitolo viene fatta un'introduzione a tutte le nozioni necessarie alla comprensione del problema preso in analisi, che consiste nell'ottimizzazione del comfort di guida percepito dai passeggeri di un veicolo. Come noto, quando si è alla guida di un veicolo l'autista ed i passeggeri possono percepire le asperità del manto stradale sotto forma di vibrazioni che vengono trasmesse fino all'abitacolo, influenzando il comfort di guida provato durante il viaggio in automobile. Per affrontare questo problema viene utilizzata la teoria dei sistemi dinamici, dove è necessario definire quali siano gli ingressi del sistema, quali siano le uscite ed infine quale sia il modello matematico che lega questi due elementi del sistema in analisi. Le tre parti appena indicate vengono analizzate in maniera separata e classificate nel seguente modo: l'ingresso del sistema viene trattato introducendo la generazione del *profilo stradale* sul quale il veicolo percorre il tragitto ad una determinata velocità, la trattazione del modello matematico viene effettuata tramite la definizione del modello *quarter-car*, che permette di determinare le dinamiche verticali che la massa dello pneumatico e la massa del corpo macchina subiscono durante il tragitto a causa delle asperità del manto stradale, e tramite la definizione del *modello di controllo* che consiste in un sistema di equazioni che permettano appunto il controllo dell'evoluzione nel tempo del sistema in analisi, mentre il segnale in uscita viene trattato tramite la definizione dei *strumenti di analisi* utilizzati per comprendere l'evoluzione del sistema, ottenuta tramite il modello di controllo.

1.1 Modellistica

In questa sezione viene trattata inizialmente la generazione del profilo stradale, successivamente verrà trattato il modello quarter-car introducendo anche le *sospensioni semi-attive a tecnologia magnetoreologica* utilizzate nell'analisi del problema ed infine verrà trattato il modello di controllo ed i parametri di controllo legati alle sospensioni semi-attive.

1.1.1 Profilo Stradale

Il profilo stradale, indicato nell'elaborato come $\mathbf{z_r}$, è l'elemento iniziale delle analisi del problema in quanto esso contiene tutte le informazioni riguardanti le perturbazioni che vanno ad influire sul comfort di guida provato dai passeggeri ed è dunque necessario generare un profilo stradale le cui asperità siano molto simili, se non uguali, a quelle presenti nella realtà. L'approccio per la generazione del profilo stradale segue dunque le normative standard indicate nell'ISO-8680 presentate in Agnostinacchio et al. (2014). [1].

Il profilo stradale viene generato, come si può vedere nella figura 1.1, prendendo inizialmente in considerazione un *rumore Gaussiano* che viene filtrato tramite la funzione di trasferimento $G_r(s)$, i cui parametri dipendono dalla velocità con cui la strada viene percorsa, dalla risoluzione spaziale della strada e dalla classe stradale considerata nell'analisi.

Figura 1.1: Schema Simulink® utilizzato per la generazione del profilo stradale $\mathbf{z_r}$

Matematicamente la funzione di trasferimento viene presentata tramite la seguente equazione

$$G_{\mathrm{r}}(s) = \frac{k_{\mathrm{r}} s}{s^2 + 2\xi_{\mathrm{r}}\omega_{\mathrm{r}} s + \omega_{\mathrm{r}}^2} \tag{1.1}$$

dove il parametro $\boldsymbol{\xi_{\mathrm{r}}}$ è costante ($\xi_{\mathrm{r}} = 0.7$) ed il parametro $\boldsymbol{\omega_{\mathrm{r}}}$ dipende dalla velocità con cui la strada viene percorsa e dalla risoluzione spaziale considerata

$$\omega_{\mathrm{r}} = \frac{2\pi v}{l_{\mathrm{c}}} \tag{1.2}$$

dove $\boldsymbol{v}$ corrisponde alla *velocità di percorrenza* mentre il parametro $\boldsymbol{l_{\mathrm{c}}}$ corrisponde alla *risoluzione spaziale* presa in considerazione nell'analizzare il problema proposto: per valori grandi di questo parametro vengono introdotte nel profilo stradale delle variazioni a bassa frequenza, che corrispondono nella realtà a colline presenti lungo il tragitto, mentre per valori piccoli queste variazioni non vengono introdotte. Nelle analisi questo parametro è costante $l_{\mathrm{c}} = 200[m]$ mentre per le velocità vengono presi in considerazione tre diversi valori ($45[km/h] \quad 90[km/h] \quad 180[km/h]$).

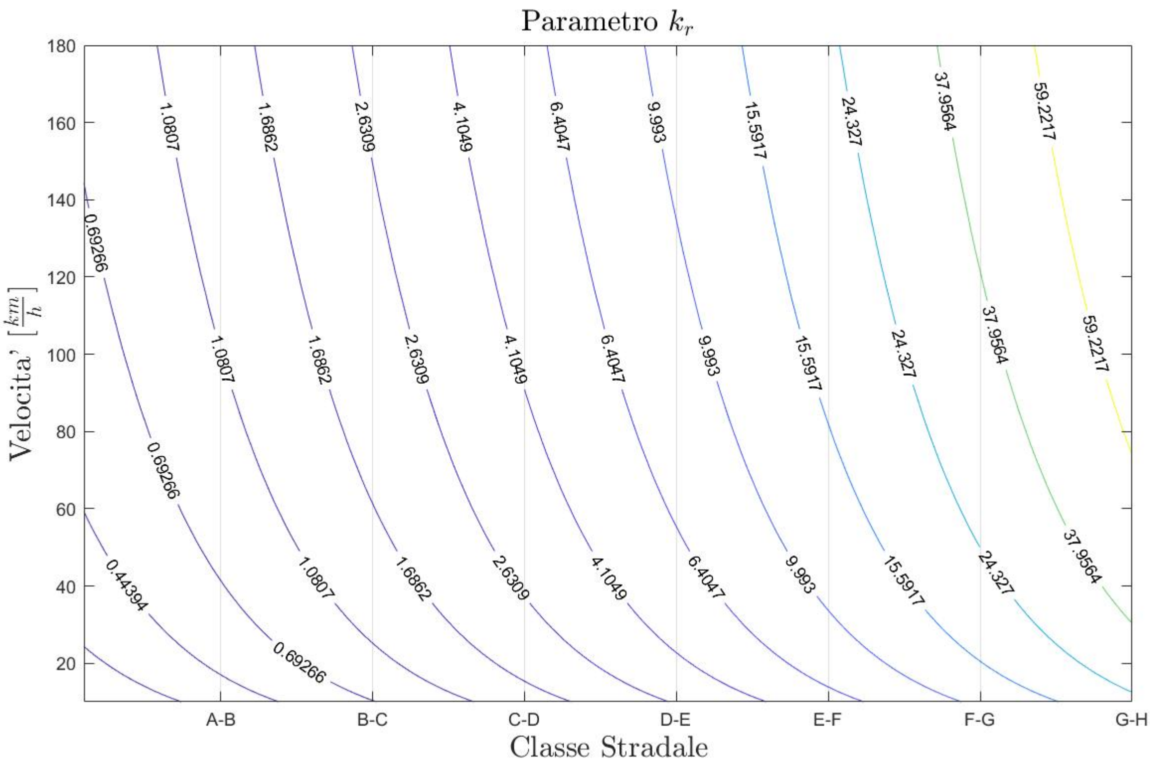


Figura 1.2: Parametro $\boldsymbol{k_{\mathrm{r}}}$ in funzione delle classi stradali e della velocità di percorrenza: le classi A-B rappresentano strade con minime asperità mentre classi G-H strade con massime asperità

Il parametro $\boldsymbol{k_r}$ a nominatore, come si può vedere in figura 1.2, rappresenta il grado di degradazione del manto stradale (quanto l'asfalto della strada è rovinato dall'usura) al variare della velocità di percorrenza considerata: per le analisi effettuate è stata considerata una classe stradale C-D, che corrisponde a delle leggere condizioni di strada sterrata come espresso in Agnostinacchio et al. (2014) [1].

Per i tre valori di velocità $\boldsymbol{v}$ considerati nell'analisi vengono prodotti i profili stradali mostrati in figura 1.3, dove si può notare come le irregolarità del manto stradale, misurate in cm, sono paragonabili per tutte e tre le velocità considerate, cioè al variare della velocità le asperità del manto stradale sono tra di esse simili, mentre la lunghezza totale del profilo stradale è sempre pari a $x = 2[km]$.

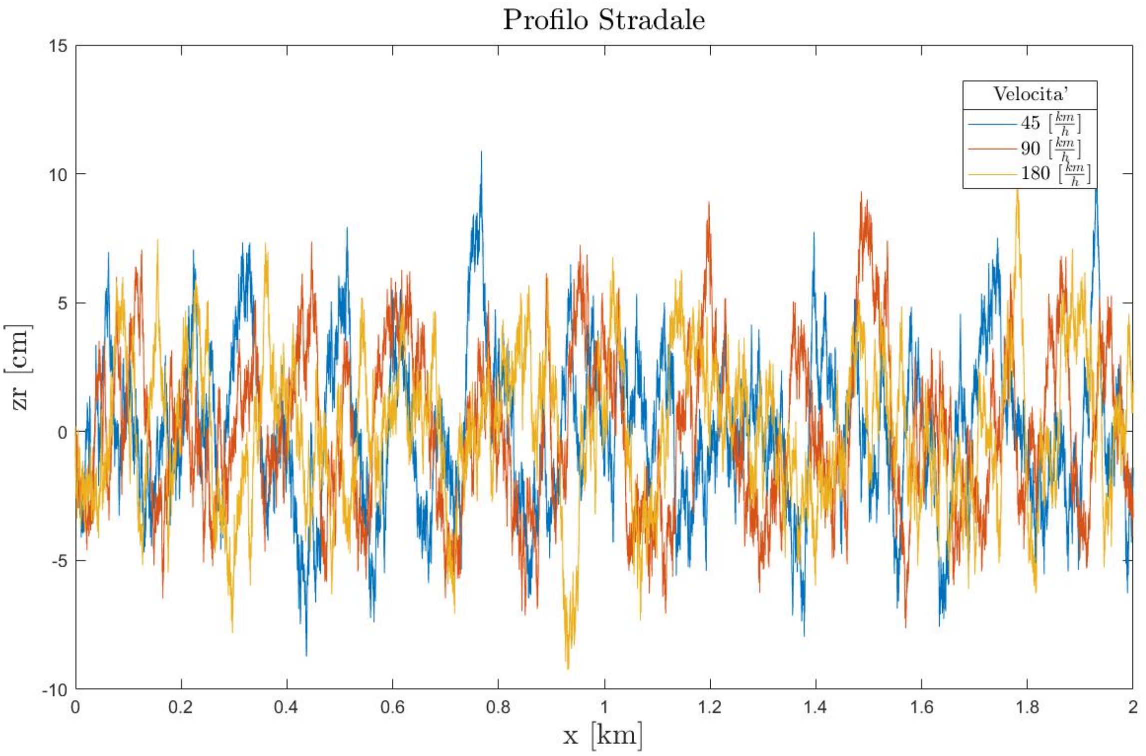


Figura 1.3: Profili stradali $\mathbf{z_r}$ generati tramite Matlab® utilizzati per l'analisi del problema proposto

La scelta di questa lunghezza è stata effettuata per permettere la convergenza della cifra di costo, che verrà introdotta in seguito, utilizzata nella valutazione delle prestazioni ottenute dall'analisi del problema, in modo che le simulazioni effettuate e i dati ottenuti siano consistenti e non dipendano dal particolare *rumore Gaussiano* considerato.

La figura 1.4 dimostra la decisione effettuata nella scelta della lunghezza del profilo stradale per analizzare il problema posto. Essa punta a mostrare come varia l'andamento della cifra di costo al variare della lunghezza di analisi per tre diversi seed iniziali considerati nel generare il *rumore Gaussiano*: come si può notare per lunghezze del profilo stradale inferiori a $x = 2000[m]$ la cifra di costo calcolata dal modello di controllo varia nell'intorno di un valore a cui tenderebbe se la lunghezza del profilo stradale fosse di molto superiore a $x = 2000[m]$. E' stata scelta $x = 2000[m]$ come lunghezza standard del profilo stradale in quanto da questa lunghezza in poi si può vedere che la cifra di costo converge verso un valore stabile, dimostrando che il valore ottenuto dal modello di controllo è un valore effettivamente utilizzabile ai fini delle analisi. Questo grafico è utile anche nel dimostrare come il *rumore Gaussiano* utilizzato nella generazione del profilo stradale non influisca nel calcolo della cifra di costo: per i tre diversi seed (numero matematico che il programma in Matlab® utilizza come base di generazione del rumore Gaussiano) e per analisi con lunghezza del profilo stradale superiore ai $x = 2000[m]$ i valori di cifra di costo sono pressoché costanti e molto simili tra loro.

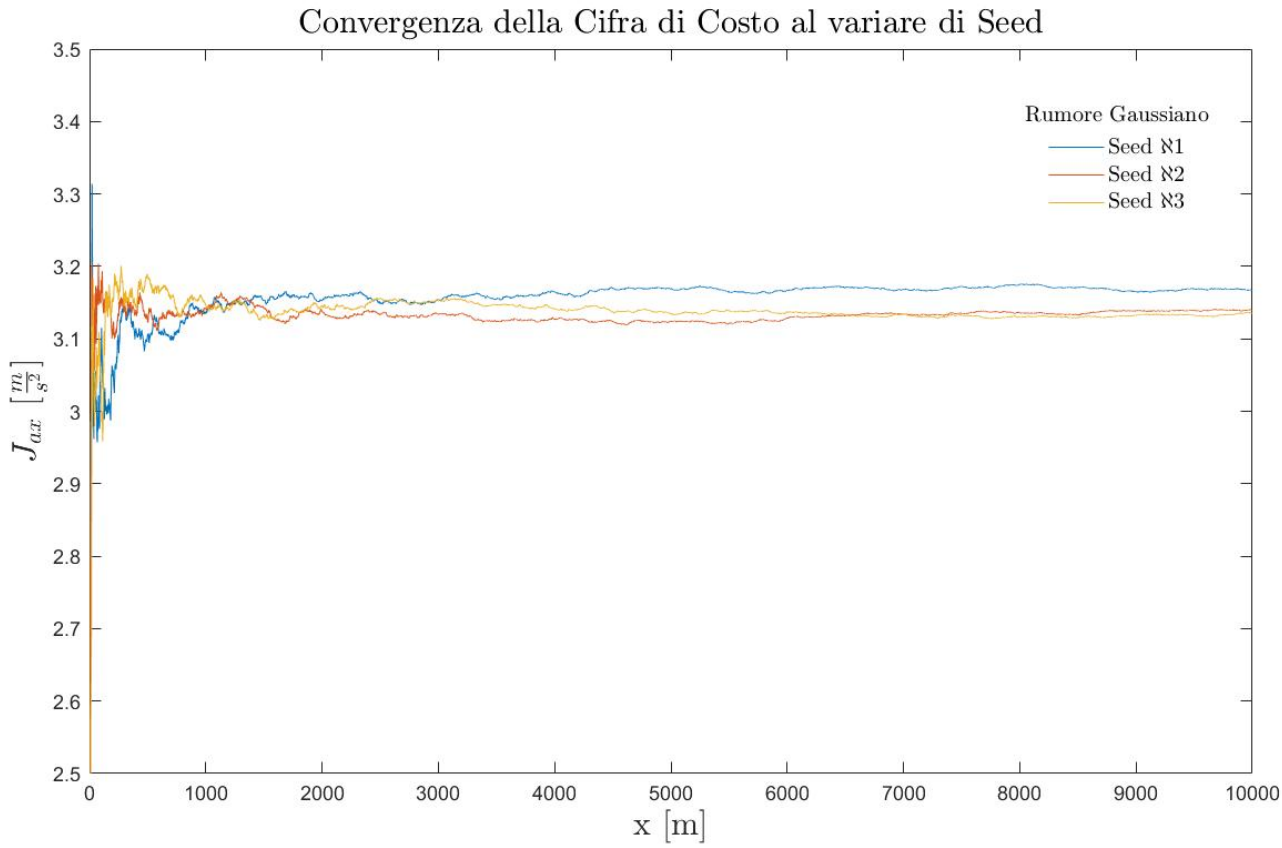


Figura 1.4: Andamento della cifra di costo $\mathbf{J_{ax}}$ al variare della lunghezza della strada considerata nell'analisi, per tre diversi valori di seed iniziale

Avendo determinato cos'è il profilo stradale e come viene modellizzato all'interno del nostro problema è possibile passare all'analisi della parte successiva, corrispondente al modello quarter-car, che possiede al suo interno le informazioni riguardanti l'ammortizzatore semi-attivo a tecnologia magnetoreologica preso in analisi e le masse sospese corrispondenti alla massa del pneumatico e alla massa dell'automobile.

1.1.2 Quarter Car

Il modello quarter-car è una delle parti più importanti all'interno del problema preso in analisi in quanto oltre a racchiudere tutti i parametri necessari per definire le parti in movimento esso contiene anche le equazioni matematiche necessarie per ottenere le posizioni spaziali delle parti medesime. In questa sezione vengono introdotti inizialmente lo schema quarter-car e successivamente vengono introdotte separatamente le sospensioni semi-attive ed il modello matematico che racchiude tutte le informazioni necessarie.

Lo schema quarter-car rappresentato in figura 1.5 indica uno dei modelli più utilizzati e consolidati per l'analisi delle dinamiche verticali (Savesi et al. (2010)) [4]. Nello schema vengono rappresentate la massa di un quarto di veicolo $\boldsymbol{m_s}$ che corrisponde alla massa sostenuta da uno dei quattro ammortizzatori del veicolo, l'ammortizzatore stesso schematizzato come un elemento elastico o molla $\boldsymbol{k}$ ed un elemento dissipativo o pistone $\boldsymbol{f_d(t)}$ in parallelo (le quali funzioni dunque sono additive) che collega la massa del veicolo alla massa di uno dei quattro pneumatici $\boldsymbol{m_u}$. Quest'ultima massa non risulta attutita dall'ammortizzatore, ma presenta un elemento elastico $\boldsymbol{k_t}$ che rappresenta gli effetti deformativi a cui può essere sottoposto il copertone dello pneumatico durante il moto del veicolo. Ad ognuno di questi parametri corrisponde una costante matematica che dipende dal veicolo preso in considerazione nell'analisi, tutte queste costanti vengono elencate nella tabella 1.1.

Nello schema quarter-car vengono anche rappresentate le posizioni verticali degli elementi elencati: $z_s(t)$ rappresenta la coordinata verticale dello chassis del veicolo, $z_u(t)$ rappresenta la coordinata verticale del centro del mozzo della ruota mentre $z_r(t)$ rappresenta la coordinata verticale del profilo stradale. Lo schema si conclude nella parte più bassa con la rappresentazione del profilo stradale stesso, su cui poggia tutto il modello quarter-car.

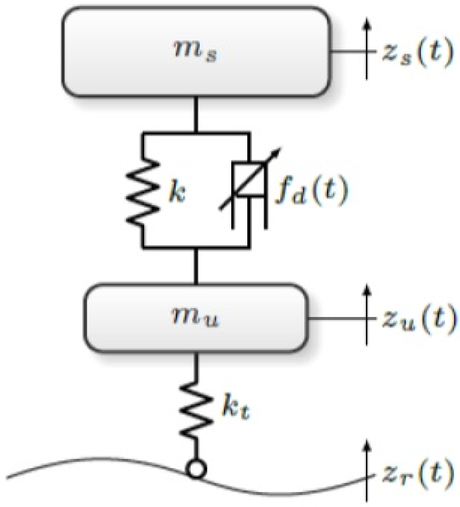


Figura 1.5: Schema del modello quarter-car comprendente la massa del veicolo $\mathbf{m_s}$ e la posizione verticale del veicolo $\mathbf{z_s(t)}$, i parametri dell'ammortizzatore $\mathbf{k}$ e $\mathbf{f_d(t)}$, la massa dello pneumatico $\mathbf{m_u}$ e la posizione dello pneumatico $\mathbf{z_u(t)}$, l'elemento elastico del copertone $\mathbf{k_t}$ e il profilo stradale $\mathbf{z_r(t)}$

Parametro	**Simbolo**	**Valore**
Massa Pneumatico	m_u	70 kg
Massa Chassis	m_s	450 kg
Rigidezza Sospensione	k	27000 $\frac{N}{m}$
Rigidezza Pneumatico	k_t	300000 $\frac{N}{m}$

Tabella 1.1: Parametri utilizzati nel modello quarter-car

Sospensioni Magnetoreologiche

Quando si vuole trattare di sospensioni utilizzate in ambito automobilistico è necessario in prima analisi fare una distinzione tra le tre diverse famiglie di ammortizzatori adottate: sospensioni passive, sospensioni semi-attive e sospensioni attive. Le sospensioni passive sono sospensioni la cui risposta dinamica non può essere modificata durante il moto del veicolo in quanto essa deve essere impostata al momento del montaggio, le sospensioni semi-attive invece sono sospensioni la cui risposta dinamica può essere modificata senza introdurre energia all'interno del sistema mentre le sospensioni attive sono sospensioni la cui risposta dinamica può essere variata inserendo energia all'interno del sistema. In questa tesi vengono analizzate delle sospensioni semi-attive basate sulla teoria dei fluidi magnetoreologici, che sottoposti ad un campo magnetico possono modificare la loro viscosità: questi fluidi hanno il vantaggio di avere una caratteristica non lineare che lega la velocità verticale e la forza esercitata quando deve essere appunto applicata una forza ammortizzante. In figura 1.6 vengono rappresentate le *curve forza-velocità* e la *non linearità associata* tipiche degli ammortizzatori magnetoreologici considerati al variare del parametro di controllo $\boldsymbol{u_{MR}}$, che rappresenta determinati livelli di corrente applicata: come si può notare, quando la corrente applicata è nulla gli ammortizzatori presentano una caratteristica lineare, mentre aumentando la corrente vengono aumentali i limiti massimi di forza applicata dagli ammortizzatori. E' importante anche notare che per qualsiasi parametro di controllo la pendenza dello "scalino" nella *curva forza-velocità* e nella *non linearità associata* è sempre la medesima.

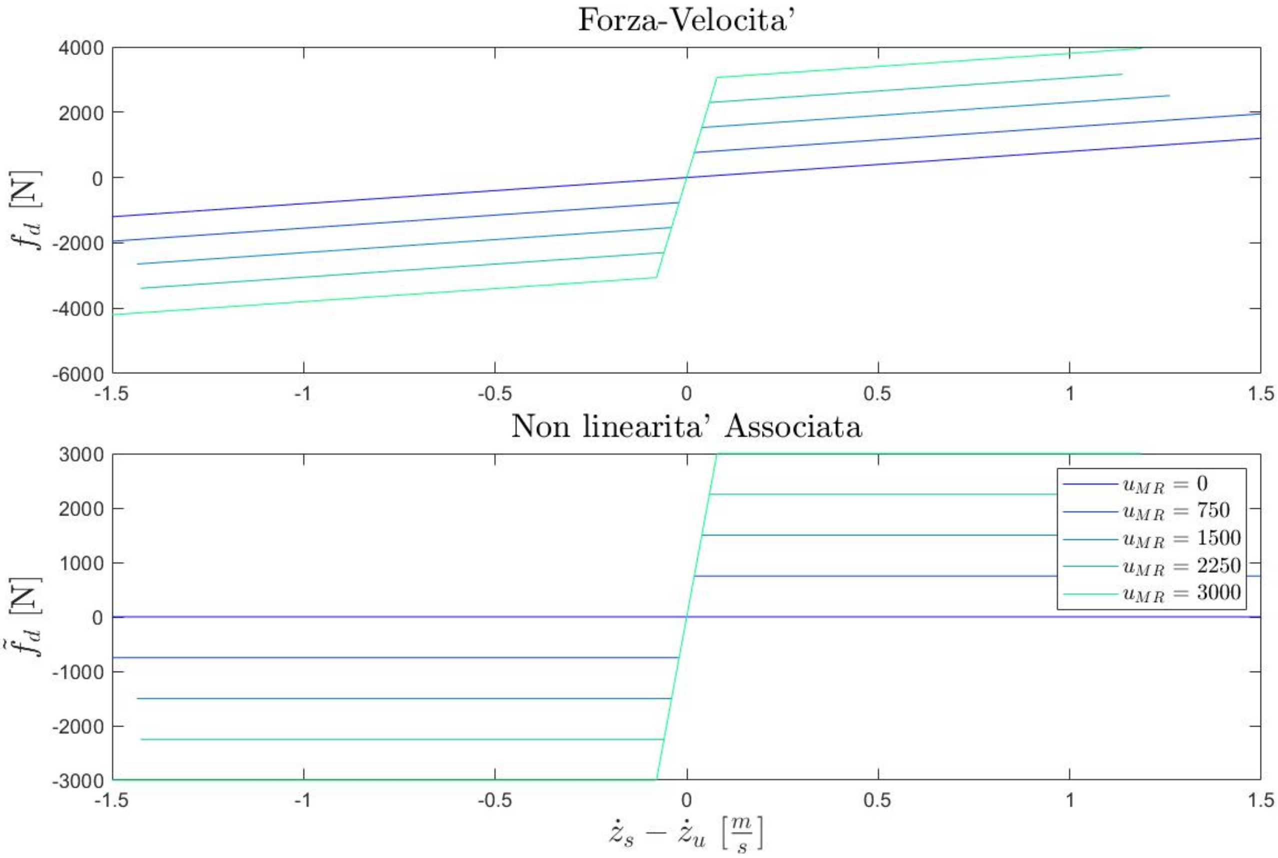


Figura 1.6: Rappresentazione del comportamento delle sospensioni magnetoreologiche

Modello Matematico Totale

Risulta molto importante definire le relazioni matematiche che intercorrono tra gli elementi del modello quarter-car rappresentati in figura 1.5 e le equazioni che definiscono i comportamenti delle sospensioni semi-attive rappresentati nei due grafici in figura 1.6, in quanto proprio da queste equazioni matematiche verrà determinato il modello di controllo utilizzato per l'analisi del problema.

