

POLITECNICO DI TORINO

Ingegneria

Corso di Laurea in Ingegneria Elettrica

Tesi di Laurea Magistrale

Algoritmi di controllo innovativi per i drives impiegati
nella trazione elettrica leggera



Relatore:

Prof. Radu Iustin Bojoi

Correlatore:

Ing. Sandro Rubino

Laureando:

Riccardo CIFALINO'

ANNO ACCADEMICO 2018-2019

Ringraziamenti

Prima di chiudere definitivamente questa tesi, una fine di un libro che racconta tante storie e tanti ricordi di questi anni, mi riservo gli ultimi istanti per scrivere sicuramente la pagina piú sentita e piú attesa: i *"Ringraziamenti"*...

Inizio nel ringraziare il mio relatore e il mio correlatore che con pazienza e dedizione hanno saputo darmi una grossa opportunità di crescita per il mio percorso di vita in generale.

E dopo questi primi ringraziamenti di rito, ma doverosi, inizio i ringraziamenti che riguardano il mio cuore e i miei pensieri di ogni giorno. Probabilmente, anche restringendo il cerchio affettivo, non riuscirei a decidere a chi dare l'onore e il merito dell'inizio, l'onore e il merito del primo nome. Una scelta troppo complicata e scelte come *"famiglia"* risulterebbero troppo generalizzate perchè io voglio scrivere nome e cognome di ognuno di voi. Voi... Penso che inizieró cosí:

"Ringrazio Voi. Ognuno di Voi che leggerá queste righe avrá avuto un ruolo fondamentale in questo tortuoso cammino. Ognuno di Voi, mentre leggerá, pronuncerà dentro di sé il suo nome e cognome, perché ognuno ha avuto un ruolo. Ruoli diversi, piú o meno presenti, piú o meno importanti, ma comunque ruoli fondamentali. Grazie a Te ora sono qui. Grazie.

Allento l'ansia di non pronunciare «primi nomi» e ne svelo solo due, ma sol perché non sono persone, ma oggetti a me particolarmente cari. Quindi concludo ringraziando la mia Sveglia rossa sul comodino,

la quale in questi lunghi e intensi 5 anni ha lavorato anche quando sul calendario i giorni erano dipinti di rosso. Ringrazio la mia Valigia, eterna compagna di viaggi, fedele sempre nel riaccompagnarmi in quel posto che chiamo Casa. Una Casa sempre troppo lontana, ma che grazie a lei forse ho sentito piú vicina.

Non manca nessuno. Ho finito. Grazie."

Simboli

Per rendere la seguente lista della simbologia piú snella si spiega che per le grandezze elettriche (correnti e tensioni) e magnetiche (flussi magnetici) si adotta il seguente criterio: lettere in maiuscolo per la notazione matriciale e lettere in minuscolo per i rispettivi valori istantanei della singola fase. Di conseguenza in questa lista non sono riportati i simboli di queste grandezze. Inoltre anche le grandezze già spiegate ed introdotte nel corso di questa tesi non sono riportate in questa lista.

L_{123} Matrice delle induttanze di statore

M_{ij} Singolo termine delle matrici delle induttanze di statore

$F_d F_q$ Proiezioni in assi dq della forza magnomotrice generata da un avvolgimento di statore (nell'ipotesi di avvolgimento puramente sinusoidale)

$\Phi_d \Phi_q$ Flussi magnetici delle forze megnetomotrici precedentemente introdotte

N_s Spire equivalenti di un avvolgimento di statore

λ_{ij} Flusso magnetico concatenato: prodotto dall'avvolgimento i -esimo e concatenato dall'avvolgimento j -esimo

R_s Resistenza di fase di statore

$L_{\sigma s}$ Induttanza di dispersione di fase di statore

- λ_M Flusso magnetico dei magneti a rotore
- λ Flusso di statore
- p Paia poli del motore elettrico
- ξ Fattore di anisotropia per i motori sincroni
- ω_R Velocità angolare meccanica del rotore
- ω Velocità angolare elettrica
- T_s Periodo di commutazione che in questo lavoro di tesi coincide anche con il periodo di campionamento (e quindi di esecuzione del codice di controllo)
- f_{sw} Frequenza di commutazione del convertitore
- V_0 Tensione di soglia degli switches o dei diodi (da specificare) del convertitore
- R_{on} Resistenza equivalente in conduzione degli switches o dei diodi (da specificare) del convertitore
- $d_s q_s$ Pedici per le grandezze elettriche per i rispetti assi $f\tau$ nel controllo DFVC
- T_e Coppia elettromagnetica del motore elettrico
- T_L Coppia di carico costante
- J_{eq} Momento d'inerzia equivalente considerando l'inerzia del motore elettrico e quella del carico meccanico a valle

Obiettivo

Questa tesi di laurea si pone il principale obiettivo di implementare in linguaggio C un innovativo controllo di macchina per un azionamento elettrico per trazione elettrica leggera. Il controllo di macchina in questione fa riferimento a quello che in letteratura viene conosciuto come *Direct Flux Vector Control (DFVC)*, ideato e sviluppato all'interno del Politecnico di Torino.

Il lavoro di tesi é diviso in due fasi principali: inizialmente si testa il controllo in due ambienti di simulazione differenti, prima *Simulink* poi *Plecs*, dove viene modellizzato l'intero azionamento elettrico; successivamente le verifiche sperimentali su un motore alternativo, ma della stessa natura di quello simulato. La motivazione di questa scelta é legata prettamente a motivi di ritardo organizzativi, in quanto l'inverter e l'azionamento in generale si trova ancora in una versione prototipale.

Indice

Ringraziamenti	II
Lista della simbologia	IV
Obiettivo	VI
1 Introduzione	1
2 Modello del Motore Sincrono IPM	7
2.1 Equazioni elettriche e magnetiche	9
2.2 Modello magnetico	16
2.3 Equazione della coppia	22
2.4 MTPA	22
3 Controllo di macchina: <i>Direct Flux Vector Control</i>	25
3.1 Equazioni del controllo DFVC	27
3.2 Calcolo della velocità	33
3.2.1 Phase-Locked Loop	33
3.3 Osservatore di flusso	34
3.4 Controllo di macchina	37
3.4.1 Anello in asse f	37
3.4.2 Anello in asse τ	40
3.5 Deflussaggio	44
3.6 Limitazione dell'angolo di carico in MTPV	46
3.7 Phase Advancing	50

4	Azionamento elettrico della Squadra Corse	53
4.1	MTPA in linearit� magnetica	58
4.2	MTPV in linearit� magnetica	61
5	Simulazione in Simulink	63
5.1	Modello meccanico del motore elettrico	65
5.2	Modello elettromagnetico del motore elettrico	65
5.3	Modello dell'encoder	68
5.4	Modello del generatore dei segnali PWM	69
5.5	Modello dell'inverter	70
5.5.1	Interfacciamento dell'inverter con il carico	71
5.6	Modello del DC-link e delle misure elettriche	73
5.7	Algoritmo di controllo	73
5.7.1	Gli stati di macchina	79
5.8	Risultati simulazione Simulink	94
5.8.1	Test in Simulink	96
6	Simulazione in Plecs	111
6.1	Modello meccanico del motore elettrico	113
6.2	Modello elettromagnetico del motore elettrico	114
6.3	Modello dell'encoder	117
6.4	Modello del generatore dei segnali PWM	119
6.5	Modello dell'inverter	119
6.5.1	Circuito elettrico dell'inverter	120
6.5.2	Modulatore PWM	121
6.6	Modello del DC-link e delle misure elettriche	124
6.7	Algoritmo di controllo	124
6.7.1	Ricostruzione delle tensioni di fase	124
6.8	Risultati simulazione Plecs	129
6.8.1	Test in Plecs	131
6.8.2	Test in condizioni di <i>UGO</i>	143
7	Conclusioni	147
	Bibliografia	149

Capitolo 1

Introduzione

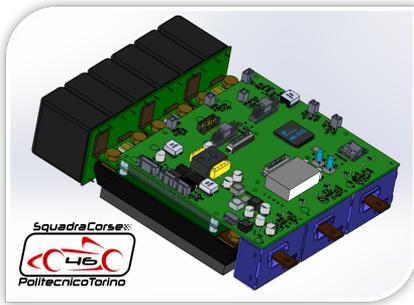
Questo progetto di tesi é parte integrante di un lavoro in team molto piú ampio e complesso, il quale nasce all'interno del Dipartimento Energia del Politecnico di Torino da una proposta della *Squadra Corse* (una realtà studentesca ormai consolidata nell'università dal 2005) per migliorare la loro monoposto da competizione a trazione elettrica integrale.

Presentando brevemente la vettura, si descrive un'architettura interna composta da un pacco batterie da cui si ricava il primo livello di tensione (600 V circa) per la trazione (un secondo livello da 24 V per gli ausiliari di bordo), quindi poi la struttura si dirama nei quattro azionamenti elettrici per ogni motoruota, composti rispettivamente da un inverter e da un motore elettrico. L'ultima versione presentata nel 2018, *SC18 integrale* (Fig. 1.1), pesa complessivamente 199 kg , una potenza di picco (massima ammissibile per le competizioni studentesche) di 80 kW , essa é in grado di raggiungere i 130 km/h e toccare i 100 km/h , da ferma, in $2,9\text{ s}$.



Figura 1.1: Monoposto a trazione elettrica integrale della Squadra Corse.

Il progetto nella sua totalità pone l'obiettivo ai ricercatori del Dipartimento di riprogettare il controllo dell'azionamento elettrico e il progetto ex novo dei quattro inverter principali per la trazione (Fig. 1.2).



(a)



(b)

Figura 1.2: Inverter di trazione progettato per la Squadra Corse dai ricercatori del Dipartimento di Energia del Politecnico: in (a) Vista CAD dell'inverter e in (b) il prototipo in fase di collaudo.

In questo lavoro questa tesi si pone il compito di simulare e testare il controllo di macchina *Direct Flux Vector Control* (DFVC) su uno dei quattro motori gemelli della monoposto. La simulazione prevede una modellistica dell'intero azionamento, in ambienti software

dedicati, e quindi successivamente il testing del controllo scritto in linguaggio C. La scelta del controllo (DFVC) é motivata dalla sua particolare efficacia in applicazioni dove sono richieste ampie zone di deflussaggio del motore elettrico, ovvero zone di lavoro a potenza costante; l'ambiente automotive naturalmente si sposa bene con questa esigenza. Il motore elettrico é un servomotore sincrono trifase, tipologia IPM (Internal Permanent Magnet), prodotto da *AMK* modello *DD5-14-10-POW-18600-B5*, una soluzione spesso adottata in campo automotive perché anch'esso si presta bene per applicazioni che richiedono una potenza costante. Segue una breve tabella che ne menziona le principali caratteristiche:

P_n	12,3	kW
V_n	350	V
I_n	41	Arms
T_n	9,8	Nm
T_{max}	21	Nm
n_{rated}	12000	rpm
n_{max}	18617	rpm
<i>Diametro</i>	96	mm
<i>Lunghezza</i>	121	mm

Tabella 1.1: Tabella introduttiva per il motore elettrico.

Nelle righe precedenti si é già accennato al fatto che la monoposto abbia quattro motoruote e per discorsi legati al regolamento per competizioni studentesche, ogni motoruota non può erogare una potenza di picco superiore a *20 kW*. Da qui si inizia a comprendere del perché (notando anche le dimensioni del motore in Fig. 1.3) si parli di controlli per azionamenti di trazione elettrica *leggera*.



Figura 1.3: Motoruota. In rosso il motore elettrico.

Una panoramica dell'intero lavoro può essere presentata in tre fasi così ordinate:

- Prima fase: simulazione in ambiente Simulink dell'intero azionamento per testare il codice del controllo implementato ad elevate prestazioni dinamiche.
- Seconda fase: simulazione in ambiente Plecs. La simulazione riproduce la simulazione Simulink e i vari test, modellando più realisticamente alcuni componenti dell'azionamento elettrico.
- Terza fase: verifica sperimentale in laboratorio su un motore alternativo per ritardi, dovuti a complicazioni tecniche, all'interno della squadra di lavoro.

Entrando nel dettaglio di questa tesi, l'elaborato è strutturato nei seguenti capitoli:

- Capitolo 2: viene ricavato ed illustrato il modello di macchina con le sue equazioni elettromagnetiche, l'importanza di un modello magnetico e l'equazione di coppia annessa al concetto di MTPA.
- Capitolo 3: viene illustrato il controllo DFVC. Si ricavano le equazioni nei suoi due assi in funzione delle due variabili di stato, le correnti. Successivamente viene spiegato ogni suo blocco per l'implementazione del suo algoritmo.

-
- Capitolo 4: si riportano i dati tecnici e mappe caratteristiche dell'azionamento elettrico, dati utili al fine della successiva modellistica.
 - Capitolo 5: si presenta la simulazione in ambiente Simulink dove si testa il controllo scritto in C e si modella l'intero azionamento elettrico.
 - Capitolo 6: si presenta la simulazione in ambiente Plecs. Essa riproduce il lavoro fatto in Simulink con la differenza che modella piú realisticamente alcune parti dell'azionamento, il convertitore in particolar modo.
 - Capitolo 7: conclusioni sul progetto.

Capitolo 2

Modello del Motore Sincrono IPM

Questo capitolo ha lo scopo di ricavare le equazioni per il modello di macchina, indispensabile per la modellistica del motore elettrico nei due diversi ambienti di simulazione precedentemente accennati.

I motori sincroni ormai rivestono gran parte delle applicazioni in campo industriali e ricoprono vari settori dell'industria, dalla generazione all'utilizzo di energia elettrica. Proprio per questa ragione si assiste ad una vasta diversificazione di modelli che vedono strutture di macchina molto diverse tra loro, ma che mantengono gli stessi principi fisici di funzionamento. In campo automotive molto spesso si ricorre a soluzioni che prevedono la presenza di magneti al fine di erogare una coppia quanto piú performante possibile. Si possono identificare le tre macro categorie di motori sincroni che prevedono l'impiego di magneti al loro interno (Fig. 2.1): SPM (*Surface Permanent Magnet*), IPM (*Internal Permanent Magnet*) e PMASR (*Permanent Magnet Assisted Synchronous Reluctance*), il secondo oggetto di questa tesi.

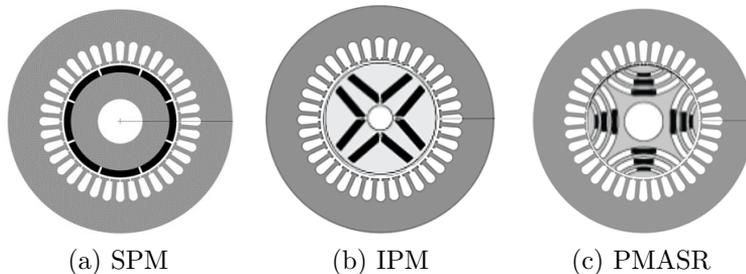


Figura 2.1: Principali soluzioni di motori sincroni adottate in campo automotive.

	Densità di coppia	Deflussaggio	Costo
SPM	++	--	++
IPM	+ -	+ -	+ -
PMASR	--	++	+ -

Tabella 2.1: Tabella comparativa per i tre tipi di motori sincroni.

Ai fini del controllo, la principale differenza tra i tre motori é la natura isotropa (SPM) o anisotropa (IPM e PMASR) di macchina. Nel secondo caso infatti non basta controllare una corrente in quadratura con il flusso dei magneti per erogare coppia, ma per ottimizzare il controllo deve essere sfruttato anche il contributo anisotropo di coppia.

Nel ricavare il modello di macchina, anche se il motore della Squadra Corse é tipo IPM, la trattazione é per una generale macchina anisotropa, inglobando quindi anche il caso del sincro PMASR.

2.1 Equazioni elettriche e magnetiche

La natura trifase del motore elettrico é stata ampiamente sottointesa e non avendo avvolgimenti a rotore le equazioni di macchina si presentano con 3 equazioni elettriche e 3 equazioni magnetiche, una coppia delle rispettive equazioni per ogni avvolgimento (immaginando gli avvolgimenti di statore come 3 avvolgimenti trifasi concentrati sinusoidali).

Prima di introdurre le equazioni elettromagnetiche si elencano le ipotesi sotto le quali sono ricavate:

- Si fa riferimento al motore sincrono di natura anisotropa con la presenza di magneti permanenti a rotore.
- Per rendere la trattazione piú snella si considera il paio poli di macchina unitario.
- Avvolgimento di statore trifase a distribuzione perfettamente sinusoidale (le armoniche sono un fenomeno secondario).
- Analogamente al punto precedente il campo di induzione dei magneti é di sola prima armonica.
- Si suppone una linearitá magnetica.

Prima di ricavare le equazioni il primo passaggio è calcolare i vari termini della matrice di induttanze di statore:

$$L_{123} = \begin{bmatrix} M_{11} & M_{12} & M_{13} \\ M_{21} & M_{22} & M_{23} \\ M_{31} & M_{32} & M_{33} \end{bmatrix}$$

Nel seguito é spiegato come si ricava il primo termine della matrice M_{11} , utile per comprendere gli altri ripetendo lo stesso ragionamento. La singola induttanza in studio é l'autoinduttanza dell'avvolgimento della prima fase. Quindi imponendo una corrente sinusoidale nell'avvolgimento e sapendo il numero di spire si giunge ad ottenere la

forza magnetomotrice. Nell'ipotesi di linearità magnetica, il flusso e la forza magnetomotrice sono direttamente proporzionali tramite il parametro della riluttanza magnetica. Ottenendo il flusso concatenato e conoscendo la corrente imposta si ricava una forma compatta dell'induttanza.

Si specifica che si adotta la convenzione di fissare il flusso dei magneti in asse d (nel sistema di riferimento a rotore).

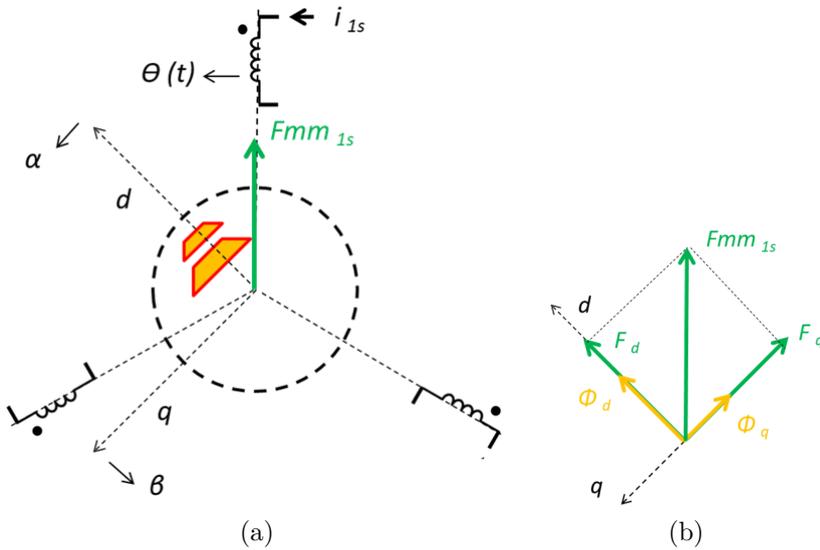


Figura 2.2: Procedimento per ricavare M_{11} imponendo solamente una corrente sinusoidale alla prima fase. Lo statore è modellizzato con i 3 avvolgimenti concentrati, mentre il rotore è solidale al sistema di riferimento dq . In verde le forze magnetomotrici e in arancione i flussi magnetici, entrambi in riferimento agli assi dq .

Appare evidente come la natura anisotropa del motore pone un problema nel definire univocamente un concetto di riluttanza, in quanto non avrebbe senso definire $\Re[\theta(t)]$, perché istantaneamente punti diversi di statore vedono una riluttanza diversa. E' immediato

pensare di spostare l'analisi nel sistema di riferimento rotorico, dove lungo i due rispettivi assi si può definire un concetto univoco di riluttanza.

Analizzando l'asse d si ottengono i seguenti andamenti in riferimento alla Fig. 2.2. Si nota subito una distorsione armonica sul flusso magnetico, quindi ai fini della conversione elettromeccanica si prenderá in esame la sola componente di prima armonica (Fig. 2.3):

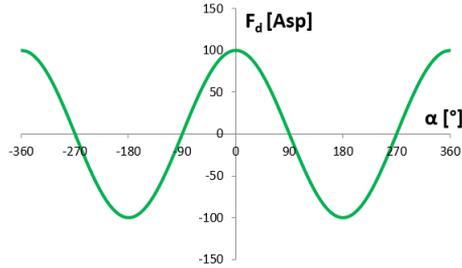
$$\Phi_{d1} = \frac{F_d}{\mathfrak{R}_{eq.d1}} \quad \text{con } \mathfrak{R}_{eq.d1} \equiv \mathfrak{R}_d$$

E' importante sottolineare come $\mathfrak{R}_d \neq \mathfrak{R}_{min}$, ma questo termine corrisponde ad un concetto medio su tutto il traferro, frutto della scomposizione in serie di Fourier.

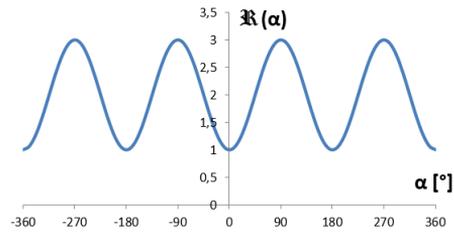
Ripetendo gli stessi ragionamenti in asse q (Fig. 2.4):

$$\Phi_{q1} = \frac{F_q}{\mathfrak{R}_{eq.q1}} \quad \text{con } \mathfrak{R}_{eq.q1} \equiv \mathfrak{R}_q$$

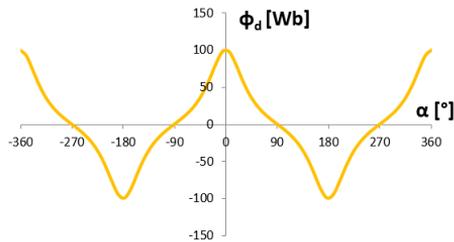
I valori sui grafici sono puramente indicativi e non influenti per l'analisi del problema.



(a)

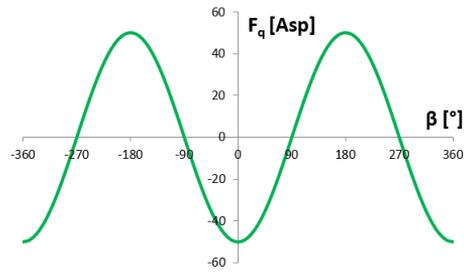


(b)

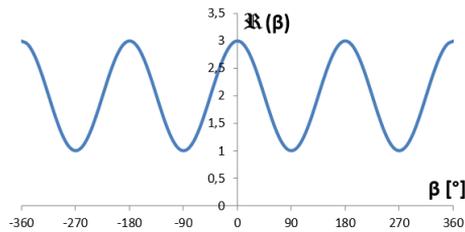


(c)

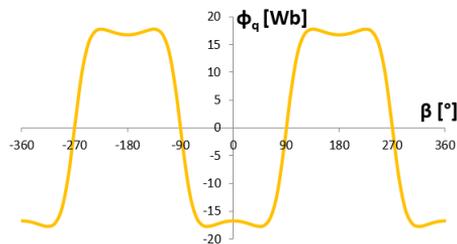
Figura 2.3: Dall'alto verso il basso: proiezione sull'asse d della forza magnetomotrice (scomposizione possibile per l'ipotesi di una distribuzione sinusoidale al traferro), riluttanza in asse d , flusso magnetico in asse d ricavato come rapporto tra la forza magnetomotrice (primo grafico) e la riluttanza (secondo grafico).



(a)



(b)



(c)

Figura 2.4: Dall'alto verso il basso: proiezione sull'asse q della forza magnetomotrice (scomposizione possibile per l'ipotesi di una distribuzione sinusoidale al traferro), riluttanza in asse q , flusso magnetico in asse q ricavato come rapporto tra la forza magnetomotrice (primo grafico) e la riluttanza (secondo grafico).