In figura 1.6 il grafico inferiore mostra l'elemento non-lineare peculiare associato alle sospensioni semi-attive a tecnologia magnetoreologica: esso consiste in una funzione di saturazione che segue le seguenti equazioni matematiche

$$\tilde{f}_{\mathrm{d}}(u_{\mathrm{MR}}, \dot{z}_{\mathrm{s}} - \dot{z}_{\mathrm{u}}) = sat_{u_{\mathrm{MR}}}(k_0(\dot{z}_{\mathrm{s}} - \dot{z}_{\mathrm{u}})) \tag{1.3}$$

$$sat_{u_{\mathrm{MR}}}(s) := max\,(\,min\,(s, u_{\mathrm{MR}}), -u_{\mathrm{MR}}) \tag{1.4}$$

che dimostrano come la forza applicata sia in funzione del parametro di controllo $\boldsymbol{u_{MR}}$ e della differenza di velocità tra lo chassis della macchina e lo pneumatico, come indicato in figura 1.5. Il parametro $\boldsymbol{\tilde{f}_d}$ determina anche la forma della curva rappresentata nel grafico superiore della figura 1.6: la curva superiore rappresenta la somma del comportamento non-lineare indicato dalle equazioni 1.3 e 1.4 e il comportamento lineare tipico che determina il minimo livello di ammortizzazione prodotto dalle sospensioni. La somma di questi due comportamenti viene rappresentata mediante la seguente equazione:

$$f_{\mathrm{d}} = c_{\mathrm{min}}(\dot{z}_{\mathrm{s}} - \dot{z}_{\mathrm{u}}) + \tilde{f}_{\mathrm{d}}(u_{\mathrm{MR}}, \dot{z}_{\mathrm{s}} - \dot{z}_{\mathrm{u}}) \tag{1.5}$$

I parametri che definiscono le equazioni 1.3, 1.4 e 1.5 sono i seguenti

Parametro	**Simbolo**	**Valore**
Ammortizzazione Minima	c_{min}	800 $\frac{Ns}{m}$
Pendenza della Saturazione	k_0	38000 $\frac{Ns}{m}$
Saturazione Massima	$\tilde{f}_{\mathrm{max}}$	3000 N

Tabella 1.2: Parametri utilizzati nelle equazioni delle sospensioni magnetoreologiche

essi sono stati scelti da Ruben Begnis [2] seguendo i risultati presentati in Gołdasz e Dzierżek (2016) [3] per ottenere un generico ammortizzatore semi-attivo a tecnologia magnetoreologica applicato in ambito automobilistico.

L'equazione 1.5 e lo schema rappresentato in figura 1.5 sono gli elementi essenziali per scrivere il sistema di equazioni matematiche differenziali del modello quarter-car, la prima rappresentante il bilancio delle forze esercitate sullo chassis del quarto di veicolo (massa ammortizzata), la seconda rappresentante il bilancio delle forze applicate sullo pneumatico (massa non ammortizzata), nelle quali il parametro $\boldsymbol{\tilde{z}}$ è definito come $\tilde{z} := z_{\mathrm{s}} - z_{\mathrm{u}}$.

$$\begin{cases} m_{\mathrm{s}}\ddot{z}_{\mathrm{s}} = -k\tilde{z} - f_{\mathrm{d}}(u_{\mathrm{MR}}, \dot{\tilde{z}}) \\ m_{\mathrm{u}}\ddot{z}_{\mathrm{u}} = -k_{\mathrm{t}}(z_{\mathrm{u}} - z_{\mathrm{r}}) + k\tilde{z} + f_{\mathrm{d}}(u_{\mathrm{MR}}, \dot{\tilde{z}}) \end{cases} \tag{1.6}$$

Si può notare che nella prima delle due equazioni la dinamica dello chassis della macchina è legata all'azione elastica e dissipativa dell'ammortizzatore semi-attivo, mentre nella seconda equazione la dinamica dello pneumatico è legata anche all'effetto elastico dello pneumatico. Una volta definite queste equazioni può essere costruito il modello di controllo implementato in Simulink® che permette la simulazione del sistema dinamico e la successiva analisi del problema, sia in modo passivo, sia utilizzando i vantaggi offerti dalle sospensione semi-attive a tecnologia magnetoreologica.

1.1.3 Modello di Controllo

Il modello di controllo rappresentato in questa sezione è il medesimo rappresentato da Ruben Begnis nell'elaborato [2]. Si tratta di un modello di controllo generalmente non lineare in quanto è presente la funzione $\boldsymbol{\tilde{f}_d}$ legata alla saturazione indicata nell'equazione 1.4: l'unico caso in cui il modello risulta lineare si ottiene quanto il parametro di controllo $\mathbf{u_{MR}}$ è nullo. Come è possibile vedere nel sistema di equazioni 1.7 l'ingresso del sistema corrisponde al profilo stradale $\mathbf{z_r}$ mentre l'uscita corrisponde all'accelerazione verticale dello chassis della macchina $\boldsymbol{\ddot{z}_s}$

$$\sum_{\mathrm{MR}} : \begin{cases} \dot{x} = A(c_{\mathrm{min}})x - B\tilde{f}_{\mathrm{d}}(u_{\mathrm{MR}}, \dot{\tilde{z}}) + Ez_{\mathrm{r}} \\ y = \ddot{z}_{\mathrm{s}} = C(c_{\mathrm{min}})x - D\tilde{f}_{\mathrm{d}}(u_{\mathrm{MR}}, \dot{\tilde{z}}) \end{cases} \tag{1.7}$$

e i coefficienti A, B, C, D, E sono matrici i cui valori sono elencati nelle seguenti equazioni

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{k}{m_\mathrm{s}} & 0 & -\frac{c_\mathrm{min}}{m_\mathrm{s}} \\ \frac{k_\mathrm{t}}{m_\mathrm{u}} & -\frac{k}{m_\mathrm{s}}-\frac{k_\mathrm{t}+k}{m_\mathrm{u}} & 0 & -\frac{c_\mathrm{min}(m_\mathrm{s}+m_\mathrm{u})}{m_\mathrm{s}m_\mathrm{u}} \end{bmatrix} \tag{1.8}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{m_\mathrm{s}} \\ \frac{(m_\mathrm{s}+m_\mathrm{u})}{m_\mathrm{s}m_\mathrm{u}} \end{bmatrix} \tag{1.9}$$

$$C = \begin{bmatrix} 0 & -\frac{k}{m_\mathrm{s}} & 0 & -\frac{c_\mathrm{min}}{m_\mathrm{s}} \end{bmatrix} \tag{1.10}$$

$$D = \begin{bmatrix} \frac{1}{m_\mathrm{s}} \end{bmatrix} \tag{1.11}$$

$$E = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -\frac{k_\mathrm{t}}{m_\mathrm{u}} \end{bmatrix} \tag{1.12}$$

mentre $x = [z_\mathrm{s} \quad \tilde{z} \quad \dot{z}_\mathrm{s} \quad \dot{\tilde{z}}]^\mathrm{T}$ è la variabile di stato del modello di controllo. Gli altri parametri sono tutti elencati nelle tabelle 1.1 e 1.2. Il parametro di controllo $\mathbf{u_{MR}}$ risulta essere un elemento molto importante del modello di controllo in quanto il metodo con cui viene definito permette di effettuare due tipi di simulazioni differenti: mantenendo costante il valore del parametro viene effettuata un analisi passiva, mentre nell'analisi semi-attiva il parametro non risulta più essere costante ma può anch'esso variare in funzione della dinamica verticale del veicolo: nel caso semi-attivo esso è definito dalla seguente equazione

$$\begin{aligned} u_\mathrm{MR} &= \frac{\tilde{f}_\mathrm{max}}{2} + sgn(\dot{\tilde{z}})sat_{\frac{\tilde{\mathrm{f}}_\mathrm{max}}{2}}([k1 \quad k2 \quad k3 \quad k4]x) \\ &= \frac{\tilde{f}_\mathrm{max}}{2} + sgn(\dot{\tilde{z}})sat_{\frac{\tilde{\mathrm{f}}_\mathrm{max}}{2}}(Kx) \end{aligned} \tag{1.13}$$

dunque nell'analisi semi-attiva $\boldsymbol{u_{MR}}$ dipende sia da $\dot{\tilde{z}}$, dalla variabile di stato del modello di controllo $x = [z_\mathrm{s} \quad \tilde{z} \quad \dot{z}_\mathrm{s} \quad \dot{\tilde{z}}]^\mathrm{T}$ e dalla matrice $K = [k1 \quad k2 \quad k3 \quad k4]$ dove ognuno dei quattro parametri può assumere un qualsiasi valore affinché sia seguita la disuguaglianza $|Kx| \geq \frac{\tilde{f}_\mathrm{max}}{2}$[1].

Nello svolgimento delle analisi i parametri *k1, k2, k3, k4* vengono selezionati utilizzando il metodo dell'*esplorazione esaustiva* o *forza bruta*: presi dei valori limite, definiti successivamente nella sezione 3.1, i parametri del vettore K vengono selezionati suddividendo l'intervallo delimitato dai valori limite in parti uguali. In questo modo le simulazioni non vanno ad analizzare tutte le possibili combinazioni dei quattro parametri, ma vanno ad analizzare solo una parte di esse: essendo però i parametri *k1, k2, k3, k4* equi distanziati tra loro si va a coprire tutta l'area selezionata e si ottene, come da nome, un'esplorazione esaustiva dell'area dati in analisi.

[1]Verrà mostrato in seguito che la ricerca dei valori massimi ammissibili è basata su di un criterio ben definito

1.2 Strumenti di Analisi

Avendo definito il profilo stradale, il modello quarter-car e il modello di controllo utilizzati nell'analisi, è opportuno definire anche gli strumenti utilizzati nel valutare l'efficacia delle sospensioni semi-attive nell'attutire le imperfezioni stradali che un veicolo incontra lungo il tragitto percorso. Il modello di controllo, una volta effettuata una simulazione, restituisce dei segnali nel tempo che seguono le variazioni dei parametri contenuti nella variabile di stato $x = [z_\mathrm{s} \quad \tilde{z} \quad \dot{z}_\mathrm{s} \quad \dot{\tilde{z}}]^\mathrm{T}$, rappresentati nei grafici in figura 1.7.

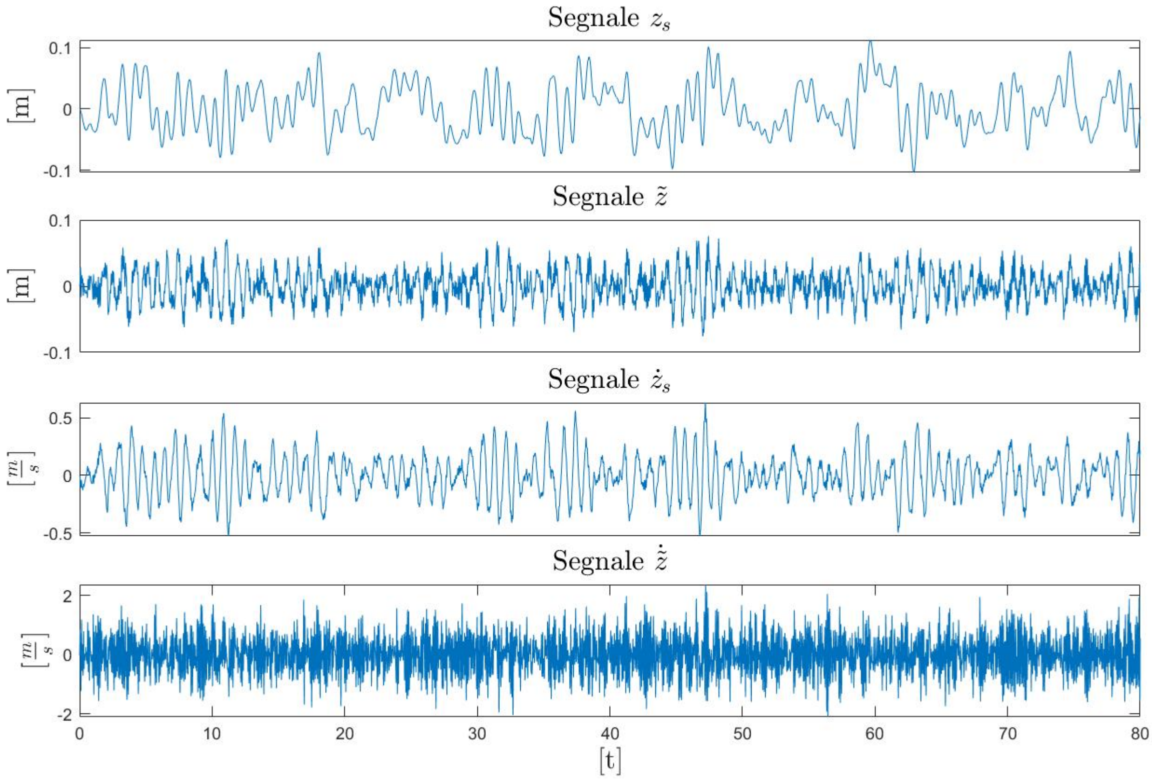


Figura 1.7: Rappresentazione dei segnali generati svolgendo una simulazione

Per determinare l'efficacia delle sospensioni è necessario elaborare i segnali nel tempo ottenuti dalle simulazioni: in questo elaborato verranno utilizzati dei parametri definiti *cifre di costo* che determinano il comfort di guida provato dai passeggeri del veicolo e le *funzioni di trasferimento* che mettono in relazione il segnale in ingresso al modello di controllo, che in questa analisi corrisponde al profilo stradale $\boldsymbol{z_r}$, ed il segnale in uscita dal modello di controllo che in questa analisi corrisponde all'accelerazione dello chassis $\boldsymbol{\ddot{z}_\mathrm{s}}$.

1.2.1 Cifre di Costo

Il parametro utilizzato in questo elaborato per identificare il comfort di guida viene denominiamo *cifra di costo*, che può essere calcolata sia sull'accelerazione verticale di cassa del veicolo $\mathbf{J_{ax}}$ oppure può essere calcolata sulla velocità verticale di cassa del veicolo $\mathbf{J_v}$. Matematicamente $\mathbf{J_{ax}}$ corrisponde all'errore quadratico medio (rms) dell'accelerazione verticale dello chassis del veicolo, considerato su tutto l'intervallo temporale di analisi considerato, come mostrato nell'equazione 1.14, mentre $\mathbf{J_v}$ corrisponde all'errore quadratico medio (rms) della velocità verticale dello chassis del veicolo come mostrato nell'equazione 1.15. Le equazioni matematiche sono le seguenti:

$$J(\ddot{z}_{\mathrm{s}}) = \sqrt{\frac{1}{T}\int_0^T |\ddot{z}_{\mathrm{s}}(\tau)|^2 d\tau} \tag{1.14}$$

$$J(\dot{z}_{\mathrm{s}}) = \sqrt{\frac{1}{T}\int_0^T |\dot{z}_{\mathrm{s}}(\tau)|^2 d\tau} \tag{1.15}$$

dove T rappresenta appunto la durata temporale della simulazione in analisi.

La cifra di costo sull'accelerazione $\mathbf{J_{ax}}$ è considerata come il parametro classico nella determinazione del comfort di guida provato dai passeggeri in quanto il corpo umano è più sensibile alle accelerazioni piuttosto che alle velocità che subisce. La cifra di costo sulla velocità $\mathbf{J_v}$ invece, combinata con i grafici delle funzioni di trasferimento corrispondenti, permette di cogliere meglio il comportamento del sistema in analisi. In questo elaborato viene utilizzata la cifra di costo sull'accelerazione $\mathbf{J_{ax}}$ per confrontare il comfort di guida ottenuto dalle simulazioni in quanto, come accennato, essa è considerata il parametro classico.

1.2.2 Funzioni di Trasferimento

Le *funzioni di trasferimento* sono dei metodi matematici che descrivono il comportamento di un sistema andando a mettere a confronto l'ingresso del sistema con la sua uscita. Implementando questo strumento matematico su *Matlab*® è possibile ottenere dei grafici dove viene rappresentato il comportamento in frequenza dei sistemi in analisi, permettendo di determinare subito quali siano le correlazioni presenti tra diversi sistemi e quali siano le componenti che pesano di più nell'ottimizzazione del problema in analisi. In figura 1.8 vengono raffigurate due curve rappresentanti le funzioni di trasferimento che mettono in confronto il segnale in entrata $\boldsymbol{z_r}$ e il segnale in uscita $\boldsymbol{\ddot{z}_s}$ dal modello di controllo per una simulazione in modalità passiva, cioè mantenendo il parametro di controllo $\mathbf{u_{MR}}$ costante. La curva arancione rappresenta la *funzione di trasferimento* ottenuta analiticamente dal modello di controllo in quanto fissando il parametro di controllo $u_{\mathrm{MR}} = 0$ il sistema di equazioni risulta lineare ed è possibile ottenere la funzione di trasferimento analizzando le sole matrici del sistema, mentre la curva blu rappresenta la funzione di trasferimento stimata andando a confrontare i segnali in entrata ed in uscita ottenuti dalla simulazione effettuata. Come è possibile notare la curva stimata mostra delle piccole fluttuazioni rispetto alla curva analitica: nonostante queste fluttuazioni entrambe le curve mostrano due picchi in corrispondenza di $f_1 \approx 1.5[Hz]$ e $f_2 \approx 11.5[Hz]$ che sono tipici di questo tipo di simulazioni.

Nella figura 1.8 vengono rappresentate le *figure di trasferimento* fino ad una componente in frequenza $f = 20[Hz]$ in quanto il modello di controllo non presenta dinamiche a frequenze elevate, dunque non ci si aspetta di trovare comportamenti notevoli ad alte frequenze. Inoltre, i picchi nelle curve delle *funzioni di trasferimento* presenti nell'intervallo di frequenze considerato sono legati alle cifre di costo introdotte nella sezione 1.2.1: il picco a frequenza $f_1 \approx 1.5[Hz]$ è legato alla cifra di costo sulla velocità $\mathbf{J_v}$ mentre il picco a frequenza $f_2 \approx 11.5[Hz]$ e legato alla cifra di costo sull'accelerazione $\mathbf{J_{ax}}$, dunque andando a confrontare le *funzioni di trasferimento* ottenute da differenti simulazioni è possibile notare vantaggi o svantaggi nel comfort di guida percepito.

Nei grafici delle *funzioni di trasferimento* che seguiranno saranno rappresentate sia le curve sui dati grezzi, sia le curve rappresentanti le *funzioni di trasferimento* sui dati anticipatamente elaborati, anche se in maniera leggera. Per elaborazione si intende l'applicazione di alcuni accorgimenti per rendere più efficaci le funzioni utilizzate nel codice *Matlab*®. In generale, il calcolo teorico delle *funzioni di trasferimento* avviene anche tramite l'applicazione della *Trasformata di Fourier*: nel calcolo computazionale viene però applicata la *DFT o Discrete Fourier Trasform* il cui funzionamento può essere ottimizzato applicando *zero-padding* e opportune *finestrature* sui segnali grezzi in entrata e in uscita da un sistema. Con *zero-padding* si intende l'aggiunta di zeri in testa e in coda al segnale per ottenere una lunghezza del nuovo segane che sia ottimale per l'applicazione della DFT, mentre con *finestratura* si intende l'applicazione di una sequenza di valori che vada a pesare in modo specifico il segnale in analisi: in generale la fenestratura ha efficacia se il vettore di valori pesanti è di lunghezza inferiore rispetto alla lunghezza del segnale analizzato.

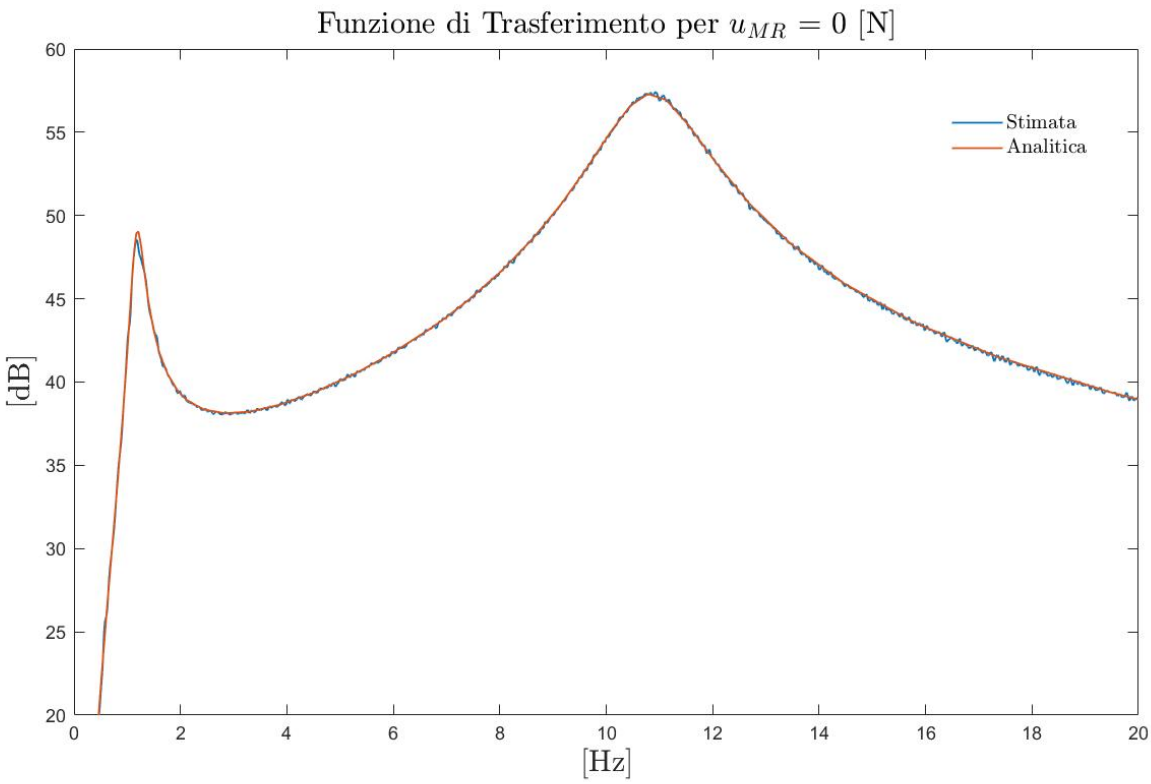


Figura 1.8: Funzioni di trasferimento analitica e stimata per una simulazione con uguale parametro di controllo

2 Analisi Passiva e Benchmark Semi-Attivo

In questa sezione viene analizzato il comfort di guida ottenuto delle sospensioni magnetoreologiche semi-attive utilizzando il modello di controllo in modalità passiva, confrontandolo con i risultati di ottimizzazione ricavati dall'analisi tramite *benchmark semi-attivo* già ottenuti in precedenza e riportati nel paper [2] utilizzando il modello di controllo in modalità semi-attiva con un definito vettore di parametri K.

L'analisi passiva consiste nell'andare a valutare le cifre di costo sull'accelerazione $\mathbf{J_{ax}}$ e sulla velocità $\mathbf{J_v}$ ottenute per valori del parametro di controllo $\mathbf{u_{MR}}$ costanti per tutta la durata di una singola simulazione. Quest'analisi è necessaria per definire dei valori base di comfort di guida provata dai passeggeri da confrontare con i dati del *benchmark semi-attivo* ed in seguito con i dati raccolti nell'analisi semi-attiva nel capitolo 3: potremo così valutare il miglioramento offerto dalle sospensioni magnetoreologiche con applicazione semi-attiva.

L'analisi effettuata tramite *benchmark semi-attivo* invece consiste nel confronto delle cifre di costo ottenute applicando il vettore K denominato *K Bench*

$$K = [-18901 \quad -45920 \quad 22704 \quad -36338] \tag{2.1}$$

nell'equazione 1.13 ed utilizzando dunque il modello di controllo effettivamente in modalità semi-attiva. *K Bench* è il vettore fornito nel paper [2] che ottimizza la cifra di costo sull'accelerazione $\mathbf{J_{ax}}$ considerata una velocità $v = 90[km/h]$.

2.1 Analisi Dati

Di seguito vengono riportati i dati ottenuti dall'analisi passiva e dall'analisi tramite *benchmark semi-attivo* per le tre differenti velocità di percorrenza indicate nella sezione 1.1.1.

Per quanto riguarda l'analisi passiva i dati sono stati raccolti andando a calcolare le cifre di costo sull'accelerazione $\mathbf{J_{ax}}$ e sulla velocità $\mathbf{J_v}$ per definiti valori di saturazione: la saturazione massima indicata in tabella 1.2 è stata suddivisa in modo che il parametro di controllo $\mathbf{u_{MR}}$ rappresentasse frazioni di essa, partendo da $u_{\mathrm{MR}} = \frac{\tilde{f}_{\mathrm{max}}}{3000}$ ed arrivando fino a $u_{\mathrm{MR}} = \tilde{f}_{\mathrm{max}}$. Questi dati raccolti sono riportati in figura 2.1 dove si possono notare delle curve parametriche rappresentanti le cifre di costo sull'accelerazione $\mathbf{J_{ax}}$ e sulla velocità $\mathbf{J_v}$ in funzione dei valori del parametro di controllo $\mathbf{u_{MR}}$ considerati. Come si può vedere nella legenda, sono state raffigurate tre differenti curve in colore blu, ognuna rappresentante i dati raccolti nell'analisi per una definita velocità di percorrenza del veicolo: è utile sottolineare che all'aumentare della velocità le cifre di costo sull'accelerazione $\mathbf{J_{ax}}$ e sulla velocità $\mathbf{J_v}$ in genere aumentano, dunque anche le curve parametriche seguono questo comportamento dove all'aumentare della velocità esse sono più distanti dall'origine. Vengono anche riportati nel grafico i valori ottimi di $\mathrm{J_{ax}}$ in colore rosso, ottenuti per le tre differenti velocità considerate.

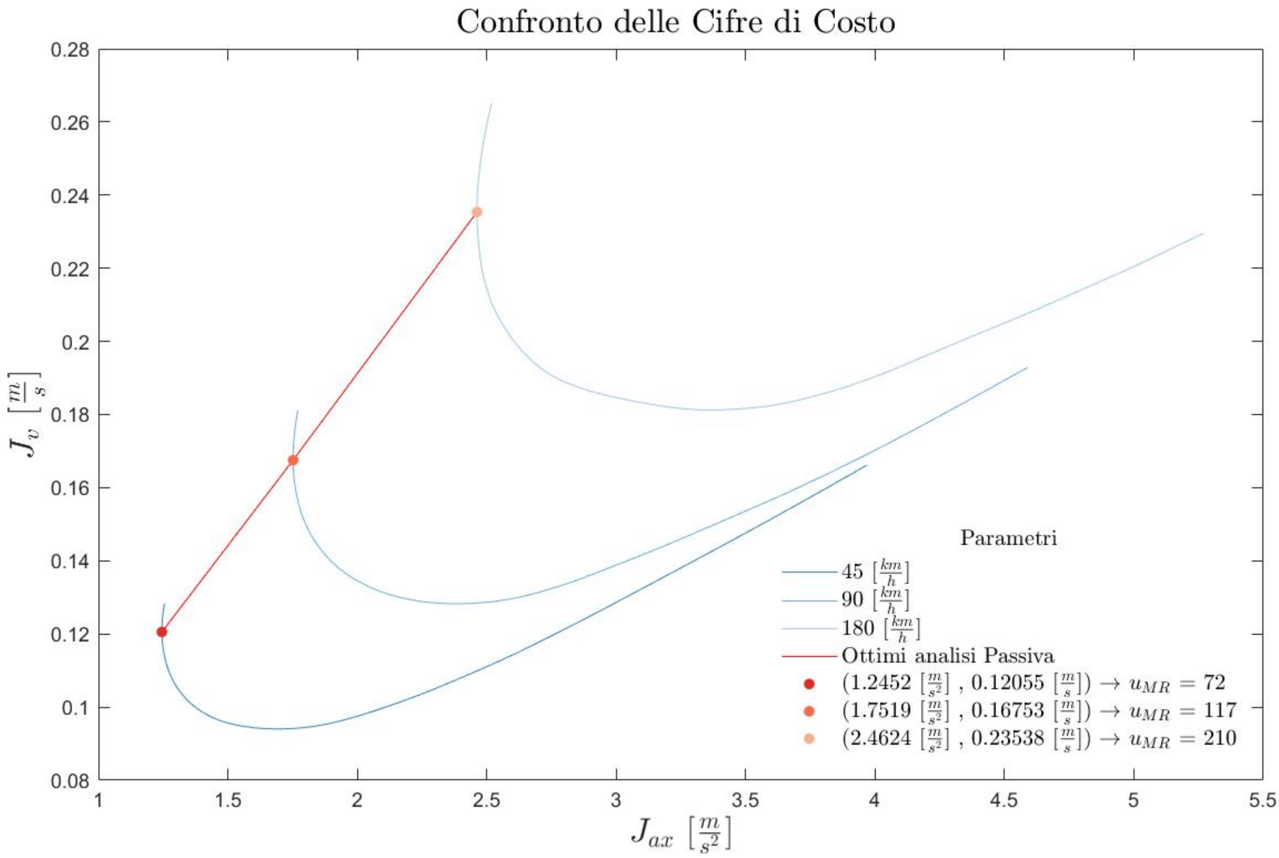


Figura 2.1: Rappresentazione dei dati raccolti nell'analisi passiva

Dal grafico in figura 2.1 si può notare che i parametri ottimi delle cifre di costo sull'accelerazione $\mathbf{J_{ax}}$ per le tre differenti velocità di percorrenza appartengono alla stessa retta, ma vengono ricavati per differenti valori del parametro di controllo $\mathbf{u_{MR}}$, ed all'aumentare della velocità di percorrenza aumenta anche il valore di tale parametro.

Per quanto riguarda invece l'analisi effettuata tramite *benchmark semi-attivo* si vuole dimostrare come il vettore *K Bench* fornito risulti essere la combinazione di parametri che ottimizza il comfort di guida, non solo per la velocità nominale di $v = 90[km/h]$, ma anche per le velocità $v = 45[km/h]$ e $v = 180[km/h]$, rispetto al comfort di guida ottenuto tramite l'analisi passiva. Come è possibile notare in figura 2.2 i valori delle cifre di costo $\mathbf{J_{ax}}$ risultano migliori rispetto ai dati raccolti nell'analisi passiva. Come era gia noto, per una velocità pari a $v = 90[km/h]$ la cifra di costo sull'accelerazione di cassa $\mathbf{J_{ax}}$ risulta significamene diminuita. E' interessante notare invece come per una velocità pari a $v = 45[km/h]$ l'ottimizzazione effettuata tramite il vettore *K Bench* non ha riportato ottimi risultati, riuscendo solo ad emulare quasi il comfrot di guida offerto dall'analisi passiva. Per una velocità pari a $v = 180[km/h]$ invece si ha un miglioramento maggiore di quello ottenuto nell'analisi a velocità nominale, dimostrando dunque il funzionamento del vettore *K Bench*.

Dal grafico 2.2 si può anche notare però l'efficacia del vettore *K Bench* nell'ottimizzare la cifra di costo sulla velocità di cassa $\mathbf{J_v}$: per tutte e tre le velocità considerate il parametro $\mathbf{J_v}$ risulta diminuito di molto, facendoci intuire dunque la ulteriore capacità di ottimizzazione che il vettore *K Bench* possiede in esso. Alla luce di questi primi risultati, si potrebbe basare l'analisi dati sul parametro $\mathbf{J_v}$ e non più sul parametro $\mathbf{J_{ax}}$, in quanto per il primo si ha un più ampio margine di ottimizzazione. Questa scelta dipende dall'individuo o dalla casa produttrice in base alla scelta di ottimizzare le velocità o le accelerazioni verticali del veicolo. In questo elaborato l'analisi continuerà ad essere basata sulla cifra di costo dell'accelerazione di cassa $\mathbf{J_{ax}}$ in quanto, come è giusto ricordare, il vettore *K Bench* fornito dal paper [2] ottimizza $\mathbf{J_{ax}}$ e non $\mathbf{J_v}$.

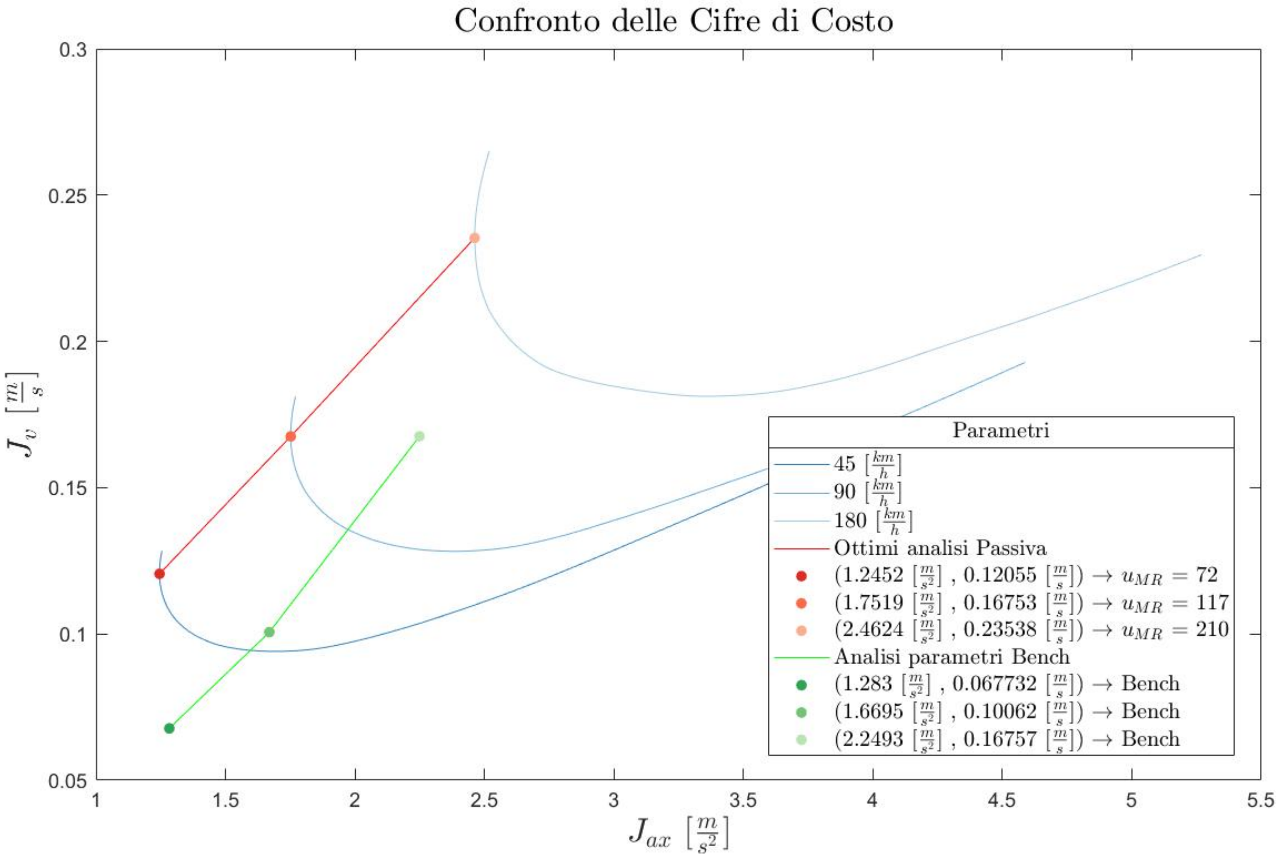


Figura 2.2: Rappresentazione dei dati raccolti nell'analisi passiva e nell'analisi tramite *benchmark semi-attivo*

Per facilità di lettura viene anche riportata una tabella contente i parametri ottimi trovati.

Velocità	**Parametro**	**Analisi Passiva**	**Analisi Benchmark**
45 $\frac{m}{s}$	J_{ax}	1.2452 $\frac{m}{s^2}$	1.283 $\frac{m}{s^2}$
	J_v	0.12055 $\frac{m}{s}$	0.067732 $\frac{m}{s}$
90 $\frac{m}{s}$	J_{ax}	1.7519 $\frac{m}{s^2}$	1.6695 $\frac{m}{s^2}$
	J_v	0.16753 $\frac{m}{s}$	0.10062 $\frac{m}{s}$
180 $\frac{m}{s}$	J_{ax}	2.4624 $\frac{m}{s^2}$	2.2493 $\frac{m}{s^2}$
	J_v	0.23538 $\frac{m}{s}$	0.16757 $\frac{m}{s}$

Tabella 2.1: Rappresentazione dei parametri ottimi trovati, suddivisi per velocità

2.2 Analisi Funzioni di Trasferimento

In questa sezione vengono rappresentate le funzioni di trasferimento, da ora in poi denominate FdT, utilizzate per confrontare l'evoluzione dei sistemi dinamici corrispondenti ai migliori valori di cifra di costo $\mathbf{J_{ax}}$ ottenuti durante l'analisi passiva e l'analisi effettuata tramite *benchmark semi-attivo*, per tutte e tre le velocità considerate. Iniziando dalla figura 2.3 è possibile notare come il picco a frequenza $\boldsymbol{f_1}$ presente nella FdT dell'analisi passiva non sia più presente nella FdT dell'analisi effettuata tramite *benchmark semi-attivo*, mentre il picco a frequenza $\boldsymbol{f_2}$ sia pressoché il medesimo.

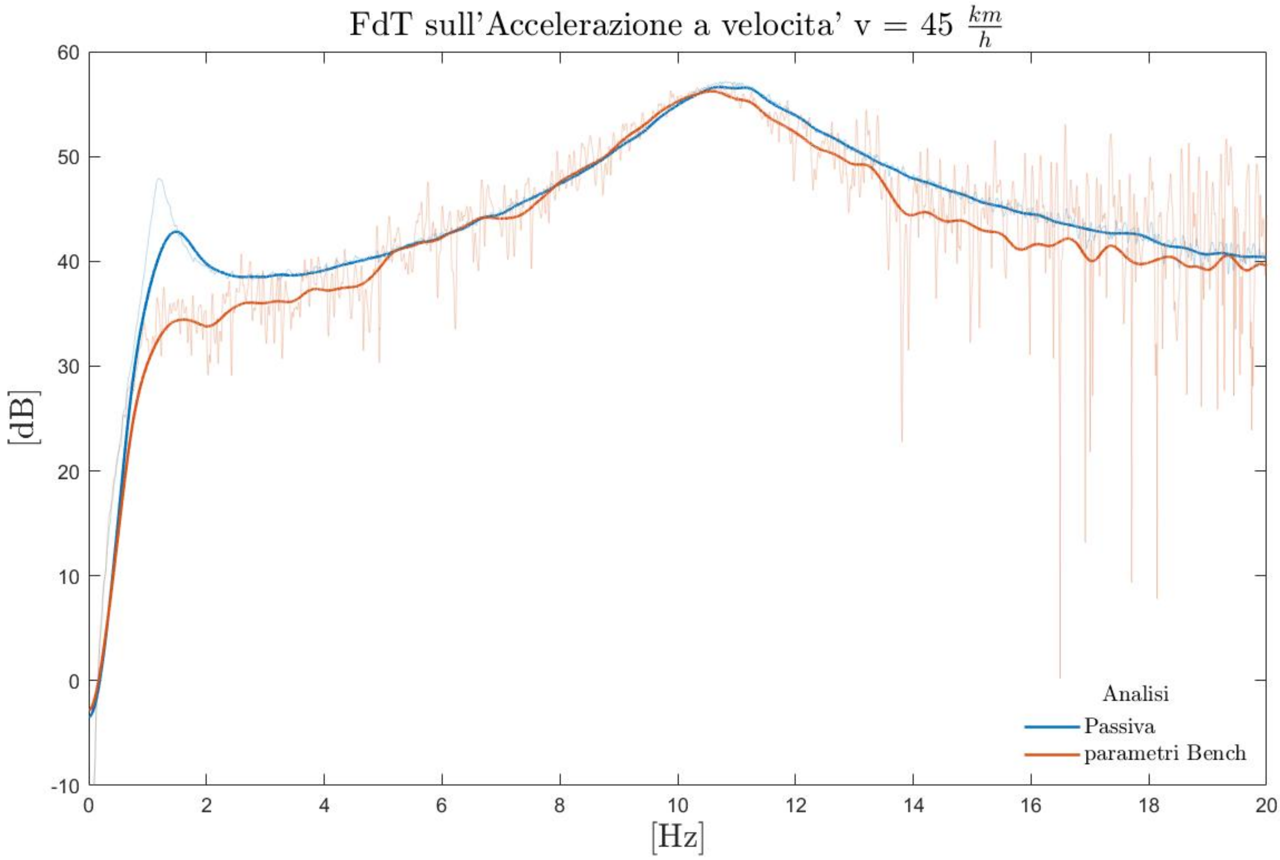


Figura 2.3: Funzioni di trasferimento rappresentati l'analisi e l'analisi effettuata tramite *benchmark semi-attivo* a velocità $v = 45[km/h]$

Inoltre dalla figura 2.3 si può notare che la curva rappresentante l'analisi effettuata tramite *benchmark semi-attivo* nell'intervallo di frequenza tra 6[Hz] e 11[Hz] va a sovrapporsi alla curva ottenuta tramite analisi passiva. Il picco smussato e la sovrapposizione vengono rispecchiati anche nella figura 2.2 dove si vede come l'analisi effettuata tramite *benchmark semi-attivo* migliori notevolmente il parametro $\mathbf{J_v}$, mentre il parametro $\mathbf{J_{ax}}$ rimane pressoché invariato.

Analizzando le figure 2.4 e 2.5 si può notare come il picco a frequenza $\boldsymbol{f_1}$ risulti sempre smussato, mentre la curva rappresentante l'analisi effettuata tramite *benchmark semi-attivo* nell'intervallo di frequenza tra 6[Hz] e 11[Hz] si distanzia dalla curva nominale: in particolare la distanza è maggiore per la velocità $v = 180[km/h]$ rispetto alla velocità $v = 90[km/h]$. Questo viene rispecchiato anche in figura 2.2 dove all'aumentare della velocità il miglioramento del parametro ottimo $\mathbf{J_{ax}}$ aumenta, mentre il miglioramento del parametro ottimo $\mathbf{J_v}$ rimane pressoché invariato.

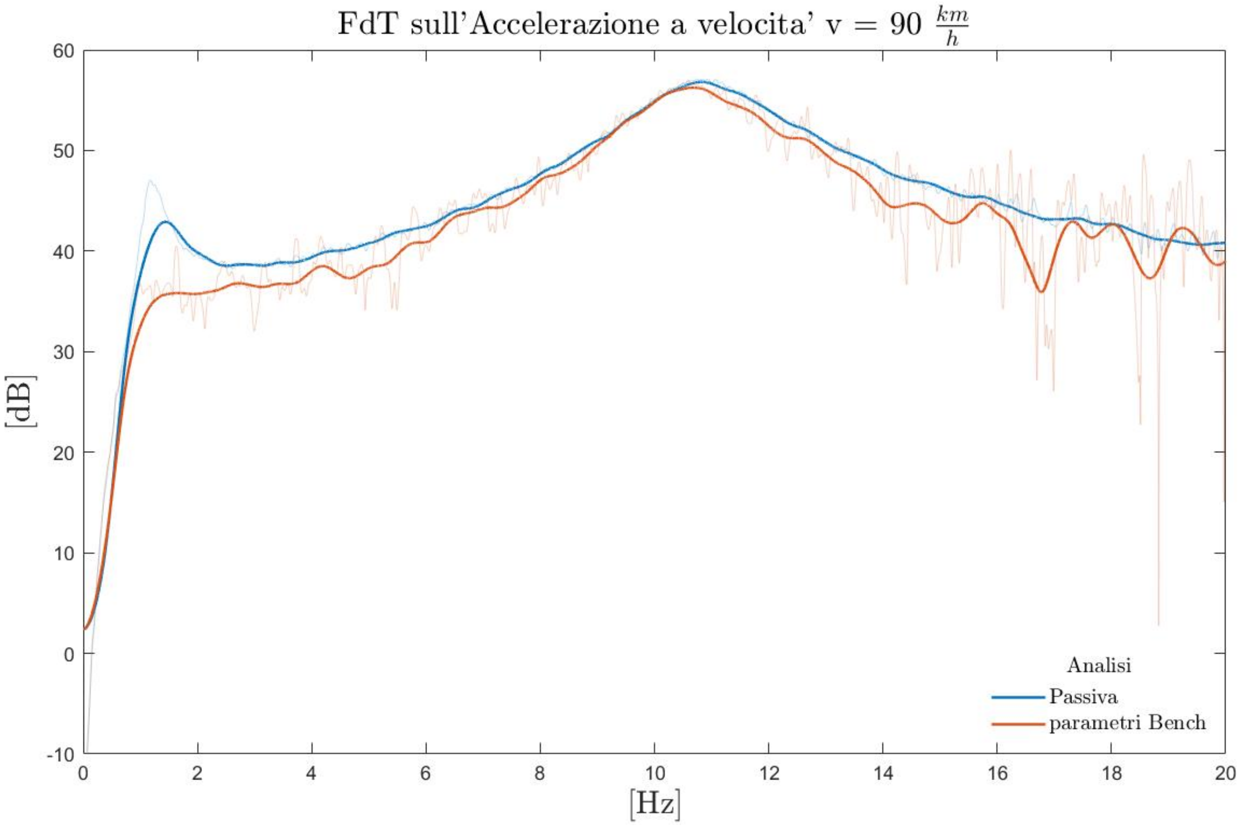


Figura 2.4: Funzioni di trasferimento rappresentati l'analisi e l'analisi effettuata tramite *benchmark semi-attivo* a velocità $v = 90[km/h]$

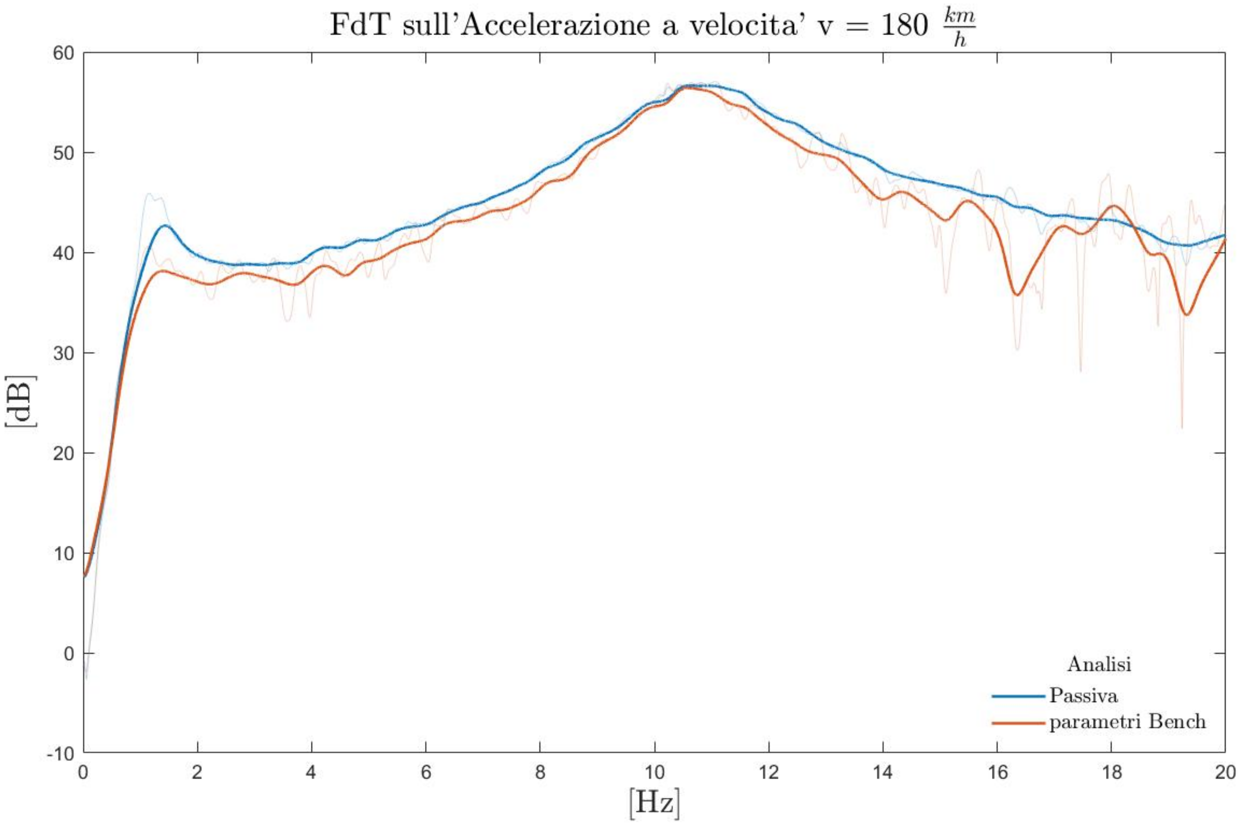


Figura 2.5: Funzioni di trasferimento rappresentati l'analisi e l'analisi effettuata tramite *benchmark semi-attivo* a velocità $v = 180[km/h]$

3 Ottimizzazione ed Analisi Semi-Attiva

Nell'analisi delle sospensioni magnetoreologiche in modalità semi-attiva viene sfruttato appieno il modello di controllo illustrato nella sezione 1.1.3, in quanto la forza ammortizzante non dipenderà solamente dalla variazione di distanza tra lo pneumatico e lo chassis della macchina, bensì dipenderà anche dalla posizione e dalla velocità verticale dello chassis stesso, come indicato dalle equazioni 1.3, 1.4, 1.5 e 1.13. Questa che verrà rappresentata è la parte più importante di tutto l'elaborato, in quanto verranno esposti i dati che appoggiano la tesi indicata nel paper [2]: in particolare verrà indicato che generalmente l'ottimizzazione ottenuta dall'analisi effettuata tramite *benchmark semi-attivo* è quella che offre il miglior livello di comfort offerto da questo metodo di controllo, rispetto agli altri casi analizzati. Verranno rappresentati due differenti casi di ottimizzazione tramite analisi semi-attiva: nella prima tutti e 4 i parametri del vettore K potranno variare, mentre nella seconda il parametro *k1* verrà mantenuto costante. Entrambi i casi verranno confrontati con i dati presentati nel capitolo 2 e verranno rappresentate anche le corrispondenti FdT.

3.1 Analisi Boundaries

In questa sezione vengono indicati i criteri con cui sono stati selezionati i parametri del vettore K per effettuare l'ottimizzazione tramite analisi semi-attiva. La ricerca delle cifre di costo ottime richiede lo svolgimento di un gran numero di simulazioni, necessarie per valutare diverse combinazioni dei parametri *k1, k2, k3, k4* appartenenti al vettore K. Risulta dunque essenziale porsi dei limiti o *Boundaries* all'interno dei quali cercare i parametri *k1, k2, k3, k4*, anche perché in questo elaborato viene trattato un problema di ottimizzazione in quattro dimensioni, essendo quattro i parametri che compongono il vettore K: se la selezione dei parametri avvenisse in maniera puramente casuale sarebbe molto difficile raggiungere delle conclusioni apprezzabili. I *Boundaries* sono dei limiti (inferiore e superiore) i quali non devono essere superati nel scegliere i parametri del vettore K per essere anche in grado di notare le possibili correlazioni presenti. Essi vengono definiti in maniera iterativa: i valori iniziali dei limiti vengono definiti secondo questa equazione

$$k_{\max} = \frac{\frac{\tilde{f}_{\max}}{2}}{q_{\max,95\%}} \tag{3.1}$$

dove il parametro $\boldsymbol{\tilde{f}_{\max}}$ è quello riportato in tabella 1.2, mentre il parametro $\boldsymbol{q_{\max,95\%}}$ é l'elemento che racchiude in se le informazioni per selezionare il limite superiore e inferiore, per ogni parametro di K, per effettuare le simulazioni. Esso viene ottenuto andando ad effettuare delle considerazioni sui dati raccolti tramite le analisi passive: vengono considerati tutti i dati raccolti dalle analisi racchiusi nel vettore $x = [z_s \quad \tilde{z} \quad \dot{z}_s \quad \dot{\tilde{z}}]^T$ variabile di stato e dopo averne calcolato il modulo viene effettuato il calcolo dei quartili al 95%. Dopo aver calcolato i quartili per ogni simulazione, essi vengono confrontati tra loro per ogni elemento della variabile di stato e vengono selezionati solo i valori massimi calcolati: dopo aver determinato questi quattro valori viene calcolato il valore di $\boldsymbol{k_{max}}$ secondo l'equazione 3.1.