Dopo questo doveroso passaggio é possibile definire il flusso concatenato, sempre in riferimento alla prova illustrata in Fig. 2.2:

I flussi magnetici fanno riferimento alla loro componente fondamentale.

$$\begin{aligned}
 \lambda_{11} &= N_s (\Phi_d \cos\Theta - \Phi_q \sin\Theta) = \\
 &= N_s \left(\frac{F_d}{\mathfrak{R}_d} \cos\Theta - \frac{F_q}{\mathfrak{R}_q} \sin\Theta \right) = \\
 &= N_s \left(\frac{N_s i_{1s} \cos\Theta}{\mathfrak{R}_d} \cos\Theta - \frac{-N_s i_{1s} \sin\Theta}{\mathfrak{R}_q} \sin\Theta \right) = \quad (2.1) \\
 &= \frac{N_s^2}{\mathfrak{R}_d} \cos^2\Theta + \frac{N_s^2}{\mathfrak{R}_q} \sin^2\Theta = M_{11} i_{1s}
 \end{aligned}$$

Definendo $M_d \equiv \frac{N_s^2}{\mathfrak{R}_d}$ come mutua induttanza principale dell'avvolgimento di statore quando l'asse d di rotore è allineato all'avvolgimento concentrato della prima fase, ed analogamente $M_q \equiv \frac{N_s^2}{\mathfrak{R}_q}$, si ricava una forma piú compatta utile ad apprezzare la duplice natura isotropa ed anisotropa del motore:

$$\begin{aligned}
 M_{11} &= M_d \cos^2\Theta + M_q \sin^2\Theta = \\
 &= \frac{M_d + M_q}{2} - \frac{M_q - M_d}{2} \cos 2\Theta = \quad (2.2) \\
 &= M_I - M_A \cos 2\Theta
 \end{aligned}$$

Ripercorrendo gli stessi passaggi per gli otto termini restanti della matrice delle induttanze si giunge finalmente al primo obiettivo di questo capitolo:

$$\begin{aligned}
 L_{123} = & M_I \begin{bmatrix} 1 & -1/2 & -1/2 \\ -1/2 & 1 & -1/2 \\ -1/2 & -1/2 & 1 \end{bmatrix} + \dots \\
 & \dots - M_A \begin{bmatrix} \cos 2\Theta & \cos(2\Theta - 2\pi/3) & \cos(2\Theta + 2\pi/3) \\ \cos(2\Theta - 2\pi/3) & \cos(2\Theta + 2\pi/3) & \cos 2\Theta \\ \cos(2\Theta + 2\pi/3) & \cos 2\Theta & \cos(2\Theta - 2\pi/3) \end{bmatrix}
 \end{aligned} \tag{2.3}$$

Seguono in forma matriciale, in ordine, le equazioni elettriche e magnetiche del motore sincrono IPM:

$$\begin{cases} \bar{V}_{123} = R_s \bar{I}_{123} + \frac{d}{dt} (\bar{\Lambda}_{123}) \\ \bar{\Lambda}_{123} = L_{\sigma s} \bar{I}_{123} + L_{123} \bar{I}_{123} + \bar{\Lambda}_M \end{cases} \tag{2.4}$$

A questo punto si tralasciano le varie operazioni matriciali e si ricordano semplicemente le matrici necessarie per ricavare il modello bifase della macchina in assi rotorici dq :

$$\text{Trasformata di Clarke } T = \begin{bmatrix} 2/3 & -1/3 & -1/3 \\ 0 & 1/\sqrt{3} & -1/\sqrt{3} \\ 1 & 1 & 1 \end{bmatrix}$$

$$\text{Matrice di rotazione (antioraria) } R = \begin{bmatrix} \cos\Theta & \sin\Theta \\ -\sin\Theta & \cos\Theta \end{bmatrix}$$

Ottenendo le canoniche equazioni matriciali in assi rotorici:

$$\begin{cases} \bar{V}_{dq} = R_s \bar{I}_{dq} + \frac{d}{dt} (\bar{\Lambda}_{dq}) + p\omega_R \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \bar{\Lambda}_{dq} \\ \bar{\Lambda}_{dq} = L_{\sigma s} \bar{I}_{dq} + \frac{3}{2} M_I \bar{I}_{dq} - \frac{3}{2} M_A \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \bar{I}_{dq} + \begin{bmatrix} \lambda_M \\ 0 \end{bmatrix} \end{cases} \tag{2.5}$$

Ultimo passaggio rimane da definire l'induttanza sincrona in asse d e in asse q :

$$\begin{aligned} L_d &= L_{\sigma s} + \frac{3}{2} (M_I - M_A) \\ L_q &= L_{\sigma s} + \frac{3}{2} (M_I + M_A) \\ L_d &< L_q \end{aligned}$$

Ora finalmente seguono le 2 equazioni elettriche e le 2 equazioni magnetiche in assi rotorici, le quali saranno utilizzate successivamente per modellare il motore nelle simulazioni implementate per questo lavoro di tesi.

$$\begin{cases} v_d = R_s i_d + L_d \frac{d i_d}{dt} - p \omega_R \lambda_q \\ v_q = R_s i_q + L_q \frac{d i_q}{dt} + p \omega_R \lambda_d \\ \lambda_d = L_d i_d + \lambda_M \\ \lambda_q = L_q i_q \end{cases} \quad (2.6)$$

2.2 Modello magnetico

Le equazioni precedentemente ricavate rimangono valide al di là se si tiene conto della saturazione nel ferro o del flusso dei magneti in funzione della temperatura. Infatti si otterrebbero semplicemente delle induttanze e un flusso dei magneti in funzione rispettivamente delle correnti e della temperatura.

Il modello magnetico quindi ha un'importanza fondamentale nel controllo e nella modellistica di un azionamento elettrico, in quanto permette una mappatura magnetica del motore potendo apprezzare la saturazione delle induttanze sui due assi bifasi e i fenomeni di non-linearità (*cross-coupling*) che fanno cadere l'ipotesi di disaccoppiamento magnetico tra i due assi.

Di seguito verranno proposte come esempio le mappe di flusso di un PMASR generico ($5,3 \text{ kW}$, 2 paia-poli) ricavate secondo la procedura dettagliatamente descritta in [1] e [2]. La natura delle seguenti mappe é la stessa che dovrebbe essere implementata nel controllo di macchina e nel software che modella il motore elettrico della Squadra Corse, solo che purtroppo ad oggi non si dispone ancora del suo modello magnetico.

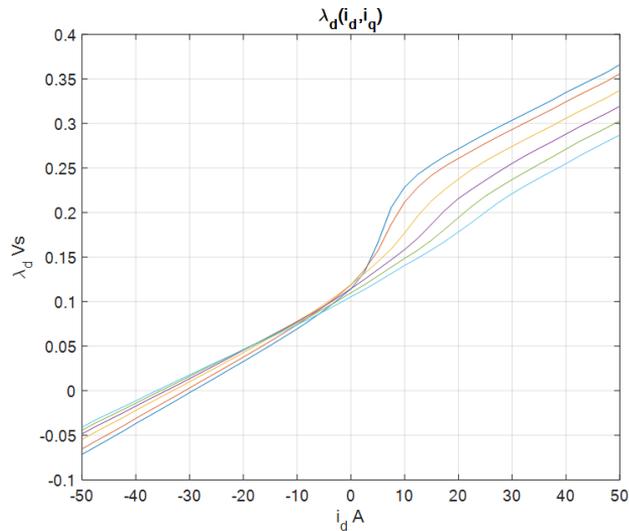


Figura 2.5: Flusso magnetico in asse d tenendo fissata la corrente in asse q . Si puó notare la saturazione dei ponticelli.

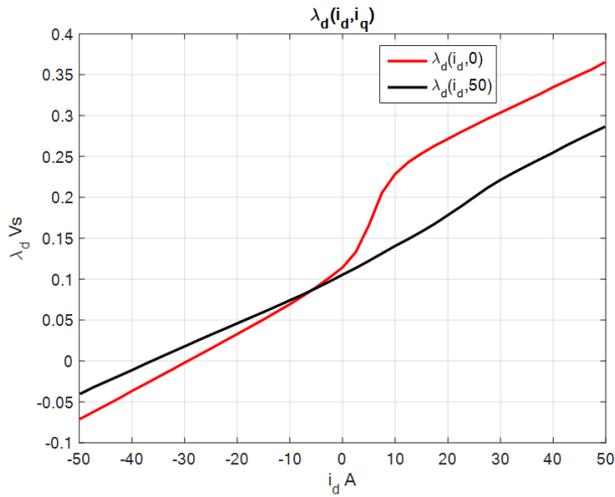


Figura 2.6: Flusso magnetico in asse d tenendo fissata la corrente in asse q nei suoi due valori limiti.

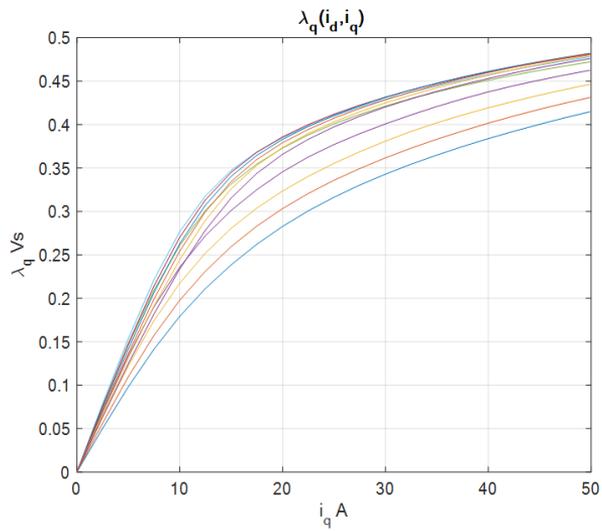


Figura 2.7: Flusso magnetico in asse q tenendo fissata la corrente in asse d .

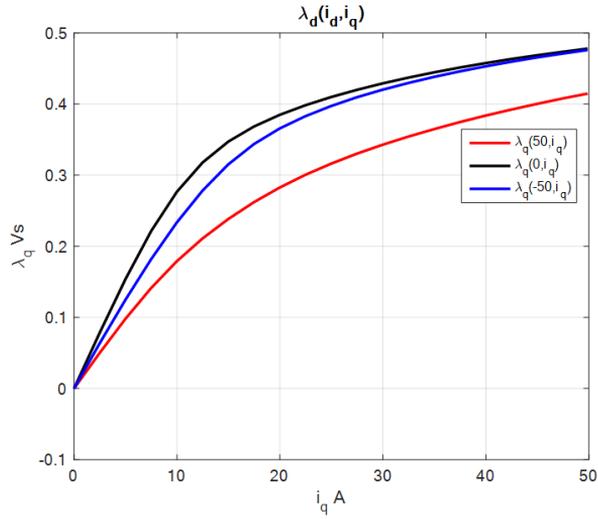


Figura 2.8: Flusso magnetico in asse q tenendo fissata la corrente in asse d nei suoi tre valori caratteristici.

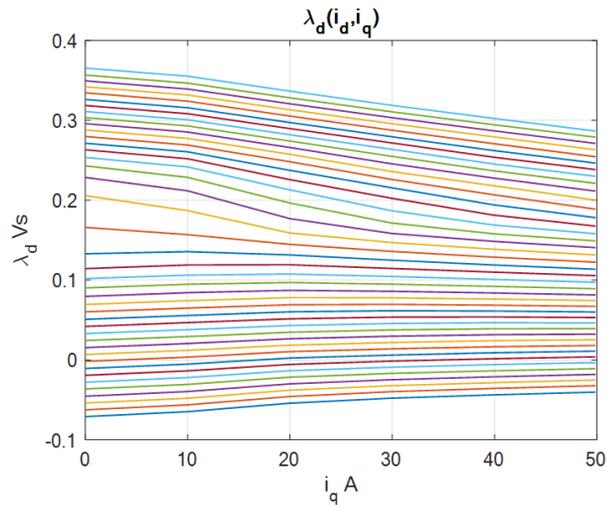


Figura 2.9: Flusso magnetico in asse d tenendo fissata la corrente in asse q .

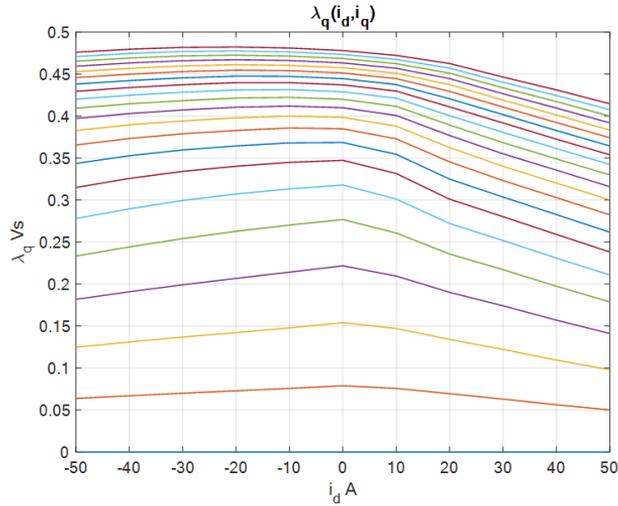


Figura 2.10: Flusso magnetico in asse q tenendo fissata la corrente in asse q .

Nei precedenti grafici si nota subito come nell'asse a maggiore induttanza (q) si osserva la classica caratteristica in saturazione di qualsiasi materiale ferromagnetico, mentre sull'asse dei magneti (d) si può notare la presenza dei magneti stessi in assenza di corrente e la saturazione molto veloce dei *ponticelli*.

Infine si fanno vedere le induttanze sui due assi, sotto forma di tabelle in due dimensioni, le stesse che potrebbero essere caricate su un software di simulazione o all'interno di un codice di controllo:

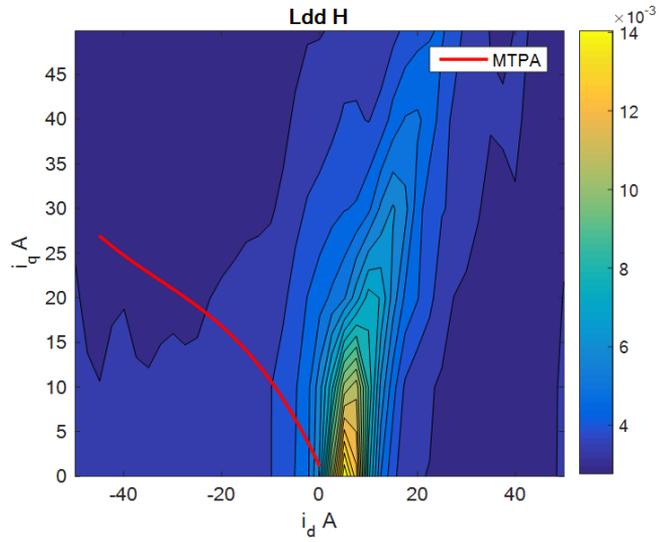


Figura 2.11: Induttanza sincrona in asse d .

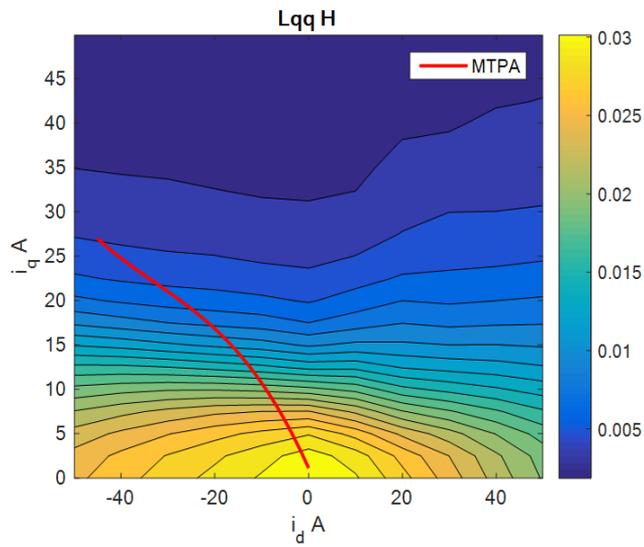


Figura 2.12: Induttanza sincrona in asse q .

2.3 Equazione della coppia

La definizione del modello bifase del motore in assi rotorici permette di ricavare l'equazione di coppia, secondo step fondamentale per implementare il modello di macchina nei prossimi ambienti di simulazione.

Si ricorda che il fattore $\frac{3}{2}$ presente nell'equazione non ha nessun significato fisico, ma è necessario per eseguire con coerenza i calcoli in bifase di una realtà fisica trifase.

$$\begin{aligned} T &= \frac{3}{2} p (\bar{\Lambda}_{dq} \wedge \bar{I}_{dq}) = \\ &= \frac{3}{2} p [(L_d - L_q) i_d i_q + \lambda_M i_q] \end{aligned} \quad (2.7)$$

Analizzando l'equazione è evidente come la coppia abbia un primo contributo dovuto alla coppia anisotropa ed un secondo contributo dovuto alla coppia isotropa della macchina.

2.4 MTPA

La natura anisotropa del motore elettrico comporta un angolo di MTPA (*Maximum Torque per Ampere*) variabile, ovvero un angolo ottimale γ in assi dq del vettore corrente per massimizzare, a corrente fissata, la coppia erogata.

Si definisce γ l'angolo generico tra il vettore corrente nel secondo quadrante e l'asse q . Nell'ipotesi semplificata di linearità magnetica e di un flusso dei magneti indipendente dalla temperatura, riscrivendo la coppia:

Osservando l'equazione di coppia (2.7) è chiaro che se si volesse generare coppia motrice bisognerebbe lavorare nel secondo quadrante del sistema di riferimento in assi dq .

$$i_d = -I \sin\gamma$$

$$i_q = I \cos \gamma$$

$$T = \frac{3}{2} p [-(L_d - L_q) I^2 \frac{\sin 2\gamma}{2} + \lambda_M I \cos \gamma]$$

$$\left[\frac{\partial T}{\partial \gamma} \right]_{I=\text{cost}} = 0 \Rightarrow \sin(\gamma_{MTPA}) = \frac{\lambda_M}{4(L_q - L_d) I} + \sqrt{\left(\frac{\lambda_M}{4(L_q - L_d) I} \right)^2 + \frac{1}{2}} \quad (2.8)$$

In un controllo reale naturalmente la caratteristica di MTPA deve tenere conto del modello magnetico del motore. Facendo riferimento alle mappe di esempio viste precedentemente per ogni valore del vettore corrente si trova l'angolo che massimizza la coppia.

Seguono alcune caratteristiche di MTPA in linearità magnetica in riferimento a vari tipi di motori sincroni già visti in Fig. 2.1:

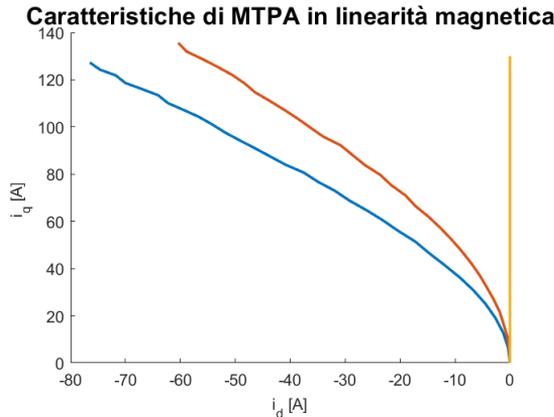


Figura 2.13: Caratteristica MTPA di un motore sincrono ad alta anisotropia con $\xi = 10$ (blu), di un motore sincrono a bassa anisotropia con $\xi = 2$ (rosso) e di un motore sincrono SPM (giallo).

Nel momento in cui si dispone, come accade nella realtà, del modello magnetico, si può adottare il procedimento indicato successivamente nel quarto capitolo integrato con le varie tabelle delle mappe di flusso.

Capitolo 3

Controllo di macchina: *Direct Flux Vector Control*

In questo capitolo é presentato il controllo di macchina, oggetto di questa tesi, conosciuto in letteratura come *Direct Flux Vector Control (DFVC)*. Sono ricavate le sue equazioni ed é spiegato ogni suo blocco fondamentale.

Il concetto di un controllo unificato é presente da piú di dieci anni, ovvero adoperare con una logica di controllo valida per qualsiasi azionamento indipendentemente dalla natura del motore elettrico; per questo lavoro di tesi si ricorda che il motore é un sincro trifase IPM. Il controllo DFVC viene chiamato nel gergo anche *controllo in $f\tau$* , in quanto il flusso di macchina é direttamente controllato in asse f , con direzione e verso del vettore di flusso di statore; mentre la coppia é controllata tramite la componente in asse τ del vettore corrente. Come si può vedere in Fig. 3.1 gli assi rotanti nel DFVC differiscono dagli assi canonici dq del controllo in field-oriented per l'angolo di carico (stimato dall'osservatore di flusso). Si anticipa quindi che il controllo trasforma il modello bifase statorico in un modello bifase rotante non coincidente con gli assi canonici di rotore, ma con un sistema di riferimento ulteriormente ruotato per l'angolo di carico.

Il punto di forza del DFVC risulta un controllo di tensione molto efficiente dinamicamente in tutta la regione di deflussaggio, per qualsiasi modello di macchina in esame.

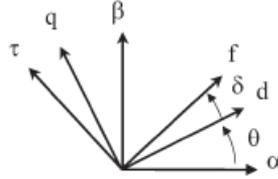


Figura 3.1: Sistema bifase fisso di statore in $\alpha\beta$, sistema rotante negli assi canonici dq e sistema rotante di riferimento $f\tau$ per il controllo DFVC. I tre sistemi differiscono tra loro per l'angolo di posizione θ del rotore (misurato dall'encoder) e per l'angolo di carico δ (stimato dall'osservatore).

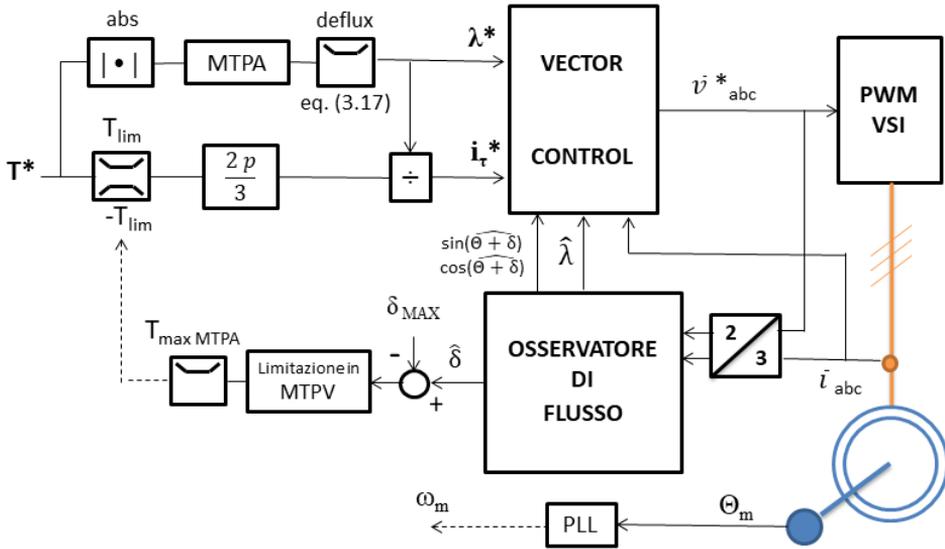


Figura 3.2: Schema generale del DFVC implementato in questo progetto di tesi.

Ora segue la presentazione dei vari blocchi fondamentali del controllo (Fig. 3.2) dopo aver ricavato le equazioni del controllo, seguendo un ordine che coincide con la sequenza di istruzioni del codice implementato in C:

- Calcolo della velocità con l'impiego di un PLL.
- Osservatore di flusso.
- Controllo di macchina (*Vector Control*).
- Deflussaggio.
- Limitazione dell'angolo di carico in MTPV.
- Controllo delle tensioni di riferimento in *Phase Advancing*.

3.1 Equazioni del controllo DFVC

Per equazioni del controllo DFVC si intende trovare le equazioni elettriche in funzione delle variabili di stato, ovvero le due correnti, su i due assi $f\tau$. Partendo da un modello elettromagnetico quanto piú generale possibile si iniziano a definire le equazioni elettriche e magnetiche di macchina su due assi rotanti qualsiasi, quindi non ci si pone sui canonici assi rotorici come é stato fatto per il modello di macchina nel precedente capitolo. Naturalmente si fa sempre riferimento ad un motore sincro IPM.

Partendo da (2.4) si procede con la trasformata di Clarke e si effettua una trasformazione di rotazione sui due assi rotanti **generici**, ruotati rispetto al riferimento statorico con l'angolo $\theta + \delta$ (dove θ é la posizione dell'asse d dei magneti, mentre δ per ora rappresenta un angolo generico), ottenendo:

Per semplificare leggermente la dimostrazione si fa riferimento ad un paio poli di macchina unitario.

$$\begin{cases} v_x = R i_x + \frac{d \lambda_x}{dt} - (\omega_R + \omega_{slip}) \lambda_y \\ v_y = R i_y + \frac{d \lambda_y}{dt} + (\omega_R + \omega_{slip}) \lambda_x \end{cases} \quad (3.1)$$

$$\begin{cases} \lambda_x = L_d i_x + M i_x (1 - \cos(2\delta)) + M i_y \sin(2\delta) + \lambda_M \cos\delta \\ \lambda_y = L_q i_y - M i_y (1 - \cos(2\delta)) + M i_x \sin(2\delta) - \lambda_M \sin\delta \end{cases}$$

$$\text{dove: } M = \frac{L_q - L_d}{2}; \quad \omega_{slip} = \frac{d \delta}{dt}$$

Si chiamano:

$$\begin{aligned} k_s &= M \sin(2\delta) \\ k_e &= M (1 - \cos(2\delta)) \\ \lambda_{Mx} &= \lambda_M \cos\delta \\ \lambda_{My} &= -\lambda_M \sin\delta \\ e_d &= L_d + k_e \\ e_q &= L_q - k_e \end{aligned}$$

E si osserva che:

$$\frac{d k_s}{d \delta} = 2 (M - k_e)$$

$$\frac{d k_e}{d \delta} = 2 k_s$$

$$\frac{d \lambda_{Mx}}{d \delta} = \lambda_{My}$$

$$\frac{d \lambda_{My}}{d \delta} = -\lambda_{Mx}$$

$$\frac{d\lambda_x}{d\delta} = e_d \frac{di_x}{d\delta} + k_s \frac{di_y}{d\delta} + 2k_s i_x + 2(M - k_e) i_y + \lambda_{My}$$

$$\frac{d\lambda_y}{d\delta} = k_s \frac{di_x}{d\delta} + e_q \frac{di_y}{d\delta} + 2(M - k_e) i_x - 2k_s i_y - \lambda_{Mx}$$

Ora si possono riscrivere le equazioni elettriche e magnetiche nelle seguenti equazioni:

$$\begin{cases} v_x = R i_x + \omega_{slip} \left(\frac{d\lambda_x}{d\delta} - \lambda_y \right) - \omega_R \lambda_y \\ v_y = R i_y + \omega_{slip} \left(\frac{d\lambda_y}{d\delta} + \lambda_x \right) + \omega_R \lambda_x \\ \lambda_x = L_d i_x + k_e i_x + k_s i_y + \lambda_{Mx} \\ \lambda_y = k_s i_x + L_q i_y - k_e i_y + \lambda_{My} \end{cases} \quad (3.2)$$

Esplicitando i termini delle equazione elettriche si ricava:

$$\begin{aligned} \frac{d\lambda_x}{d\delta} - \lambda_y &= e_d \frac{di_x}{d\delta} + k_s \frac{di_y}{d\delta} + k_s i_x + 2 \left(M - k_e - \frac{e_q}{2} \right) i_y \\ \frac{d\lambda_y}{d\delta} + \lambda_x &= e_q \frac{di_y}{d\delta} + k_s \frac{di_x}{d\delta} - k_s i_y + 2 \left(M - k_e + \frac{e_d}{2} \right) i_x \end{aligned}$$

Si definiscono altri due ulteriori termini:

$$\begin{aligned} 2 \left(M - k_e - \frac{e_q}{2} \right) &= M (1 + \cos(2\delta)) - L_q = k_q \\ 2 \left(M - k_e + \frac{e_d}{2} \right) &= M (1 + \cos(2\delta)) + L_d = k_d \end{aligned}$$

Ricordando semplicemente che in generale $\omega_{slip} \frac{di}{d\delta} = \frac{di}{dt}$, ora si possono presentare le due equazioni elettriche esplicitando le derivate di

corrente, ma non esplicitando i rispettivi flussi:

$$\left\{ \begin{array}{l} (e_d - \frac{k_s^2}{e_q}) \frac{d i_x}{dt} = - [R + \omega_{slip} (k_s - \frac{k_s k_d}{e_q})] i_x + \dots \\ \dots - [\omega_{slip} (k_q + \frac{k_s^2}{e_q}) - \frac{k_s}{e_q} R] i_y + \dots \\ \dots + (v_x + \omega_R \frac{k_s}{e_q} \lambda_x) - (\frac{k_s}{e_q} v_y - \omega_R \lambda_y) \\ (e_q - \frac{k_s^2}{e_d}) \frac{d i_y}{dt} = - [\omega_{slip} (k_d - \frac{k_s^2}{e_d}) - \frac{k_s}{e_d} R] i_x + \dots \\ \dots - [R - \omega_{slip} (k_s + \frac{k_s k_q}{e_d})] i_y + \dots \\ \dots - (\frac{k_s}{e_d} v_x + \omega_R \lambda_x) + (v_y - \omega_R \frac{k_s}{e_d} \lambda_y) \end{array} \right. \quad (3.3)$$

Ora si conclude questa dimostrazione nel ricavare le equazioni di stato delle correnti (diventano equazioni di stato se si esplicitano i rispettivi flussi) per concentrarsi invece sullo specifico sistema di riferimento del DFVC. Adesso la rotazione é eseguita con un angolo sempre pari a $\theta + \delta$, dove però δ coincide con l'angolo di carico (Fig. 3.3).