Successivamente esso viene ulteriormente maggiorato del 10%, indicato $\boldsymbol{k^*_{\mathbf{max}}}$, per poter considerare un margine di sicurezza ulteriore, e viene utilizzato per settare il limiti superiore e inferiore delle analisi da effettuare secondo la seguente disequazione 3.2.

$$-k^*_{\mathrm{max_n}} \leq k_\mathrm{n} \leq k^*_{\mathrm{max_n}} \tag{3.2}$$

Una volta svolte le analisi, se le cifre di costo ottime corrispondono a delle analisi effettuate tramite i parametri $\boldsymbol{k^*_{\mathbf{max}}}$, questi vengono maggiorati ulteriormente in modo iterativo finché le simulazioni non restituiscono delle cifre di costo ottime i cui parametri *k1, k2, k3, k4* non corrispondono ai parametri maggiorati, oppure se le cifre di costo ottime tra due simulazioni consecutive presentano un errore percentuale inferiore al 0.5%. Questo metodo iterativo, inoltre, permette di determinare il comportamento e le tendenze dei valori *k1, k2, k3, k4* nel determinare le cifre di costo ottime ma soprattutto permette di determinare il numero di passi necessari per ottenere la convergenza della cifra di costo $\mathbf{J_{ax}}$ verso il miglior valore ottenibile dalle simulazioni sotto le ipotesi considerate. Per determinare la convergenza delle analisi verso la migliore cifra di costo sull'accelerazione $\mathbf{J_{ax}}$ ottenibile è stato scelto come errore limite tra due misurazioni consecutive, come indicato sopra, un errore pari a $\epsilon_{\mathrm{lim}} = 0.5\%$, per tutte e tre le velocità di percorrenza considerate, in modo da vedere per l'appunto la velocità di convergenza delle simulazioni alle tre velocità.

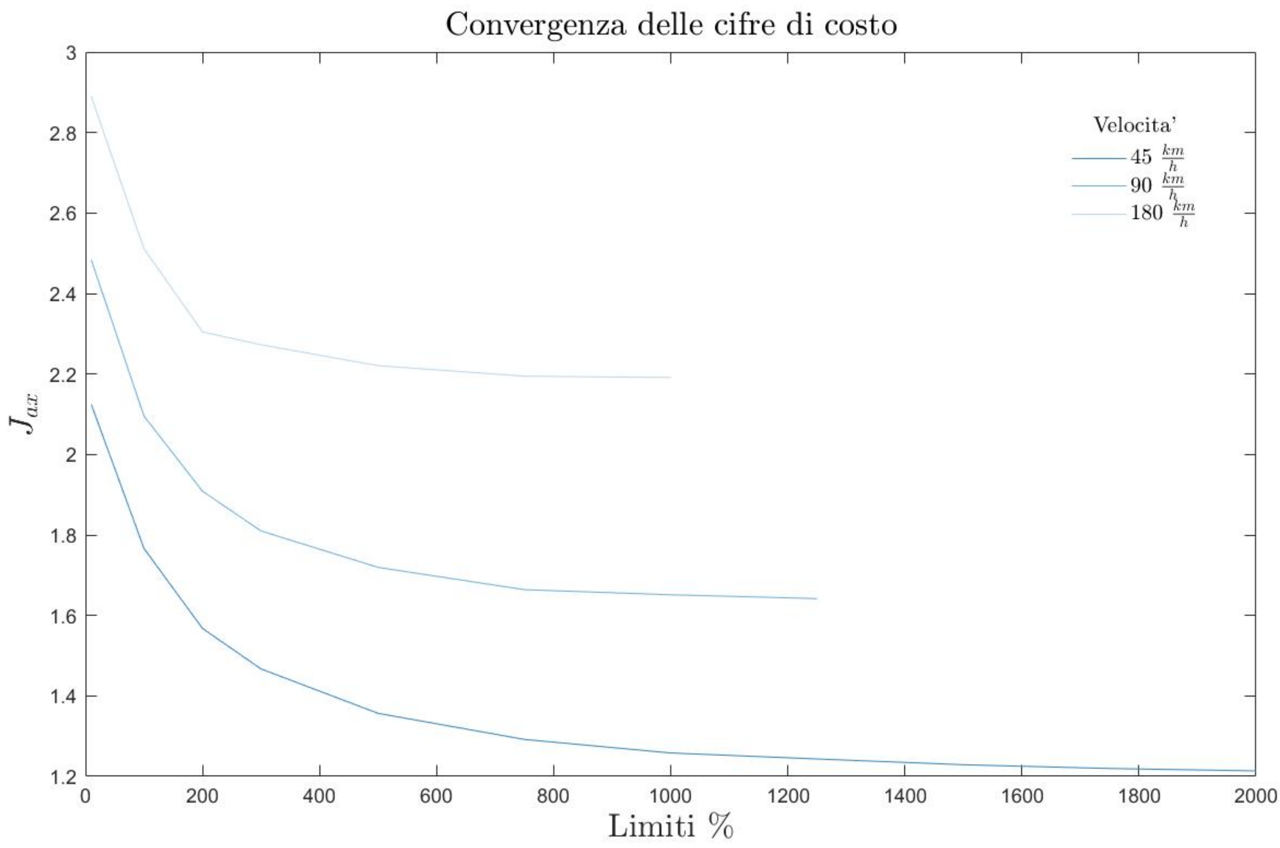
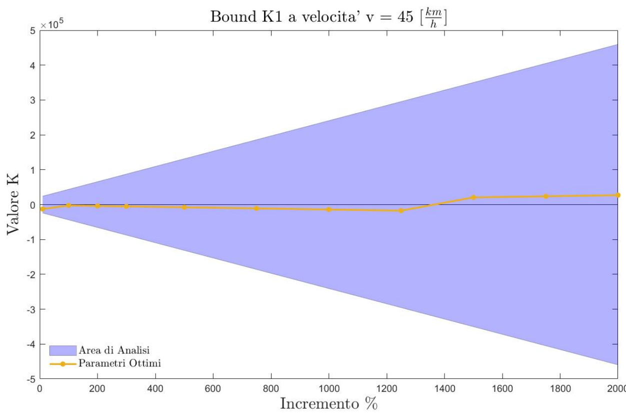
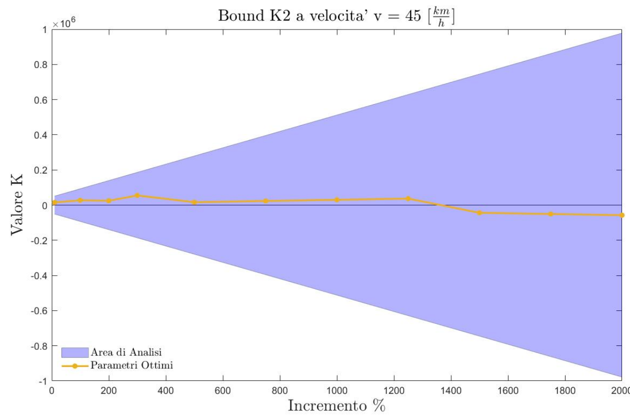


Figura 3.1: Rappresentazione delle convergenze del parametro $\mathbf{J_{ax}}$ ottimo

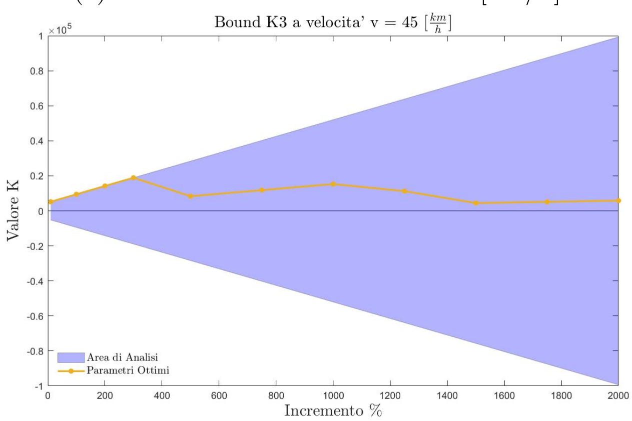
Sotto questa ipotesi si possono notare in figura 3.1 tre curve con convergenze a percentuali di maggiorazione differenti: per una velocità pari a $v = 45[km/h]$ si può notare come la convergenza avvenga per una maggiorazione pari al 2000% con un errore $\epsilon = 0.48\%$, invece per $v = 90[km/h]$ la convergenza avviene alla maggiorazione del 1250% con un errore $\epsilon \approx 0.5\%$ mentre alla velocità di $v = 180[km/h]$ la convergenza avviene alla maggiorazione del 1000% con un errore $\epsilon = 0.17\%$. Oltre a rappresentare il grafico in figura 3.1 risulta essere comodo anche rappresentare l'evoluzione dei parametri *k1, k2, k3, k4*, per le tre differenti velocità di percorrenza, durante la convergenza in analisi. Di seguito vengono dunque rappresentati due gruppi di grafici rappresentanti l'evoluzione dei parametri del vettore K per le tre velocità di percorrenza considerate.



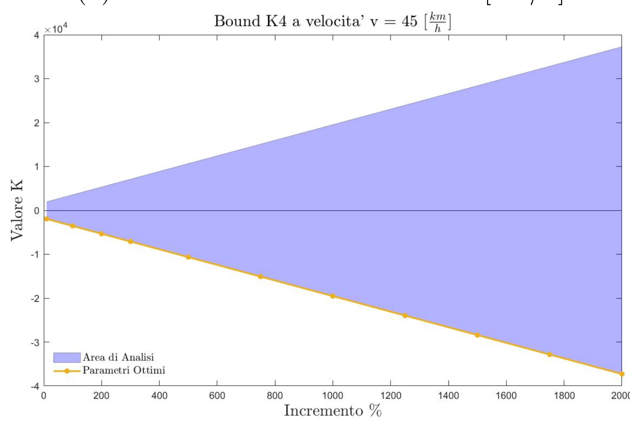
(a) Parametro k1 a velocità $v = 45[km/h]$



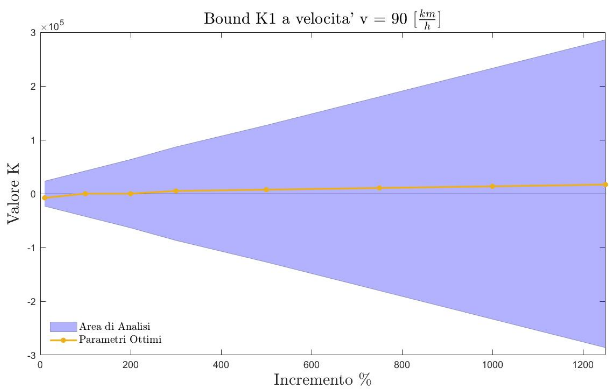
(b) Parametro k2 a velocità $v = 45[km/h]$



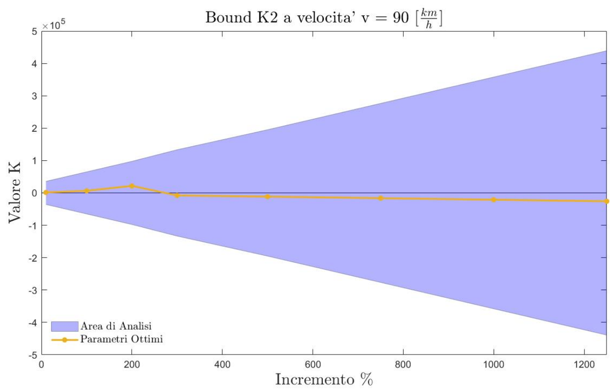
(c) Parametro k3 a velocità $v = 45[km/h]$



(d) Parametro k4 a velocità $v = 45[km/h]$

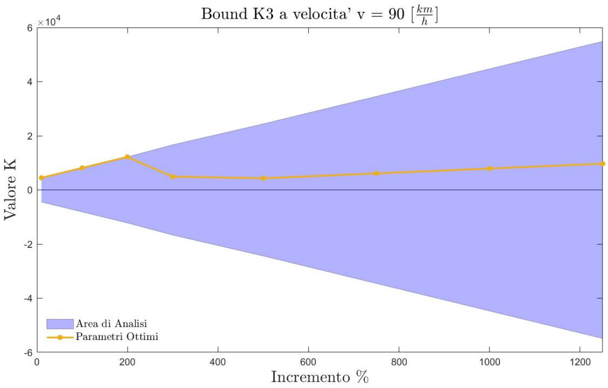


(e) Parametro k1 a velocità $v = 90[km/h]$

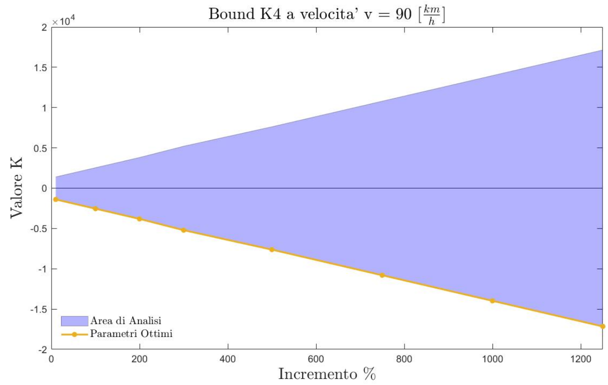


(f) Parametro k2 a velocità $v = 90[km/h]$

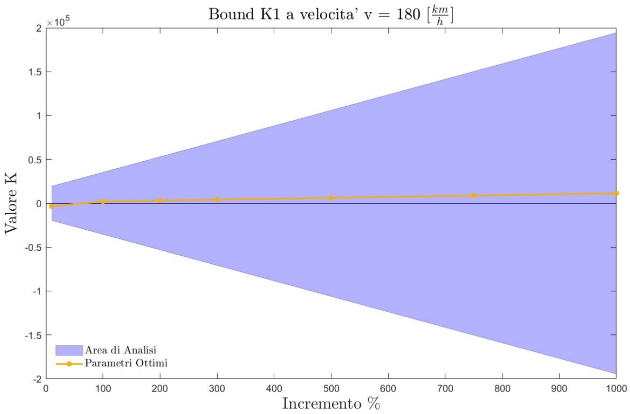
Figura 3.2: Elementi del vettore K



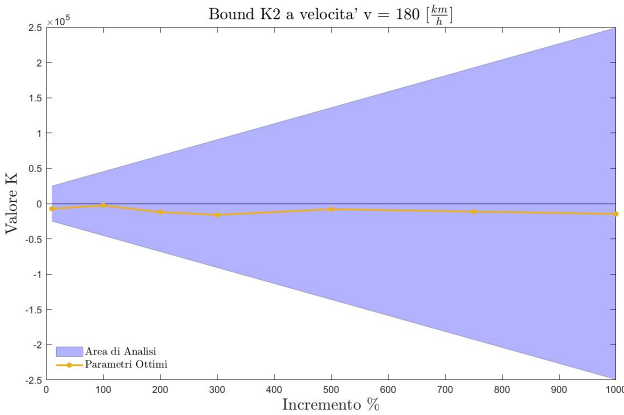
(a) Parametro k3 a velocità $v = 90[km/h]$



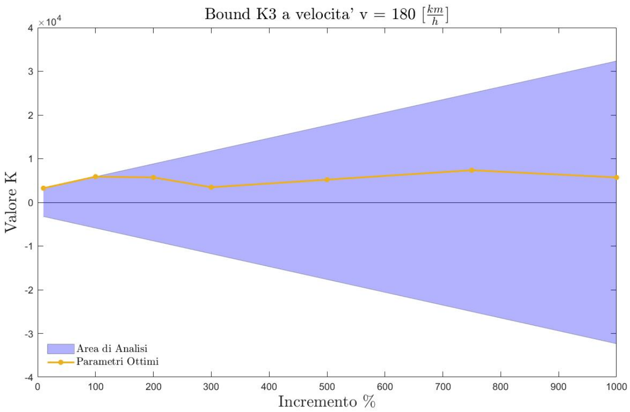
(b) Parametro k4 a velocità $v = 90[km/h]$



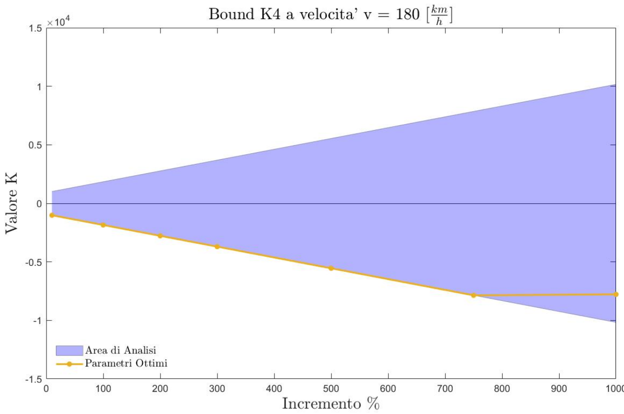
(c) Parametro k1 a velocità $v = 180[km/h]$



(d) Parametro k2 a velocità $v = 180[km/h]$



(e) Parametro k3 a velocità $v = 180[km/h]$



(f) Parametro k4 a velocità $v = 180[km/h]$

Figura 3.3: Elementi del vettore K

Nelle figure 3.2 e 3.3 rappresentanti i parametri *k1, k2, k3, k4* per le tre velocità prese in analisi (i grafici riguardanti la velocità $v = 90[km/h]$ sono stati divisi nei due gruppi di grafici per comodità di impaginazione) si può vedere sia l'area di parametri coperta dall'analisi, sia i valori che ottimizzano il problema ad ogni simulazione: si nota che per i parametri *k1* e *k2* non sono presenti delle correlazioni tra i valori ottenuti, per il parametro *k3* esiste una correlazione solo per i primi valori raccolti mentre per il parametri *k4* esiste una forte correlazione tra quasi tutti i valori raccolti in quanto presentano lo stesso comportamento. Risulta utile, a questo punto, analizzare i valori assunti dal parametro *k4* nelle analisi effettuate: si nota che questo parametro cade quasi sempre (c'è un unico caso in cui il parametro non è sul bordo, visibile nel grafico 3.3f) sul bordo dell'area di analisi, indicando che ci sia margine di miglioramento non ancora coperto o considerato dalle analisi svolte, però ricordando i dati rappresentati nel grafico 3.1 è facile convincersi che in realtà non ha senso aumentare l'area dati presa in analisi in quanto il miglioramento del comfort di guida sarebbe irrisorio.

Risulta anche utile rappresentare i vettori K che ottimizzano la cifra di costo J_{ax}:

$$K_{45} = [27035 \quad -57532 \quad 5841 \quad -37281] \tag{3.3}$$

$$K_{90} = [16832 \quad -25833 \quad 9687 \quad -17153] \tag{3.4}$$

$$K_{180} = [11435 \quad -14659 \quad 5710 \quad -7775] \tag{3.5}$$

come è possibile vedere anche qua è presente una correlazione, cioè per tutte e tre le velocità in analisi i parametri *k1* e *k3* sono sempre positivi mentre i parametri *k2* e *k4* risultano essere sempre negativi. Ricordando quali sono le componenti della variabile di stato $x = [z_{\mathrm{s}} \quad \tilde{z} \quad \dot{z}_{\mathrm{s}} \quad \dot{\tilde{z}}]^{\mathrm{T}}$ si può vedere come le correlazioni sopra indicate non siano casuali: la posizione e la velocità dello chassis della macchina vengono premoltiplicati per un valore positivo, mentre la distanza tra lo chassi e lo pneumatico e la variazione di distanza tra lo chassi e lo pneumatico vengono premoltiplicati per un valore negativo. Essendo queste correlazioni presenti tra velocità differenti tra loro, è possibile pensare che questa sia una regola applicabile per altre velocità oltre a quelle prese in considerazioni, anche se i valori di *K Bench* 2.1 vanno contro a quanto espresso fin'ora.

Va ricordato che nel caso della velocità $v = 180[km/h]$ il parametro *k4* non cade sul bordo dell'area di analisi e come verrà indicato nella sezione seguente questo dato rappresenta lo strappo alla regola, in quanto il valore della cifra di costo sulla velocità J_v anch'essa andrà contro alla tendenza dei dati raccolti.

3.2 Analisi risultati di Ottimizzazione

In questa sezione si entra nella vera e propria analisi dei dati delle simulazioni in quanto in questa sezione verranno introdotti proprio i dati in interesse di questo elaborato e verranno confrontati con i dati dell'analisi effettuata tramite *benchmark semi-attivo* e i dati delle simulazioni passive.

Come è possibile vedere nel grafico 3.4, vengono ripresi i dati mostrati nel grafico 2.2 e vengono posti a confronto con i dati raccolti nelle simulazioni semi-attive effettuate seguendo le regole indicate nella sezione 3.1. Si può notare che le cifre di costo ottenute tramite le simulazioni semi-attive hanno un margine di miglioramento minimo per quanto riguarda la cifra di costo sull'accelerazione $\mathbf{J_{ax}}$ rispetto alle simulazioni semi-attive effettuate tramite *benchmark semi-attivo*, mentre le cifre di costo sulla velocità $\mathbf{J_v}$ risultano essere peggiori in due dei tre casi rappresentati. Per comodità nella tabella 3.1 vengono inserite le cifre di costo $\mathbf{J_{ax}}$ e $\mathbf{J_v}$ indicate nel grafico 3.4 che rappresentano l'ottimizzazione del problema preso in analisi:

Velocità	**Parametro**	**Analisi Passiva**	**Analisi Benchmark**	**Analisi Semi-Attiva**
45 $\frac{m}{s}$	J_{ax}	1.2452 $\frac{m}{s^2}$	1.283 $\frac{m}{s^2}$	1.2137 $\frac{m}{s^2}$
	J_v	0.12055 $\frac{m}{s}$	0.067732 $\frac{m}{s}$	0.091745 $\frac{m}{s}$
90 $\frac{m}{s}$	J_{ax}	1.7519 $\frac{m}{s^2}$	1.6695 $\frac{m}{s^2}$	1.6418 $\frac{m}{s^2}$
	J_v	0.16753 $\frac{m}{s}$	0.10062 $\frac{m}{s}$	0.13456 $\frac{m}{s}$
180 $\frac{m}{s}$	J_{ax}	2.4624 $\frac{m}{s^2}$	2.2493 $\frac{m}{s^2}$	2.1912 $\frac{m}{s^2}$
	J_v	0.23538 $\frac{m}{s}$	0.16757 $\frac{m}{s}$	0.14873 $\frac{m}{s}$

Tabella 3.1: Rappresentazione di tutti i parametri ottimi suddivisi per velocità

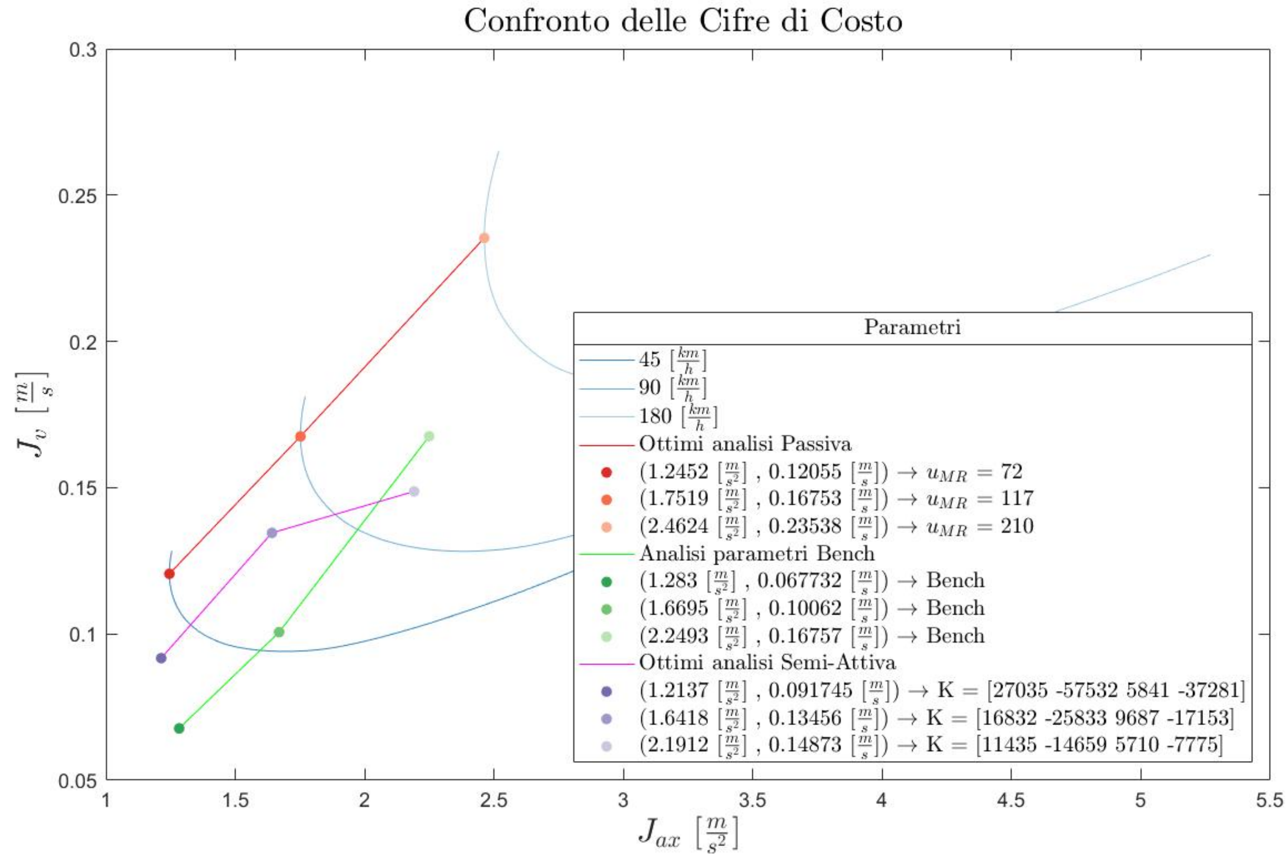


Figura 3.4: Rappresentazione dei dati raccolti nell'analisi passiva, nell'analisi tramite *benchmark semi-attivo* e nell'analisi semi-attiva

Andando a prendere in considerazione i valori della tabella 3.1 per la velocità $v = 45[km/h]$ è possibile vedere come il valore di $\mathbf{J_{ax}}$ peggiore venga ottenuto tramite l'analisi effettuata tramite *benchmark semi-attivo* mentre il valore migliore è ottenuto tramite l'analisi semi-attiva con l'utilizzo del vettore K indicato nell'espressione 3.3. Per quanto riguarda invece i valori di $\mathbf{J_v}$, il valore migliore risulta proprio dall'analisi effettuata tramite *benchmark semi-attivo* mentre, come è possibile aspettarsi, il valore peggiore è ottenuto tramite l'analisi passiva. Vedendo però il grafico 3.4 è possibile notare che l'analisi effettuata tramite *benchmark semi-attivo* offre valori di $\mathbf{J_v}$ molto più bassi rispetto ai valori dell'analisi passiva: come è giusto ricordare però, lo scopo dell'analisi è l'ottimizzazione del parametro $\mathbf{J_{ax}}$ e quindi è necessario riferire tutti i valori utili all'analisi ad esso, nonostante non siano essi magari i migliori ottenuti. A fronte dei dati elencati è possibile dire che l'analisi effettuata tramite *benchmark semi-attivo* in realtà effettua il lavoro richiesto in quanto il parametro $\mathbf{J_{ax}}$ non si distacca di molto da quello ottimo trovato tramite l'analisi dati effettuata; per quanto riguarda il parametro $\mathbf{J_v}$ invece viene effettuata un ottimizzazione molto evidente anche se non veniva ipotizzata inizialmente nel paper [2] da cui e stato tratto il problema in analisi.

Andando invece a prendere in considerazione i valori della tabella 3.1 per la velocità $v = 90[km/h]$ è possibile vedere come il valore di $\mathbf{J_{ax}}$ peggiore è ottenuto dall'analisi passiva, come era possibile aspettarsi, mentre il valore migliore è ottenuto tramite l'analisi semi-attiva con l'utilizzo del vettore K indicato nell'espressione 3.4. Per quanto riguarda invece i valori di $\mathbf{J_v}$, la tendenza è medesima a quella vista per la velocità $v = 45[km/h]$ con la predominanza dell'analisi effettuata tramite *benchmark semi-attivo*. Anche in questo caso, nonostante la presenza di un minimo margine di ottimizzazione, l'analisi effettuata tramite *benchmark semi-attivo* svolge quello che viene richiesto ed oltre in quanto ottimizza anche il parametro $\mathbf{J_v}$.

Infine, andando a prendere in considerazioni i valori della tabella 3.1 per la velocità $v = 180[km/h]$ le tendenze sopra indicate vanno un pò a sfumare. Come si può notare, l'analisi semi-attiva effettuata tramite l'utilizzo del vettore K indicato nell'espressione 3.5 restituisce sia il valore ottimo per il parametro $\mathbf{J_{ax}}$, sia per il parametro $\mathbf{J_v}$, indicando dunque un grado ulteriore di ottimizzazione fornita.

Questo è il caso che esce dalla regola e rovina le constatazioni effettuate fin'ora, in quanto l'applicazione del vettore *K Bench* viene superata, anche se di poco, dall'applicazione del vettore K trovato tramite le simulazioni.

Dopo aver analizzato questi dati è possibile certamente affermare che il vettore *K Bench* risulta essere uno tra i migliori parametri trovati in quanto ottimizza sia il parametro $\mathbf{J_{ax}}$ sia il parametro $\mathbf{J_v}$ lasciando solo minimi margini di miglioramento, per tutte tre le velocità considerate. Applicando il metodo di ricerca effettuato per trovare il vettore *K Bench* alla velocità $v = 90[km/h]$ anche ad altre velocità di certo restituirebbe vettori che eliminerebbero i margini di miglioramento sopraindicati.

3.3 Analisi risultati di Ottimizzazione con k1 fissato

In questa sezione viene presentata un'ulteriore analisi del modello di controllo che si differenzia da quella presentata nella sezione 3.2 per il metodo di selezione del parametro *k1* appartenente al vettore K. Per tutte le simulazioni effettuate per ottenere i dati rappresentati di seguito il parametro *k1* è stato mantenuto costante e pari a $k1 = 0$, per andare a vedere il comportamento del modello di controllo in questo specifico caso. Analizzando la variabile di stato $\boldsymbol{x}$ presentata nella sezione 1.1.3 è possibile notare come il parametro *k1* sia legato alla posizione verticale dello chassis $\mathbf{z_s}$: impostare il parametro $k1 = 0$ vuol dire dunque andare ad annullare il contributo della posizione verticale dello chassis $\mathbf{z_s}$ all'interno del modello di controllo. Di seguito viene effettuata un'analisi dei *Boundaries* ottenuti per questa specifica analisi e viene effettuata un'analisi dei dati ottenuti dalle simulazioni effettuate.

3.3.1 Analisi Boundaries

I parametri *k2, k3, k4*, nei dati presentati di seguito, sono stati selezionati all'interno dei *Boundaries* definiti seguendo le medesime regole presentate nella sezione 3.1 e anche in questo caso per determinare la convergenza delle analisi verso la migliore cifra di costo sull'accelerazione $\mathbf{J_{ax}}$ ottenibile è stato scelto come errore limite tra due misurazioni consecutive un errore pari a $\epsilon_{\text{lim}} = 0.5\%$, per tutte e tre le velocità di percorrenza considerate, in modo da vedere per l'appunto la velocità di convergenza delle simulazioni alle tre velocità. Solo il parametro *k1* è stato mantenuto costante. In figura 3.5 è possibile vedere due famiglie di curve separate: le curve in blu sono le medesime rappresentate in figura 3.1 mentre le curve in rosso rappresentano i dati ottenuti tramite le simulazioni con parametro *k1* fissato. Come è possibile notare le maggiorazioni applicate necessarie per ottenere la convergenza sono le medesime per le velocità $v = 45[km/h]$ e $v = 180[km/h]$, mentre per una velocità pari a $v = 90[km/h]$ è stata necessaria una maggiorazione pari al 1500%. Per una velocità pari a $v = 45[km/h]$ si può notare come la convergenza avvenga anche in questo caso per una maggiorazione pari al 2000% ma con un errore $\epsilon = 0.42\%$, invece per $v = 90[km/h]$ la convergenza avviene alla maggiorazione del 1500% con un errore inferiore al caso precedente e pari a $\epsilon \approx 0.31\%$ mentre alla velocità di $v = 180[km/h]$ la convergenza avviene alla maggiorazione del 1000% ma con un errore $\epsilon = 0.35\%$, maggiore rispetto al caso precedente. Oltre a rappresentare il grafico in figura 3.5 risulta essere comodo anche rappresentare l'evoluzione dei parametri *k1, k2, k3, k4*, per le tre differenti velocità di percorrenza, durante la convergenza in analisi effettuata con parametro $k1 = 0$.

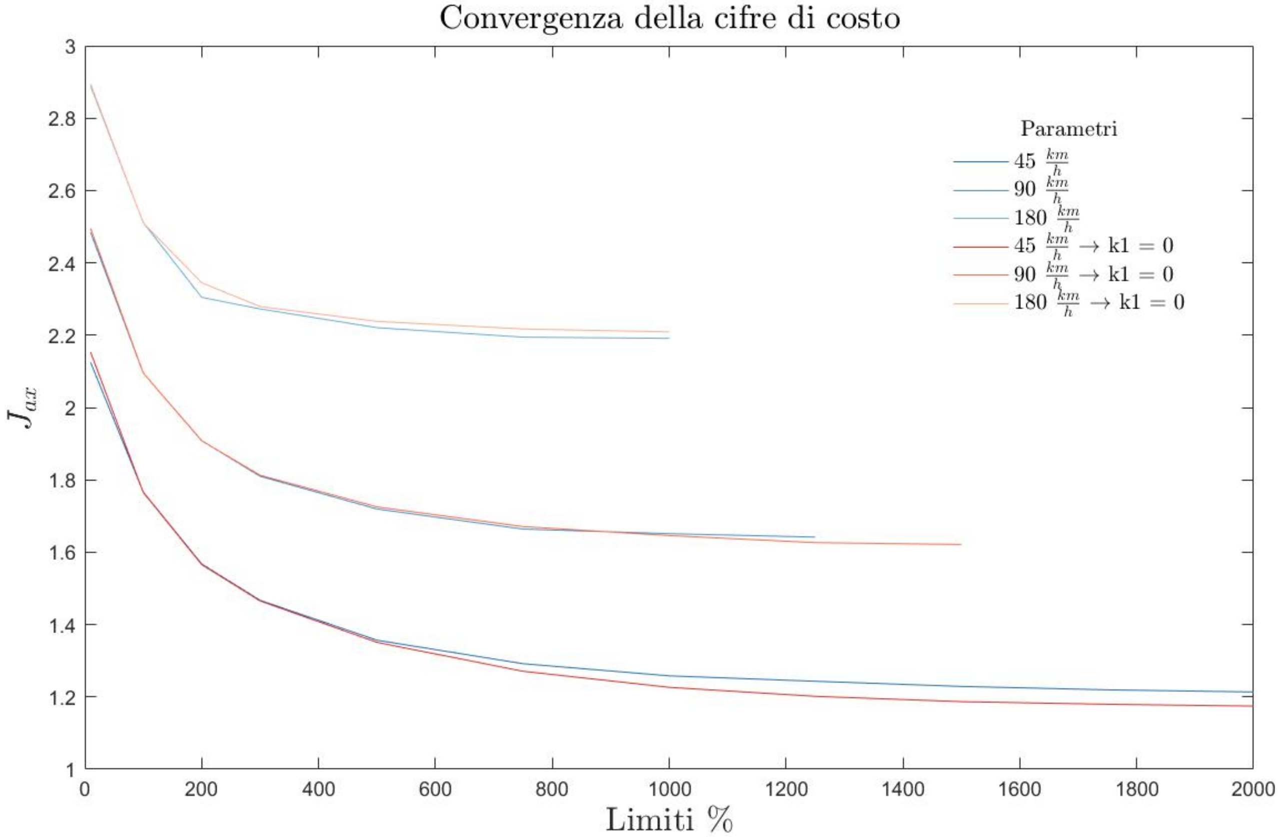
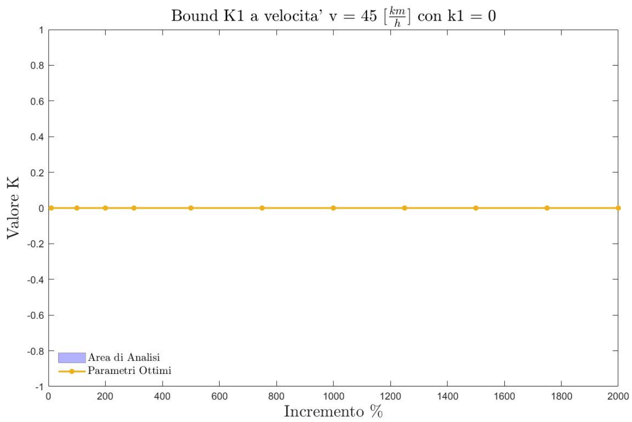


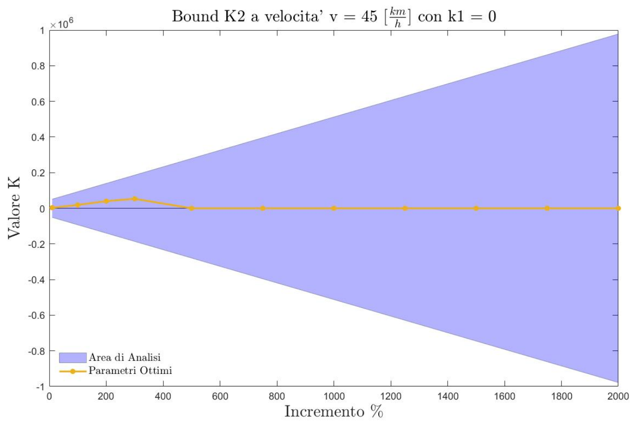
Figura 3.5: Rappresentazione delle convergenze del parametro $\mathbf{J_{ax}}$ ottimo per simulazioni con parametro *k1* variabile e costante

Come nella sezione 3.1, nelle figure 3.6 e 3.7 viene rappresentata l'evoluzione dei parametri *k1, k2, k3, k4*: per queste simulazioni si nota la stessa correlazione sui parametri *k4* mostrata nei grafici 3.6 e 3.7 mentre per il parametro *k2* si nota una nuova correlazione in quanto anche questo parametro assume valori pari a $k2 = 0$.

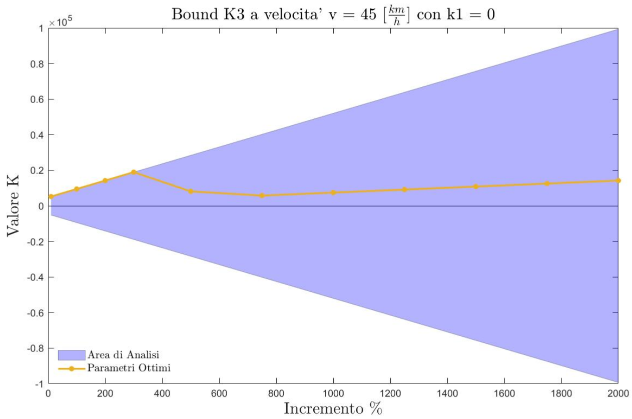
Nelle seguenti pagine vengono rappresentati due gruppi di grafici rappresentanti l'evoluzione dei parametri del vettore K per le tre velocità di percorrenza considerate per le simulazioni effettuate con parametro $k1 = 0$. Nelle figure 3.6 e 3.7 rappresentanti i parametri *k1, k2, k3, k4* per le tre velocità prese in analisi si può vedere, come nel caso precedente, sia l'area di parametri coperta dall'analisi, sia i valori che ottimizzano il problema ad ogni gruppo di simulazioni: per queste simulazioni si nota la stessa correlazione sui parametri *k4* mostrata nei grafici 3.6 e 3.7, per il parametro *k3* esiste una correlazione solo per i primi valori raccolti, mentre per il parametro *k2* si nota una nuova correlazione in quanto anche questo parametro assume valori pari a $k2 = 0$. Risulta utile, a questo punto, analizzare i valori assunti dal parametro *k2* nelle analisi effettuate: si nota che questo parametro assume quasi sempre valore pari a $k2 = 0$ tranne nelle simulazione effettuate a velocità $v = 180[km/h]$ dove questo valore viene raggiunto solo per le maggiorazioni pari a $100\%, 200\%, 750\%, 1000\%$.



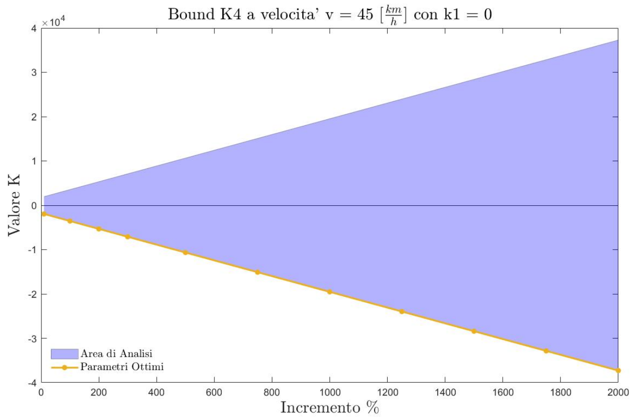
(a) Parametro k1 a velocità $v = 45[km/h]$



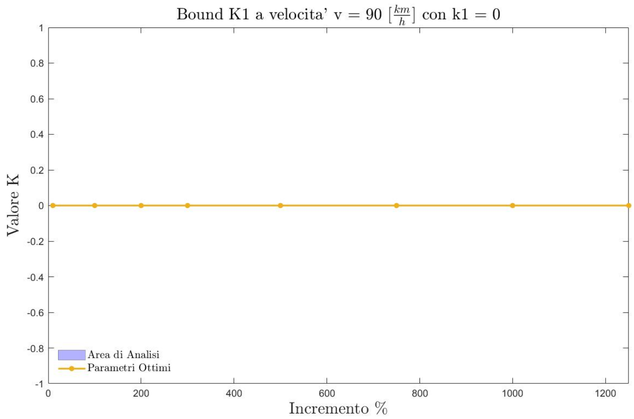
(b) Parametro k2 a velocità $v = 45[km/h]$



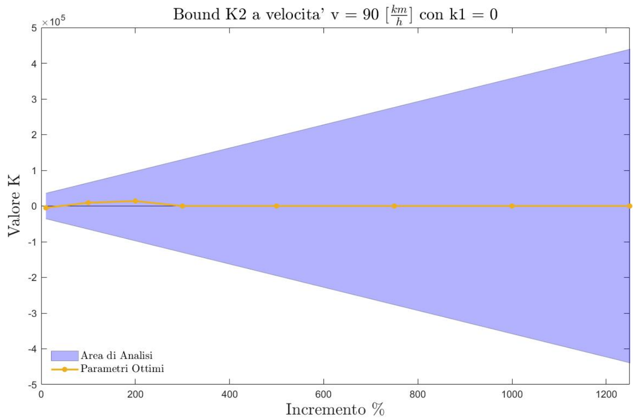
(c) Parametro k3 a velocità $v = 45[km/h]$



(d) Parametro k4 a velocità $v = 45[km/h]$

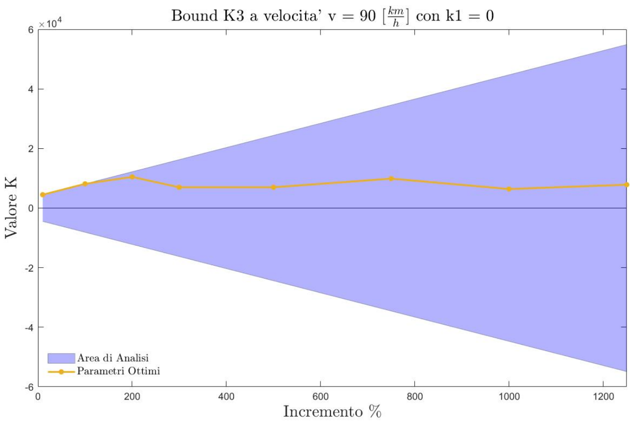


(e) Parametro k1 a velocità $v = 90[km/h]$

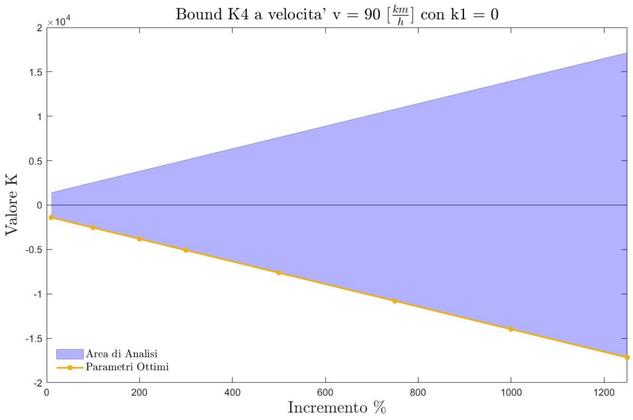


(f) Parametro k2 a velocità $v = 90[km/h]$

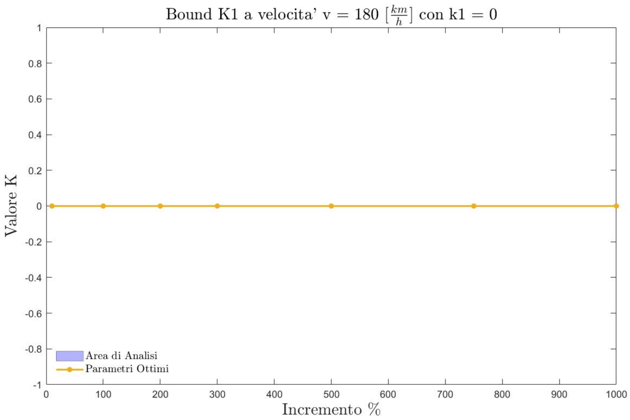
Figura 3.6: Elementi del vettore K



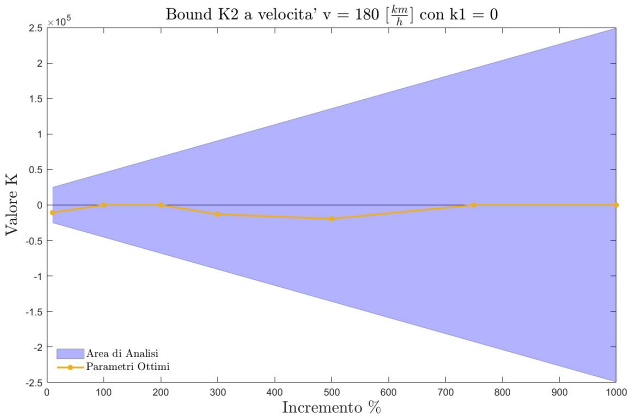
(a) Parametro k3 a velocità $v = 90[km/h]$



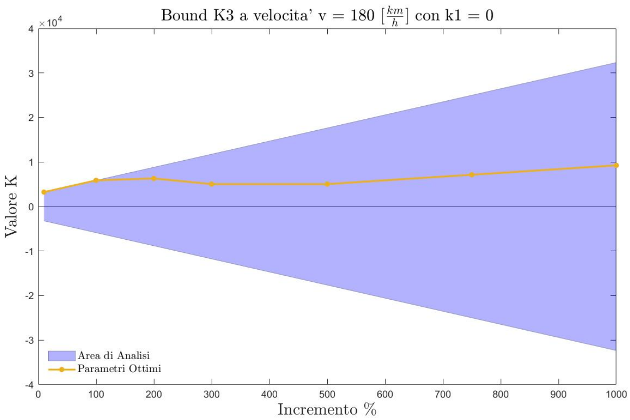
(b) Parametro k4 a velocità $v = 90[km/h]$



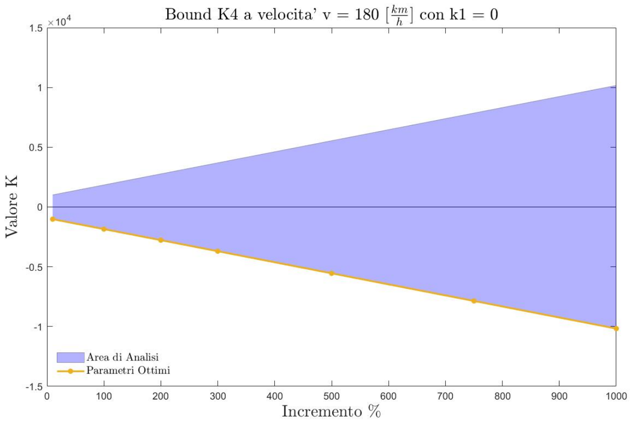
(c) Parametro k1 a velocità $v = 180[km/h]$



(d) Parametro k2 a velocità $v = 180[km/h]$



(e) Parametro k3 a velocità $v = 180[km/h]$



(f) Parametro k4 a velocità $v = 180[km/h]$

Figura 3.7: Elementi del vettore K

Anche in questo caso è anche utile rappresentare i vettori K che ottimizzano la cifra di costo $\mathbf{J_{ax}}$:

$$K_{45} = [0 \quad 0 \quad 14186 \quad -37281] \tag{3.6}$$

$$K_{90} = [0 \quad 0 \quad 9294 \quad -17425] \tag{3.7}$$

$$K_{180} = [0 \quad 0 \quad 9245 \quad -10168] \tag{3.8}$$

come è possibile vedere anche qua è presente una correlazione, cioè per tutte e tre le velocità in analisi i parametri *k1* e *k2* sono sempre nulli, i parametri *k3* sono sempre positivi mentre i parametri *k4* risultano essere sempre negativi. Ricordando quali sono le componenti della variabile di stato $x = [z_s \quad \tilde{z} \quad \dot{z}_s \quad \dot{\tilde{z}}]^T$ si può vedere come nel caso delle simulazioni effettuate con parametro *k1* fissato nullo il contributo della posizione dello chassis $\boldsymbol{z_s}$ e della distanza tra lo chassis e lo pneumatico $\boldsymbol{\tilde{z}}$ nel modello di controllo sia nullo, dunque risultano importanti solo le informazioni legate alle velocità dello chassis $\boldsymbol{\dot{z}_s}$ e alla variazione di distanza tra lo chassis e lo pneumatico $\boldsymbol{\dot{\tilde{z}}}$. Guardando i parametri *k3* e *k4* invece, la velocità dello chassis della macchina viene premoltiplicata per un valore positivo e la variazione di distanza tra lo chassis e lo pneumatico viene premoltiplicato per un valore negativo.

3.3.2 Analisi Dati

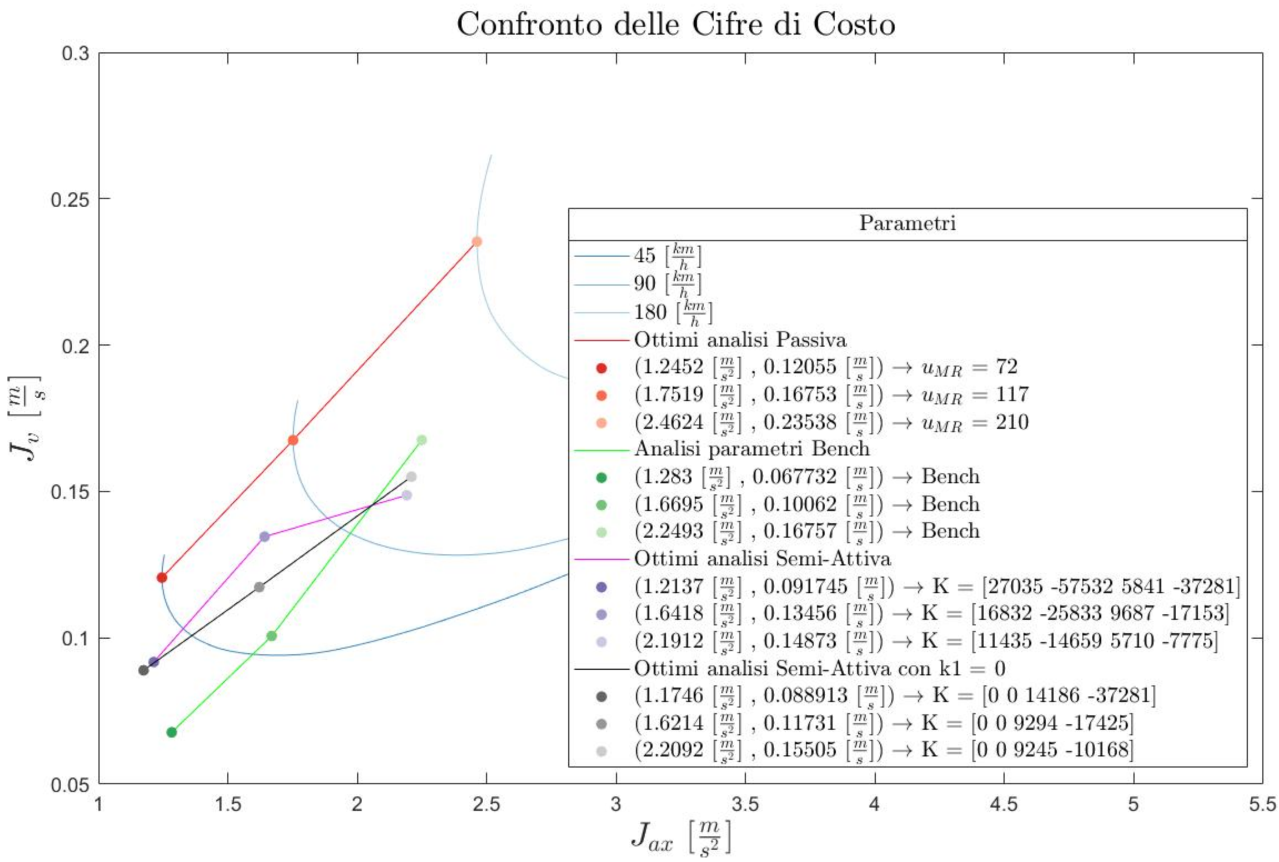


Figura 3.8: Rappresentazione dei dati raccolti nell'analisi passiva, nell'analisi tramite *benchmark semi-attivo*, nell'analisi semi-attiva e nell'analisi semi-attiva con parametro $k1 = 0$

In figura 3.8 vengono ripresi i dati mostrati nel grafico 3.4 e vengono posti a confronto con i dati raccolti nelle simulazioni semi-attive effettuate seguendo le regole indicate nella sezione 3.1, ma con parametro $k1 = 0$. Le cifre di costo ottenute tramite queste simulazioni hanno un margine di miglioramento minimo per quanto riguarda la cifra di costo sull'accelerazione $\mathbf{J_{ax}}$ rispetto alle simulazioni effettuate tramite *benchmark semi-attivo* per tutte e tre le velocità prese in considerazione, mentre rispetto alle simulazioni semi-attive, per una velocità $v = 180[km/h]$, presentano un peggioramento. Per quanto riguarda le cifre di costo sulla velocità $\mathbf{J_v}$, essere risultano migliori rispetto al caso semi-attivo e peggiori rispetto alle simulazioni effettuate tramite *benchmark semi-attivo*, tranne che per la veloctià $v = 180[km/h]$ dove il risultato viene invertito. Per comodità nella tabella 3.2 vengono inserite le cifre di costo $\mathbf{J_{ax}}$ e $\mathbf{J_v}$ indicate nel grafico 3.8 che rappresentano l'ottimizzazione del problema preso in analisi.