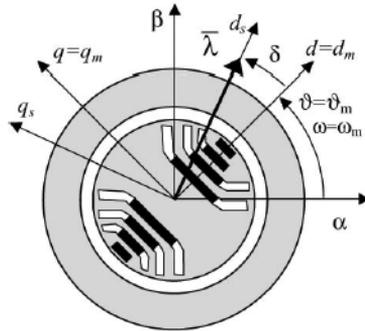


Figura 3.3: Assi di riferimento del DFVC in $f\tau$ sfasati dell'angolo di carico δ rispetto ai canonici assi dq in un motore sincrono IPM. Figura proposta in [3].

Si procede parallelamente iniziando prima a ricavare le equazioni elettriche con le opportune trasformate partendo da (2.4):

$$\begin{cases} v_f = R_s i_f + \frac{d\lambda}{dt} \\ v_\tau = R_s i_\tau + (\omega + \omega_{slip}) \lambda \end{cases} \quad (3.4)$$

Mentre ora si ricavano le equazioni di stato, esplicitando i flussi, nel sistema di riferimento $f\tau$. Si riprendono le equazioni (3.3) adattate al nuovo sistema di riferimento e i flussi sono esplicitati tramite le due equazioni magnetiche presenti in (3.2).

Il sistema di riferimento del DFVC allinea l'asse generico x , utilizzato fin ora, con il versore del flusso di statore, in questo modo si ottiene $\lambda_y = 0$.

$$\left\{ \begin{array}{l} (e_d - \frac{k_s^2}{e_q}) \frac{di_f}{dt} = - [R + \omega_{slip} (k_s - \frac{k_s k_d}{e_q}) - \omega_R (k_s \frac{e_d}{e_q})] i_f + \dots \\ \dots - [\omega_{slip} (k_q + \frac{k_s^2}{e_q}) - \omega_R (\frac{k_s^2}{e_q}) - \frac{k_s}{e_q} R] i_\tau + \dots \\ \dots + (v_f + \omega_R \frac{k_s}{e_q} \lambda_{Mf}) - \frac{k_s}{e_q} v_\tau \\ (e_q - \frac{k_s^2}{e_d}) \frac{di_\tau}{dt} = - [\omega_{slip} (k_d - \frac{k_s^2}{e_d}) + \omega_R e_d - \frac{k_s}{e_d} R] i_f + \dots \\ \dots - [R - \omega_{slip} (k_s + \frac{k_s k_q}{e_d}) + \omega_R k_s] i_\tau + \dots \\ \dots - (\frac{k_s}{e_d} v_f + \omega_R \lambda_{Mf}) + v_\tau \end{array} \right. \quad (3.5)$$

I due ultimi sistemi ricavati, (3.4) e (3.5), devono coincidere. Infatti le rispettive espressioni a regime portano alla medesima formulazione.

Inoltre guardando alla (3.5) si ricavano le induttanze nei rispettivi due assi, analizzando i coefficienti delle derivate di corrente. Entrambe, come ci si aspetta, risultano essere funzioni dell'angolo di carico

δ :

$$L_f = e_d - \frac{k_s^2}{e_q} = \frac{L_d L_q}{\frac{L_d + L_q}{2} + \frac{L_q - L_d}{2} \cos(2\delta)} \quad (3.6)$$

$$L_\tau = e_q - \frac{k_s^2}{e_d} = \frac{L_d L_q}{\frac{L_d + L_q}{2} - \frac{L_q - L_d}{2} \cos(2\delta)}$$

Facendo uno studio di funzione delle precedenti formule si nota come entrambe le induttanze variano tra L_d e L_q , ma questo aspetto sarà ripreso successivamente nella sezione dedicata alla taratura degli anelli di controllo.

Riprendendo invece la (3.4), si conclude questa sezione con il diagramma vettoriale a regime delle equazioni elettriche in $f\tau$:

$$\bar{V}_{f\tau} = R_s \bar{I}_{f\tau} + j\omega \bar{\Lambda}_{f\tau}$$

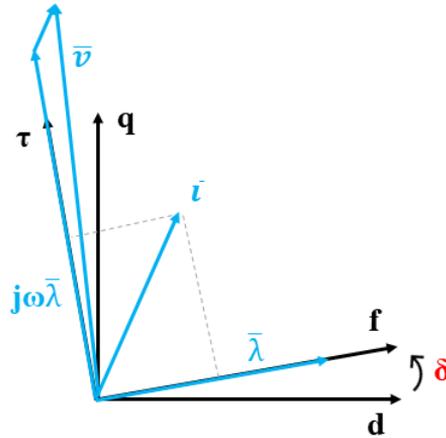


Figura 3.4: Schema vettoriale a regime delle equazioni elettriche del controllo in DFVC.

3.2 Calcolo della velocità

Nel controllo il calcolo della velocità meccanica, e quindi elettrica, del rotore é molto importante perché essa é impiegata nel deflussaggio, nel feed-forward degli anelli e nel *Phase Advancing*. Nel mondo degli azionamenti sostanzialmente si adoperano due diversi metodi di calcolo: il metodo derivativo o tramite PLL (*Phase-Locked Loop*). Entrambi prevedono come dato in ingresso l'angolo meccanico della posizione di rotore misurata dall'encoder. Il metodo derivativo prevede un filtro passa-basso, ma questo non si sposa con un azionamento per elevate dinamiche che non può ammettere un ritardo nella stima della velocità; per questo motivo si é scelto un PLL per il calcolo della velocità.

3.2.1 Phase-Locked Loop

Un PLL teoricamente genera un vettore rotante inseguitore con l'obiettivo di agganciarsi al vettore rotante da misurare. Praticamente si raggiunge l'obiettivo chiudendo un anello sull'angolo da stimare:

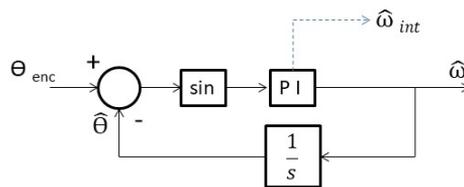


Figura 3.5: Schema a blocchi di un PLL.

In ingresso al regolatore PI entra il seno dell'errore da annullare, questo per non avere problemi di discontinuitá (l'angolo dell'encoder si resetta ad ogni giro meccanico) che porterebbero ad errori del controllo non giustificati. Mentre all'interno del controllo é consigliabile usare la parte integrativa del PI, questo per ottenere una velocità già filtrata e quindi meno rumorosa. Si specifica che la scelta del tipo di integrazione (in retroazione) nell'implementazione del codice

in C, con il metodo di Eulero o alla Tustin, é ininfluente anche per prestazioni di azionamenti ad alta dinamica.

Si riporta il calcolo dei guadagni dell'anello omettendo la loro dimostrazione:

$$\begin{aligned}k_p_{PLL} &= \frac{\omega_c^2}{\sqrt{1 + \tan^2 \Phi_M}} \\k_i_{PLL} &= \frac{\omega_c^2 \tan \Phi_M}{\sqrt{1 + \tan^2 \Phi_M}}\end{aligned}\tag{3.7}$$

Dove ω_c é la pulsazione di crossover in *rad/s* e Φ_M il margine di fase in *rad*.

Prima di proseguire, si specifica solamente che le bande per un PLL non richiedono dinamiche molto spinte, in quanto il vettore rotante generato, una volta agganciato al riferimento, necessita una dinamica molto modesta. Quantificando queste ultime parole si può tarare il PLL per una banda dell'ordine delle decine di Hertz.

3.3 Osservatore di flusso

Nella logica del controllo, a monte degli anelli da controllare, si posiziona l'osservatore di flusso, ovvero un sistema in grado di adoperare con due diversi stimatori di flusso. L'osservatore proposto in Fig. 3.6 calcola una stima del vettore di flusso osservato come risultato di una compromesso continuo tra due modelli di macchina differenti. Infatti dalla teoria dei controlli, come é dimostrato nel seguito, il blocco "g" corrisponde alla pulsazione per cui avviene una graduale transizione tra i due diversi stimatori. Il primo stimatore di flusso (il canale superiore) in assi stazionari é sensibile all'errore per basse velocità per due motivi: la resistenza é un parametro fisso che non tiene conto della sua variazione in funzione della temperatura, quindi la sua caduta di tensione ha un peso maggiore per bassi valori di tensione; l'integrale, essendo retroazionato, a basse velocità non

ha un comportamento integrativo. Il secondo stimatore di flusso (il canale inferiore), invece, é sensibile all'errore per elevate velocità, in quanto le tabelle per il modello magnetico non tengono conto delle perdite del ferro. Le tabelle quindi risultano valide a basse velocità, come viene spiegato in [1], infatti le perdite nel ferro danno un contributo significativo alle alte frequenze. In fase di simulazione si potrebbe anche optare per un modello magnetico di macchina lineare (trascuando quindi la saturazione del ferro), ovvero implementare semplicemente le equazioni magnetiche in assi rotanti non tenendo conto delle mappe di flusso del motore.

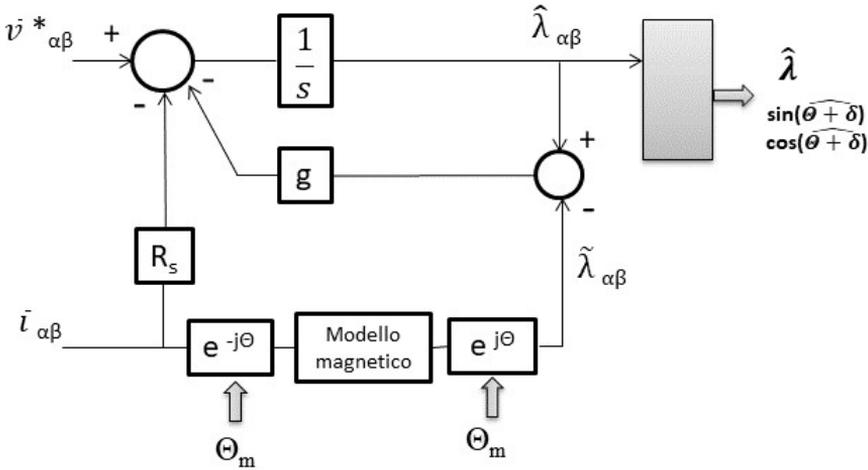


Figura 3.6: Schema a blocchi dell'osservatore di flusso del DFVC.

Nel controllo DFVC l'osservatore é sicuramente uno dei blocchi fondamentali. L'ampiezza del vettore di flusso osservato é il feedback in retroazione per l'anello di flusso, mentre la sua fase nel sistema di riferimento dq é necessaria per effettuare le dovute trasformazioni nel sistema di riferimento fr .

Ora si dimostra brevemente come avviene la transizione tra i due stimatori nell'intorno della pulsazione g . Si inizia non considerando lo stimatore di flusso in assi rotanti con il modello magnetico del

canale inferiore, quindi il flusso stimato in assi stazionari presenta la seguente funzione di trasferimento (evidenziata nei passaggi):

$$\begin{aligned}\widehat{\lambda}_{\alpha\beta} &= \frac{1}{s+g} (\widehat{v}_{\alpha\beta} - R_s \widehat{i}_{\alpha\beta}) = \\ &= \frac{s}{s+g} \frac{(\widehat{v}_{\alpha\beta} - R_s \widehat{i}_{\alpha\beta})}{s} = \\ &= \frac{s}{s+g} \widetilde{\lambda}_{VI}\end{aligned}$$

Analogamente, non considerando il canale superiore, si ricava la funzione di trasferimento del flusso osservato con il flusso stimato con il modello magnetico:

$$\begin{aligned}\widehat{\lambda}_{\alpha\beta} &= \frac{\frac{g}{s}}{1 + \frac{g}{s}} \widetilde{\lambda}_{\alpha\beta \text{ modello magnetico}} = \\ &= \frac{g}{s+g} \widetilde{\lambda}_{\alpha\beta \text{ modello magnetico}}\end{aligned}$$

Ottendo una combinazione lineare dinamica del flusso osservato:

$$\widehat{\lambda}_{\alpha\beta} = \frac{s}{s+g} \widetilde{\lambda}_{VI} + \frac{g}{s+g} \widetilde{\lambda}_{\alpha\beta \text{ modello magnetico}} \quad (3.8)$$

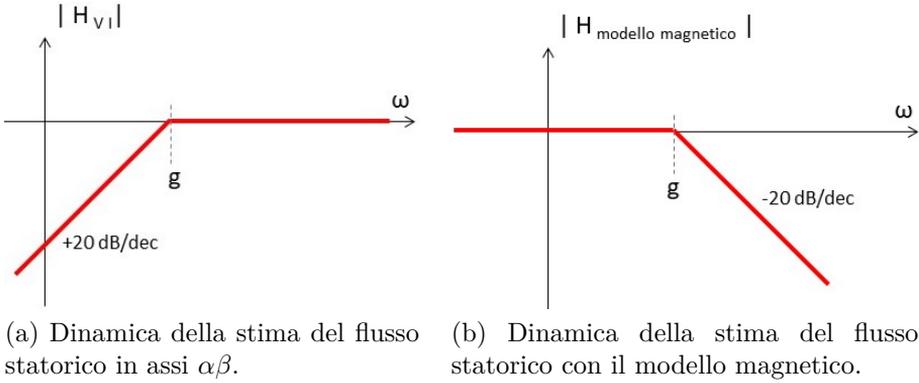


Figura 3.7: Diagrammi di Bode delle due diverse stime di flusso. Si nota come la pulsazione g decreta il passaggio da una stima all'altra.

In Fig. 3.7 risulta piú chiaro come il parametro g deve essere tarato accuratamente in base a quali frequenze il modello magnetico risulta attendibile e decreta la zona di transizione da un modello all'altro. In generale la pulsazione g assume valori dalle decine alle centinaia di rad/s .

3.4 Controllo di macchina

Il cuore del controllo sono i due anelli retroazionati che forzano i rispettivi riferimenti sui due assi di riferimento $f\tau$. In asse f é controllato il flusso di statore, mentre in asse τ la componente del vettore rotante della corrente. Nel sistema di riferimento del DFVC é piú utile riscrivere l'equazione di coppia (2.7) nel modo seguente:

$$T = \frac{3}{2} p \lambda i_\tau \quad (3.9)$$

3.4.1 Anello in asse f

In un controllo di coppia il riferimento di flusso é generato dalle tabelle in MTPA (dalla mappatura magnetica del motore), saturato

dall'equazione di deflussaggio. Dall'equazione elettrica in asse f si costruisce il suo diagramma a blocchi per il successivo calcolo dei guadagni dell'anello:

$$\text{Equazione elettrica asse } f: v_f = R_s i_f + \frac{d\lambda}{dt} \quad (3.10)$$

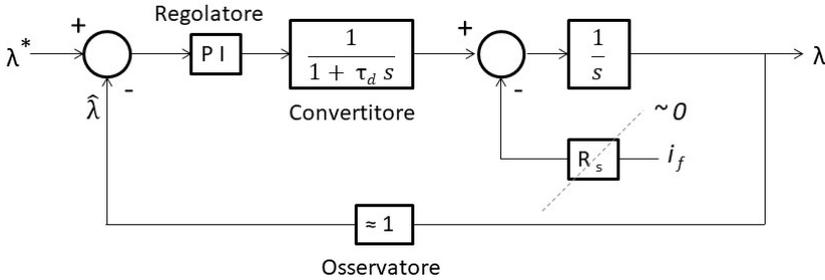


Figura 3.8: Schema a blocchi dell'asse f del controllo DFVC.

Trascurando cadute resistive di macchina, il convertitore é modellizzato con un ritardo approssimato con un filtro del primo ordine. Il ritardo é calcolato come la somma di un periodo di commutazione (il ritardo intrinseco del controllo tra il campionamento delle grandezze e l'invio dei comandi) ed il ritardo medio di attuazione nel periodo successivo, ovvero metà del periodo di commutazione.

$$\tau_d = \frac{3}{2} T_s$$

La funzione di trasferimento in anello aperto in asse d_s risulta:

$$H_{OL}(\omega) = \frac{k_P}{s^2} \frac{\frac{k_I}{k_P} + s}{1 + \tau_d s} \quad \text{con } \frac{k_I}{k_P} = \omega_z$$

Si rappresenta il corrispondente diagramma di Bode del modulo, posizionando una frequenza di crossover (equivalente circa alla frequenza di banda) in un tratto a pendenza di -20 dB/dec, *sufficientemente lontana* dai due punti di cambio pendenza (-40 dB/dec), al

fine di avere un margine di fase accettabile in linea con la teoria dei controlli.

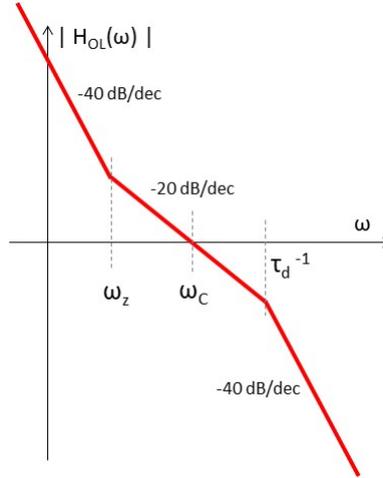


Figura 3.9: Diagramma di Bode del modulo della funzione di trasferimento in asse f . In figura si approssima l'intersezione con l'asse delle ascisse al fine di ottenere una frequenza di crossover (ω_c) in linea con un margine di fase accettabile.

Nell'intorno della frequenza di crossover la funzione di trasferimento in anello aperto si può approssimare come segue:

$$|H_{OL}(\omega_c)| \approx \frac{k_P}{\omega_c} = 1 \Rightarrow k_P \equiv \omega_c$$

Infine si giunge al calcolo dei guadagni dell'anello ricordando che tra le tre pulsazioni (zero del regolatore, pulsazione di banda, polo del convertitore) ci sia almeno una decade di distanza. Nell'ipotesi che la frequenza di crossover coincida con la frequenza di banda (f_b) del sistema:

$$\begin{aligned} k_P &= 2\pi f_b \\ k_I &= \frac{1}{10} k_p 2\pi f_b \end{aligned} \tag{3.11}$$

Feed-forward

Nel campo dell'alta dinamica é ormai comune inserire un feed-forward per aiutare ed efficientare il controllo a raggiungere piú velocemente il riferimento. In questo specifico caso, facendo riferimento a (3.10), si puó trascurare la dinamica di flusso e porre semplicemente la caduta resistiva come feed-forward:

$$ffw_f = R_s i_f \quad (3.12)$$

Saturazione della tensione in uscita

Osservando (3.10) si puó saturare la tensione di riferimento in uscita dall'anello al valore della sola componente resistiva. In linea di massima, per ottenere un valore di riferimento, si consiglia di basarsi su una caduta resistiva con la corrente massima ammessa dal controllo (generalmente non uguale, ma minore, della sovracorrente massima di protezione). Si pone attenzione come sia giusto che l'anello necessiti solo di una minima parte della tensione sul DC-link, in quanto il riferimento di flusso non richiederá mai una dinamica particolarmente spinta.

3.4.2 Anello in asse τ

Il riferimento di corrente si ricava da (3.9) conoscendo il riferimento di coppia voluto e il flusso, il quale come accennato prima é generato dalle tabelle in MTPA (dalla mappatura magnetica del motore), saturato dall'equazione di deflussaggio. Il riferimento di corrente potrebbe essere ulteriormente saturato dalla limitazione dell'angolo di carico in MTPV, ma per il momento é un aspetto che non influenza la taratura dell'anello e sará apporofondito piú avanti.

Nel campo degli azionamenti elettrici, un controllo di corrente in assi rotanti (quindi grandezze continue) presenta il canonico calcolo dei guadagni, di cui si riporta una breve dimostrazione. Si immagina di voler controllare la corrente in un modello dove si trascurano cadute resistive e quindi é presente solamente l'induttanza equivalente

del circuito e si trascurano inoltre i vari disturbi additivi che non devono essere considerati per la taratura dell'anello. Risulta il seguente diagramma a blocchi:

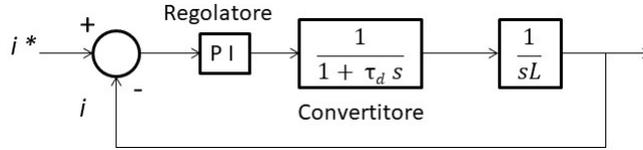


Figura 3.10: Schema a blocchi semplificato di un generale controllo di corrente in un induttore.

Con la corrispondente funzione in anello aperto:

$$H_{OL}(\omega) = \frac{k_P}{L} \frac{1}{s^2} \frac{\frac{k_I}{k_P} + s}{1 + \tau_d s} \quad \text{con} \quad \frac{k_I}{k_P} = \omega_z$$

Il rispettivo diagramma di Bode coincide con quello già illustrato in Fig. 3.9, quindi nell'intorno della frequenza di crossover la funzione di trasferimento in anello aperto si può approssimare come segue:

$$|H_{OL}(\omega_c)| \approx \frac{k_P}{\omega_c L} = 1 \Rightarrow k_P \equiv \omega_c L$$

$$\begin{aligned} k_P &= \omega_c L_\tau \\ k_I &= \frac{1}{10} k_p \omega_c \end{aligned} \tag{3.13}$$

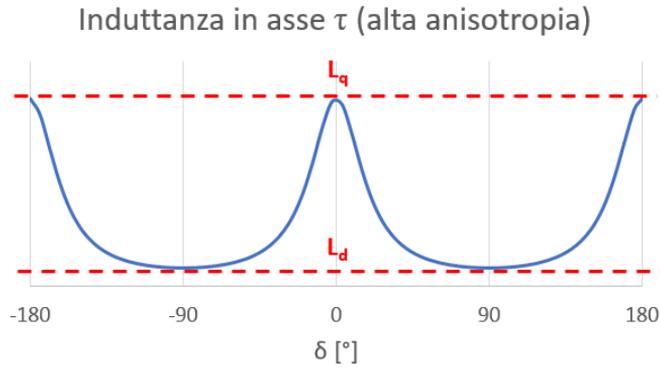
Dove L_τ é l'induttanza equivalente dell'equazione elettrica della corrente da controllare in asse τ .

Negli assi f τ però, come già anticipato e dimostrato in (3.6), le induttanze sui rispettivi assi non sono costanti, perché funzioni dell'angolo di carico. Anche prendendo semplicemente alcuni valori notevoli dell'angolo si apprezza la sua variabilità:

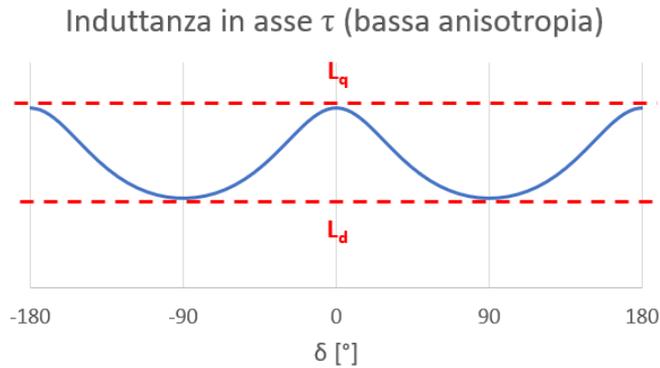
$$\text{se } \delta = 0^\circ \Rightarrow L_\tau = L_q$$

$$\text{se } \delta = 90^\circ \Rightarrow L_\tau = L_d$$

Riprendendo le formulazioni in (3.6) si plotta l'induttanza in asse τ (l'induttanza in asse f presenta gli stessi andamenti solamente traslati di 90° di angolo di carico) per due diversi ξ fattori di anisotropia:



(a)



(b)

Figura 3.11: Induttanza in asse τ in funzione dell'angolo di carico δ . L'induttanza come si può notare è sempre compresa tra L_q e L_d . In (a) il grafico è plottato con un fattore di anisotropia $\xi = 10$; mentre in (b) $\xi = 2$.

Implementare un guadagno proporzionale variabile in funzione dell'angolo di carico, sarebbe un appesantimento del codice non giustificato. Si trova soluzione a questo problema ponendo molto praticamente:

$$L_\tau = \frac{L_d + L_q}{2} \quad (3.14)$$

Feed-forward

Come analogamente spiegato nella precedente taratura dell'altro anello, per una prestazione efficiente alle alte dinamiche é necessario una retroazione in feed-forward.

$$\text{Equazione elettrica asse } \tau: v_\tau = R_s i_\tau + \lambda \left(\omega + \frac{d\delta}{dt} \right) \quad (3.15)$$

In feed-forward si ritiene trascurabile la dinamica del vettore di flusso, e quindi si considera nullo il suo scorrimento rispetto agli assi canonici dq .

$$ffw_\tau = R_s i_\tau + \lambda \omega \quad (3.16)$$

Saturazione della tensione in uscita

L'anello sar  saturato in tensione con quella rimanente dalla saturazione precedente in asse f . Si pone questa volta attenzione come ora sia giusto che l'anello necessiti della maggior parte della tensione sul DC-link, in quanto il riferimento di coppia, e quindi di corrente in asse τ , possa richiedere una dinamica particolarmente spinta.

3.5 Deflussaggio

Si era gi  precedentemente accennato come il controllo in deflussaggio fosse uno dei punti di forza del DFVC, in quanto vi sono solamente relazioni algebriche (prive di dinamica) che regolano il flusso di riferimento, permettendo di conseguenza elevate dinamiche per alte velocit . Questo aspetto si apprezzer  solo in parte in questa sezione, perch  verr  percepito in maniera molto pi  significativa quando saranno illustrate le varie simulazioni.

Ricordando lo schema generale del controllo in Fig. 3.2 si nota come il deflussaggio corrisponde ad una semplice saturazione del flusso in MTPA. La saturazione inizia ad intervenire solo per velocit  superiori alla velocit  base, iniziando a limitare il flusso sulla classica

curva iperbolica dell'isopotenza; caratteristica dipendente dalla massima tensione disponibile al DC-link del convertitore, il quale limita in deflussaggio la coppia massima per ogni velocità.

L'obiettivo rimane ricavare un'equazione per calcolare il flusso massimo disponibile, sfruttando tutta la tensione disponibile al DC-link:

Si fa riferimento all'equazione elettrica (3.10) in asse f e (3.15) in asse τ trascurando i vari termini dinamici.

$$\begin{aligned} \frac{V_{DC-link}^2}{3} &= V_{max}^2 = v_f^2 + v_\tau^2 \\ v_\tau &= R_s i_\tau + \omega \lambda \\ \sqrt{V_{max}^2 - v_f^2} &= R_s i_\tau + \omega \lambda \\ \sqrt{V_{max}^2 - (R_s i_f)^2} &= R_s i_\tau + \omega \lambda \\ \lambda^* &\leq \alpha \frac{\sqrt{V_{max}^2 - (R_s i_f)^2} - R_s i_\tau \operatorname{sign}(\omega)}{|\omega|} \end{aligned} \quad (3.17)$$

La (3.17) si può trovare in varie forme più semplificate, a seconda di quanto si vuole essere accurati nel considerare o meno i termini resistivi. Mentre il coefficiente α è consigliabile mantenerlo come proposto in [4] per lasciare margine di tensione ai regolatori dei due assi per evitare di saturare i duty-cycle, quindi il suo valore è sempre minore dell'unità.

3.6 Limitazione dell'angolo di carico in MTPV

In [3] si sottolinea l'importanza di non superare l'angolo di MTPV (*Maximum Torque per Voltage*) di macchina sul piano delle correnti, ovvero il luogo dei punti che massimizza la coppia per ogni ampiezza del vettore tensione, altrimenti si inizia ad esplorare una regione di instabilità del controllo. La macchina eroga una coppia minore di quella massima estraibile e di conseguenza il regolatore così non riesce più ad annullare l'errore in entrata; infatti a rigore non è il motore a diventare un sistema instabile (anzi le macchine elettriche, per loro natura, risultano essere sistemi stabili), ma diventerebbe instabile un eventuale anello di velocità fuori l'anello di coppia. Quindi per qualsiasi modello di un motore elettrico esiste un angolo di carico massimo per il quale si sbatte sul luogo dei punti in MTPV, ma se per i motori asincroni e sincroni SPM o riluttanza puri, questo angolo si ricava dal modello di macchina (come si vede anche in [3]), per i sincroni IPM e PMASR è necessaria una mappatura magnetica. Infatti con le convenzioni adottate:

$$90^\circ < \delta_{max,IPM/PMASR} < 135^\circ \quad (3.18)$$

Tuttavia si può ricavare una formulazione chiusa dell'angolo di MTPV (dove motori asincroni e sincroni SPM risultano sottocasi particolari) sotto le ipotesi di linearità magnetica e flusso dei magneti non dipendente dalla temperatura, come è proposta in [5]:

La formula dell'articolo è riadattata in quanto si sono usate convenzioni degli assi differenti, ma questo naturalmente non ne preclude la sua validità.

$$\left[\frac{dT}{d\delta}\right]_{\lambda=const} = 0 \Rightarrow \delta_{MTPV} = \frac{\pi}{2} + a \sin\left(\frac{-\beta + \sqrt{\beta^2 + 8}}{4}\right) \quad (3.19)$$

$$\text{con: } \beta = \frac{\lambda_M}{\lambda} \frac{\xi}{\xi - 1}; \xi = \frac{L_q}{L_d}$$

Prima di spiegare la taratura di questo anello, rimane spiegare come si può calcolare in prima approssimazione δ_{max} nell'ipotesi di linearità magnetica. Dopo varie simulazioni sull'azionamento in condizioni di deflussaggio spinto (massima coppia trascinando il rotore a velocità massima, con un livello minimo di tensione al DC-link) si registra il valore di minimo del flusso. Con questo stesso valore si ricava $\delta_{MTPV \lambda_{min}}$ da (3.19). Questo angolo massimo di MTPV dovrà poi essere confrontato con quello della simulazione in cui non vi è stata nessuna limitazione in MTPV, per capire meglio se il motore raggiunge realmente la curva di MTPV (e quindi necessita una limitazione dell'angolo di carico) o meno.

Si continua con l'analisi dinamica attraverso la quale si regola effettivamente l'angolo di carico. In [3] si ricorda che limitando la corrente in asse τ è come se si limitasse contemporaneamente l'angolo di carico del motore, scendendo sui punti di lavoro dell'MTPV; ma questa limitazione deve intervenire solo se la macchina è arrivata a lavorare in MTPV, altrimenti questa limitazione della corrente non avrebbe senso. Infatti in Fig. 3.12 si nota che se l'errore in entrata è negativo, la limitazione di corrente è saturata a zero, non avendo nessun effetto sul controllo. Generalmente un motore arriva a toccare l'MTPV per punti di lavoro con velocità elevate, dove è richiesto generalmente un deflussaggio molto spinto.

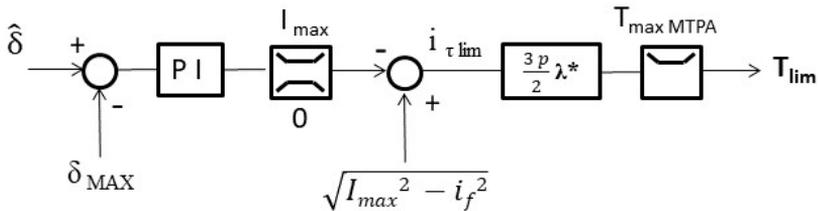


Figura 3.12: Diagramma a blocchi del blocco "Limitazione in MTPV" raffigurato nello schema generale del DFVC in Fig. 3.2.

Si riporta il diagramma a blocchi in [3] per regolare dinamicamente l'anello che limita l'anello di carico:

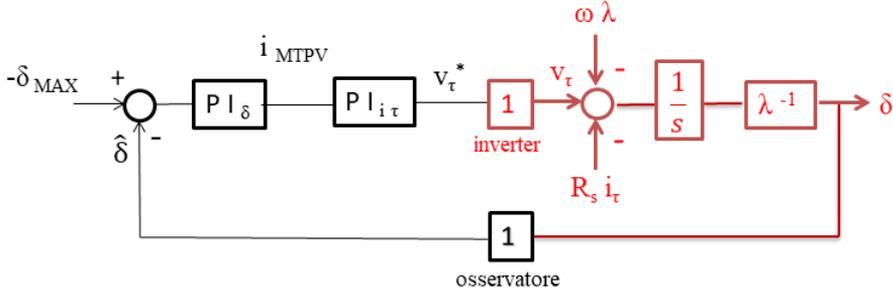


Figura 3.13: Diagramma a blocchi dell'anello di controllo per limitare l'angolo di carico in MTPV. Il controllo limita la corrente in asse q_s solo se l'errore in entrata è positivo (vedi Fig. 3.12). In rosso la dinamica del sistema fisico reale fuori dall'algoritmo di controllo.

Si continua nel calcolo dei guadagni dell'anello esplicitando la funzione di trasferimento in anello aperto e il rispettivo diagramma di Bode del modulo, posizionando la frequenza di crossover in modo da tale da ottenere un margine di fase accettabile.

$$\begin{aligned}
 H_{OL}(\omega) &= \frac{k_{P\delta} s + k_{I\delta}}{s} \frac{k_{Pq} s + k_{Iq}}{s} \frac{1}{s} \frac{1}{\lambda} \\
 &= \frac{k_{P\delta} k_{Pq} s^2 + (k_{P\delta} k_{Iq} + k_{I\delta} k_{Pq}) s + k_{I\delta} k_{Iq}}{\lambda s^3}
 \end{aligned}$$

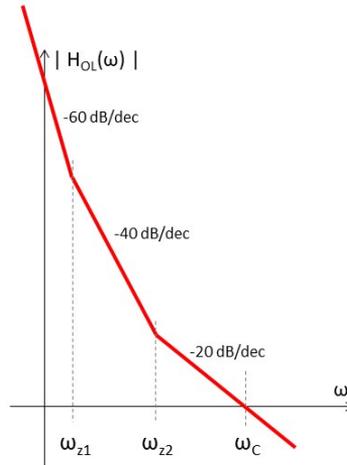


Figura 3.14: Diagramma di Bode del modulo della funzione di trasferimento in anello aperto del controllo per limitare l'angolo di carico in MTPV. Si posiziona qualitativamente una frequenza di crossover (ω_c) a valle dei due zeri (ω_{z1} e ω_{z2}) per ottenere un margine di fase accettabile.

Si procede con lo stesso approccio adottato per la taratura dei due principali anelli di controllo:

$$|H_{OL}(\omega_c)| \approx \frac{k_{P\delta} k_{Pq}}{\lambda \omega_c} = 1 \Rightarrow k_{P\delta} \equiv \frac{\lambda \omega_c}{k_{Pq}}$$

E riproponendo l'approssimazione di confondere la frequenza di crossover con la banda:

$$\begin{aligned} k_{P\delta} &= \frac{\lambda 2\pi f_b}{k_{Pq}} \\ k_{I\delta} &= \frac{1}{10} k_{P\delta} 2\pi f_b \end{aligned} \quad (3.20)$$

Si nota subito come la banda sia in funzione del flusso di statore, pertanto si calcolano i guadagni facendo riferimento al flusso minimo raggiungibile alla massima velocità. Ci si limita, senza dimostrarlo, ad informare che questo anello può portare all'instabilità del controllo

indipendentemente dall'angolo di carico massimo consentito; tuttavia si suggerisce una banda almeno inferiore di una decade rispetto alla banda del controllo di corrente.

3.7 Phase Advancing

Ultimo punto che chiude la presentazione del controllo DFVC é la manipolazione in uscita delle tensioni di riferimento generate con una tecnica di *Phase Advancing*. Questa procedura é implementata nel controllo di questo particolare lavoro di tesi, ma in realtà é un principio applicabile in qualsiasi logica di controllo, effettivamente utile se l'azionamento é progettato per raggiungere elevate velocità.

Il controllo digitale ha un intrinseco ritardo di un periodo di campionamento (che in questo lavoro di tesi coincide con il periodo di commutazione), da quando effettua i calcoli del controllo a quando essi sono effettivamente inviati all'inverter. In un periodo di campionamento il rotore si sposterá di un $\Delta\theta = T_s \omega_R$ facendo spostare gli assi rotanti su cui il controllo vuole generare le tensioni di riferimento. La soluzione a questo problema é una rotazione oraria (se il rotore gira in senso antiorario) almeno dello stesso $\Delta\theta$ delle tensioni generate, per compensare lo spostamento degli assi rotanti che in quel frangente di tempo ruoteranno. Si sottolinea *almeno* perché si potrebbe considerare un ulteriore ritardo di metà periodo, se si vuole tenere conto del tempo di attuazione medio dei comandi. Anche se questo procedimento puó sembrare anti-intuitivo, in questo modo si controllano le grandezze correttamente, evitando che le stesse abbiano componenti sull'asse complementare nell'attuazione dei riferimenti.

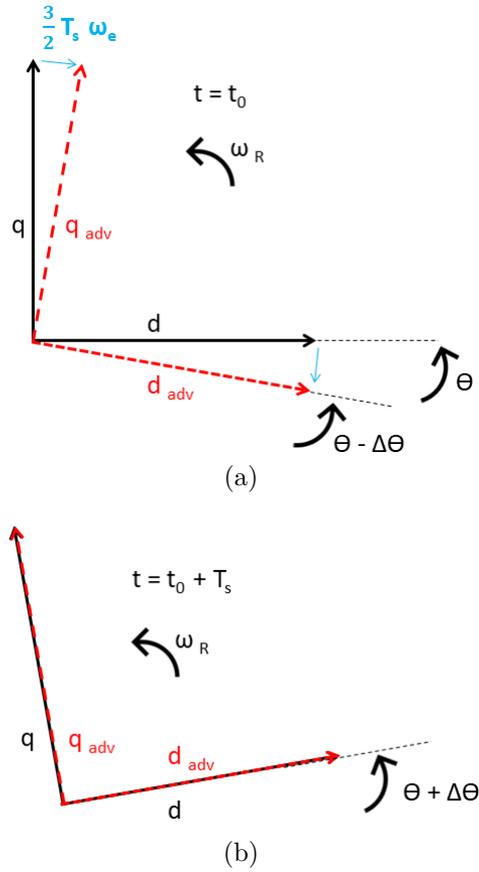


Figura 3.15: In (a) il controllo genera due assi di riferimento (rosso) in ritardo rispetto agli assi rotanti di rotore (nero). Dopo un periodo di campionamento in (b) vengono inviati all’inverter i comandi precedentemente generati, ma gli assi di riferimento del controllo subiscono delle trasformazioni sul motore reale con un angolo pari a $\theta + \Delta\theta$. Questo fa combaciare gli assi del controllo con gli assi di rotore, controllandoli correttamente.

Capitolo 4

Azionamento elettrico della Squadra Corse

In questo capitolo si presenterá l'azionamento della Squadra Corse, oggetto di questa tesi. Riportando i suoi dati tecnici che caratterizzeranno le varie simulazioni che seguono nei due ambienti di simulazione.

La struttura adottata dalla squadra corse prevede due livelli di tensione: un livello per la potenza da 540 V nominali impiegato per la trazione elettrica e un livello di bassa tensione da 24 V per gestire l'elettronica di segnale e alimentare vari ausiliari della monoposto. La trazione elettrica é organizzata con quattro azionamenti gemelli per ogni ruota, composti da un motore elettrico e un inverter. Mentre la batteria per la trazione é montata con un accoppiamento diretto ai quattro DC-link dell'azionamento, senza nessuna conversione nel mezzo (potenzialmente utile per evitare di sovradimensionare gli inverter di trazione che vedono la varianza di tensione della



Figura 4.1

batteria).

Le simulazioni in Simulink, prima, e in Plecs, dopo, si prefissano di modellizzare il controllo di una singola linea di trazione elettrica, ignorando l'interazione tra le quattro linee. Il principale obiettivo sarà testare il codice di controllo in C del DFVC, con il quale si è deciso di controllare l'azionamento. Quindi ora nel seguito saranno riportati i vari dati tecnici dell'azionamento utili per la successiva modellistica.

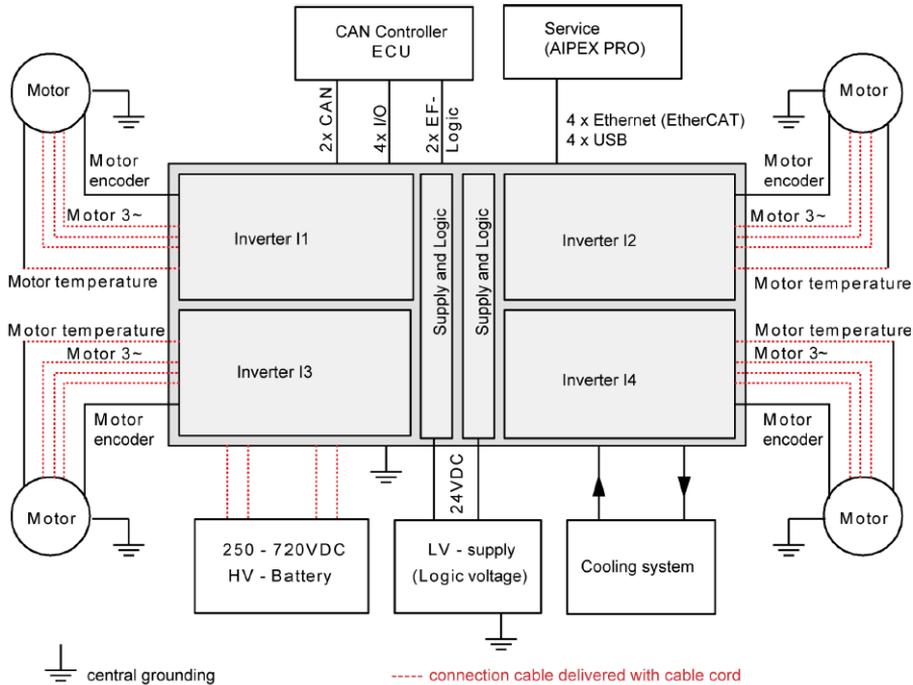


Figura 4.2: Panoramica generale dell'azionamento elettrico della monoposto elettrica della Squadra Corse.

L'inverter dell'azionamento é stato progettato interamente da alcuni ricercatori all'interno del Dipartimento Energia del Politecnico di Torino, mentre il motore elettrico é un sincro IPM prodotto da *AMK*, modello *DD5-14-10-POW-18600-B5*:

DD5 La taglia del motore nella categoria dei servomotori sincroni prodotti da *AMK*.

14 Valore della massima coppia continuativa in *Nm*.

10 Numero di poli del motore elettrico.

P Encoder assoluto tipo *single-turn*, ovvero che non conta il numero di giri.

O Assenza di freno.

W Raffreddamento a liquido.

18600 Velocità massima a vuoto in *rpm*.

B5 Tipo di montaggio meccanico del motore.

La simbologia adottata nella seguente tabella riprende quella elencata prima dell'introduzione di questa tesi e quella presentata nei due precedenti capitoli:

Inverter	V_{DC} nominale	540	V
	Range di tensione al DC-link	250 ÷ 720	V
	f_{sw}	20	kHz
	Tipi di swithces: MOSFET		
	R_{on} MOSFET	20	m Ω
	R_{on} diodo	10	m Ω
	V_0 diodo	0,9	V
Motore Elettrico	P_n	12,3	kW
	V_n	350	V
	I_n	41	Arms
	I_{max} (per 1,24 s)	105	Arms
	p	5	-
	R_s	67,5	m Ω
	L_d	0,12	mH
	L_q	0,24	mH
	λ_M	28,9	mVs
	T_n	9,8	Nm
	T_{max}	21	Nm
	n_{rated}	12000	rpm
	n_{max}	18617	rpm
J	2,74	kgcm ²	

Tabella 4.1: Principali caratteristiche dell'azionamento elettrico utili per la modellistica nei prossimi ambienti di simulazione.

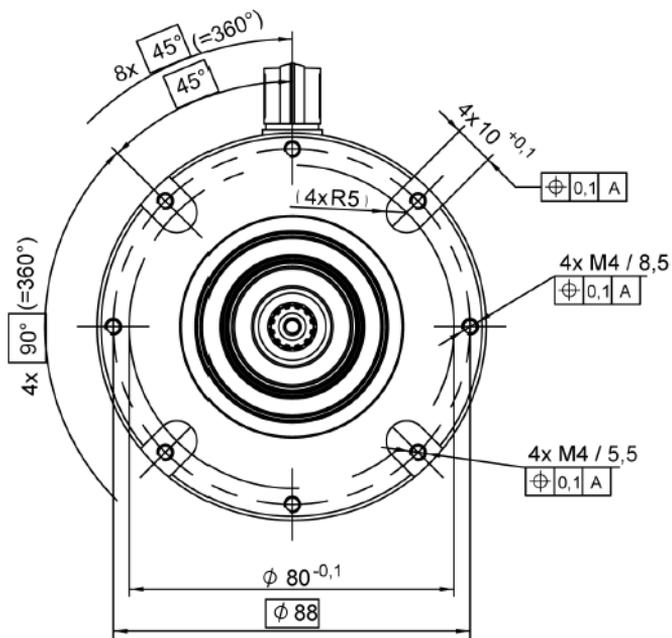


Figura 4.3: Vista frontale quotata del motore elettrico.

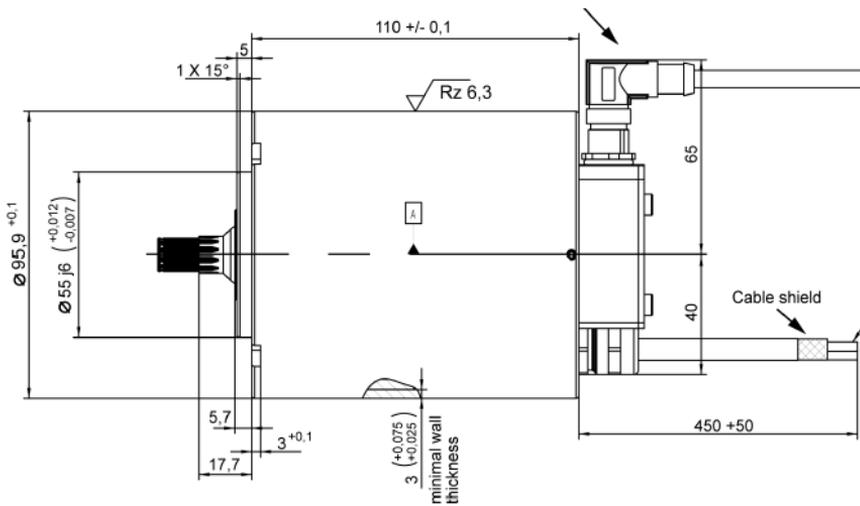


Figura 4.4: Vista laterale quotata del motore elettrico.

4.1 MTPA in linearit  magnetica

Osservando anche 3.2 (tra la coppia di riferimento e come viene generato il flusso di riferimento in asse f),   naturale per qualsiasi controllo cercare di far lavorare il motore in MTPA. La caratteristica di MTPA in questa sezione sar  ricavata con un'ipotesi di linearit  magnetica e con flusso dei magneti costanti.

L'algoritmo per calcolare l'MTPA   stato implementato su un foglio di lavoro in ambiente Matlab. Viene riportato in degli step sequenziali il procedimento adottato, facendo riferimento a (2.6) e (2.7) ricavate precedentemente:

Si ricorda che per le convenzioni adottate, per generare coppia motrice, si deve lavorare sul secondo quadrante sul piano delle correnti.

1. Si definisce un vettore corrente fino al massimo valore ammesso dal controllo e un altro vettore dove si fraziona, a discrezione dell'operatore, l'angolo retto del secondo quadrante sul piano delle correnti.
2. Ora si costruisce, utilizzando l'equazione di coppia e il modello magnetico lineare, una matrice $T(\theta; I)$, ottenendo tutte le combinazioni possibili per la coppia facendo variare il modulo della corrente e l'angolo nel secondo quadrante.
3. Adesso, per ogni valore del modulo di corrente, si seleziona nella matrice della coppia (appena creata) l'angolo per cui si registra il valore di coppia massimo. Alla fine di questa procedura si pu  plottare la caratteristica di MTPA sul piano delle correnti (Fig. 4.5).
4. Infine, tramite il modello magnetico lineare, si ricava a ritroso il flusso magnetico di MTPA in funzione della coppia desiderata, ottenendo la curva che poi dovr  essere caricata in C per il DFVC (Fig. 4.6).

I grafici sono stati ricavati partendo da un'ampiezza massima della corrente di 200 A, mentre i parametri di macchina fanno riferimento ai dati tecnici di Tab. 4.1.

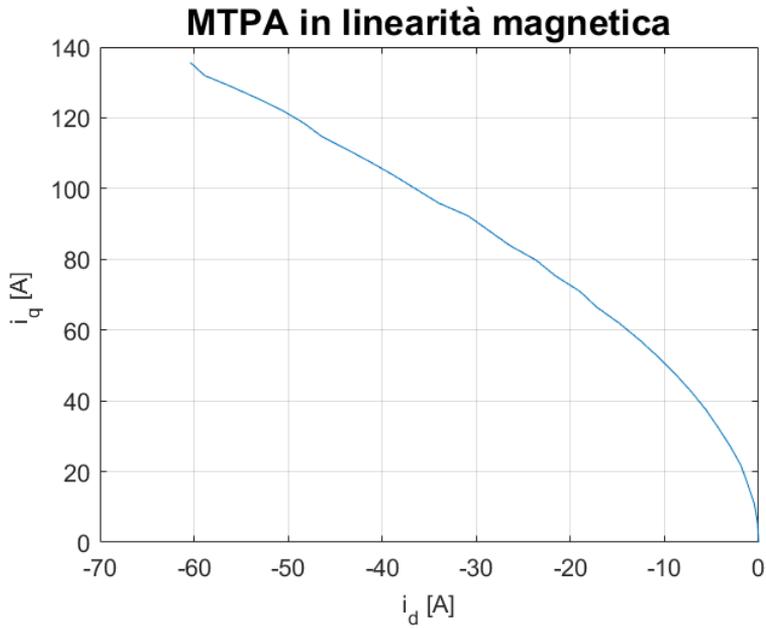


Figura 4.5: Caratteristica di MTPA in linearità magnetica sul secondo quadrante del piano delle correnti.

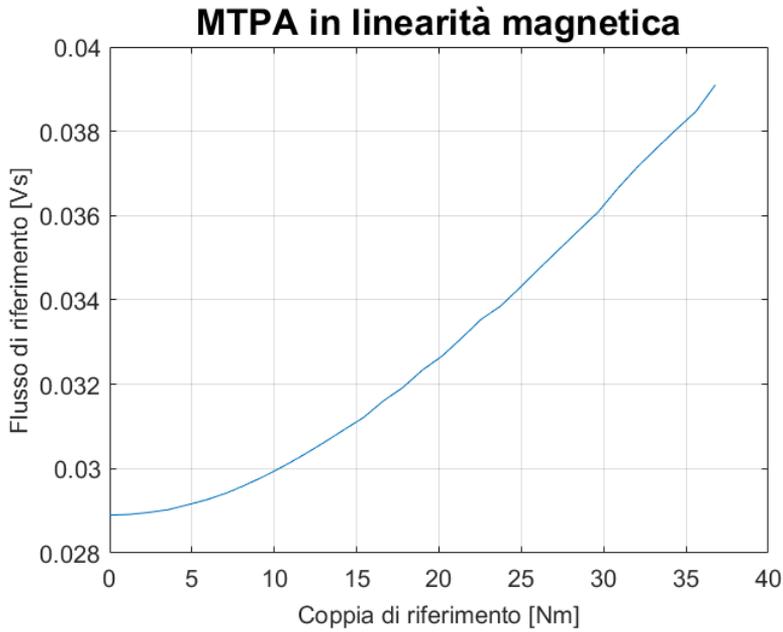


Figura 4.6: Caratteristica del flusso di macchina in funzione della coppia di riferimento in MTPA, nell'ipotesi di linearità magnetica.