Considerando i valori della tabella 3.1 e della tabella 3.2 per la velocità $v = 45[km/h]$ è possibile notare come il valore di $\mathbf{J_{ax}}$ peggiore è ottenuto tramite l'analisi effettuata tramite *benchmark*

Velocità	Parametro	Analisi Benchmark	Analisi Semi-Attiva ($k1 = 0$)
45 $\frac{m}{s}$	J_{ax}	1.283 $\frac{m}{s^2}$	1.1746 $\frac{m}{s^2}$
	J_v	0.067732 $\frac{m}{s}$	0.088913 $\frac{m}{s}$
90 $\frac{m}{s}$	J_{ax}	1.6695 $\frac{m}{s^2}$	1.6214 $\frac{m}{s^2}$
	J_v	0.10062 $\frac{m}{s}$	0.11731 $\frac{m}{s}$
180 $\frac{m}{s}$	J_{ax}	2.2493 $\frac{m}{s^2}$	2.2092 $\frac{m}{s^2}$
	J_v	0.16757 $\frac{m}{s}$	0.15505 $\frac{m}{s}$

Tabella 3.2: Rappresentazione di tutti i parametri ottimi suddivisi per velocità

semi-attivo mentre il valore migliore è ottenuto tramite l'analisi semi-attiva con $k1 = 0$, cioè con l'utilizzo del vettore K indicato nell'espressione 3.6. Per quanto riguarda invece i valori di $\mathbf{J_v}$, il valore migliore risulta proprio dall'analisi effettuata tramite *benchmark semi-attivo* mentre, come è possibile aspettarsi, il valore peggiore è ottenuto tramite l'analisi passiva.

Andando invece a prendere in considerazione i valori della tabella 3.1 e della tabella 3.2 per la velocità $v = 90[km/h]$ è possibile vedere come il valore di $\mathbf{J_{ax}}$ peggiore è ottenuto dall'analisi passiva, come era possibile aspettarsi, mentre il valore migliore è ottenuto tramite l'analisi semi-attiva con $k1 = 0$, cioè con l'utilizzo del vettore K indicato nell'espressione 3.7. Per quanto riguarda invece i valori di $\mathbf{J_v}$, la tendenza è medesima a quella vista per la velocità $v = 45[km/h]$ con la predominanza dell'analisi effettuata tramite *benchmark semi-attivo*.

Infine, andando a prendere in considerazioni i valori della tabella 3.1 e della tabella 3.2 per la velocità $v = 180[km/h]$ l'analisi semi-attiva effettuata tramite l'utilizzo del vettore K indicato nell'espressione 3.5 restituisce ancora sia il valore ottimo per il parametro $\mathbf{J_{ax}}$, sia per il parametro $\mathbf{J_v}$.

3.4 Analisi Funzioni di Trasferimento

In questa sezione vengono rappresentate tutte le FdT utilizzate per confrontare l'evoluzione dei sistemi dinamici corrispondenti ai migliori valori di cifra di costo $\mathbf{J_{ax}}$ ottenuti durante l'analisi passiva, l'analisi effettuata tramite *benchmark semi-attivo*, l'analisi semi-attiva e l'analisi semi-attiva con $k1 = 0$ per tutte e tre le velocità considerate. In figura 3.9 vengono riportate le curve rappresentate in figura 2.3 e vengono aggiunte le FdT legate all'analisi semi-attiva ed all'analisi semi-attiva con $k1 = 0$. In corrispondenza della frequenza $\boldsymbol{f_1}$ è possibile notare come le curve gialla e verde presentino un picco più smussato rispetto all'analisi passiva, ma comunque con un valore massimo maggiore rispetto all'analisi effettuata tramite *benchmark semi-attivo*: questo è rispecchiato anche nelle tabelle 3.1 e 3.2 in quanto la migliore cifra di costo $\mathbf{J_v}$ è proprio ottenuta dall'analisi effettuata tramite *benchmark semi-attivo*. In corrispondenza della frequenza $\boldsymbol{f_2}$, invece, il valore ottenuto dalle tre simulazioni semi-attive è pressoché uguale, però nell'intervallo di frequenza tra 6[Hz] e 11[Hz] la curva verde risulta essere quella con valori minori di tutti e questo viene rispecchiato nelle tabelle 3.1 e 3.2 in quanto la migliore cifra di costo $\mathbf{J_{ax}}$ è ottenuta dall'analisi semi-attiva con $k1 = 0$. Nella figura 3.10 si può notare che il comportamento delle curve è medesimo a quello mostrato in figura 3.9, però in questo caso le curve arancione, gialla e verde risultano più compatte tra loro nell'intervallo di frequenza tra 6[Hz] e 11[Hz] in quanto le corrispondenti cifre di costo $\mathbf{J_{ax}}$ ottime in tabella 3.1 e in tabella 3.2 sono molto vicine tra loro.

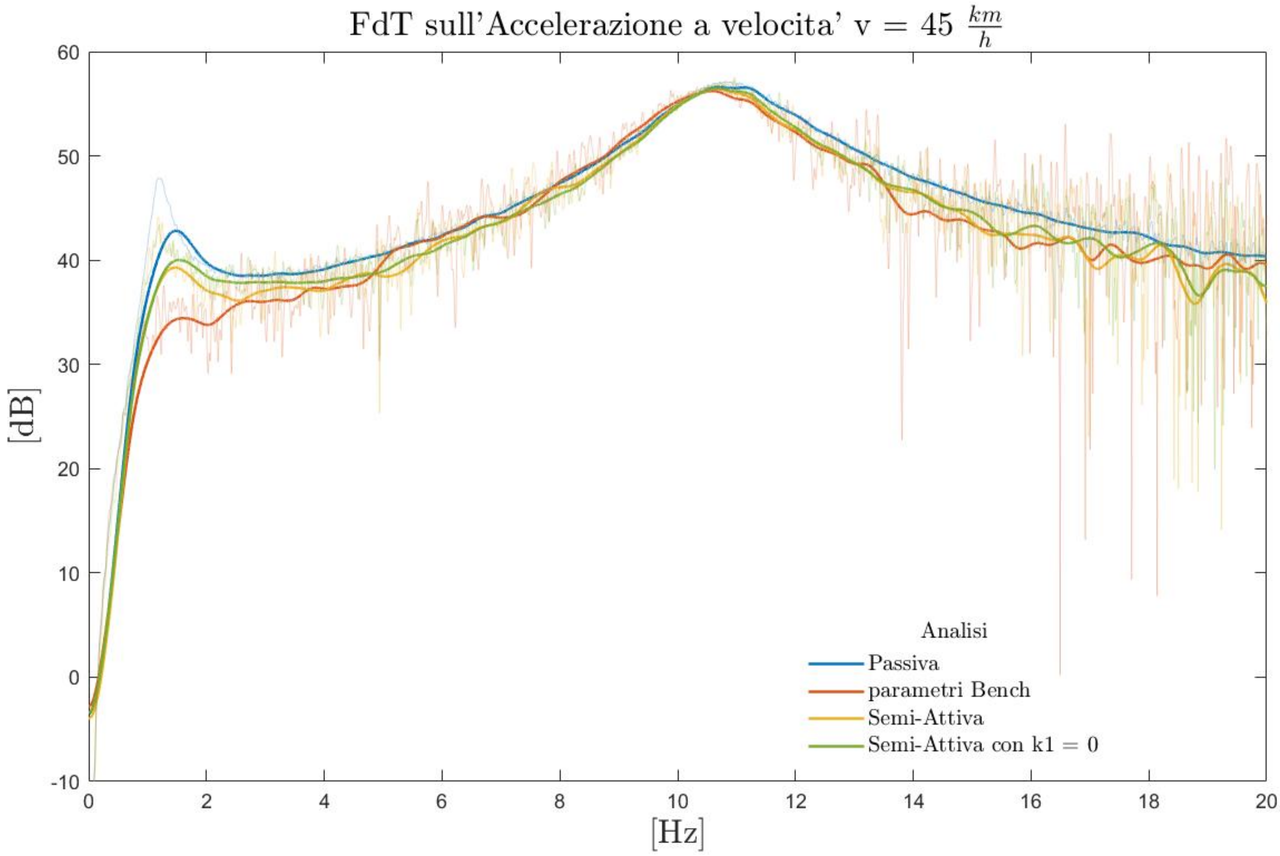


Figura 3.9: Funzioni di trasferimento rappresentati le analisi che restituiscono i valori ottimi indicati nelle tabelle 3.1 e 3.2 per la velocità $v = 45[km/h]$

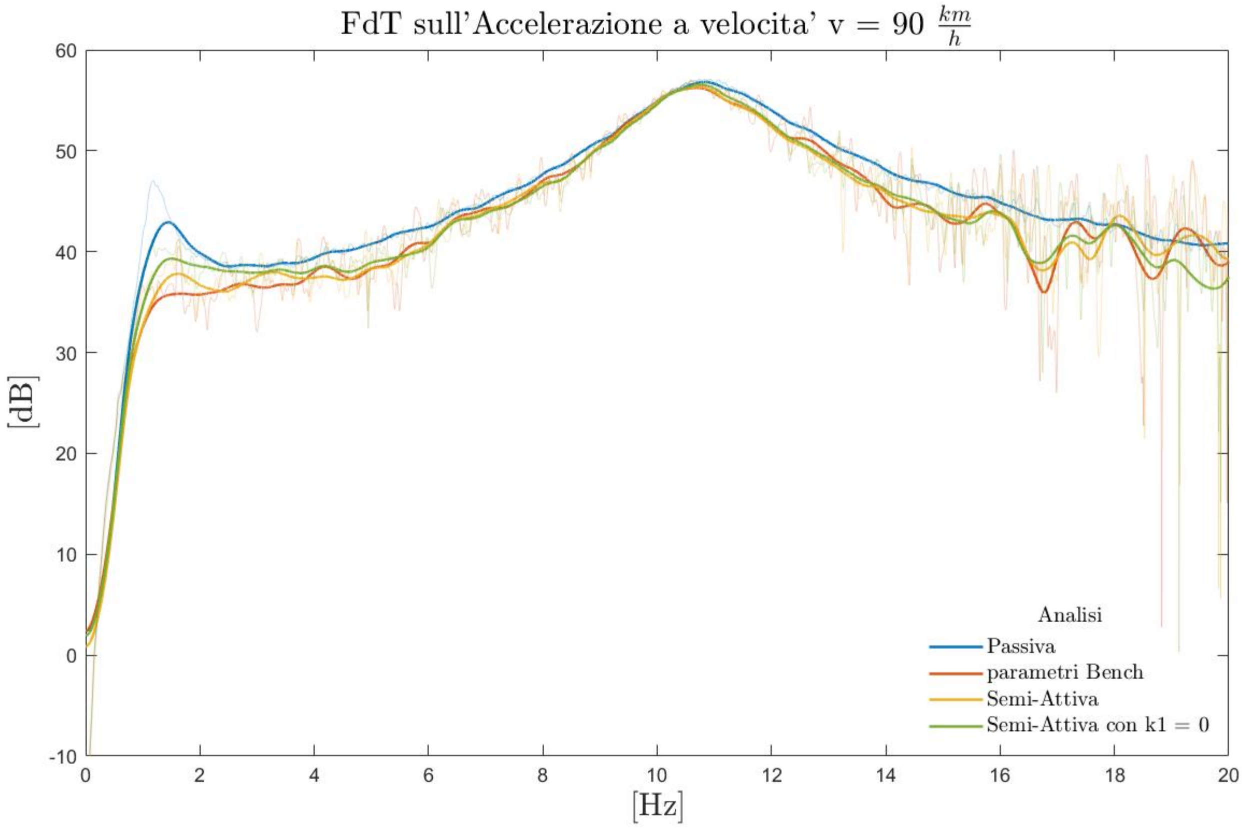


Figura 3.10: Funzioni di trasferimento rappresentati le analisi che restituiscono i valori ottimi indicati nelle tabelle 3.1 e 3.2 per la velocità $v = 90[km/h]$

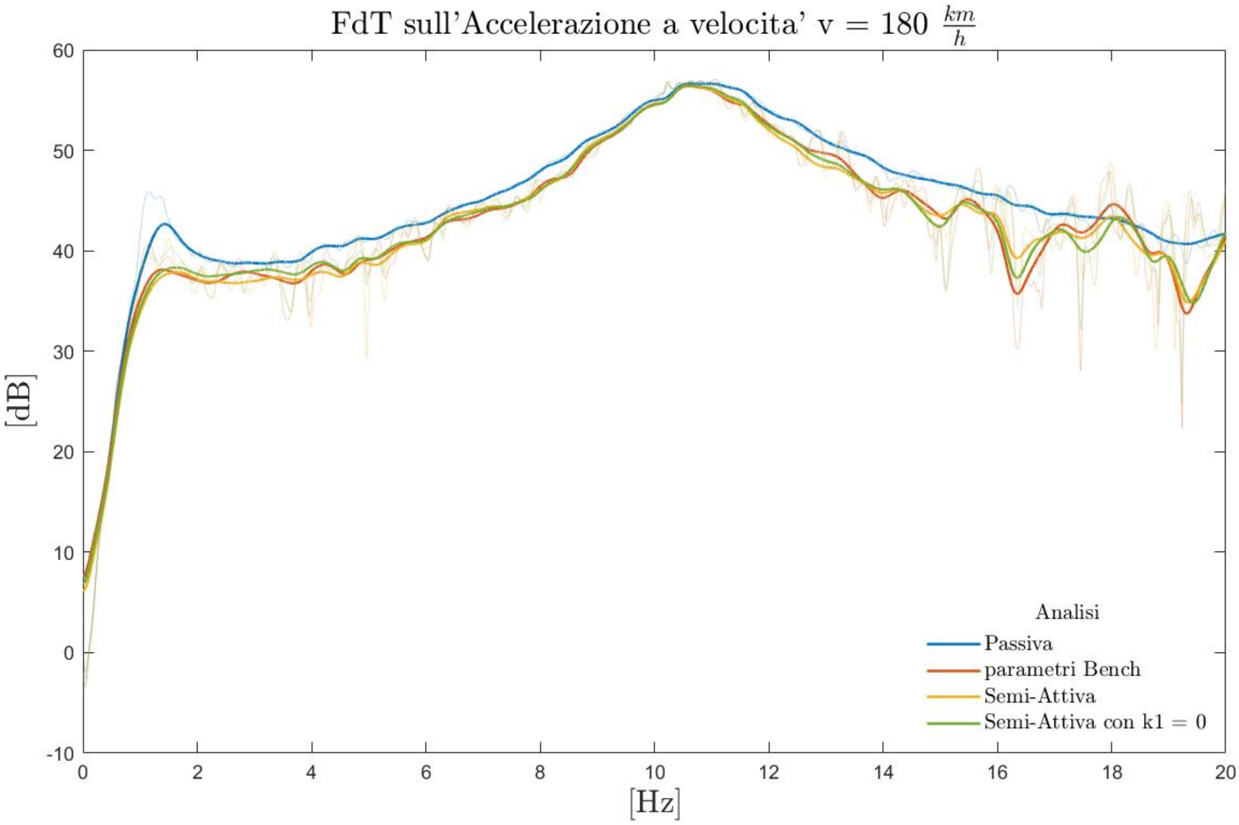


Figura 3.11: Funzioni di trasferimento rappresentati le analisi che restituiscono i valori ottimi indicati nelle tabelle 3.1 e 3.2 per la velocità $v = 180[km/h]$

Questa tendenza cambia nelle analisi effettuate a $v = 180[km/h]$, come mostrato in figura 3.11, in quanto le curve arancione, gialla e verde si sovrappongono per tutto l'Intervallo di frequenze analizzato. Questo fenomeno viene rispecchiato anche nel grafico 3.8 dove i valori ottimi delle cifre di costo sono molto simili sia per $\mathbf{J_{ax}}$ sia per $\mathbf{J_v}$. Dai grafici in figura 3.9, 3.10 e 3.11 è dunque intuibile che più i picchi in corrispondenza delle frequenze $\boldsymbol{f_1}$ e $\boldsymbol{f_2}$ sono smussati, più le cifre di costo corrispondenti risultano basse e più il comfort di guida percepito aumenta.

4 Conclusioni

A fronte dei dati presentati in questo elaborato nella figura 3.8 dove viene mostrato il confronto tra le cifre di costo $\mathbf{J_{ax}}$ e $\mathbf{J_v}$ e nelle figure 3.9, 3.10 e 3.11 dove vengono mostrate le *funzioni di trasferimento* per le tre velocità di percorrenza considerate, è possibile affermare che lo scopo dell'elaborato è stato raggiunto. L'elaborato punta a mostrare l'efficacia del vettore *K Bench* e delle analisi tramite *benchmark semi-attvo* nell'ottimizzare il comfort di guida provato dai passeggeri di un veicolo in moto, non solo alla velocità di $v = 90[km/h]$, ma anche a velocità superiori e inferiori. Come già mostrato in figura 3.8, tramite il vettore *K Bench* si raggiungono valori ottimali per tutte e tre le velocità considerate: a basse velocità, alla velocità nominale di $v = 90[km/h]$ e ad alte velocità è stato mostrato come siano possibili ulteriori ottimizzazioni della cifra di costo $\mathbf{J_{ax}}$, che però risultano irrisorie rispetto al valore ottenuto grazie alle simulazioni tramite *benchmark semi-attvo*, mentre per quanto riguarda la cifra di costo $\mathbf{J_v}$ è stata mostrata la forte efficacia del vettore *K Bench* a basse velocità ed alla velocità nominale ma non alla massima velocità considerata, dove l'analisi effettuata tramite *benchmark semi-attvo* veniva battuta dalle analisi semi-attiva e semi-attiva con parametro $k1 = 0$. Per quanto riguarda quest'ultimo tipo di simulazioni, è stato mostrato un comportamento molto interessante in quanto due dei quattro parametri del vettore K risultano nulli: quatto potrebbe suggerire l'esistenza di un ulteriore modello di controllo in grado di ottimizzare il problema considerato in analisi. Quanto detto fin'ora viene anche mostrato dalle figure 3.9, 3.10 e 3.11 dove è possibile notare comportamenti in frequenza molto simili per le tre tipologie di simulazioni semi-attive effettuate e analizzate. In conclusione, si può affermare che utilizzando il vettore *K Bench* è possibile ottimizzare il problema del comfort di guida per differenti velocità considerate, a differenza delle simulazioni semi-attiva e semi-attiva con $k1 = 0$ dove si ottengono differenti vettori K per ogni caso considerato.

Bibliografia

[1] Michele Agostinacchio, Donato Ciampa, and Saverio Olita. The vibrations induced by surface irregularities in road pavements–a matlab® approach. *European Transport Research Review*, 6(3):267–275, 2014.

[2] Ruben Begnis, Giulio Panzani, Mirko Brentari, and Luca Zaccarian. An lmi-based approach for the control of semi-active magnetorheological suspensions. *IFAC-PapersOnLine*, 53(2):14363–14368, 2020.

[3] J Gołdasz and S Dzierżek. Parametric study on the performance of automotive mr shock absorbers. In *IOP Conference Series: Materials Science and Engineering*, volume 148, page 012004. IOP Publishing, 2016.

[4] Sergio M Savaresi, Charles Poussot-Vassal, Cristiano Spelta, Olivier Sename, and Luc Dugard. *Semi-active suspension control design for vehicles.* Elsevier, 2010.

Allegato A Codice Matlab®

A.1 Codice Principale

```matlab
%% SCRIPT TESI DI LAUREA (Versione Finale)

% Lo script svolge l'analisi dati necessaria per la tesi TRIENNALE IN
% INGENIERIA MECCATRONICA all'Universita di Trento di Serafini Daniele. Lo
% scopo dello script e' quello di analizzare il Modello di QUARTER CAR
% sviluppato per la Tesi intitolata "An LMI-based approach for the control
% of semi-active magnetorheological suspensions" e rappresentato nel
% modello Simulink "ProgettoTDL.slx"

clc
close all
clear

%% DEFINIZIONE PARAMETRI ---------------------------------------------------
%--------------------------------------------------------------------------
% L'analisi viene svolta variando la velocita' della macchina, e'
% necessario variare road_omega e road_kr per mantenere costante l'energia
% del profilo stradale.

vel_vec   = [45 90 180]; % [\frac{km}{h}] Velocita da Analizzare
kr_vec    = [0.0177 0.0250 0.0354]; % ottenuti tramite script road_example
lc        = 200; % [m] Lunghezza caratteristica (risoluzione della strada)
                 % Per valori grandi di lc vengono considerate variazioni a
                 % bassa freqeunza della strada come colline, per valori
                 % bassi la strada e' "piatta".
omega_vec = 2*pi*(vel_vec/3.6)/lc;
time_vec  = 2*1000./(vel_vec/3.6); % [s] lughezza della strada 2 km

data = [kr_vec; vel_vec; time_vec];

%% PRELIMINARI -------------------------------------------------------------
%--------------------------------------------------------------------------
%% Generazione del profilo stradale

in = Simulink.SimulationInput.empty;
c = 1;

for i = 1 : 3

    in(c) = Simulink.SimulationInput('ProgettoTDL');
    in(c) = in(c).setBlockParameter('ProgettoTDL/ctr','Value','1');
    in(c) = in(c).setBlockParameter('ProgettoTDL/Fcn',...
            'Numerator',mat2str([data(1,i) 0]));
    wr = 2*pi*data(2,i)/(3.6*200);
    in(c) = in(c).setBlockParameter('ProgettoTDL/Fcn',...
            'Denominator',mat2str([1 2*0.7*wr wr*wr]));
    in(c) = setModelParameter(in(c),'StopTime',num2str(data(3,i)));
    in(c) = in(c).setBlockParameter('ProgettoTDL/u_mr',...
            'Value',num2str(0));
```

```
    in(c) = setBlockParameter(in(c),'ProgettoTDL/Subsystem/K',...
            'Multiplication','Matrix(K*u)');
    in(c) = setBlockParameter(in(c),'ProgettoTDL/Subsystem/K',...
            'Gain',mat2str([0 0 0 0]));
    in(c) = setPostSimFcn(in(c),...
            @(x) x.setUserString(num2str([data(:,i)' 0])));

    c = c+1;

end

out = parsim(in);

close(figure(1));
figure(1);

for i = 1 : 3

    p1(i) = plot(out(i).tout*vel_vec(i)/3600,out(i).zr*100); hold on;

end

xlabel('x [km]'); ylabel('zr [cm]'); title('Profilo Stradale');
title(legend,'Velocita''');
legend('45 $[\frac{km}{h}]$','90 $[\frac{km}{h}]$','180 $[\frac{km}{h}]$');
title(legend,'Velocita''');
hold off;

%% Convergenza dell'analisi

close(figure(24))

in = Simulink.SimulationInput.empty;
for c = 1 : 3

    rng(c)
    in(c) = Simulink.SimulationInput('ProgettoTDL');
    in(c) = in(c).setBlockParameter('ProgettoTDL/Random Source',...
            'seed',num2str(rand*1000));
    in(c) = in(c).setBlockParameter('ProgettoTDL/ctr','Value','1');
    in(c) = in(c).setBlockParameter('ProgettoTDL/Fcn',...
            'Numerator',mat2str([0.0250 0]));
    wr = 2*pi*90/(3.6*200);
    in(c) = in(c).setBlockParameter('ProgettoTDL/Fcn',...
            'Denominator',mat2str([1 2*0.7*wr wr*wr]));
    in(c) = setModelParameter(in(c),'StopTime',num2str(400)); % 10 km
    in(c) = in(c).setBlockParameter('ProgettoTDL/u_mr',...
            'Value',num2str(1500));
    in(c) = setBlockParameter(in(c),'ProgettoTDL/Subsystem/K',...
            'Multiplication','Matrix(K*u)');
    in(c) = setBlockParameter(in(c),'ProgettoTDL/Subsystem/K',...
            'Gain',mat2str([0 0 0 0]));
    in(c) = setPostSimFcn(in(c),...
            @(x) x.setUserString(num2str([0.0250 90 400 1500])));

end

out = parsim(in);

J_vec = zeros(10000,3);
z1 = out(1).zs_dd;
z2 = out(2).zs_dd;
z3 = out(3).zs_dd;

for i = 1 : 10000