Guardando l'ultimo grafico si prende atto che la massima coppia teoricamente erogabile dal motore in MTPA in linearità magnetica, risulta essere maggiore della massima coppia erogabile riportata nel data-sheet del motore (vedi Tab. 4.1).

4.2 MTPV in linearit  magnetica

Per il luogo dei punti in MTPV, sotto l'ipotesi di linearit  magnetica, si plotta (3.19) sul piano dei flussi magnetici in assi dq calcolando la caratteristica fino al valore di flusso massimo trovato in MTPA.

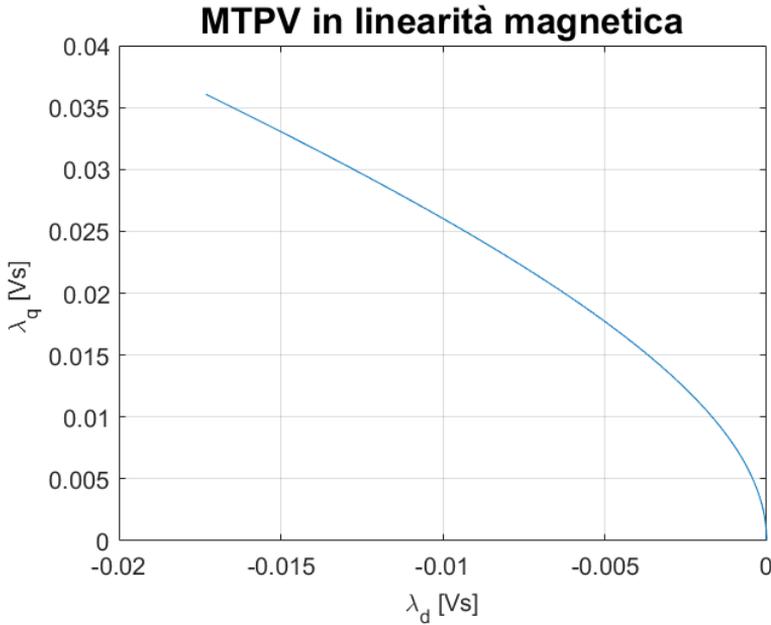


Figura 4.7: Caratteristica di MTPV in linearit  magnetica sul secondo quadrante del piano dei flussi. Si pu  notare come per un motore sincrono IPM: $90^\circ < \delta_{MTPV} < 135^\circ$.

Questo grafico sar  utile quando nella prima simulazione si osserver  il valore del flusso di macchina alla massima velocit  senza nessuna limitazione dell'angolo di carico, la quale invece   prevista nel DFVC. In questo modo, calcolando l'angolo di carico nel suddetto punto di lavoro, si pu  capire se il motore arriva a lavorare in MTPV, e se lo raggiunge, fissare δ_{MAX} per il DFVC.

Capitolo 5

Simulazione in Simulink

In questo capitolo si trova tutta l'implementazione in ambiente Simulink (una libreria di Matlab), al fine di modellizzare ogni componente principale dell'azionamento elettrico in studio. Principale scopo é quello di testare il codice del controllo DFVC scritto in C, ragion per cui per alcuni componenti dell'azionamento é implementata una modellistica piú semplificata rispetto a quella del secondo ambiente di simulazione (Plecs).

Il codice scritto in C si combina con la logica a blocchi di Simulink grazie ad una funzione chiamata *S-Function*, la quale permette di abbinare un blocco della simulazione a delle istruzioni scritte in C, C++ o Fortran. In questo lavoro di tesi il codice é stato scritto in C e come editor per la compilazione é stato utilizzato lo stesso Matlab. Prima di presentare la modellistica di ogni singolo blocco, si fa vedere una panoramica della simulazione omettendo nell'immagine i segnali di riferimento per il controllo e il condizionamento per la visualizzazione dei vari grafici.

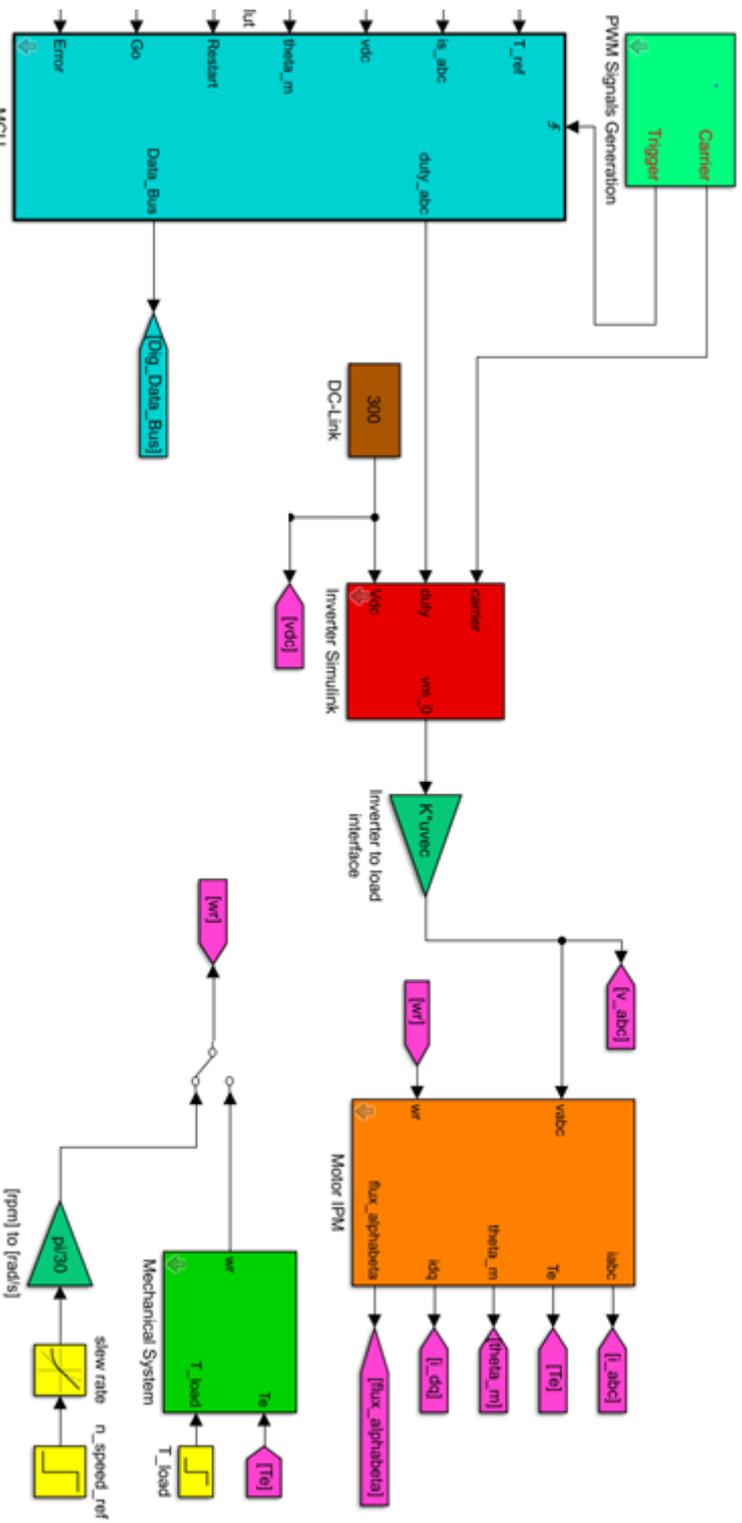


Figura 5.1: Una panoramica dei blocchi principali della simulazione implementata in ambiente Simulink. Da sinistra verso destra: il blocco (*CPU*) che contiene il codice in C della S-Function e il blocco della PWM (celeste e verde-acqua), il DC-link (marrone), l'inverter (rosso), il motore dal punto di vista elettromagnetico (arancione) e meccanico (verde).

5.1 Modello meccanico del motore elettrico

Nella parte riferita al modello meccanico, come si può anche osservare in Fig. 5.1 (blocco verde), si nota uno switch da settare manualmente. Il canale inferiore simula semplicemente il trascinamento del rotore indotto da un motore primo, quindi si forza un certo andamento della velocità meccanica. Il canale superiore, invece, implementa la canonica equazione di Newton della meccanica, omettendo forze resistive dovute all'attrito viscoso (proporzionali alla velocità) o attrito dovuto alla ventilazione (proporzionali al quadrato della velocità).

$$T_e - T_L = J_{eq} \frac{d\omega_R}{dt} \quad (5.1)$$

La modellistica a blocchi in Simulink risulta:

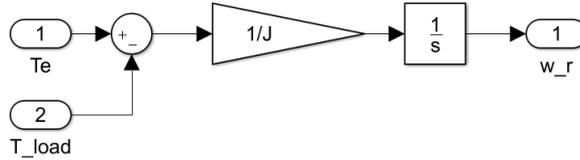


Figura 5.2: Modello Simulink del motore elettrico dal punto di vista meccanico.

5.2 Modello elettromagnetico del motore elettrico

Continuando nella presentazione si giunge subito al cuore dell'azionamento, ovvero il motore elettrico (in Fig. 5.1 corrisponde al blocco arancione). All'interno di questo blocco si trova una prima parte dove dalla velocità meccanica si ricavano l'angolo meccanico (ingresso dell'encoder), l'angolo elettrico e la velocità elettrica entrambi necessari nel modello del motore. La posizione iniziale del rotore risulta essere casuale.

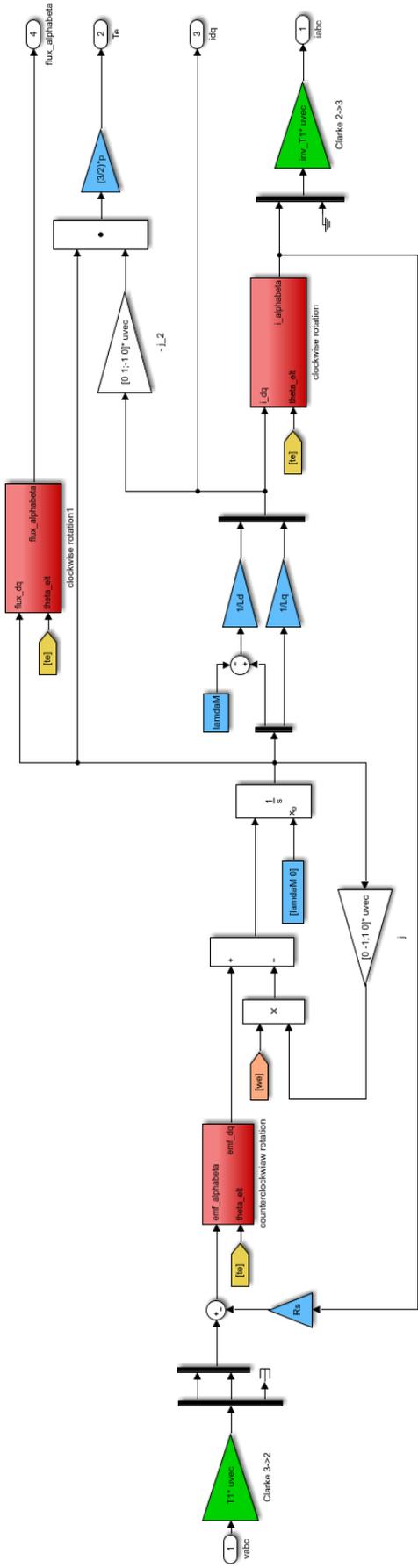


Figura 5.4: Modello Simulink del motore sincrono IPM dal punto di vista elettromagnetico. Si osservano le matrici per le trasformazioni di Clarke (verde), le matrici di rotazione oraria e antioraria (rosso), i vari parametri del modello lineare di macchina (celeste), l'angolo (giallo) e la velocità (rosa) elettrici precedentemente calcolati. In entrata le tensioni di fase. In uscita (in ordine) i flussi magnetici in assi statorici $\alpha\beta$, la coppia elettromagnetica, le correnti negli assi canonici di rotore dq e le correnti di fase.

5.3 Modello dell'encoder

Nel campo degli azionamenti *sensored* la misura della posizione angolare rotorica é fondamentale sia per il calcolo della velocità meccanica, sia per le trasformazioni all'interno della logica di controllo. L'impiego di un encoder é generalmente la soluzione piú frequente. Esistono due tipi di encoder: encoder assoluto e encoder incrementale. In questa simulazione Simulink é implementato solo il primo, per una semplice ragione di praticitá in quanto é piú immediato da modellizzare.

Un encoder assoluto é caratterizzato da N led e rispettivamente N photo transistor, nel mezzo si trova un disco rotante con N piste diverse. Gli N bit in uscita corrispondono univocamente a una delle 2^N decodifiche possibili dell'angolo meccanico da misurare. In questo modo appena acceso il sistema é in grado di dare la posizione assoluta.

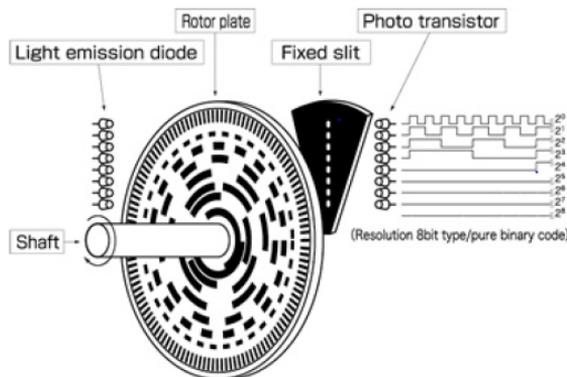


Figura 5.5: Struttura semplificata di un encoder assoluto.

In Simulink risulta facile modellizzare un encoder assoluto. Prima si scala l'angolo reale sulla base di quante divisioni (2^N) dispone l'encoder e dopo con una quantizzazione ad intervalli unitari si ottiene la posizione digitalizzata dell'angolo, con numeri interi da zero fino al numero massimo di divisioni.

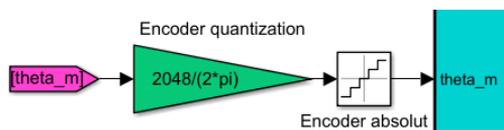


Figura 5.6: Modello Simulink di un encoder assoluto con 2048 divisioni. Da sinistra verso destra: l'angolo meccanico reale di rotore (viola), la scalatura in base al numero di divisioni (verde), la quantizzazione in intervalli unitari (bianco) e il l'angolo digitale entra in ingresso al blocco del codice di controllo (celeste).

5.4 Modello del generatore dei segnali PWM

Il generatore dei segnali PWM (in Fig. 5.1 il blocco verde-acqua) ha la funzione di generare la portante necessaria all'inverter per la modulazione e il trigger per sincronizzare l'esecuzione del codice con i vertici della portante. Questo è molto importante perché così facendo si campionano le correnti nei loro punti medi (considerando un ripple di PWM) e si inviano i comandi di riferimento all'inverter in istanti *lontani* dalle commutazioni.

La portante scelta è una triangola isoscele unipolare, quindi compresa tra zero e l'unità. Mentre è possibile scegliere tre diversi tipi di sincronizzazione per l'esecuzione del codice:

- Sincronizzazione con un singolo periodo di commutazione in corrispondenza dello zero della portante.
- Sincronizzazione con un singolo periodo di commutazione in corrispondenza dell'unità della portante.
- Sincronizzazione con un doppio periodo di commutazione, quindi in corrispondenza in maniera alternata tra lo zero e l'unità della portante. In questo lavoro di tesi si è deciso il secondo tipo di sincronizzazione e non ci saranno cambiamenti nel seguito.

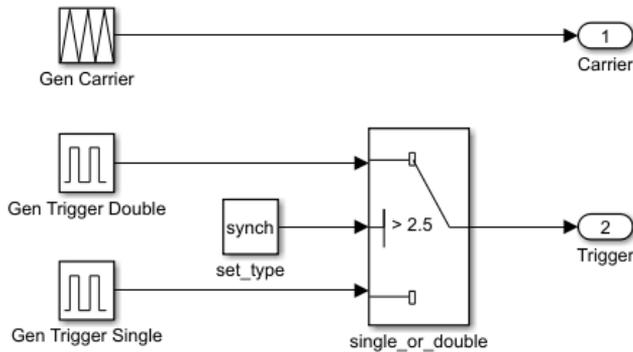


Figura 5.7: Modello Simulink del generatore dei segnali PWM. Nel canale superiore la generazione della portante. Nel canale inferiore la generazione del trigger per la sincronizzazione per l'esecuzione del codice di controllo. Lo switch che si vede differenzia una singola o una doppia sincronizzazione, mentre all'interno del secondo generatore si apprezza il tipo di sincronizzazione con un singolo periodo.

5.5 Modello dell'inverter

In questo ordine adottato, risalendo a monte dell'azionamento, si trova l'inverter (in Fig. 5.1 il blocco rosso).

La simulazione implementa la possibilità di generare delle tensioni PWM o al valor medio. La modulazione PWM é effettuata con un comparatore ad isteresi che vede in ingresso la differenza tra la modulante e la portante.

Nel mondo della conversione statica si preferisce descrivere il problema con gli *indici di modulazione* (m) invece che con i *duty-cycle* (d), perché i primi hanno una natura simmetrica e quindi sono più vicini al fenomeno fisico delle tensioni sinusoidali. La realtà fisica non cambia, ma cambia come la si descrive. Infatti in Simulink viene fatta la canonica trasformazione:

$$m = 2d - 1$$

Usando gli indici di modulazione l’inverter impone una tensione di fase riferita al suo punto medio virtuale del DC-link. Infine viene riscaldato tutto per il fattore $\frac{V_{DC-link}}{2}$; naturalmente avendo a disposizione una tensione massima $V_{DC-link}$, un campo di regolazione simmetrico ha l’estremo negativo e positivo pari alla metà della tensione sul DC-link.

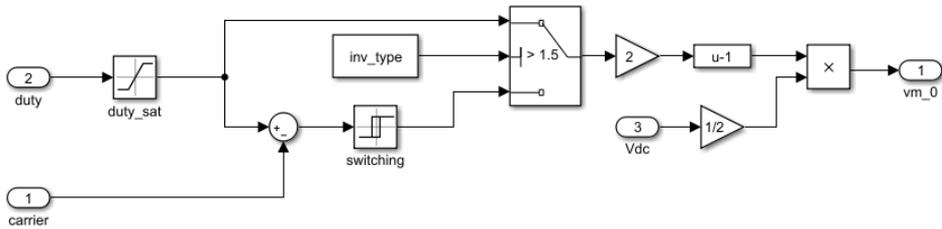


Figura 5.8: Modello Simulink dell’inverter. In ingresso i duty di riferimento generati dal controllo e la portante generata dal generatore di segnali PWM. In uscita le tre tensioni di fase riferite al punto medio virtuale del DC-link.

La modellistica adottata per l’inverter trascura cadute di tensione dovute alla resistenza equivalente, alla tensione di soglia e al dead-time. Questo perché, come anche spiegato per altri blocchi, in Simulink il fine principale è quello di testare il codice di controllo. Questa scelta si riflette su un controllo leggermente semplificato perché non deve tenere conto di compensazioni dovute agli errori di tensioni causate da un inverter reale (quindi non vengono implementate soluzioni proposte in [7]); questo aspetto è approfondito nel successivo ambiente di simulazione.

5.5.1 Interfacciamento dell’inverter con il carico

L’interfacciamento dell’inverter con il carico in ambiente Simulink è un aspetto che in Fig. 5.1 si può apprezzare nel guadagno (blocco verde) immediatamente a valle dell’inverter (blocco rosso). Come

spiegato prima l'inverter impone tre tensioni di fase riferite al punto medio virtuale del DC-link, ma il modello di macchina richiede tre tensioni di fase riferite al punto di neutro. In una logica a blocchi si viene a perdere l'informazione fondamentale della natura trifase del sistema elettrico e la **reiezione al modo comune** é l'equazione elettrotecnica necessaria da implementare per simulare correttamente la natura trifase del sistema elettrico che altrimenti si perderebbe. Si definiscono con le seguenti lettere i tre punti elettrici caratteristici dell'azionamento:

M Punto medio virtuale della tensione di DC-link

N Neutro del sistema elettrico trifase (per analizzare solo le componenti responsabili della conversione elettromeccanica)

f Punto sulla fase dove l'inverter impone la tensione all'uscita del convertitore stesso, generalmente riferita a *M*

Ipotesi di un sistema elettrico bilanciato e simmetrico.

$$\begin{aligned}v_{fM} &= v_{fN} + v_{NM} \\ \sum_{f=1}^3 v_{fM} &= \sum_{f=1}^3 v_{fN} + 3v_{NM} \\ v_{NM} &= \frac{1}{3} \sum_{f=1}^3 v_{fM}\end{aligned}$$

Trovata l'espressione della tensione di modo comune si é ricavata automaticamente la relazione tra le tensioni imposte dall'inverter e la tensioni di fase riferite al neutro:

$$\vec{v}_{fN} = \frac{1}{3} \begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix} \vec{v}_{fM} \quad (5.2)$$

5.6 Modello del DC-link e delle misure elettriche

La scala tempi della simulazione é volta ad analizzare transitori di tempo ad elevata dinamica del motore. A queste frequenze la tensione sul DC-link sarebbe vista dal motore come un generatore di tensione costante, in Simulink si traduce come un blocco di una costante fissa (in Fig. 5.1 il blocco marrone) in entrata all’inverter.

Il controllo prevede quattro misure elettriche: la tensione del DC-link e le tre correnti di fase. Su tutte e quattro é possibile impostare un offset casuale per rendere la misura piú reale. Per simulare una misura digitale si imposta il fondo scala di ogni grandezza e il numero di bit sulla quale deve essere quantizzata (usando il quantizzatore giá impiegato nel modello dell’encoder).

5.7 Algoritmo di controllo

Questo blocco Simulink (in Fig. 5.1 il blocco celeste) grazie alla funzione *S-Function* associa il codice di controllo al blocco in questione che si trova all’interno della simulazione.

La logica di controllo adotta un’astrazione di una macchina a stati, stati sequenziali tra loro dal punto di vista temporale. Essi possono essere stati dinamici o statici, se statici significa che sono in attesa di un comando esterno:

- *Reset*: stato di riposo della macchina. Statico.
- *Wake Up*: stato di inizializzazione del DC-link. Dinamico.
- *Drive Init*: stato in cui si calcolano gli errori di misura. Dinamico.
- *Bootstrap Load*: stato in cui viene simulato l’inizializzazione dei driver dell’inverter. Dinamico.

- *Stop Motor*: stato in cui l'azionamento é pronto per l'avviamento. Statico.
- *Go Motor*: stato in cui si controlla l'azionamento. Statico e dinamico allo stesso tempo.
- *Error*: stato di errore dove l'azionamento é pilotato in una condizione di sicurezza. Statico.

Questo blocco, il quale emula sostanzialmente la CPU di controllo, produce in uscita i tre duty di riferimento verso l'inverter. Il blocco risulta essere un sistema triggerato, per ragioni spiegate precedentemente in *Modello del generatore dei segnali PWM*.

Adesso si passa a presentare il codice in C del controllo DFVC implementato in questa simulazione, seguendo l'ordine con cui é stato scritto.

A monte della logica a stati di macchina, vi sono pezzi di codice fondamentali necessari al controllo: l'acquisizione delle misure e dei parametri con il rispettivo condizionamento, le trasformate, il calcolo della velocità, le protezioni dell'azionamento. Si ricorda inoltre come all'interno del codice si trovano le righe necessarie per configurare il blocco Simulink in *S-Function* con Matlab, ma questa parte non é descritta in questo lavoro di tesi.

Acquisizione dei parametri e degli ingressi

La prima fase preliminare é naturalmente gestire l'acquisizione degli ingressi al blocco del codice e dei vari parametri passati via maschera. In questa sezione si ometterá di riportare righe di codice e si trova semplicemente un elenco delle variabili iniziali del codice.

In ingresso al blocco del codice si trova (in ordine): l'andamento di coppia di riferimento, le misure delle tre correnti di fase e della tensione sul DC-link, la misura dell'angolo meccanico proveniente dall'encoder e i tre comandi impiegati negli stati statici del controllo. I parametri passati via maschera sono i valori inerenti alla taratura

dei due principali anelli di controllo, il tempo di dead-time, il fondo scala e il numero bit rispettivamente per le correnti e la tensione del DC-link, per simulare una misura digitale.

Condizionamento delle misure

Le misure delle quattro grandezze (correnti e tensione) sono simulate con errori di offset. Gli stessi sono calcolati in uno degli stati di macchina nella fase iniziale di inizializzazione. Si possono calcolare solo offset di grandezze periodiche, quindi l'errore sulla tensione del DC-link, non essendo rimovibile, può influenzare anche significativamente le prestazioni del controllo.

```
// Rimozione degli errori di offset sui primi tre canali di misura
// dedicati alle tre correnti di fase
iabc.a = input.ch0 - offset_in.ch0;
iabc.b = input.ch1 - offset_in.ch1;
iabc.c = input.ch2 - offset_in.ch2;
```

Trasformazioni di Clarke

Anche in questa fase si è deciso di omettere righe di codice in quanto si eseguono semplicemente le canoniche trasformazioni di Clarke delle tre correnti e delle tre tensioni di fase. Le tre tensioni di fase sono ricostruite dai tre duty-cycle generati dal controllo al passo precedente. Ricordando l'idealità dell'inverter nella simulazione Simulink, la ricostruzione non deve tenere conto di eventuali compensazioni dovute all'errore di tensione causato da un inverter reale.

```

// Funzione che ricostruisce le tre tensioni di fase a partire da i
// 3 duty-cycle generati al passo precedente.
// Si passano le 3 tensioni, i 3 duty e la misura di tensione
// al DC-link.

void real_Vref (Xabc *v, Xabc d, float VDC) {

    float tmp1;
    tmp1 = one_over_three * VDC;

    v->a = tmp1 * (2.0 * d.a - d.b - d.c);
    v->b = tmp1 * (2.0 * d.b - d.c - d.a);
    v->c = tmp1 * (2.0 * d.c - d.a - d.b);
}

```

Calcolo dell'angolo meccanico (e elettrica) e della velocità meccanica (e elettrica)

Nel terzo capitolo sono state spiegate le motivazioni che portano a calcolare la velocità con un PLL. Si pone solamente attenzione che l'angolo in ingresso al controllo è un angolo digitalizzato, quindi è necessaria una conversione in radianti del medesimo. Il fattore di conversione esegue l'operazione inversa presente nel modello dell'encoder ed è una costante del controllo in relazione al numero di divisioni dell'encoder stesso. Di seguito si riporta la parte di codice sull'argomento:

```

// Conversione in radianti dell'angolo digitale in entrata
// al controllo.
// <ENCFACOR> e' una costante del controllo
theta_m = theta_enc * ENCFACOR;

// Normalizzazione dell'angolo tra 180 e -180 gradi.
// <p> (paia poli del motore) e' una costante del controllo
theta_m = angle_normalization(theta_m);
theta_e = p*theta_m;
theta_e = angle_normalization(theta_e);

```

```
SinCos_thetaE.cos = cos(theta_e);
SinCos_thetaE.sin = sin(theta_e);
SinCos_thetaM.cos = cos(theta_m);
SinCos_thetaM.sin = sin(theta_m);

// Calcolo della velocita' con il PLL
// In ingresso al PI si deve avere:
// sin(<referimento> - <feedback>)
speed_var.ref = SinCos_thetaM.sin * cos(thetaPLL);
speed_var.fbk = SinCos_thetaM.cos * sin(thetaPLL);

PI_control (&speed_var, &speed_par);

// Si prende la parte integrativa per ottenere una
// velocita' piu' pulita (come se fosse filtrata).
speed_m = speed_var.intg;

speed_e = p * speed_m;

// Integrazione con Eulero
thetaPLL += Ts*speed_var.out;

thetaPLL = angle_normalization(thetaPLL);
```

Protezione dell'azionamento

Le ultime righe, prima dei vari stati di macchina, sono la protezione dell'azionamento dove sono eseguiti vari check di verifica, confrontando correnti, tensione del DC-link e velocità del motore con i rispettivi valori di riferimento, definiti come costanti nel codice. Appena si registra un superamento della soglia, per qualsiasi grandezza, si pone l'azionamento in sicurezza forzando lo stato di macchina in *Error* e si segnala l'errore con un apposito *flag*.

```
// Protezione di corrente sulla fase A
    if (fabs(iabc.a) > CRT_PROT) {
// Reset Duty-Cycles
        duty.a = 0.5;
        duty.b = 0.5;
        duty.c = 0.5;

        StateDrive = ERROR;
        protection_flag = 1.0;
    }

// Protezione di corrente sulla fase B
    if (fabs(iabc.b) > CRT_PROT) {
// Reset Duty-Cycles
        duty.a = 0.5;
        duty.b = 0.5;
        duty.c = 0.5;

        StateDrive = ERROR;
        protection_flag = 2.0;
    }

// Protezione di corrente sulla fase C
    if (fabs(iabc.c) > CRT_PROT) {
// Reset Duty-Cycles
        duty.a = 0.5;
        duty.b = 0.5;
        duty.c = 0.5;

        StateDrive = ERROR;
        protection_flag = 3.0;
    }

// Protezione di tensione sul DC-link
    if (vdc > VOLT_PROT) {
        StateDrive = ERROR;
        protection_flag = 4.0;
    }

// Protezione di velocita' del rotore
```

```
    if (speed_m > SPEED_PROT) {  
// Reset Duty-Cycles  
    duty.a = 0.5;  
    duty.b = 0.5;  
    duty.c = 0.5;  
  
    StateDrive = ERROR;  
    protection_flag = 5.0;  
    }  
}
```

5.7.1 Gli stati di macchina

Dopo i pezzi di codice descritti precedentemente si arriva alla logica dei vari stati di macchina già elencati nell'introduzione di questa sezione. La presentazione dei vari stati segue l'ordine temporale con cui si susseguono nel controllo, naturalmente ad eccezione dello stato in *Error* (descritto per ultimo) che si potrebbe verificare in qualunque fase della simulazione (Fig. 5.9).

Lo stato di macchina è una variabile *StateDrive* che assume valori diversi a seconda dello stato di macchina presente all'esecuzione del codice.

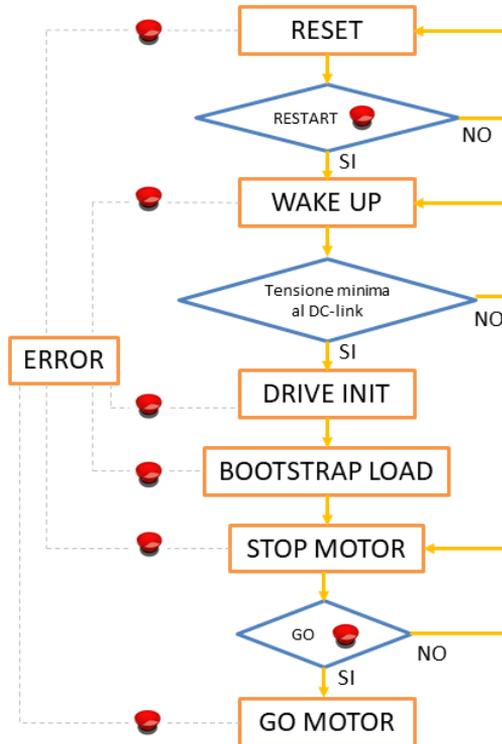


Figura 5.9: Flow-chart della logica a stati di macchina implementata nel codice di controllo. I pulsanti rossi simboleggiano eventuali comandi in arrivo dall'operatore.

Reset

Lo stato di *Reset* é uno stato statico e che pone l'azionamento in una condizione iniziale di riposo.

```

case RESET:

// Reset Duty-Cycles
duty.a = 0.5;
duty.b = 0.5;
duty.c = 0.5;

```

```

// Reset generico contatore
counter = 0;

// Va in WAKE_UP se arriva il comando dall'esterno
if (Restart > 0.5)
    StateDrive = WAKE_UP;
else
    StateDrive = RESET;

// Va in ERROR se arriva il comanda dall'esterno
if (Error > 0.5)
    StateDrive = ERROR;

// PWM OFF
pwm_enable = 0.0;

break;

```

Wake Up

In questo stato é presente un check se la tensione sul DC-link ha raggiunto o meno una tensione minima di soglia. E' uno stato dinamico e non necessita particolari attenzioni dal momento che in questa simulazione non é stata simulata la pre-carica del banco di condensatori del DC-link.

```

case WAKE_UP:

// Reset delle principali variabili di controllo
ResetVars();

// PWM OFF
pwm_enable = 0.0;

// Check sulla tensione del DC-link misurata in ingresso
if (vdc > VDC_WAKE_UP) {
    StateDrive = DRIVE_INIT;
}

```

```

        protection_flag = 0.0;
    }
else {
    StateDrive = WAKE_UP;
    protection_flag = 6.0;
}

// Ritorna in ERROR se arriva il comando dall'esterno
if (Error > 0.5)
    StateDrive = ERROR;

break;

```

Drive Init

Segue sempre uno stato dinamico di macchina in cui sono calcolati i possibili offset sulle misure in ingresso delle tre correnti di fase.

```

case DRIVE_INIT:

// Il numero di periodi di campionamento per calcolare gli
// offset e' una scelta a discrezione dell'utente.
if (counter <= 127) {
    // Acquisizione dai tre canali
    offset.ch0 += input.ch0;
    offset.ch1 += input.ch1;
    offset.ch2 += input.ch2;
}
else {
if (counter == 128) {
    offset_in.ch0 = offset.ch0 / 128.0;
    offset_in.ch1 = offset.ch1 / 128.0;
    offset_in.ch2 = offset.ch2 / 128.0;

    counter = -1;
    // Va in Bootstrap
    StateDrive = BOOTSTRAP_LOAD;
}
}
}

```

```
// PWM OFF
pwm_enable = 0.0;

counter++;

// Ritorna in ERROR se arriva il comando dall'esterno
if (Error > 0.5)
    StateDrive = ERROR;

break;
```

Bootstrap Load

Si continua a giungere ad uno stato di macchina dinamico dove viene accesa la PWM e si simula il caricamento dei condensatori dei driver del convertitore, i quali nella realtà comandano gli switches dell'inverter.

```
case BOOTSTRAP_LOAD:

// PWM ON
pwm_enable = 1.0;

// Duty-Cycles per il Bootstrap
duty.a = 0.05;
duty.b = 0.05;
duty.c = 0.05;

counter++;

// Uscita dal BOOTSTRAP_LOAD
if (counter == 256) {
    StateDrive = STOP_MOTOR;
    counter = 0;
}

break;
```

Stop Motor

Finalmente é tutto pronto per poter iniziare a controllare l'azionamento. Lo stato di *Stop Motor* é uno stato statico dove si attende il comando esterno per poter imporre al motore l'andamento della coppia di riferimento.

```
case STOP_MOTOR:

// PWM ON
pwm_enable = 1.0;

// Inizializzazioni di TUTTE le variabili di controllo
InitVarsCtrl();

// Inizia l'algoritmo di controllo se arriva il comando
// dall'esterno
if (Go > 0.5) {
    StateDrive = GO_MOTOR;
    protection_flag = 0.0;
    pwm_enable = 1.0;
}

// Ritorna in ERROR se arriva il comando dall'esterno
if (Error > 0.5)
    StateDrive = ERROR;

break;
```

Go Motor

Il tempo trascorso nelle precedenti fasi é stato piú che sufficiente nel far andare a regime il PLL per il calcolo della velocità. Ora però bisogna fare andare a regime anche il regolatore dell'osservatore di flusso prima di poter controllare il motore. Questa prima fase si chiama *Flux Build* ed ha una durata in relazione alla costante di tempo della dinamica imposta all'osservatore.

```

case GO_MOTOR:

// FLUX BUILD

// La coppia di riferimento in ingresso e' forzata a zero per far
// agganciare l'osservatore di flusso
if (counter<2000 && flag_flux==0) {
    T_ref = 0.0;
    counter++;

    if (counter==1){ // si inizializza l'osservatore
        // <FM> e' il flusso dei magneti costante nel controllo
        fluxobs_old.alpha = SinCos_thetaE.cos * FM;
        fluxobs_old.beta = SinCos_thetaE.sin * FM;
        fluxobs.alpha = SinCos_thetaE.cos * FM;
        fluxobs.beta = SinCos_thetaE.sin * FM;
    }
}
if (counter==2000) {
    counter = 0.0; // reset counter
    flag_flux=1;
}

```

L'osservatore di flusso (Fig. 3.6) é sicuramente uno degli aspetti piú nevralgici del controllo. La teoria in merito é stata esposta nei capitoli precedenti, tuttavia si ritiene doveroso approfondire quale metodo di integrazione scegliere da scrivere in C, per l'integrale del modello di macchina in assi stazionari, il canale superiore in Fig. 3.6.

In digitale il passo di integrazione coinciderá con il passo di campionamento delle misure richieste dall'azionamento. Generalmente si opta per l'integrazione secondo Eulero oppure secondo Tustin. In breve si ricorda:

$$\text{Eulero : } y(n) = y(n-1) + T_s x(n-1)$$

$$\text{Tustin : } y(n) = y(n-1) + \frac{T_s}{2} [x(n) + x(n-1)]$$

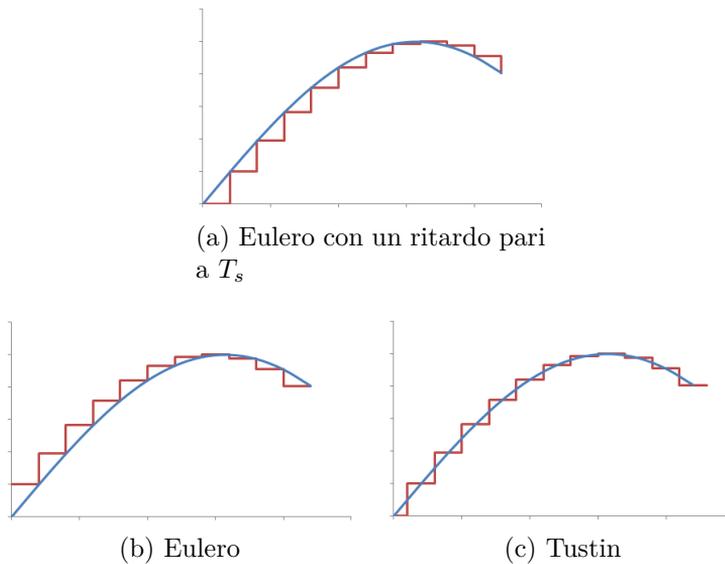


Figura 5.10: Confronto tra tre diversi tipi di integrazione digitale (rosso) rispetto ad una stessa integrazione nel mondo del continuo, non discreto (blu).

Apparentemente la scelta tra i due metodi (Fig. 5.10(b)-(c)) sembrerebbe indifferente. Questo é corretto per azionamenti che non richiedono prestazioni ad alta dinamica ad alte velocità. Infatti in questo caso, la sinusoide di flusso osservata inizia ad essere calcolata con un numero veramente esiguo di punti, situazione che porterebbe ad ogni istante di campionamento T_s un errore sul flusso particolarmente significativo, causando una perdita di controllo del motore. Una soluzione a questo problema (Fig. 5.10(a)) é quella di ritardare l'integrazione euleriana, al fine di avere un errore nullo tra la il flusso discreto osservato e il flusso di macchina reale.

```
// OSSERVATORE DI FLUSSO

// Flusso stimato in assi rotanti con il modello magnetico

// Trasformazioni delle correnti nei canonici assi rotorici
alphabetadq (SinCos_thetaE, &iabFbk, &idqMotor);
```

```

// Modello magnetico lineare di macchina
fluxdq_est.d = Ld * idqMotor.d + FM;
fluxdq_est.q = Lq * idqMotor.q;
// Trasformazione dei flussi in assi statorici
dq_alphabeta (SinCos_thetaE, &fluxdq_est,
              &fluxalphabeta_est);

// Flusso stimato in assi statorici

// Errore del flusso osservato
err_fluxobs.alpha = fluxobs.alpha - fluxalphabeta_est.alpha;
err_fluxobs.beta = fluxobs.beta - fluxalphabeta_est.beta;

// Integrazione con Eulero ritardata di una passo
// di esecuzione.
// <G> e' la pulsazione di transizione tra i due stimatori
// ed e' una costante del controllo.

// asse alpha
emf.alpha = vabRec.alpha - Rs*iabFbk.alpha -
            G*err_fluxobs.alpha;
fluxobs_old.alpha = fluxobs.alpha;
fluxobs.alpha += Ts*emf.alpha;
// asse beta
emf.beta = vabRec.beta - Rs*iabFbk.beta -
            G*err_fluxobs.beta;
fluxobs_old.beta = fluxobs.beta;
fluxobs.beta += Ts*emf.beta;

// Calcolo del vettore di flusso osservato
fluxobs_amp = sqrt(fluxobs_old.alpha*fluxobs_old.alpha
                  + fluxobs_old.beta*fluxobs_old.beta);
if (fluxobs_amp>0.0001) { // evitare divisioni per 0
    fluxobsAngle.sin = fluxobs_old.beta / fluxobs_amp;
    fluxobsAngle.cos = fluxobs_old.alpha / fluxobs_amp;
}
else {
    fluxobsAngle.sin = 0.0;
    fluxobsAngle.cos = 1.0;
}

```

Riguardando Fig. 3.2 si vede come in uscita dall'osservatore si ottiene l'angolo per il DFVC e il l'ampiezza del flusso da retroazionare in uno degli anelli principali del controllo. Infatti ora segue proprio il cuore del controllo con i due anelli, di flusso e di corrente. In questa sezione é incluso anche l'anello di limitazione di carico in MTPV, perché se il motore dovesse arrivare a lavorare in MTPV, esso limiterá ulteriormente il riferimento di corrente (per limitare l'angolo di carico). Si specifica, infine, che la caratteristica di MTPA in linearitá magnetica (Fig. 4.6) é stata caricata sul codice discretizzandola su 32 punti.

```
// GENERAZIONE DELLE 3 TENSIONI DI RIFERIMENTO DI FASE

// Si inizializzano le coordinate angolari necessarie per il DFVC,
// calcolate precedentemente nell'osservatore
SinCos_thetaFs.cos = fluxobsAngle.cos;
SinCos_thetaFs.sin = fluxobsAngle.sin;

// Trasformazione delle correnti negli assi rotanti del DFVC
alpha_beta_dq (SinCos_thetaFs, &iabFbk, &idqFbk);

// Dalle mappe di MTPA si genera il flusso di riferimento
// dalla coppia di riferimento in entrata
flux_ref = LUT_1d(&fluxS_tab[0], fabs(T_ref),
                 Tmax, Tmin, DT, INV_DT);

// Deflussaggio
tmp1 = speed_e/(fabs(speed_e)+0.001); //sign(velocita')
// Il coeff. 0,9 utile per evitare la saturazione
// dei duty-cycles
flux_weak = 0.9 * (vdc*sqrt3_inv -
                  Rs*idqFbk.q*tmp1)/(fabs(speed_e) + 0.001);
// per evitare divisioni per 0

// Saturazione del flusso di riferimento sul flusso
// massimo in MTPA <FluxMax> e sul flusso calcolato
// nella legge di deflussaggio <flux_weak>
if( flux_weak > FluxMax)
    flux_weak = FluxMax;
```

```

    if( flux_ref > flux_weak)
        flux_ref = flux_weak;

// Anello di controllo del flusso statorico in asse f
PI_FluxS_var.ref = flux_ref;
PI_FluxS_var.fbk = fluxobs_amp;
PI_FluxS_par.limUP = 1.5*Rs*Imax;//limitazione tensione
PI_FluxS_par.limDOWN = -PI_FluxS_par.limUP;
PI_FluxS_var.ffw = Rs*idqFbk.d; // feed-forward

PI_control(&PI_FluxS_var, &PI_FluxS_par);
vdqRef.d = PI_FluxS_var.out;

// Anello di controllo dell'angolo di carico in MTPV
// sin(angolo di carico)
tmp1 = fluxobsAngle.sin*SinCos_thetaE.cos -
        fluxobsAngle.cos*SinCos_thetaE.sin;
// cos(angolo di carico)
tmp2 = fluxobsAngle.cos*SinCos_thetaE.cos +
        fluxobsAngle.sin*SinCos_thetaE.sin;
// Angolo di carico osservato
PI_angleloadVar.ref = fabs(atan(tmp1/tmp2));

// <angleloadMAX> e' il massimo angolo di carico
// ammesso dal controllo per il motore ed e' una
// costante del codice
PI_angleloadVar.fbk = angleloadMAX;

// L'anello deve saturare la corrente solo se l'angolo
// di carico raggiunge il valore massimo consentito
PI_angleloadPar.limUP = sqrt(Imax*Imax -
                             idqFbk.d*idqFbk.d);
PI_angleloadPar.limDOWN = 0.0;

PI_control(&PI_angleloadVar, &PI_angleloadPar);

// Saturazione della corrente di riferimento
iq_ref_lim = sqrt(Imax*Imax - idqFbk.d*idqFbk.d)
              - PI_angleloadVar.out;

// Saturazione della coppia di riferimento

```

```

T_lim = 1.5*p*flux_ref*iq_ref_lim;
if (T_lim >= Tmax) {
    T_lim = Tmax;
}

// La doppia saturazione se l'azionamento lavora con
// coppie negative
saturation2side(&T_ref, T_lim);

// Anello di controllo della corrente in asse tau
PI_CurrentVar_Q.ref = T_ref/(1.5*p*flux_ref);
saturation2side(&PI_CurrentVar_Q.ref, iq_ref_lim);
PI_CurrentVar_Q.fbk = idqFbk.q;
PI_CurrentPar_Q.limUP = sqrt(vdc*vdc*one_over_three -
                             vdqRef.d*vdqRef.d);
PI_CurrentPar_Q.limDOWN = -PI_CurrentPar_Q.limUP;
//feed-forward
PI_CurrentVar_Q.ffw = speed_e*fluxobs_amp +
                    Rs*idqFbk.q;

PI_control(&PI_CurrentVar_Q, &PI_CurrentPar_Q);
vdqRef.q = PI_CurrentVar_Q.out;

// Phase Advancing
// angolo per il Phase Advancing
phase_adv = 1.5*speed_e*Ts;
vdqRef_adv.d = vdqRef.d*cos(phase_adv) -
              vdqRef.q*sin(phase_adv);
vdqRef_adv.q = vdqRef.d*sin(phase_adv) +
              vdqRef.q*cos(phase_adv);

// Trasformazioni delle tensioni di riferimento dagli assi rotanti
// del DFVC al sistema trifase
dq_alphabeta (SinCos_thetaFs, &vdqRef_adv, &vabRef);
InverseClarke00 (&vabRef, &vabcRef);

```

Il controllo si conclude con la generazione dei tre duty-cycles di riferimento in uscita del blocco in *S-Function* a comandare l'inverter. La tecnica di modulazione adottata é la BEM (*Balancement Envelope Modulation*), al fine di massimizzare la tensione di fase quando si é prossimi alla saturazione dei duty-cycles.

```

//GENERAZIONE DEI 3 DUTY-CYCLES DI RIFERIMENTO

// Generazione dei tre Duty-Cycles
duty.a = 0.5 + (vabcRef.a / vdc);
duty.b = 0.5 + (vabcRef.b / vdc);
duty.c = 0.5 + (vabcRef.c / vdc);

// Tecnica di modulazione BEM
BEM (&duty);

// Saturazione dei Duty-Cycles tra 0 e 1
saturation (&duty);

// Ritorna in ERROR se arriva il comando dall'esterno
if (Error > 0.5)
    StateDrive = ERROR;

break;

```

Si ricorda brevemente la tecnica di modulazione BEM riportando direttamente le righe di codice dell'omonima funzione:

```

// Funzione impiegata per la BEM alla funzione si passano i tre
// duty-cycles generati

void BEM (Xabc *Abc) {
    float max; // duty max
    float min; // duty min

    // Si trova il duty-cycle massimo tra i 3
    if (Abc->a >= Abc->b && Abc->a >= Abc->c)
        max = Abc->a;
    if (Abc->b >= Abc->a && Abc->b >= Abc->c)
        max = Abc->b;
    if (Abc->c >= Abc->a && Abc->c >= Abc->b)
        max = Abc->c;
    // Si trova il duty-cycle minimo tra i 3
    if (Abc->a <= Abc->b && Abc->a <= Abc->c)
        min = Abc->a;

```

```

if (Abc->b <= Abc->a && Abc->b <= Abc->c)
    min = Abc->b;
if (Abc->c <= Abc->a && Abc->c <= Abc->b)
    min = Abc->c;

// Balancement Envelope Modulation
Abc->a = Abc->a + 0.5 -(max+min)*0.5;
Abc->b = Abc->b + 0.5 -(max+min)*0.5;
Abc->c = Abc->c + 0.5 -(max+min)*0.5;
}

```

Nel riportare il codice si sono omesse l'esplicitazione di tutte le funzioni, la dichiarazione delle variabili, l'inizializzazione delle stesse e le costanti impiegate, in quanto appesantirebbero solamente la trattazione non portando nulla di utile nel comprendere meglio questo lavoro di tesi. Si ritiene utile, invece, riportare solamente la funzione del regolatore PI come chiusura dello stato di macchina *Go Motor*.

```

// Funzione impiegata per il regolatore PI

// Sono passate due strutture: variabili e parametri.
// In variabili: il riferimento, il feedback, la parte
// proporzionale, la parte integrativa, il feed-forward e l'uscita
// generata dal PI.
// In parametri: il coefficiente proporzionale, il coefficiente
// integrativo, il limite inferiore e superiore per la saturazione.

void PI_control (Xpi_var *var, Xpi_par *par) {

    var->err = var->ref - var->fbk;

    // Parte proporzionale (con feed-forward)
    var->prop = var->ffw + par->kp * var->err;

    //Saturazione parte proporzionale
    if (var->prop > par->limUP)
        var->prop = par->limUP;
    if (var->prop < par->limDOWN)
        var->prop = par->limDOWN;
}

```

```

// Parte integrativa
// <ki> = Ts * <coeff_intg>
var->intg += par->ki*var->err;

// Anti Wind-Up simmetrico
if ((var->intg + var->prop) > par->limUP)
    var->intg = par->limUP - var->prop;
if ((var->intg + var->prop) < par->limDOWN)
    var->intg = par->limDOWN - var->prop;

// Output del PI
var->out = var->prop + var->intg;
}

```

Error

Come conclusione alla presentazione dei vari stati di macchina si trova lo stato di *Error*. Stato statico in cui ogni stato precedente può porre l'azionamento in una condizione di sicurezza, in seguito a valori anomali delle grandezze o a comandi esterni del controllore.

```

case ERROR:

// Reset Duty-Cycles
duty.a = 0.5;
duty.b = 0.5;
duty.c = 0.5;

// Reset generico contatore
counter = 0;

// PWM OFF
pwm_enable = 0.0;

// Va in WAKE_UP se arriva un comando dall'esterno
if (Restart > 0.5)
    StateDrive = WAKE_UP;

```

```
else
    StateDrive = ERROR;
break;
```

5.8 Risultati simulazione Simulink

In questa sezione conclusiva si riportano i risultati finali della simulazione in Simulink appena presentata. Come risultati finali si intende imporre una coppia di riferimento al motore e analizzare la risposta del controllo sulla base di varie grandezze osservate.

Prima di riportare i seguenti risultati, si introducono le varie prove con una tabella riassuntiva dove si esplicitano numericamente i guadagni dei vari anelli di controllo (dettagliatamente descritta nel terzo capitolo) o dove si completa la descrizione di alcuni componenti dell'azionamento. L'azionamento nei suoi dati tecnici, riferiti al convertitore e al motore, é stato già presentato nel precedente capitolo nella Tab. 4.1.

Per alcuni parametri si rimanda (scritta a lato) all'equazione di riferimento.

Il controllo é sincronizzato con l' "1" della portante generata dal generatore dei segnali PWM.