```

```
    J_vec(i,1) = rms(z1(1:i*40));
    J_vec(i,2) = rms(z2(1:i*40));
    J_vec(i,3) = rms(z3(1:i*40));

end

figure(24)
hold on
title('Convergenza della Cifra di Costo al variare di Seed')
plot((1:1:10000),J_vec(:,1),(1:1:10000),J_vec(:,2),(1:1:10000),J_vec(:,3));
xlabel('x [m]'); ylabel('$J_{ax}$ $[\frac{m}{s^2}]$');
title(legend,'Rumore Gaussiano');
legend('Seed $\aleph1$','Seed $\aleph2$','Seed $\aleph3$');
ylim([2.5 3.5]);
hold off

%% ANALISI SOSPENSIONE PASSIVA -------------------------------------------
%-----------------------------------------------------------------------
%% Simulazioni Controllo Passivo

u_mr_vec = (0:1:3000)';

q = parsenv3(data,u_mr_vec);

save('Quartili.mat','q');

%% Confronto prestazioni $J_{ax}$ e J_v

close(figure(2))
close(figure(3))
close(figure(4))

col_mat = brewermap(6,'*Blues');

for i = 1 : 3

    matJ = getvaluesfin('Passive',data(:,i)');
    matJ = sortrows(matJ,6);

    figure(2)
    hold on
    p2(i) = plot(matJ(:,6),matJ(:,1),'Color',col_mat(i+1,:));
    hold off

    figure(3)
    hold on
    p3(i) = plot(matJ(:,1),matJ(:,2),'Color',col_mat(i+1,:));
    hold off

    figure(4)
    hold on
    p4(i) = plot(matJ(:,2),matJ(:,1),'Color',col_mat(i+1,:));
    hold off

end

figure(2)
title('Cifra di costo sull''Accelerazione di Cassa');
title(legend,'Velocita''');
legend('45 $[\frac{km}{h}]$','90 $[\frac{km}{h}]$','180 $[\frac{km}{h}]$',...
    'Location','east'); xlabel('$u_{MR}$ [N]');
ylabel('$J_{ax}$ $[\frac{m}{s^2}]$');

figure(3)
colormap(brewermap([],'Blues'))
title('Confronto delle Cifre di Costo ')
```

```
title(legend,'Parametri')
legend('45 $[\frac{km}{h}]$','90 $[\frac{km}{h}]$','180 $[\frac{km}{h}]$',...
    'Location','east'); xlabel('$J_{ax}$ $[\frac{m}{s^2}]$');
ylabel('$J_{v}$ $[\frac{m}{s}]$');

figure(4)
title('Confronto delle Cifre di Costo')
title(legend,'Velocita''')
legend('45 $[\frac{km}{h}]$','90 $[\frac{km}{h}]$','180 $[\frac{km}{h}]$',...
    'Location','east'); xlabel('$J_{v}$ $[\frac{m}{s}]$');
ylabel('$J_{ax}$ $[\frac{m}{s^2}]$');

%% Plot Saturazione Ammortizzatore

close(figure(25))
u_mr_vec = (0:1:3000)';

in = Simulink.SimulationInput.empty;
c = 1;
i = 2;

for o = 1 : 750 : length(u_mr_vec)

    in(c) = Simulink.SimulationInput('ProgettoTDL');
    in(c) = in(c).setBlockParameter('ProgettoTDL/ctr','Value','1');
    in(c) = in(c).setBlockParameter('ProgettoTDL/Fcn',...
            'Numerator',mat2str([data(1,i) 0]));
    wr = 2*pi*data(2,i)/(3.6*200);
    in(c) = in(c).setBlockParameter('ProgettoTDL/Fcn',...
            'Denominator',mat2str([1 2*0.7*wr wr*wr]));
    in(c) = setModelParameter(in(c),'StopTime',num2str(data(3,i)));
    in(c) = in(c).setBlockParameter('ProgettoTDL/u_mr',...
            'Value',num2str(u_mr_vec(o)));
    in(c) = setBlockParameter(in(c),'ProgettoTDL/Subsystem/K',...
            'Multiplication','Matrix(K*u)');
    in(c) = setBlockParameter(in(c),'ProgettoTDL/Subsystem/K',...
            'Gain',mat2str([0 0 0 0]));
    in(c) = setPostSimFcn(in(c),...
            @(x) x.setUserString(num2str([data(:,i)' u_mr_vec(o)])));

    c = c+1;

end

out = parsim(in,'ShowProgress','on','StopOnError','on');

colo = winter(5);

for i = 1 : 5

    q = out(i);

    f = 800*q.ztilda_d + q.f_tilda;

    m = zeros(length(q.tout),3);
    m(:,1) = q.ztilda_d;
    m(:,2) = q.f_tilda;
    m(:,3) = f;
    M(:,:) = sortrows(m);

    figure(25)
    hold on
    subplot(2,1,1)
    plot(M(:,1),M(:,3),'Color',colo(i,:));
    hold off
    figure(25)
```

```
    hold on
    subplot(2,1,2)
    plot(M(:,1),M(:,2),'Color',colo(i,:),'DisplayName',['$u_{MR}$ = ',...
        num2str(u_mr_vec(i)*750)]);
    hold off
    legend

end

figure(25)
subplot(2,1,1)
title('Forza-Velocita''')
ylabel('$f_{d}$ [N]')
xlim([-1.5 1.5])
subplot(2,1,2)
title('Non linearita'' Associata')
xlim([-1.5 1.5])
xlabel('$\dot{z}_{s}-\dot{z}_{u}$ $[\frac{m}{s}]$')
ylabel('$\tilde{f}_{d}$ [N]')


%% Rappresentazione del Parametro Ottimo Passivo

close(figure(5))
close(figure(6))
colo = brewermap(5,'*Reds');

ottJ_ax = zeros(3,6);
ottJ_v  = zeros(3,6);

for i = 1 : 3

    % Vado a ricercare l'ottimo in funzione delle velocita'

    matJ = getvaluesfin('Passive',data(:,i)');

    a = sortrows(matJ,1); ottJ_ax(i,:) = a(1,:);
    a = sortrows(matJ,2); ottJ_v(i,:) = a(1,:);

end

figure(3)
hold on;
plot(ottJ_ax(1:3,1),ottJ_ax(1:3,2),'r','DisplayName','Ottimi analisi Passiva');
hold off;

for i = 1 : 3

    lgd1 = ['(',num2str(ottJ_ax(i,1)),' $[\frac{m}{s^2}]$ , ',...
        num2str(ottJ_ax(i,2)),' $[\frac{m}{s}]$',...
            ') $\rightarrow$ $u_{MR}$ = ',num2str(ottJ_ax(i,6))];
    lgd2 = ['(',num2str(ottJ_v(i,1)),' $[\frac{m}{s^2}]$ , ',...
        num2str(ottJ_v(i,2)),' $[\frac{m}{s}]$',...
            ') $\rightarrow$ $u_{MR}$ = ',num2str(ottJ_v(i,6))];

    figure(3)
    hold on
    p5(i) = plot(ottJ_ax(i,1),ottJ_ax(i,2),'.','Color',colo(i+1,:),...
        'MarkerSize',20,'DisplayName',lgd1);
%     p6(i) = plot(ottJ_v(i,1),ottJ_v(i,2),'.','MarkerSize',20,...
%             'DisplayName',lgd2);
    hold off

    figure(5)
    bj = bar(vel_vec,[ottJ_ax(:,1)'; ottJ_v(:,2)']);

```

```
    figure(6)
    bu = bar(vel_vec,[ottJ_ax(:,6)'; ottJ_v(:,6)']);

end

figure(5)
hold on;
xtips1 = bj(1).XEndPoints;
ytips1 = bj(1).YEndPoints;
labels1 = string(bj(1).YData);
text(xtips1,ytips1,labels1,'HorizontalAlignment','center',...
    'VerticalAlignment','bottom'); xlabel('$[\frac{m}{s}]$');

xtips2 = bj(2).XEndPoints;
ytips2 = bj(2).YEndPoints;
labels2 = string(bj(2).YData);
text(xtips2,ytips2,labels2,'HorizontalAlignment','center',...
    'VerticalAlignment','bottom'); xlabel('$[\frac{m}{s}]$');

legend('$J_{ax}$ $[\frac{m}{s^2}]$','$J_{v}$ $[\frac{m}{s}]$',...
    'Location','northwest');
title('Rappresentazione delle Cifre di Costo ottime')
hold off;

figure(6)
hold on;
xtips1 = bu(1).XEndPoints;
ytips1 = bu(1).YEndPoints;
labels1 = string(bu(1).YData);
text(xtips1,ytips1,labels1,'HorizontalAlignment','center',...
    'VerticalAlignment','bottom'); xlabel('$[\frac{m}{s}]$');

xtips2 = bu(2).XEndPoints;
ytips2 = bu(2).YEndPoints;
labels2 = string(bu(2).YData);
text(xtips2,ytips2,labels2,'HorizontalAlignment','center',...
    'VerticalAlignment','bottom'); xlabel('$[\frac{m}{s}]$');

legend('$J_{ax}$ $[\frac{m}{s^2}]$','$J_{v}$ $[\frac{m}{s}]$',...
    'Location','northwest');
title('Rappresentazione dei Parametri $u_{MR}$ ottimi')
hold off;

%% ANALISI SOSPENSIONE SEMIATTIVA ----------------------------------------
%-----------------------------------------------------------------------
%% Simulazioni controllo semiattivo Bench

Kbench = [-18901 -45920 22704 -36338];

in = Simulink.SimulationInput.empty;
c = 1;

for i = 1 : 3

    in(c) = Simulink.SimulationInput('ProgettoTDL');
    in(c) = in(c).setBlockParameter('ProgettoTDL/ctr','Value','0');
    in(c) = in(c).setBlockParameter('ProgettoTDL/Fcn',...
            'Numerator',mat2str([data(1,i) 0]));
    wr = 2*pi*data(2,i)/(3.6*200);
    in(c) = in(c).setBlockParameter('ProgettoTDL/Fcn',...
            'Denominator',mat2str([1 2*0.7*wr wr*wr]));
    in(c) = setModelParameter(in(c),'StopTime',num2str(data(3,i)));
    in(c) = in(c).setBlockParameter('ProgettoTDL/u_mr',...
            'Value','1');
    in(c) = setBlockParameter(in(c),'ProgettoTDL/Subsystem/K',...
            'Multiplication','Matrix(K*u)');
```

```
    in(c) = setBlockParameter(in(c),'ProgettoTDL/Subsystem/K',...
            'Gain',mat2str(Kbench));
    in(c) = setPostSimFcn(in(c),...
            @(x) x.setUserString(num2str([data(:,i)' Kbench])));
    c = c+1;

end

out = parsim(in);
matbench = zeros(3,9);

for o = 1 : length(out)

    ustr = str2num(out(o).SimulationMetadata.UserString); %#ok<*ST2NM>
    title = mat2str(ustr);
    mat(1,1) = rms(out(o).zs_dd);
    mat(1,2) = rms(out(o).zs_d);
    writematrix([mat ustr],['..\Valuesfinal\',title,'.txt']);
    matbench(o,:) = [mat ustr];

end

%% Save MatBench

save('matBench.mat','matbench');

%% Rappresentazione del Parametro Bench

load 'matBench.mat';
colo = brewermap(5,'*Greens');
figure(3)
hold on;
plot(matbench(:,1),matbench(:,2),'g','DisplayName','Analisi parametri Bench');
hold off;

for i = 1 : 3

    lgd3 = ['(',num2str(matbench(i,1)),' $[\frac{m}{s^2}]$ , ',...
            num2str(matbench(i,2)),' $[\frac{m}{s}]$',...
                ') $\rightarrow$ Bench'];

    figure(3)
    hold on
    p7(i) = plot(matbench(i,1),matbench(i,2),'.','Color',colo(i+1,:),...
        'MarkerSize',20,'DisplayName',lgd3);
    hold off

end

close(figure(7))
figure(7)
bj = bar(vel_vec,[matbench(:,1)'; matbench(:,2)']);

figure(7)
hold on;
xtips1 = bj(1).XEndPoints;
ytips1 = bj(1).YEndPoints;
labels1 = string(bj(1).YData);
text(xtips1,ytips1,labels1,'HorizontalAlignment','center',...
    'VerticalAlignment','bottom'); xlabel('$[\frac{m}{s}]$');

xtips2 = bj(2).XEndPoints;
ytips2 = bj(2).YEndPoints;
labels2 = string(bj(2).YData);
text(xtips2,ytips2,labels2,'HorizontalAlignment','center',...
    'VerticalAlignment','bottom'); xlabel('$[\frac{m}{s}]$');
```

```
legend('$J_{ax}$ [$[\frac{m}{s^2}]$]','$J_{v}$ [$[\frac{m}{s}]$]',...
    'Location','northwest');
title('Confronto delle Cifre di Costo');

hold off;

%% Ottimizzazione upper/lower bound con quartili

load 'Quartili.mat';

steps = 14;

% Sezione v = 90 \frac{km}{h}

q_90 = q(q(:,6) == 90,:);
qmax_90 = max(q_90(:,1:4));
Kmax_90 = (1500./qmax_90)*16; % aumento valore 1500%


% Definisco vettore K per simulazione


k1 = 0; % in questo modo effettuo l'analisi finale
% k1 = (-Kmax_90(1):2*Kmax_90(1)/steps:Kmax_90(1)); % qua analisi iniziale

k2 = (-Kmax_90(2):2*Kmax_90(2)/steps:Kmax_90(2));
k3 = (-Kmax_90(3):2*Kmax_90(3)/steps:Kmax_90(3));
k4 = (-Kmax_90(4):2*Kmax_90(4)/steps:Kmax_90(4));

K_90 = zeros(length(k1)*length(k2)*length(k3)*length(k4),4);
c = 1;

for i = 1 : length(k1)
    for j = 1 : length(k2)
        for k = 1 : length(k3)
            for l = 1 : length(k4)
                K_90(c,:) = [k1(i) k2(j) k3(k) k4(l)];
                c = c+1;
            end
        end
    end
end

K_90 = fix(K_90); % fix per avere interi

% Sezione v = 45 \frac{km}{h}

q_45 = q(q(:,6) == 45,:);
qmax_45 = max(q_45(:,1:4));
Kmax_45 = (1500./qmax_45)*21; % aumento valore 2000%

% Definisco vettore K per simulazione

k1 = 0; % in questo modo effettuo l'analisi finale
% k1 = (-Kmax_45(1):2*Kmax_45(1)/steps:Kmax_45(1)); % qua analisi iniziale

k2 = (-Kmax_45(2):2*Kmax_45(2)/steps:Kmax_45(2));
k3 = (-Kmax_45(3):2*Kmax_45(3)/steps:Kmax_45(3));
k4 = (-Kmax_45(4):2*Kmax_45(4)/steps:Kmax_45(4));

K_45 = zeros(length(k1)*length(k2)*length(k3)*length(k4),4);
c = 1;

for i = 1 : length(k1)
    for j = 1 : length(k2)
```

```
        for k = 1 : length(k3)
            for l = 1 : length(k4)
                K_45(c,:) = [k1(i) k2(j) k3(k) k4(l)];
                c = c+1;
            end
        end
    end
end

K_45 = fix(K_45); % fix per avere interi

% Sezione v = 180 \frac{km}{h}

q_180 = q(q(:,6) == 180,:);
qmax_180 = max(q_180(:,1:4));
Kmax_180 = (1500./qmax_180)*11; % aumento valore 1000%

% Definisco vettore K per simulazione

k1 = 0; % in questo modo effettuo l'analisi finale
%k1 = (-Kmax_180(1):2*Kmax_180(1)/steps:Kmax_180(1)); %qua analisi iniziale

k2 = (-Kmax_180(2):2*Kmax_180(2)/steps:Kmax_180(2));
k3 = (-Kmax_180(3):2*Kmax_180(3)/steps:Kmax_180(3));
k4 = (-Kmax_180(4):2*Kmax_180(4)/steps:Kmax_180(4));

K_180 = zeros(length(k1)*length(k2)*length(k3)*length(k4),4);
c = 1;

for i = 1 : length(k1)
    for j = 1 : length(k2)
        for k = 1 : length(k3)
            for l = 1 : length(k4)
                K_180(c,:) = [k1(i) k2(j) k3(k) k4(l)];
                c = c+1;
            end
        end
    end
end

K_180 = fix(K_180); % fix per avere interi

% Matrici K
save('MatriciK.mat','K_180','K_90','K_45');

clear k1 k2 k3 k4 i j k l c ...
      q q_45 q_90 q_180 q_max45 q_max90 q_max180 steps;

%% Simulazione Controllo Semiattivo v = 45 \frac{km}{h}

load('MatriciK.mat');
load('matJcontv45.mat');
clear matJv45;

[matJv45,¬] = parsenfilev2(data(:,1),K_45);

matJcontv45 = unique([matJcontv45; matJv45],'rows');

save('matJcontv45.mat','matJcontv45','matJv45');

writematrix(matJcontv45,'ValK45.txt');

% Rappresentazione Minimi su K3 e K4

tic
figure(1)
```

```
h1 = scatter3(matJv45(:,8),matJv45(:,9),matJv45(:,1),[],matJv45(:,1));
xlabel('K3');
ylabel('K4');
zlabel('J');
colormap(parula);
figure(2)
h2 = scatter3(matJcontv45(:,8),matJcontv45(:,9),matJcontv45(:,1),...
     [],matJcontv45(:,1));
xlabel('K3');
ylabel('K4');
zlabel('J');
colormap(parula);
h1.Marker = '.';
h2.Marker = '.';
toc

%% Simulazione Controllo Semiattivo v = 45 \frac{km}{h} con k1 = 0

load('MatriciK.mat');
load('matJcontv45k1.mat');
clear matJv45k1;

[matJv45k1,¬] = parsenfilev2(data(:,1),K_45);

matJcontv45k1 = unique([matJcontv45k1; matJv45k1],'rows');

save('matJcontv45k1.mat','matJcontv45k1','matJv45k1');

writematrix(matJcontv45k1,'ValK45k1.txt');

%% Simulazione Controllo Semiattivo v = 90 \frac{km}{h}

load('MatriciK.mat');
load('matJcontv90.mat');
clear matJv90;

[matJv90,¬] = parsenfilev2(data(:,2),K_90);

matJcontv90 = unique([matJcontv90; matJv90],'rows');

save('matJcontv90.mat','matJcontv90','matJv90');

writematrix(matJcontv90,'ValK90.txt');

% Rappresentazione Minimi su K3 e K4

tic
figure(26)
h1 = scatter3(matJv90(:,8),matJv90(:,9),matJv90(:,1),[],matJv90(:,1));
xlabel('K3');
ylabel('K4');
zlabel('J');
colormap(parula);
figure(27)
h2 = scatter3(matJcontv90(:,8),matJcontv90(:,9),matJcontv90(:,1),...
     [],matJcontv90(:,1));
xlabel('K3');
ylabel('K4');
zlabel('J');
colormap(parula);
h1.Marker = '.';
h2.Marker = '.';
toc

%% Simulazione Controllo Semiattivo v = 90 \frac{km}{h} con k1 = 0

```

```
load('MatriciK.mat');
load('matJcontv90k1.mat');
clear matJv90k1;

[matJv90k1,¬] = parsenfilev2(data(:,2),K_90);

matJcontv90k1 = unique([matJcontv90k1; matJv90k1],'rows');

save('matJcontv90k1.mat','matJcontv90k1','matJv90k1');

writematrix(matJcontv90k1,'ValK90k1.txt');

%% Simulazione Controllo Semiattivo v = 180 \frac{km}{h}

load('MatriciK.mat');
load('matJcontv180.mat');
clear matJv180;

[matJv180,¬] = parsenfilev2(data(:,3),K_180);

matJcontv180 = unique([matJcontv180; matJv180],'rows');

save('matJcontv180.mat','matJcontv180','matJv180');

writematrix(matJcontv180,'ValK180.txt');

% Rappresentazione Minimi su K3 e K4

tic
figure(1)
h1 = scatter3(matJv180(:,8),matJv180(:,9),matJv180(:,1),[],matJv180(:,1));
xlabel('K3');
ylabel('K4');
zlabel('J');
colormap(parula);
figure(2)
h2 = scatter3(matJcontv180(:,8),matJcontv180(:,9),matJcontv180(:,1),...
     [],matJcontv180(:,1));
xlabel('K3');
ylabel('K4');
zlabel('J');
colormap(parula);
h1.Marker = '.';
h2.Marker = '.';
toc

%% Simulazione Controllo Semiattivo v = 180 \frac{km}{h} con k1 = 0

load('MatriciK.mat');
load('matJcontv180k1.mat');
clear matJv180k1;

[matJv180k1,¬] = parsenfilev2(data(:,3),K_180);

matJcontv180k1 = unique([matJcontv180k1; matJv180k1],'rows');

save('matJcontv180k1.mat','matJcontv180k1','matJv180k1');

writematrix(matJcontv180k1,'ValK180k1.txt');

%% Rappresentazione del Parametro Ottimo Semi-Attivo

load 'matAttivo.mat';
colo = brewermap(5,'*Purples');
figure(3)
hold on;
```

```
plot(matattivo(:,1),matattivo(:,2),'m','DisplayName','Ottimi analisi Semi-Attiva');
hold off;

for i = 1 : 3

    lgd3 = ['(',num2str(matattivo(i,1)),' $[\frac{m}{s^2}]$ , ',...
            num2str(matattivo(i,2)),' $[\frac{m}{s}]$',...
                ') $\rightarrow$ K = ',mat2str(matattivo(i,6:9))];

    figure(3)
    hold on
    p7(i) = plot(matattivo(i,1),matattivo(i,2),'.','Color',colo(i+1,:),...
        'MarkerSize',20,'DisplayName',lgd3);
    hold off

end

close(figure(7))

figure(7)
bj = bar(vel_vec,[matattivo(:,1)'; matattivo(:,2)']);

figure(7)
hold on;
xtips1 = bj(1).XEndPoints;
ytips1 = bj(1).YEndPoints;
labels1 = string(bj(1).YData);
text(xtips1,ytips1,labels1,'HorizontalAlignment','center',...
    'VerticalAlignment','bottom'); xlabel('$[\frac{m}{s}]$');

xtips2 = bj(2).XEndPoints;
ytips2 = bj(2).YEndPoints;
labels2 = string(bj(2).YData);
text(xtips2,ytips2,labels2,'HorizontalAlignment','center',...
    'VerticalAlignment','bottom'); xlabel('$[\frac{m}{s}]$');

legend('$J_{ax}$ [$[\frac{m}{s^2}]$]','$J_{v}$ [$[\frac{m}{s}]$]',...
    'Location','northwest');
title('Confronto delle Cifre di Costo');

hold off;

%% Rappresentazione del Parametro Ottimo Semi-Attivo con k1 = 0

load 'matAttivok1.mat';
colo = brewermap(5,'*Greys');
figure(3)
hold on;
plot(matattivok1(:,1),matattivok1(:,2),'k','DisplayName',...
    'Ottimi analisi Semi-Attiva con k1 = 0');
hold off;

for i = 1 : 3

    lgd3 = ['(',num2str(matattivok1(i,1)),' $[\frac{m}{s^2}]$ , ',...
            num2str(matattivok1(i,2)),' $[\frac{m}{s}]$',...
                ') $\rightarrow$ K = ',mat2str(matattivok1(i,6:9))];

    figure(3)
    hold on
    p7(i) = plot(matattivok1(i,1),matattivok1(i,2),'.','Color',colo(i+1,:),...
        'MarkerSize',20,'DisplayName',lgd3);
    hold off

end