Encoder assoluto	N° divisioni	2048	-
Inverter	f_{sw}	20	kHz
	T_s	0,05	ns
	Dead-time	0,00	μs
Regolatore PI asse f	Banda	1	kHz
	k_{Pf} (3.11)	6283	s^{-1}
	k_{If} (3.11)	$3,948 \times 10^3$	s^{-2}
Regolatore PI asse τ	Banda	1	kHz
	$k_{P\tau}$ (3.13)	1,131	V/A
	$k_{I\tau}$ (3.13)	$710,6 \times 10^3$	V/As
Regolatore PI angolo di carico	Banda	100	Hz
	δ_{MAX}	100	$^\circ$
	$k_{P\delta}$ (3.20)	6,111	A
	$k_{I\delta}$ (3.20)	384,0	A/s
Regolatore PI PLL	Banda	30	Hz
	Margine di fase	60	$^\circ$
	k_{PPLL} (3.7)	130,6	rad/s
	k_{IPLL} (3.7)	$3,948 \times 10^3$	rad/s ²
Osservatore di flusso	g (3.8)	90	rad/s
Deflussaggio	α (3.17)	0,95	-
Protezioni	I_{max} per l'azionamento	350	A
	$I_{ref.max}$ per il controllo	250	A
	V_{DCmax}	700	V
	V_{DCmin} (stato WAKE-UP)	220	V
	ω_{Rmax}	22000	rpm

Tabella 5.1: In tabella si riportano le tarature degli anelli di controllo e si completano le ultime caratteristiche dell'azionamento in esame.

Prima di presentare la prova eseguita in Simulink si pone l'attenzione su alcuni aspetti principali, utili a chiarire i dati riportati in Tab. 5.1:

- Le protezioni di corrente sul controllo implementato non soddisfano pienamente i requisiti di corrente massima del motore, riportati in Tab. 4.1. Questo perché caricando le mappe in linearità magnetica si arriva a chiedere una coppia maggiore di quella realmente erogabile, perciò solo ai fini della simulazione si ammetteranno correnti più elevate.
- I valori dei guadagni per l'anello di carico necessitano la conoscenza del flusso minimo che il motore raggiunge alla massima velocità. Questo dato non è stato ancora dichiarato e si ricaverà solo nell'osservazione delle prossime simulazioni. Tuttavia si anticipa già da adesso come questo motore non arrivi mai a lavorare in MTPV, quindi l'anello di controllo per l'angolo di carico non entrerà mai in funzione.
- Il parametro g non ha una taratura univoca, pertanto è frutto dell'esperienza o di alcune prove a tentativo.

5.8.1 Test in Simulink

In questo lavoro di tesi il seguente test, il quale si andrà ad illustrare a breve, sarà la prova principale ad elevatissima dinamica per validare il controllo DFVC.

Visto l'impiego di questo azionamento in esame in campo automotive, ci si è spinti nel testare il controllo con l'obiettivo di raggiungere prestazioni ad elevatissima dinamica, sicuramente anche oltre la sua applicazione che avrà realmente. A questo fine il test prevede di trascinare il rotore (imponendo una velocità) e di dare un riferimento a gradino di coppia come viene mostrato nel seguente grafico:

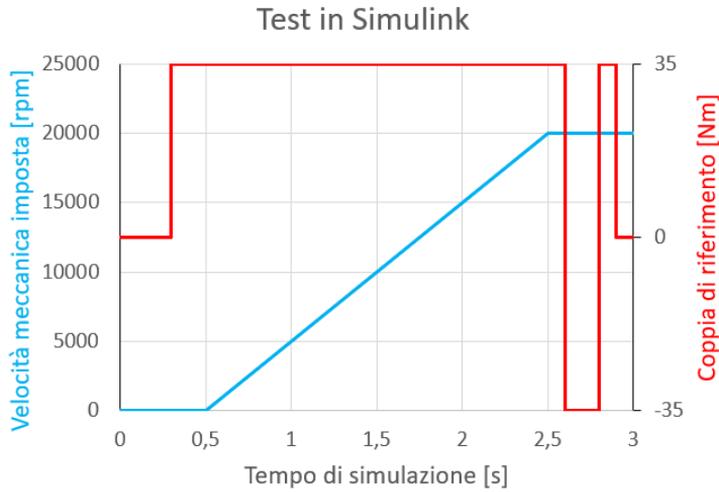


Figura 5.11: Andamenti della velocità meccanica con cui si trascina forzatamente il rotore e della coppia di riferimento nel test principale in Simulink.

La criticità di questo test sta nel fatto sicuramente di un'inversione a gradino della coppia a velocità massime, impensabile per un motore reale. La rampa di velocità impone un'accelerazione di 10000 rpm/s , anch'esso un valore molto alto per la realtà. Per apprezzare come il controllo si comporti per zone di deflussaggio spinto e quindi apprezzare i punti di forza del DFVC, si è imposto:

$$V_{DC} = 300 \text{ V}$$

Per questa prima fase di simulazione, avendo in Simulink il solo fine principale di testare il controllo, si opta per l'inverter un modello ai valori medi, quindi privo della modulazione PWM. Ora finalmente possono essere presentati tutti i risultati ottenuti del test effettuato in Simulink.

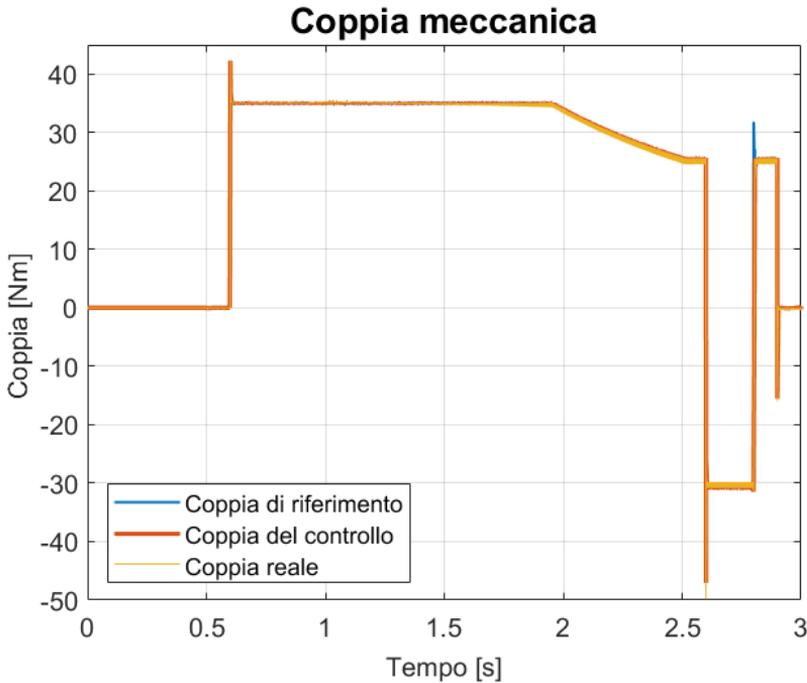


Figura 5.12: In questo plot si é presa la coppia di riferimento impostata dall'operatore all'esterno del codice e NON quella saturata all'interno del codice.

In Fig. 5.12 si nota come la coppia di riferimento sia ritardata rispetto all'andamento imposto dal controllore (Fig. 5.11), questo perché la fase di *Flux Build* nel codice di controllo forza a zero il riferimento per inizializzare l'osservatore di flusso. Inoltre si possono osservare due aspetti: primo, ad un certo punto la coppia nel controllo viene saturata a causa dell'effetto del deflussaggio; secondo, l'inversione di coppia non é simmetrica perché cambiando il verso della corrente, questo aiuta ad ottenere un margine maggiore di tensione, e quindi un margine maggiore in termini di flusso in deflussaggio.

Adesso le due grandezze principali controllate nei rispettivi assi $f\tau$:

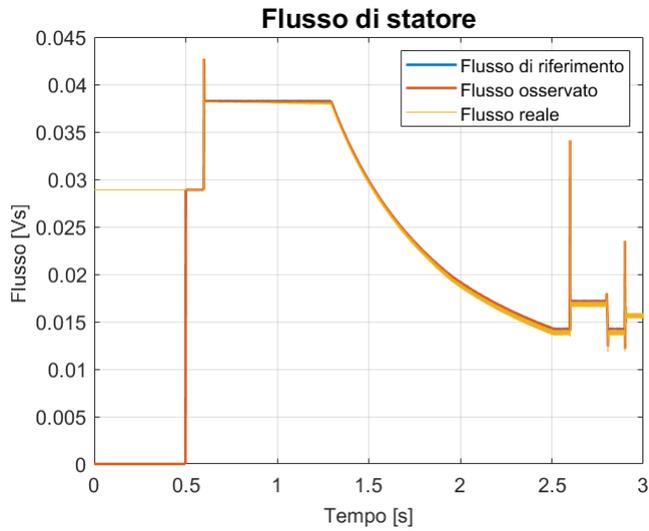


Figura 5.13: Flusso di statore controllato in asse f .

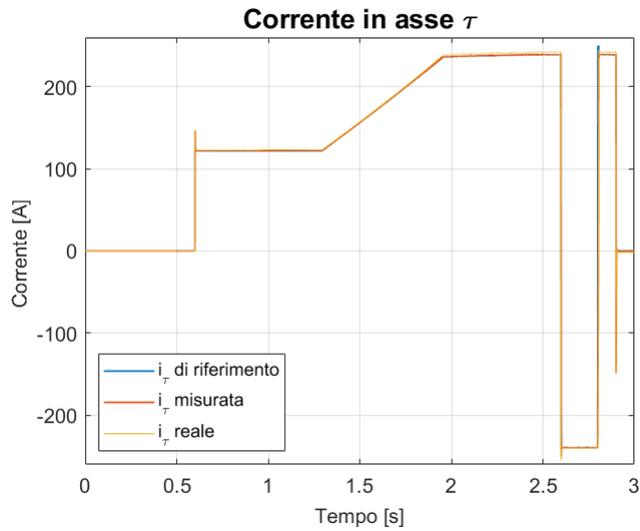


Figura 5.14: Corrente controllata in asse τ .

E' interessante vedere (Fig. 5.15) come l'inversione di coppia produca un'inversione di potenza non altrettanto simmetrica, a causa della caduta di tensione sulla resistenza di fase:

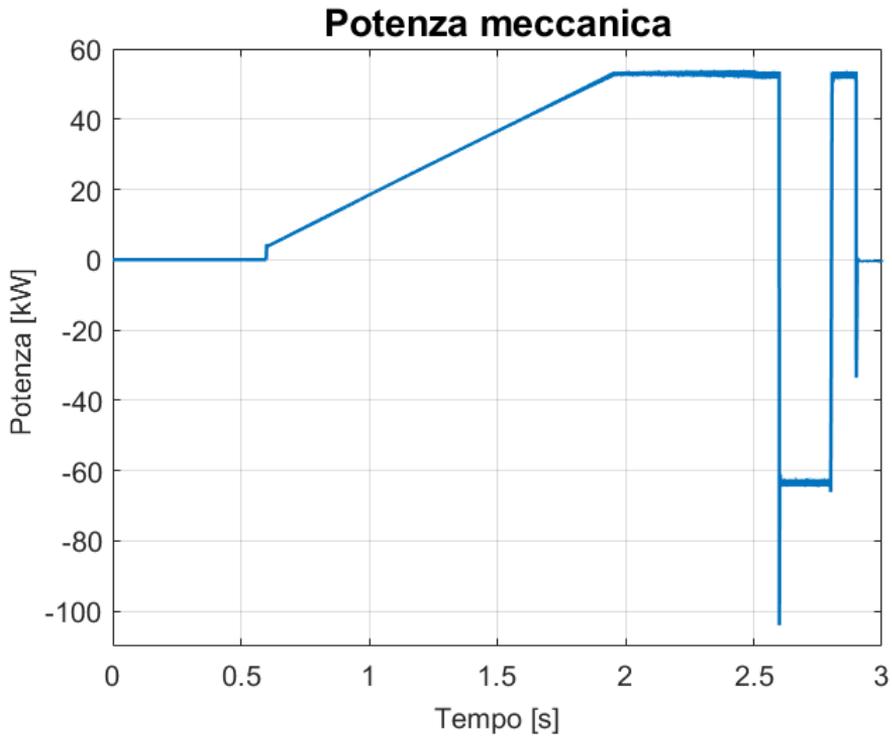


Figura 5.15

In Fig. 5.16 si verifica come un calcolo della velocità tramite PLL risulti molto efficace, perché una volta che il vettore generato va a regime nei primi istanti, rimane sempre agganciato al riferimento anche per elevati valori di velocità (testato fino a 20000 rpm).

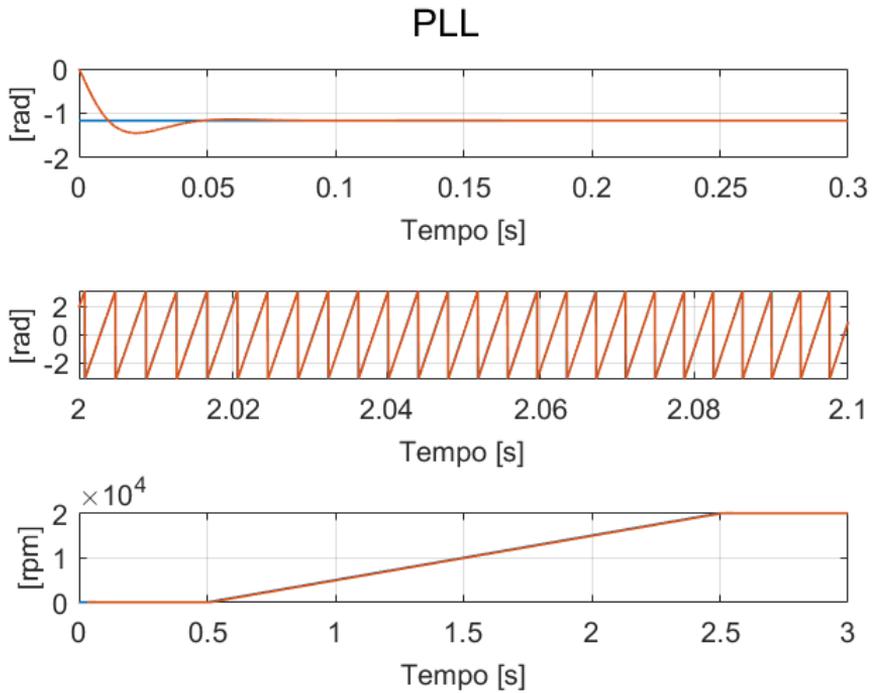
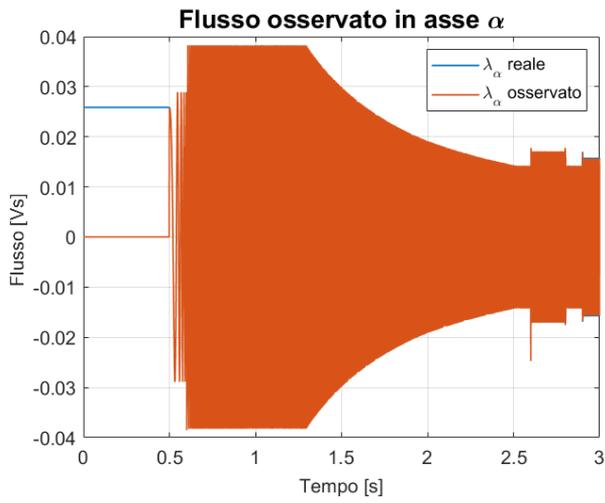


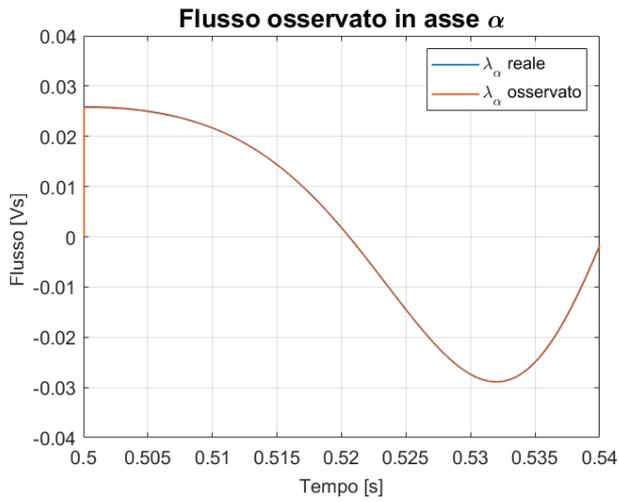
Figura 5.16: Si plottano le grandezze meccaniche reali (angolo e velocità) in blu e le rispettive grandezze calcolate dal PLL in arancione.

Arriva il momento di presentare uno dei blocchi che é stato sempre mostrato come uno tra i piú principali del controllo: l'osservatore di flusso.

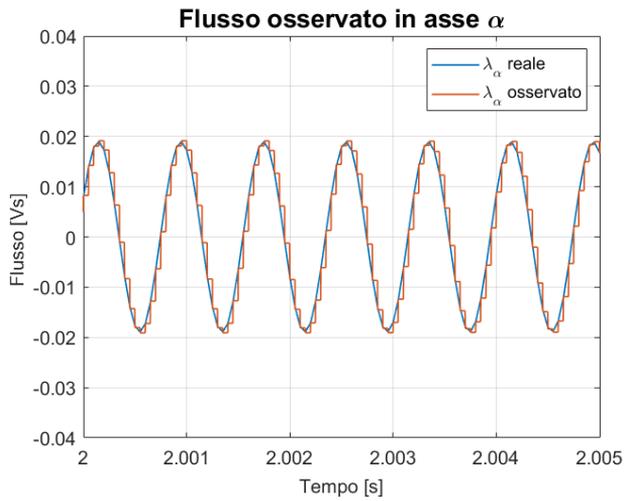
Nella Fig. 5.13 mostrata precedentemente si puó apprezzare come l'osservatore lavori bene e come cambiando il verso di coppia cambi anche il margine di flusso in deflussaggio, secondo anche (3.17). Ora però nei successivi grafici riguardanti l'osservatore si pone l'attenzione su ulteriori due aspetti. Il primo riguarda la velocità di transizione tra i due modelli di flusso stimati, infatti osservando Fig. 5.16 si calcola intorno a $0,52$ s una velocità elettrica di 90 rad/s (ricordando $p = 5$), ovvero proprio l'esatto valore g tarato all'interno del codice per l'osservatore. La transizione sembra essere quasi perfetta, a confermare una corretta taratura del parametro (Fig. 5.17(b) e 5.18(b)). Secondo aspetto é che si invita a notare un metodo di integrazione euleriano ritardato di un periodo di campionamento, per ragioni già esposte prima nella sezione *Algoritmo di controllo* (Fig. 5.17(c) e 5.18(c)).



(a)

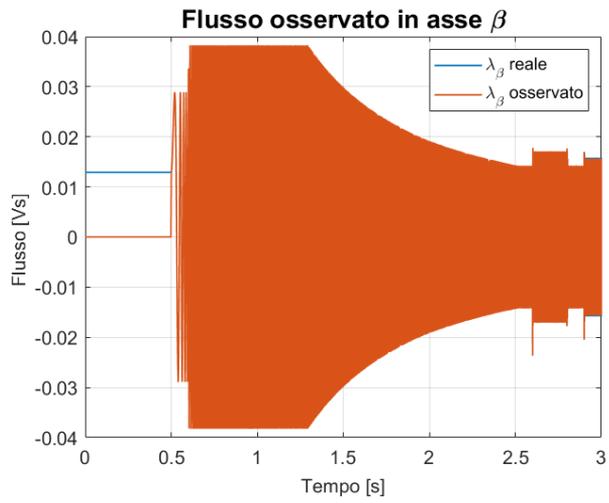


(b)

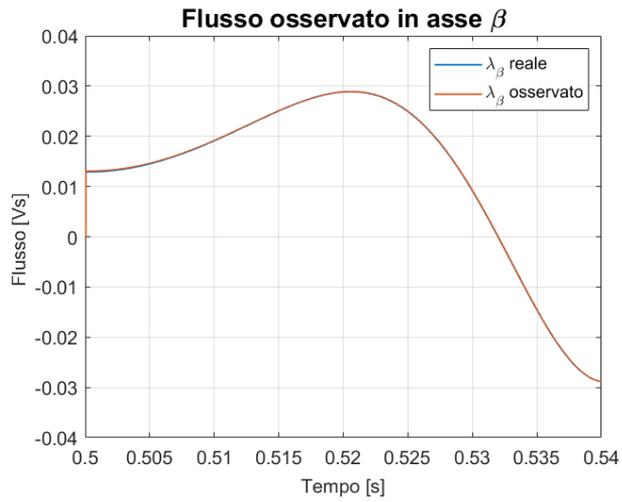


(c)

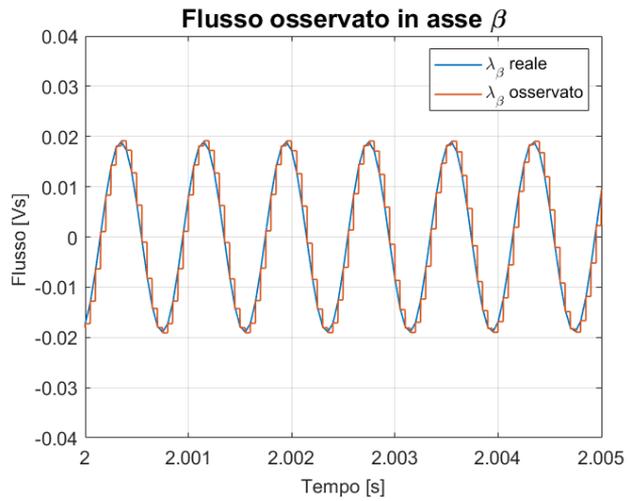
Figura 5.17



(a)



(b)



(c)

Figura 5.18

Confrontando il valore di flusso minimo alla massima velocità (Fig. 5.13) e la caratteristica di MTPV in linearità magnetica (Fig. 4.7) si giustifica finalmente il valore di δ_{MAX} (100°) nella Tab. 5.1. Però, come già anticipato, osservando l'evoluzione dell'angolo di carico (Fig. 5.19), questo motore elettrico non arriverà mai a lavorare in MTPV e quindi l'anello di controllo per la limitazione in corrente rimarrà inutilizzato.

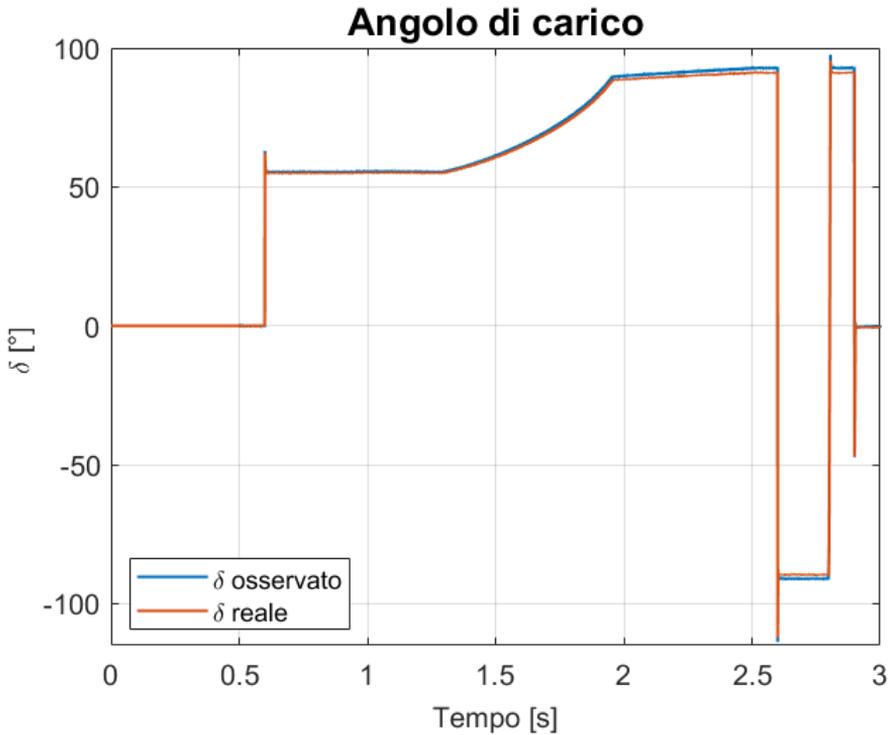
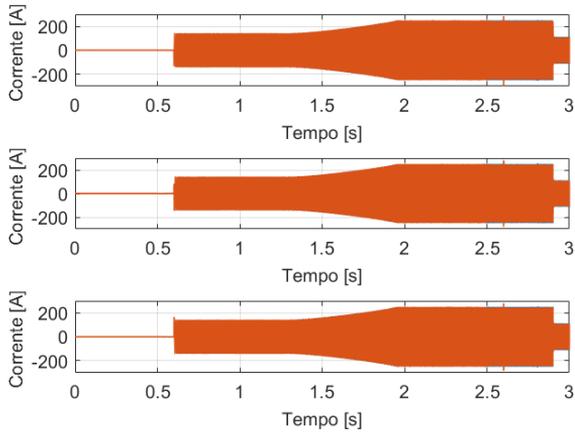
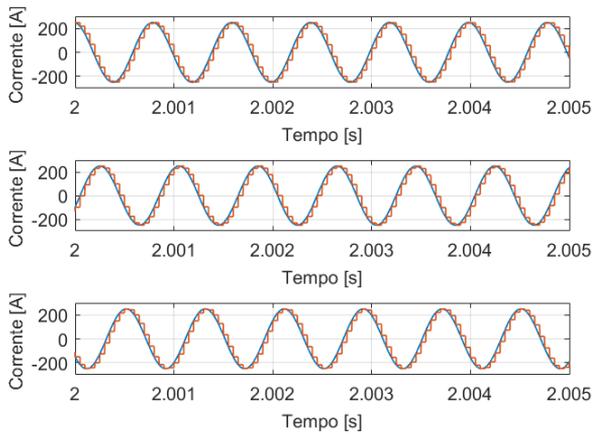


Figura 5.19: Si osservi come l'angolo di carico non superi mai δ_{MAX} di 100° , non facendo mai intervenire l'anello di controllo di limitazione in MTPV.

In chiusura si plottano le due grandezze di fase principali e l'andamento dei duty-cycles imposti dall'inverter.