```

```
close(figure(30))

figure(30)
bj = bar(vel_vec,[matattivok1(:,1)'; matattivok1(:,2)']);

figure(30)
hold on;
xtips1 = bj(1).XEndPoints;
ytips1 = bj(1).YEndPoints;
labels1 = string(bj(1).YData);
text(xtips1,ytips1,labels1,'HorizontalAlignment','center',...
    'VerticalAlignment','bottom'); xlabel('$[\frac{m}{s}]$');

xtips2 = bj(2).XEndPoints;
ytips2 = bj(2).YEndPoints;
labels2 = string(bj(2).YData);
text(xtips2,ytips2,labels2,'HorizontalAlignment','center',...
    'VerticalAlignment','bottom'); xlabel('$[\frac{m}{s}]$');

legend('$J_{ax}$ [$[\frac{m}{s^2}]$]','$J_{v}$ [$[\frac{m}{s}]$]',...
    'Location','northwest');
title('Confronto delle Cifre di Costo');

hold off;

%% ANALISI DATI EXCEL -----------------------------------------------------
%------------------------------------------------------------------------

load 'MExcel.mat';
load 'MExcelBound.mat';
load 'MExcelPercent.mat';
load 'MExcelOttimi.mat';

load 'MExcelk1.mat';
load 'MExcelBoundk1.mat';
load 'MExcelPercentk1.mat';
load 'MExcelOttimik1.mat';

%% velocita'' 45 \frac{km}{h}

for i = 1 : 4

    close(figure(i+7))

    figure(i+7)
    hold on
    area(MExcel45Percent,MExcel45Bound(:,i),'FaceColor','b',...
        'FaceAlpha',.3,'EdgeAlpha',.3,'DisplayName','Area di Analisi');
    h = area(MExcel45Percent,-MExcel45Bound(:,i),'FaceColor','b',...
        'FaceAlpha',.3,'EdgeAlpha',.3,'DisplayName','');
    h.Annotation.LegendInformation.IconDisplayStyle = 'off';
    plot(MExcel45Percent,MExcel45(:,i),'Marker','.','MarkerSize',20,...
        'LineWidth',2,'DisplayName','Parametri Ottimi')
    hold off
    title(['Bound K',num2str(i),' a velocita'' v = 45 $[\frac{km}{h}]$'])
    xlabel('Incremento $\%$')
    ylabel('Valore K')
    legend('Location','southwest')

end

%% velocita'' 45 \frac{km}{h} k1 = 0

for i = 1 : 4

    close(figure(i+7))
```

```
    figure(i+7)
    hold on
    area(MExcel45Percentk1,MExcel45Boundk1(:,i),'FaceColor','b',...
        'FaceAlpha',.3,'EdgeAlpha',.3,'DisplayName','Area di Analisi');
    h = area(MExcel45Percentk1,-MExcel45Boundk1(:,i),'FaceColor','b',...
        'FaceAlpha',.3,'EdgeAlpha',.3,'DisplayName','');
    h.Annotation.LegendInformation.IconDisplayStyle = 'off';
    plot(MExcel45Percentk1,MExcel45k1(:,i),'Marker','.','MarkerSize',20,...
        'LineWidth',2,'DisplayName','Parametri Ottimi')
    hold off
    title(['Bound K',num2str(i),' a velocita'' v = 45 $[\frac{km}{h}]$ con k1 = 0'])
    xlabel('Incremento $\%$')
    ylabel('Valore K')
    legend('Location','southwest')

end

%% velocita'' 90 \frac{km}{h}

for i = 1 : 4

    close(figure(i+11))

    figure(i+11)
    hold on
    area(MExcel90Percent,MExcel90Bound(:,i),'FaceColor','b',...
        'FaceAlpha',.3,'EdgeAlpha',.3,'DisplayName','Area di Analisi');
    h = area(MExcel90Percent,-MExcel90Bound(:,i),'FaceColor','b',...
        'FaceAlpha',.3,'EdgeAlpha',.3,'DisplayName','');
    h.Annotation.LegendInformation.IconDisplayStyle = 'off';
    plot(MExcel90Percent,MExcel90(:,i),'Marker','.','MarkerSize',20,...
        'LineWidth',2,'DisplayName','Parametri Ottimi')
    hold off
    title(['Bound K',num2str(i),' a velocita'' v = 90 $[\frac{km}{h}]$'])
    xlabel('Incremento $\%$')
    ylabel('Valore K')
    xlim([0 1250])
    legend('Location','southwest')

end

%% velocita'' 90 \frac{km}{h} k1 = 0

for i = 1 : 4

    close(figure(i+11))

    figure(i+11)
    hold on
    area(MExcel90Percentk1,MExcel90Boundk1(:,i),'FaceColor','b',...
        'FaceAlpha',.3,'EdgeAlpha',.3,'DisplayName','Area di Analisi');
    h = area(MExcel90Percentk1,-MExcel90Boundk1(:,i),'FaceColor','b',...
        'FaceAlpha',.3,'EdgeAlpha',.3,'DisplayName','');
    h.Annotation.LegendInformation.IconDisplayStyle = 'off';
    plot(MExcel90Percentk1,MExcel90k1(:,i),'Marker','.','MarkerSize',20,...
        'LineWidth',2,'DisplayName','Parametri Ottimi')
    hold off
    title(['Bound K',num2str(i),' a velocita'' v = 90 $[\frac{km}{h}]$ con k1 = 0'])
    xlabel('Incremento $\%$')
    ylabel('Valore K')
    xlim([0 1250])
    legend('Location','southwest')

end
```

```matlab
%% velocita'' 180 \frac{km}{h}

for i = 1 : 4

    close(figure(i+15))

    figure(i+15)
    hold on
    area(MExcel180Percent,MExcel180Bound(:,i),'FaceColor','b',...
        'FaceAlpha',.3,'EdgeAlpha',.3,'DisplayName','Area di Analisi');
    h = area(MExcel180Percent,-MExcel180Bound(:,i),'FaceColor','b',...
        'FaceAlpha',.3,'EdgeAlpha',.3,'DisplayName','');
    h.Annotation.LegendInformation.IconDisplayStyle = 'off';
    plot(MExcel180Percent,MExcel180(:,i),'Marker','.','MarkerSize',20,...
        'LineWidth',2,'DisplayName','Parametri Ottimi')
    hold off
    title(['Bound K',num2str(i),' a velocita'' v = 180 $[\frac{km}{h}]$'])
    xlabel('Incremento $\%$')
    ylabel('Valore K')
    legend('Location','southwest')

end

%% velocita'' 180 \frac{km}{h} k1 = 0

for i = 1 : 4

    close(figure(i+15))

    figure(i+15)
    hold on
    area(MExcel180Percentk1,MExcel180Boundk1(:,i),'FaceColor','b',...
        'FaceAlpha',.3,'EdgeAlpha',.3,'DisplayName','Area di Analisi');
    h = area(MExcel180Percentk1,-MExcel180Boundk1(:,i),'FaceColor','b',...
        'FaceAlpha',.3,'EdgeAlpha',.3,'DisplayName','');
    h.Annotation.LegendInformation.IconDisplayStyle = 'off';
    plot(MExcel180Percentk1,MExcel180k1(:,i),'Marker','.','MarkerSize',20,...
        'LineWidth',2,'DisplayName','Parametri Ottimi')
    hold off
    title(['Bound K',num2str(i),' a velocita'' v = 180 $[\frac{km}{h}]$ con k1 = 0'])
    xlabel('Incremento $\%$')
    ylabel('Valore K')
    legend('Location','southwest')

end

%% Grafici Tendenze

close(figure(20))
figure(20)
hold on
plot(MExcel45Percent,MExcel45Ott,MExcel90Percent,MExcel90Ott,...
    MExcel180Percent,MExcel180Ott)
xlabel('Limiti $\%$');ylabel('$J_{ax}$');
title('Convergenza delle cifre di costo')
title(legend,'Velocita''')
legend('45 $\frac{km}{h}$','90 $\frac{km}{h}$','180 $\frac{km}{h}$')
hold off

%% Grafici Tendenze k1 = 0

col_mat1 = brewermap(7,'*Blues');
col_mat2 = brewermap(5,'*Reds');
close(figure(20))
figure(20)
hold on
```

```
plot(MExcel45Percent,MExcel45Ott,'Color',col_mat1(2,:))
plot(MExcel90Percent,MExcel90Ott,'Color',col_mat1(3,:))
plot(MExcel180Percent,MExcel180Ott,'Color',col_mat1(4,:))
plot(MExcel45Percentk1,MExcel45Ottk1,'Color',col_mat2(2,:))
plot(MExcel90Percentk1,MExcel90Ottk1,'Color',col_mat2(3,:))
plot(MExcel180Percentk1,MExcel180Ottk1,'Color',col_mat2(4,:))
legend('45 $\frac{km}{h}$','90 $\frac{km}{h}$','180 $\frac{km}{h}$',...
    '45 $\frac{km}{h}$ $\rightarrow$ k1 = 0',...
    '90 $\frac{km}{h}$ $\rightarrow$ k1 = 0',...
    '180 $\frac{km}{h}$ $\rightarrow$ k1 = 0')
xlabel('Limiti $\%$');ylabel('$J_{ax}$');
title('Convergenza della cifre di costo')
title(legend,'Parametri')
hold off

%% Funzione di Trasferimento dei parametri ottimi per ogni velocita''

load 'matBench.mat';
load 'matAttivo.mat';
load 'matAttivok1.mat';

bench = matbench(1,6:9);

% Passiva

inp = Simulink.SimulationInput.empty;

for c = 1 : 3

    inp(c) = Simulink.SimulationInput('ProgettoTDL');
    inp(c) = inp(c).setBlockParameter('ProgettoTDL/ctr','Value','1');
    inp(c) = inp(c).setBlockParameter('ProgettoTDL/Fcn',...
            'Numerator',mat2str([data(1,c) 0]));
    wr = 2*pi*data(2,c)/(3.6*200);
    inp(c) = inp(c).setBlockParameter('ProgettoTDL/Fcn',...
            'Denominator',mat2str([1 2*0.7*wr wr*wr]));
    inp(c) = setModelParameter(inp(c),'StopTime',num2str(data(3,c)));
    inp(c) = inp(c).setBlockParameter('ProgettoTDL/u_mr',...
            'Value',num2str(ottJ_ax(c,6)));
    inp(c) = setBlockParameter(inp(c),'ProgettoTDL/Subsystem/K',...
            'Multiplication','Matrix(K*u)');
    inp(c) = setBlockParameter(inp(c),'ProgettoTDL/Subsystem/K',...
            'Gain',mat2str([0 0 0 0]));
    inp(c) = setPostSimFcn(inp(c),...
            @(x) x.setUserString(num2str([data(:,c)' ottJ_ax(c,6)])));

end

outp = parsim(inp);

% Bench

inb = Simulink.SimulationInput.empty;

for c = 1 : 3

    inb(c) = Simulink.SimulationInput('ProgettoTDL');
    inb(c) = inb(c).setBlockParameter('ProgettoTDL/ctr','Value','0');
    inb(c) = inb(c).setBlockParameter('ProgettoTDL/Fcn',...
            'Numerator',mat2str([data(1,c) 0]));
    wr = 2*pi*data(2,c)/(3.6*200);
    inb(c) = inb(c).setBlockParameter('ProgettoTDL/Fcn',...
            'Denominator',mat2str([1 2*0.7*wr wr*wr]));
    inb(c) = setModelParameter(inb(c),'StopTime',num2str(data(3,c)));
    inb(c) = inb(c).setBlockParameter('ProgettoTDL/u_mr',...
            'Value','1');
```

```
    inb(c) = setBlockParameter(inb(c),'ProgettoTDL/Subsystem/K',...
            'Multiplication','Matrix(K*u)');
    inb(c) = setBlockParameter(inb(c),'ProgettoTDL/Subsystem/K',...
            'Gain',mat2str(bench));
    inb(c) = setPostSimFcn(inb(c),...
            @(x) x.setUserString(num2str([data(:,c)' bench])));

end

outb = parsim(inb);

% Semi-Attiva

ina = Simulink.SimulationInput.empty;

for c = 1 : 3

    ina(c) = Simulink.SimulationInput('ProgettoTDL');
    ina(c) = ina(c).setBlockParameter('ProgettoTDL/ctr','Value','0');
    ina(c) = ina(c).setBlockParameter('ProgettoTDL/Fcn',...
            'Numerator',mat2str([data(1,c) 0]));
    wr = 2*pi*data(2,c)/(3.6*200);
    ina(c) = ina(c).setBlockParameter('ProgettoTDL/Fcn',...
            'Denominator',mat2str([1 2*0.7*wr wr*wr]));
    ina(c) = setModelParameter(ina(c),'StopTime',num2str(data(3,c)));
    ina(c) = ina(c).setBlockParameter('ProgettoTDL/u_mr',...
            'Value','1');
    ina(c) = setBlockParameter(ina(c),'ProgettoTDL/Subsystem/K',...
            'Multiplication','Matrix(K*u)');
    ina(c) = setBlockParameter(ina(c),'ProgettoTDL/Subsystem/K',...
            'Gain',mat2str(matattivo(c,6:9)));
    ina(c) = setPostSimFcn(ina(c),...
            @(x) x.setUserString(num2str([data(:,c)' matattivo(c,6:9)])));

end

outa = parsim(ina);

% Semi-Attiva k1 = 0

inak1 = Simulink.SimulationInput.empty;

for c = 1 : 3

    inak1(c) = Simulink.SimulationInput('ProgettoTDL');
    inak1(c) = inak1(c).setBlockParameter('ProgettoTDL/ctr','Value','0');
    inak1(c) = inak1(c).setBlockParameter('ProgettoTDL/Fcn',...
            'Numerator',mat2str([data(1,c) 0]));
    wr = 2*pi*data(2,c)/(3.6*200);
    inak1(c) = inak1(c).setBlockParameter('ProgettoTDL/Fcn',...
            'Denominator',mat2str([1 2*0.7*wr wr*wr]));
    inak1(c) = setModelParameter(inak1(c),'StopTime',num2str(data(3,c)));
    inak1(c) = inak1(c).setBlockParameter('ProgettoTDL/u_mr',...
            'Value','1');
    inak1(c) = setBlockParameter(inak1(c),'ProgettoTDL/Subsystem/K',...
            'Multiplication','Matrix(K*u)');
    inak1(c) = setBlockParameter(inak1(c),'ProgettoTDL/Subsystem/K',...
            'Gain',mat2str(matattivok1(c,6:9)));
    inak1(c) = setPostSimFcn(inak1(c),...
            @(x) x.setUserString(num2str([data(:,c)' matattivok1(c,6:9)])));

end

outak1 = parsim(inak1);

for i = 1 : 3
```

```
    len = 2^nextpow2(length(outp(i).zr));
    winlen = 2500;

    pr = [zeros(round((len-length(outp(i).zr))/2),1); outp(i).zr;...
        zeros(round((len-length(outp(i).zr))/2)-1,1)];
    pd = [zeros(round((len-length(outp(i).zs_dd))/2),1); outp(i).zs_dd;...
        zeros(round((len-length(outp(i).zs_dd))/2)-1,1)];
    br = [zeros(round((len-length(outb(i).zr))/2),1); outb(i).zr;...
        zeros(round((len-length(outb(i).zr))/2)-1,1)];
    bd = [zeros(round((len-length(outb(i).zs_dd))/2),1); outb(i).zs_dd;...
        zeros(round((len-length(outb(i).zs_dd))/2)-1,1)];
    ar = [zeros(round((len-length(outa(i).zr))/2),1); outa(i).zr;...
        zeros(round((len-length(outa(i).zr))/2)-1,1)];
    ad = [zeros(round((len-length(outa(i).zs_dd))/2),1); outa(i).zs_dd;...
        zeros(round((len-length(outa(i).zs_dd))/2)-1,1)];
    ak1r = [zeros(round((len-length(outak1(i).zr))/2),1); outak1(i).zr;...
        zeros(round((len-length(outak1(i).zr))/2)-1,1)];
    ak1d = [zeros(round((len-length(outak1(i).zs_dd))/2),1); outak1(i).zs_dd;...
        zeros(round((len-length(outak1(i).zs_dd))/2)-1,1)];

    [Txy,f] = tfestimate([outp(i).zr outb(i).zr outa(i).zr outak1(i).zr],...
        [outp(i).zs_dd outb(i).zs_dd outa(i).zs_dd outak1(i).zs_dd],[],[],...
            (0:0.001:20),1000);

    [Txy1,f1] = tfestimate([pr br ar ak1r],[pd bd ad ak1d],winlen,[],...
        (0:0.001:20),1000);


    close(figure(i+20))
    figure(i+20)
    hold on
    pp = plot(f1,mag2db(abs(Txy1)),'LineWidth',1.5);
    pp1 = plot(f,mag2db(abs(Txy)));
    pp1(1).Color = [0 0.4470 0.7410];
    pp(1).Color = [0 0.4470 0.7410];
    pp1(2).Color = [0.8500 0.3250 0.0980];
    pp(2).Color = [0.8500 0.3250 0.0980];
    pp1(3).Color = [0.9290 0.6940 0.1250];
    pp(3).Color = [0.9290 0.6940 0.1250];
    pp1(4).Color = [0.4660 0.6740 0.1880];
    pp(4).Color = [0.4660 0.6740 0.1880];
    pp1(1).Color(4) = 0.25;
    pp1(2).Color(4) = 0.25;
    pp1(3).Color(4) = 0.25;
    pp1(4).Color(4) = 0.25;
    hold off
    xlim([0 20])
    ylim([-10 60])
    title(['FdT sull''Accelerazione a velocita'' v = ',...
        num2str(data(2,i)),' $\frac{km}{h}$']);
    title(legend,'Analisi')
    legend('Passiva','parametri Bench','Semi-Attiva',...
        'Semi-Attiva con k1 = 0','Location','southeast');
    xlabel('[Hz]');
    ylabel('[dB]');

end

%% Generazione FdT u_MR = 0

in = Simulink.SimulationInput.empty;
c = 1;
i = 2;
in(c) = Simulink.SimulationInput('ProgettoTDL');
in(c) = in(c).setBlockParameter('ProgettoTDL/ctr','Value','1');
```

```
in(c) = in(c).setBlockParameter('ProgettoTDL/Fcn',...
'Numerator',mat2str([data(1,i) 0]));
wr = 2*pi*data(2,i)/(3.6*200);
in(c) = in(c).setBlockParameter('ProgettoTDL/Fcn',...
'Denominator',mat2str([1 2*0.7*wr wr*wr]));
in(c) = setModelParameter(in(c),'StopTime',num2str(data(3,i)));
in(c) = in(c).setBlockParameter('ProgettoTDL/u_mr',...
'Value',num2str(0));
in(c) = setBlockParameter(in(c),'ProgettoTDL/Subsystem/K',...
'Multiplication','Matrix(K*u)');
in(c) = setBlockParameter(in(c),'ProgettoTDL/Subsystem/K',...
'Gain',mat2str([0 0 0 0]));
in(c) = setPostSimFcn(in(c),...
@(x) x.setUserString(num2str([data(:,i)' 0])));
out = parsim(in);
[Txy,f] = tfestimate(out.zr,out.zs_dd,[],[],...
(0:0.001:20),1000);

A =[0 0 1 0; 0 0 0 1; 0 -60 0 -(16/9); (300000/70) -(33120/7) 0 -(832/63)];
B =[0; 0; 0; -(300000/7)];
C =[0 -60 0 -(16/9)];
D = 0;

[b,a] = ss2tf(A,B,C,D);

sys=tf(b,a);

[mag,phase,wout] = bode(sys);

close(figure(28))
figure(28)
hold on
semilogx(f,20*log10(abs(Txy)),'LineWidth',1.1);
semilogx(wout(:,1)/(2*pi), 20*log10(squeeze((mag)))-20,'LineWidth',1.1);
grid off;
title('Funzione di Trasferimento per $u_{MR}$ = 0 [N]'); xlabel('[Hz]');
ylabel('[dB]');
xlim([0 20])
ylim([20 60])
legend('Stimata','Analitica')
hold off

%% Immagini

for i = 1 : 28

    myfig(0,figure(i),'FontLegend',12,'FontTick',11,'FontLabel',18,...
        'Interpreter','latex','LegendBox','off')

end
```

[1]

A.2 Funzione Aggiuntiva

```
function [matJ,q] = parsenfilev2(data,vector)

% Funzione per l'Analisi del Modello Simulink "ProgettoTDL.slx"

```

[1]Impaginazione basata su Florian Knorn (2021). M-code LaTeX Package (https://www.mathworks.com/matlabcentral/fileexchange/8015-m-code-latex-package), MATLAB Central File Exchange. Retrieved August 18, 2021.

```
tic

if length(vector(1,:)) == 1

    in = Simulink.SimulationInput.empty;
    matJ = zeros(length(vector(:,1)),6);
    c = 1;

    for i = 1 : 3
    for o = 1 : length(vector(:,1))

        in(c) = Simulink.SimulationInput('ProgettoTDL');
        in(c) = in(c).setBlockParameter('ProgettoTDL/ctr','Value','1');
        in(c) = in(c).setBlockParameter('ProgettoTDL/Fcn',...
                'Numerator',mat2str([data(1,i) 0]));
        wr = 2*pi*data(2,i)/(3.6*200);
        in(c) = in(c).setBlockParameter('ProgettoTDL/Fcn',...
                'Denominator',mat2str([1 2*0.7*wr wr*wr]));
        in(c) = setModelParameter(in(c),'StopTime',num2str(data(3,i)));
        in(c) = in(c).setBlockParameter('ProgettoTDL/u_mr',...
                'Value',num2str(vector(o)));
        in(c) = setBlockParameter(in(c),'ProgettoTDL/Subsystem/K',...
                'Multiplication','Matrix(K*u)');
        in(c) = setBlockParameter(in(c),'ProgettoTDL/Subsystem/K',...
                'Gain',mat2str([0 0 0 0]));
        in(c) = setPostSimFcn(in(c),...
                @(x) x.setUserString(num2str([data(:,i)' vector(o)])));

        c = c+1;

    end
    end

    q = zeros(c-1,8);

elseif length(vector(1,:)) == 4

    in = Simulink.SimulationInput.empty;
    c = 1;
    q = 0;

    for o = 1 : length(vector(:,1))

        in(c) = Simulink.SimulationInput('ProgettoTDL');
        in(c) = in(c).setBlockParameter('ProgettoTDL/ctr','Value','0');
        in(c) = in(c).setBlockParameter('ProgettoTDL/Fcn',...
                'Numerator',mat2str([data(1) 0]));
        wr = 2*pi*data(2)/(3.6*200);
        in(c) = in(c).setBlockParameter('ProgettoTDL/Fcn',...
                'Denominator',mat2str([1 2*0.7*wr wr*wr]));
        in(c) = setModelParameter(in(c),'StopTime',num2str(data(3)));
        in(c) = in(c).setBlockParameter('ProgettoTDL/u_mr',...
                'Value','1');
        in(c) = setBlockParameter(in(c),'ProgettoTDL/Subsystem/K',...
                'Multiplication','Matrix(K*u)');
        in(c) = setBlockParameter(in(c),'ProgettoTDL/Subsystem/K',...
                'Gain',mat2str(vector(o,:)));
        in(c) = setPostSimFcn(in(c),...
                @(x) x.setUserString(num2str([data' vector(o,:)])));
        c = c+1;

    end

    matJ = zeros(length(in),9);

else
```

```

    error('Error on allcompute');

end

if isempty(in)

    return

end

c = 1;
length(in)
len = length(in)-(rem(length(in),1500));
num = (length(in)-(rem(length(in),1500)))/1500;
inn = reshape(in(1:len),num,[]);
innf = in(len+1:length(in));
clear in;
mat = zeros(1,2);
j = 1;


for m = 1 : (num)

    out = parsim(inn(m,:),'ShowProgress','on','StopOnError','on');

    for o = 1 : 1500

        ustr = str2num(out(o).SimulationMetadata.UserString); %#ok<*ST2NM>
        mat(1,1) = rms(out(o).zs_dd);
        mat(1,2) = rms(out(o).zs_d);
        matJ(j,:) = [mat ustr];
        j = j + 1;

        %Calcolo quantili

        if length(vector(1,:)) == 1
            q(c,1:4) = quantile(abs([out(o).zs out(o).ztilda out(o).zs_d...
                      out(o).ztilda_d]),0.95);
            q(c,5:8) = ustr;
            c = c+1;
        end


    end
        clear out;
end

out = parsim(innf,'ShowProgress','on','StopOnError','on');

for o = 1 : length(out)

    ustr = str2num(out(o).SimulationMetadata.UserString); %#ok<*ST2NM>
    mat(1,1) = rms(out(o).zs_dd);
    mat(1,2) = rms(out(o).zs_d);
    matJ(j,:) = [mat ustr];
    j = j + 1;

    %Calcolo quantili

    if length(vector(1,:)) == 1
        q(c,1:4) = quantile(abs([out(o).zs out(o).ztilda out(o).zs_d...
                  out(o).ztilda_d]),0.95);
        q(c,5:8) = ustr;
        c = c+1;
    end
```

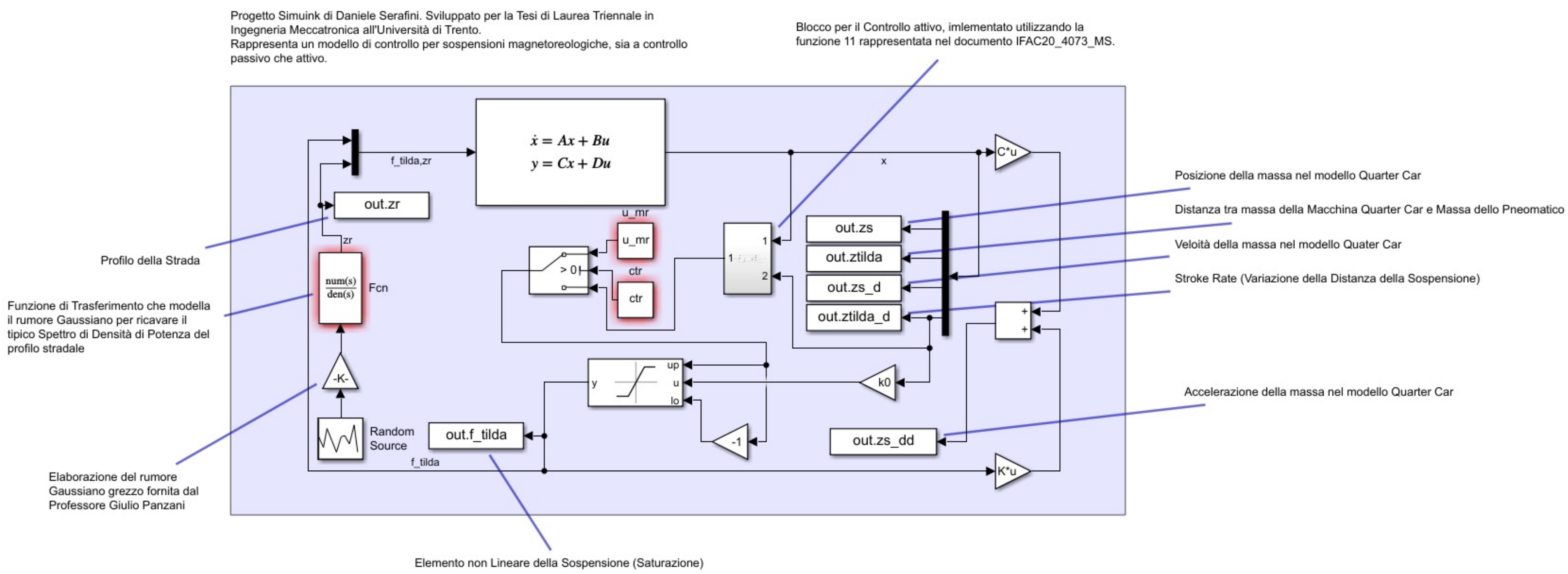
```
end
clear out;
clc
toc
end
```

[2]

[2] Impaginazione basata su Florian Knorn (2021). M-code LaTeX Package (https://www.mathworks.com/matlabcentral/fileexchange/8015-m-code-latex-package), MATLAB Central File Exchange. Retrieved August 18, 2021.

Allegato B Schema Simulink®

Nella facciata successiva viene inserito lo schema Simulink® utilizzato nell'analisi del problema.



Progetto Simuink di Daniele Serafini. Sviluppato per la Tesi di Laurea Triennale in Ingegneria Meccatronica all'Università di Trento.
Rappresenta un modello di controllo per sospensioni magnetoreologiche, sia a controllo passivo che attivo.
Blocco per il Controllo attivo, imlementato utilizzando la funzione 11 rappresentata nel documento IFAC20_4073_MS.
f_tilda,zr
$\dot{x} = Ax + Bu$
$y = Cx + Du$
x
C*u
out.zr
zr
u_mr
ctr
> 0
1
2
out.zs
out.ztilda
out.zs_d
out.ztilda_d
Posizione della massa nel modello Quarter Car
Distanza tra massa della Macchina Quarter Car e Massa dello Pneomatico
Veloità della massa nel modello Quater Car
Stroke Rate (Variazione della Distanza della Sospensione)
Profilo della Strada
Funzione di Trasferimento che modella il rumore Gaussiano per ricavare il tipico Spettro di Densità di Potenza del profilo stradale
num(s)
den(s)
Fcn
-K-
Random Source
up
u
lo
y
k0
-1
+
+
Accelerazione della massa nel modello Quarter Car
out.f_tilda
out.zs_dd
f_tilda
K*u
Elaborazione del rumore Gaussiano grezzo fornita dal Professore Giulio Panzani
Elemento non Lineare della Sospensione (Saturazione)