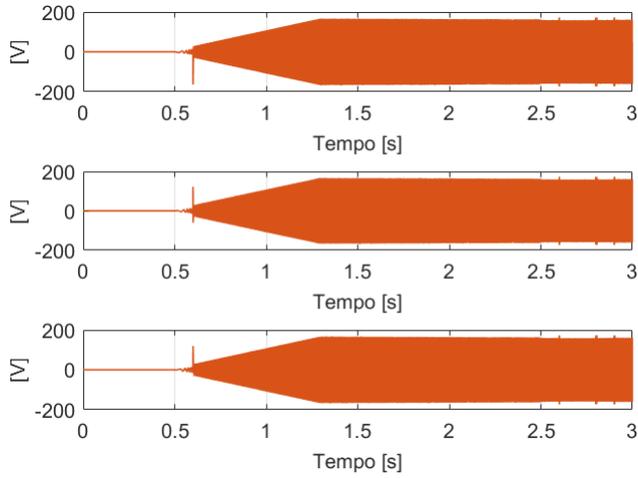


(a)

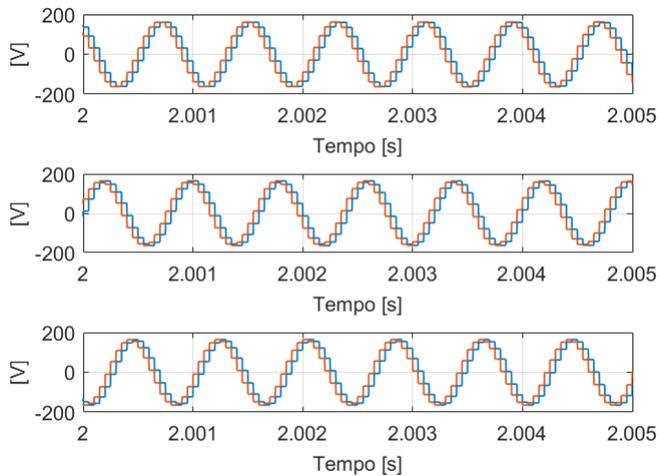


(b)

Figura 5.20: In blu le correnti reali di fase e in arancione le correnti di fase misurate per il controllo.

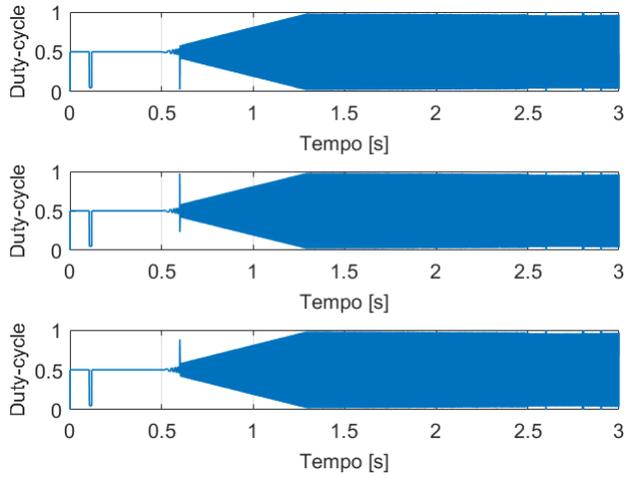


(a)

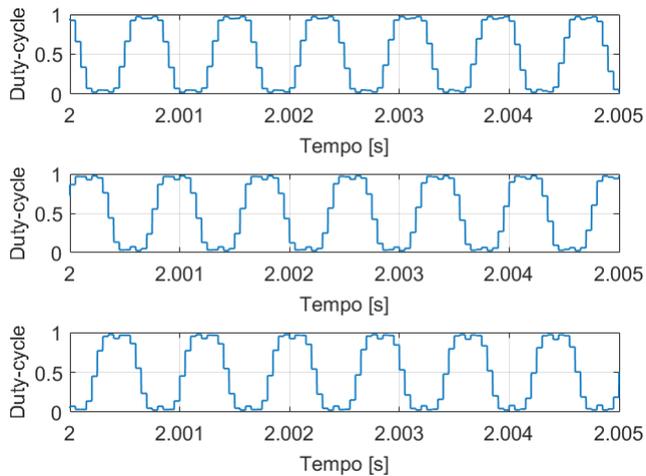


(b)

Figura 5.21: Si ricorda in Simulink un modello ai valori medi dell'inverter. In arancione le tensioni di riferimento generate e in blu le tensioni reali di fase opportunamente sfasate in ritardo di un periodo di campionamento.



(a)



(b)

Figura 5.22: Si può notare in (b), anche se per velocità che comportano un numero esiguo di campioni al periodo, l'impiego di una tecnica di modulazione BEM.

Si vuole chiudere questo capitolo riguardo la simulazione in Simulink nel far apprezzare gli effetti di un corretto controllo in *Phase Advancing*. Rimandando alla teoria esposta, tramite un semplice calcolo, si fa apprezzare l'errore di angolo che ci sarebbe nel sistema di riferimento rotante bifase del motore se non si controllasse con un'opportuna tecnica di Phase Advancing alla massima velocità raggiunta:

$$\begin{aligned}\Delta\theta_{errore} &= 1,5 * n * \frac{\pi}{30} * T_s = \\ &= 1,5 * 20000 \text{ rpm} * \frac{\pi}{30} * 0,00005 \text{ s} = 0,157 \text{ rad} = 9^\circ\end{aligned}$$

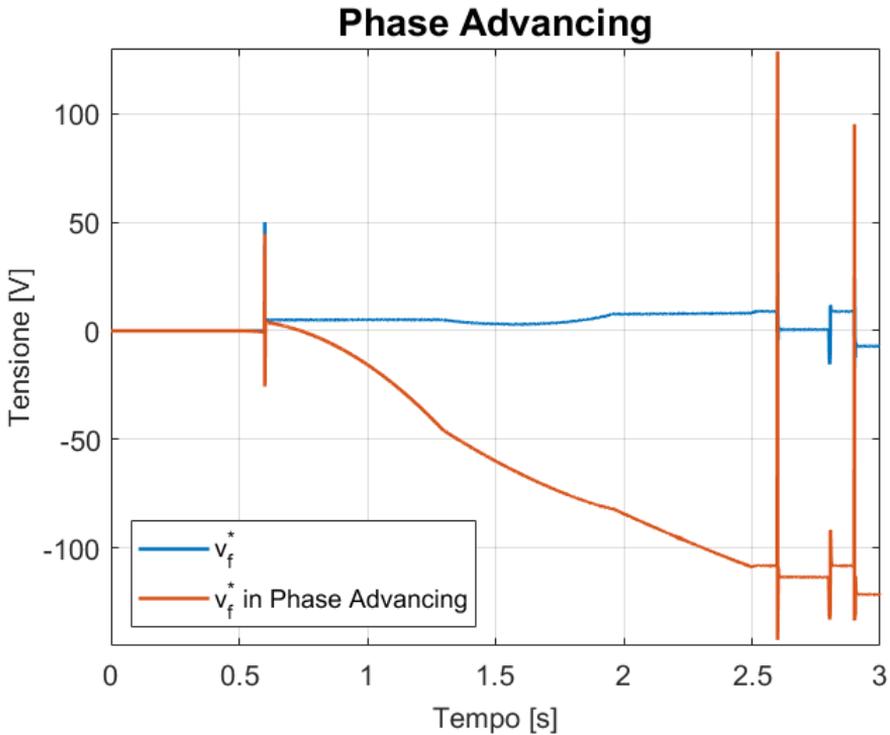


Figura 5.23

Capitolo 6

Simulazione in Plecs

In questo capitolo si trova tutta l'implementazione in ambiente Plecs. Nel seguente software, come piú volte accennato, si modella in maniera piú accurata l'intero azionamento elettrico migliorando soprattutto il modello dell'inverter e della gestione dei suoi comandi, dopo che il controllo in C é stato correttamente validato nella precedente simulazione Simulink.

Plecs risulta essere un potente software di simulazione che trova particolare impiego nel mondo dell'elettronica di potenza e si avvia verso l'intento di unificare, nello stesso ambiente, una logica a blocchi insieme a dei circuiti elettrici, magnetici, termici e meccanici. Ad esclusione della modellistica magnetica e termica, in questa simulazione saranno implementati il resto dei modelli che Plecs offre.

L'immagine seguente presenta una panoramica della simulazione (omettendo il blocco del controllo, il generatore dei segnali PWM e i vari *scope*, per ragioni legate a rendere piú chiara l'immagine) che ricorda molto quella precedente di Simulink (Fig. 5.1).

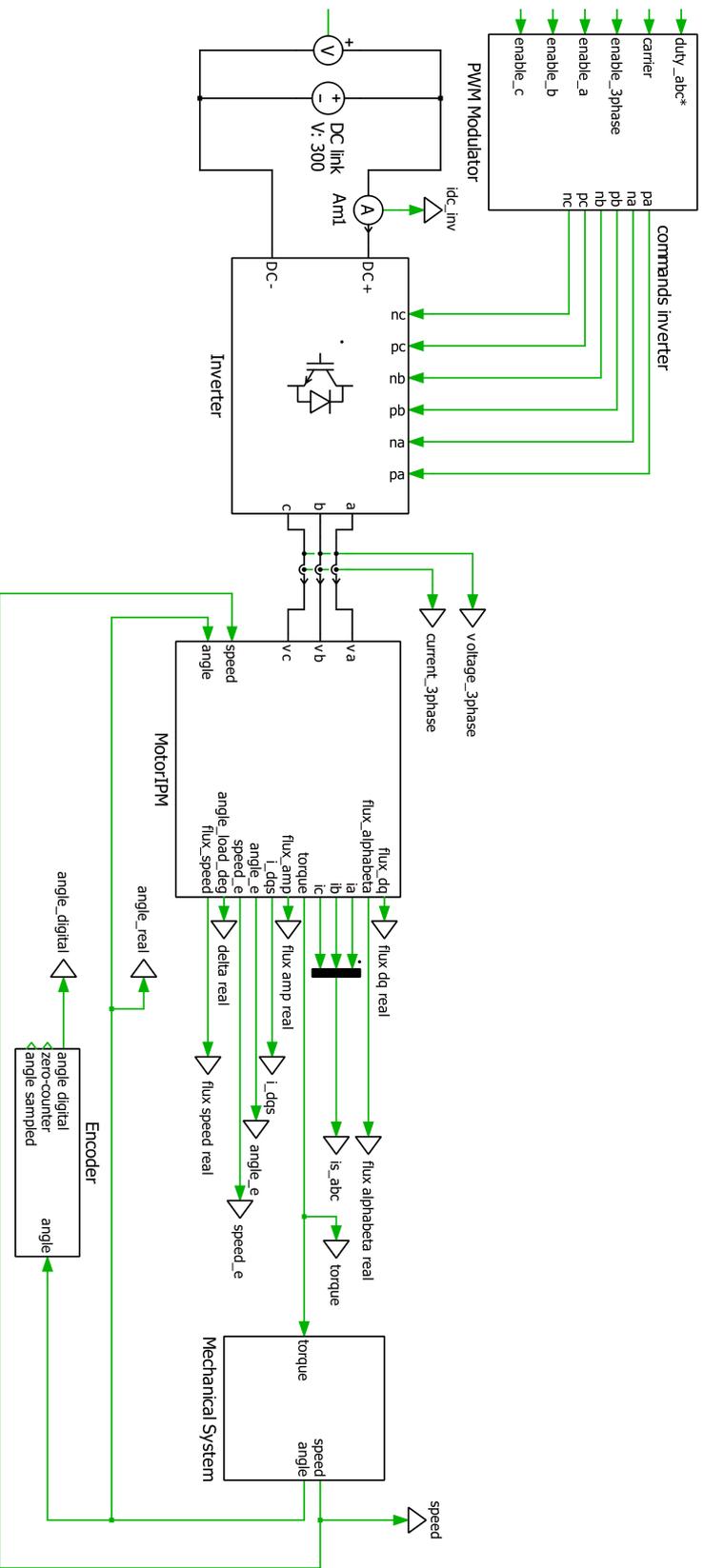


Figura 6.1: Una panoramica dei blocchi principali della simulazione implementata in ambiente Plecs. Si può notare già da questa panoramica come la simulazione sia composta da una logica a blocchi (linee verdi) e da un modellistica elettrica (linee nere). All'interno del modello meccanico si trova anche una semplice modellistica meccanica.

6.1 Modello meccanico del motore elettrico

Procedendo con lo stesso ordine con cui sono stati illustrati i vari blocchi in ambiente Simulink, si inizia dal modello del sistema meccanico.

Si perde la comodità di poter settare tramite uno switch le due diverse configurazioni (trascinamento con una velocità imposta o evoluzione libera), infatti l'utente deve attuare delle banali modifiche per implementare una delle due configurazioni possibili. Tuttavia in Plecs, per un'evoluzione libera del sistema meccanico, si è implementata la canonica equazione di Newton della meccanica tenendo conto, questa volta, di due ulteriori contributi resistivi alla coppia motrice: attriti lineari con la velocità (condensati in un unico coefficiente k) e attriti dovuti alla ventilazione (anch'essi rappresentati da un loro rispettivo coefficiente k_v).

$$T_e - T_L - k \omega_R - k_v \omega_R^2 = J_{eq} \frac{d\omega_R}{dt} \quad (6.1)$$

Di seguito la modellistica implementata in Plecs:

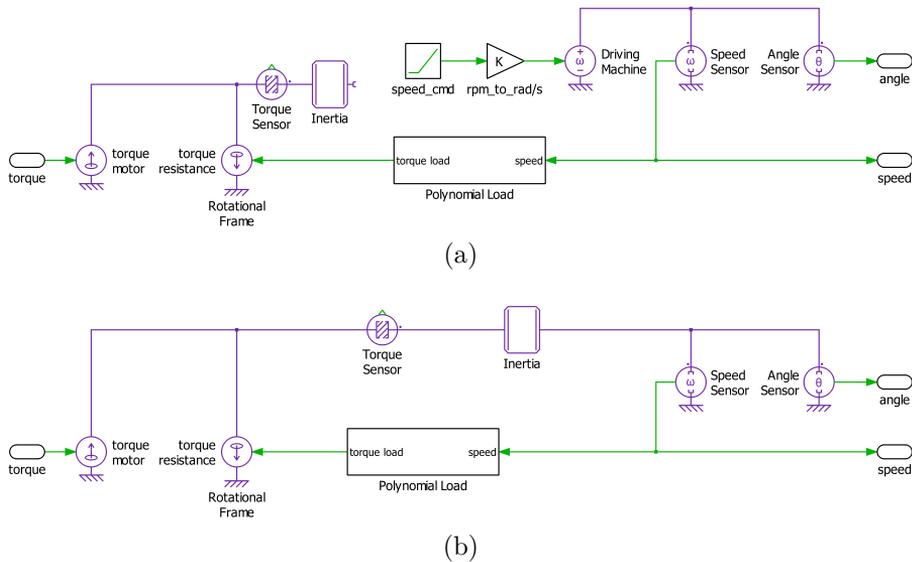


Figura 6.2: Modellistica meccanica in Plecs in due possibili configurazioni: (a) trascinamento del rotore con una velocità imposta; (b) il sistema meccanico evolve secondo l'equazione di Newton della meccanica. Il blocco meccanico sfrutta dei modelli meccanici (linee viola) insieme ad una logica a blocchi (linee verdi).

6.2 Modello elettromagnetico del motore elettrico

Procedendo con lo stesso ordine del capitolo precedente si arriva ad implementare il modello di macchina. All'interno di questo blocco si trova una prima parte dove si passa dalle grandezze meccaniche a quelle elettriche e una modellistica elettrica del circuito dove si disegna la linea trifase del motore che termina con il suo centro-stella. Nel circuito trifase sono iniettate solo due correnti di linea scelte in modo arbitrario; questo perché se si iniettassero tutte e tre le correnti, calcolate dal modello di macchina, un minimo errore di calcolo da parte del software potrebbe portare a non soddisfare

l'equazione elettrica al nodo del centro-stella, ovvero $i_a + i_b + i_c = 0$, con conseguente errore di procedura nella simulazione.

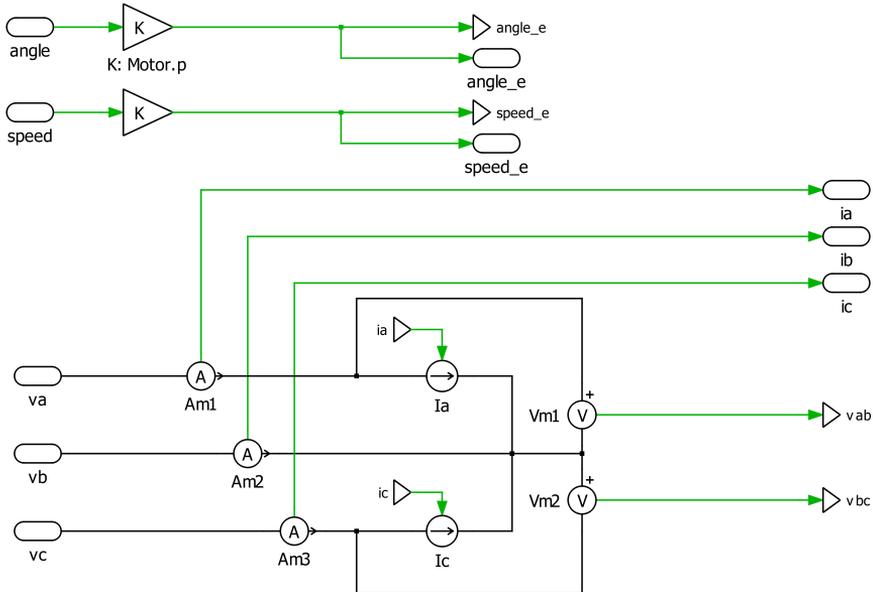


Figura 6.3: Prima parte della modellistica elettromagnetica del motore elettrico in PLECS. Nel circuito trifase sono iniettate solo due correnti di linea calcolate nel modello di macchina e misurate due delle tre tensioni concatenate (ingressi del modello elettromagnetico).

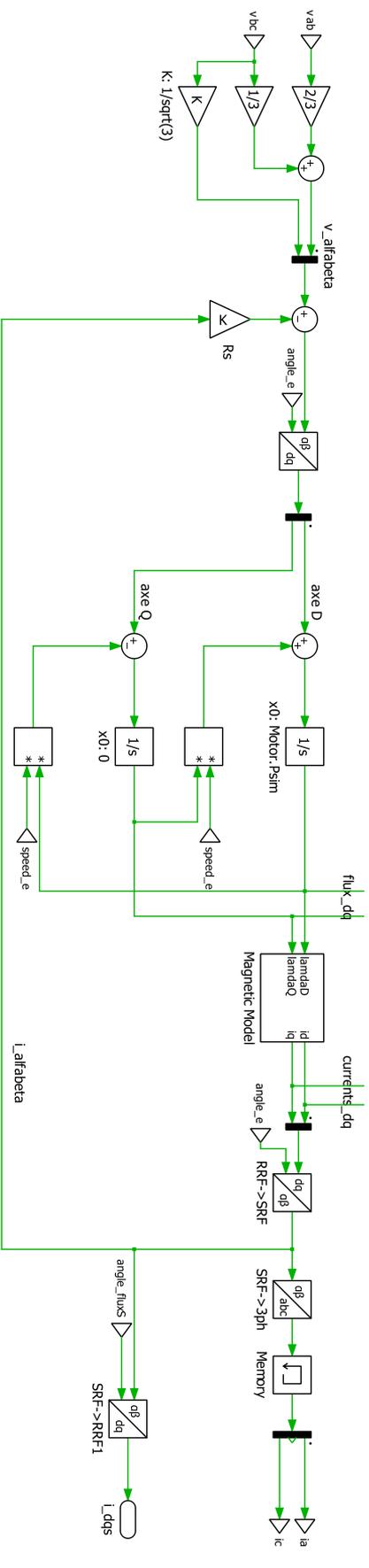


Figura 6.4: Modellistica del modello elettromagnetico del motore in PLECS. La logica a blocchi implementa le equazioni di macchina (2.6). I flussi e le correnti nei canonici assi dq sono usati per calcolare la coppia elettromeccanica e ulteriori manipolazioni utili per alcuni scope; questi blocchi non sono riportati per rendere piú chiara la figura.

In conclusione a questa sezione si vuole porre l'attenzione come questa simulazione preveda un modello magnetico molto piú realistico, in quanto é possibile caricare le tabelle inerenti alla mappatura magnetica del motore, tenendo conto in questo modo di fenomeni non lineari come la saturazione del ferro e l'accoppiamento magnetico tra i due assi rotanti (*cross-coupling*):

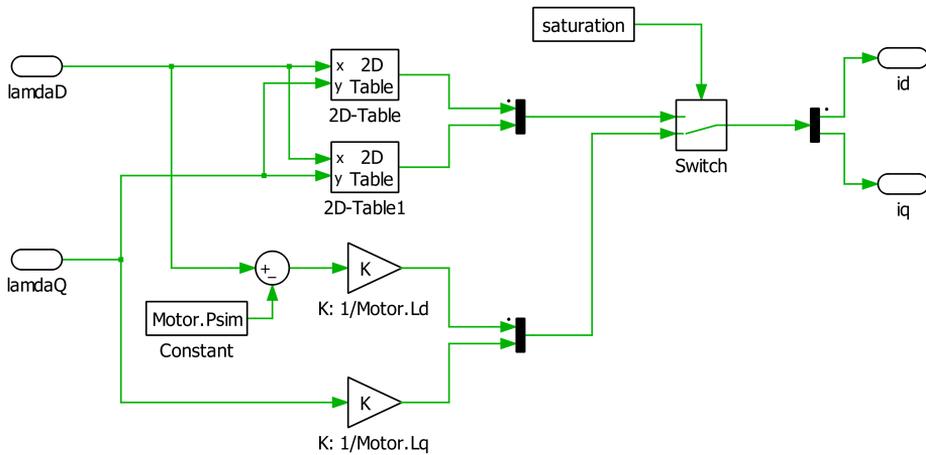


Figura 6.5: Modellistica del modello magnetico in Plecs. Tramite uno switch logico é possibile scegliere un modello magnetico lineare oppure un modello realistico che tiene conto della mappatura magnetica tramite due *Look-Up-Table*.

6.3 Modello dell'encoder

In ambiente Simulink era stato implementato un encoder assoluto. In ambiente Plecs ci si spinge a modellizzare un encoder, il quale, a discrezione dell'utente puó essere assoluto o incrementale. Il principio di funzionamento per l'encoder assoluto é stato giá illustrato, quindi ora si passa a spiegare il principio su cui si fonda quello incrementale.

L'encoder incrementale si basa su due tracce, sfasate tra loro di 90° , per poter discriminare il senso di rotazione. Con un counter

sensibile ai fronti di salita si può ricavare una posizione digitalizzata. Venendo a mancare un riferimento assoluto è necessaria una terza traccia, la quale presenta solamente una tacca per poter identificare lo zero di reset per il counter.

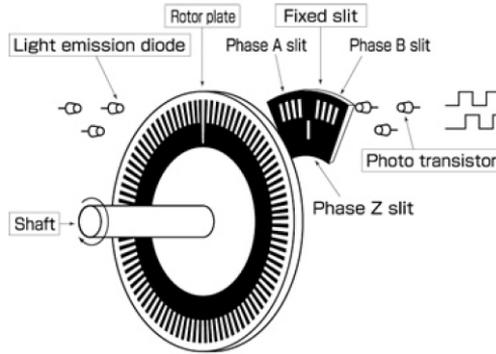


Figura 6.6: Struttura semplificata di un encoder incrementale.

Un encoder di questa natura comporta degli accorgimenti importanti in un qualsiasi controllo di macchina, in quanto fino a quando non vi è il primo reset, lo strumento misura un angolo completamente privo di senso, quindi il controllo deve avere un'intelligenza tale da iniziare l'algoritmo solo dopo che il counter viene resettato (si può pensare di far muovere il rotore lentamente fino a quando viene eseguito il primo giro meccanico).

Il modello in Plecs qui proposto presenta un offset di angolo costante, quindi un codice dovrebbe misurare questo offset fino a quando il counter non viene resettato per la prima volta al primo giro. Tuttavia, in questo lavoro di tesi, la logica di controllo implementata si sposa solo con un'idea di dover impiegare un encoder assoluto nel reale azionamento elettrico, ragion per cui l'opzione di un encoder incrementale è modellizzata ma non sarà mai impiegata.

L'angolo in entrata è già compreso tra $-\pi$ e π ed è discretizzato, come in Simulink, sulla base del numero di divisioni dell'encoder.

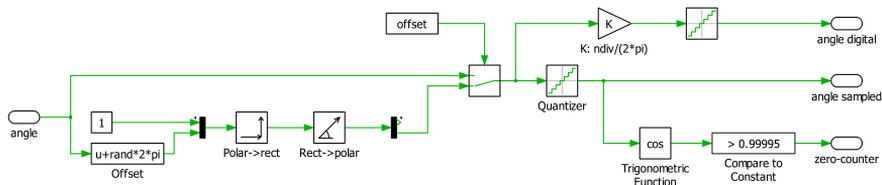


Figura 6.7: Modellistica dell’encoder in Plecs. Uno swicht logico discrimina la scelta tra un encoder assoluto o incrementale, mentre il blocco trigonometrico sulla destra decreta il reset del counter per un eventuale encoder incrementale.

6.4 Modello del generatore dei segnali PWM

Il blocco del modello del generatore dei segnali, che non compare nella panoramica iniziale (Fig. 6.1), é praticamente identico a quello già illustrato e proposto nella simulazione Simulink, con in uscita la stessa portate per la modulazione PWM e gli stessi tipi di sincronizzazione per il trigger del blocco di controllo. Pertanto si preferisce passare subito al prossimo blocco.

6.5 Modello dell’inverter

Arrivati al modello dell’inverter, probabilmente in questa sezione si apprezzerá in maniera molto piú significativa una modellistica nettamente differente con quella presentata in Simulink.

Il modello dell’inverter é implementato su due blocchi differenti: uno é il modulatore PWM e presenta una logica a blocchi, l’altro é proprio il circuito elettrico dell’inverter che sfrutta la modellistica elettrica proposta da Plecs.

6.5.1 Circuito elettrico dell'inverter

Il blocco del circuito elettrico dell'inverter si pone tra il DC-link e la linea trifase del motore. Data la natura trifase del convertitore sono necessari in ingresso sei comandi binari (due per gamba) che andranno a comandare gli switches del sistema. In questa simulazione é stato modellizzato l'inverter trifase con dei MOSFET e dei diodi. I primi presentano semplicemente un puro modello resistivo con una resistenza costante, mentre i secondi hanno i due parametri caratteristici della caduta di soglia e della resistenza equivalente, anch'essi costanti.

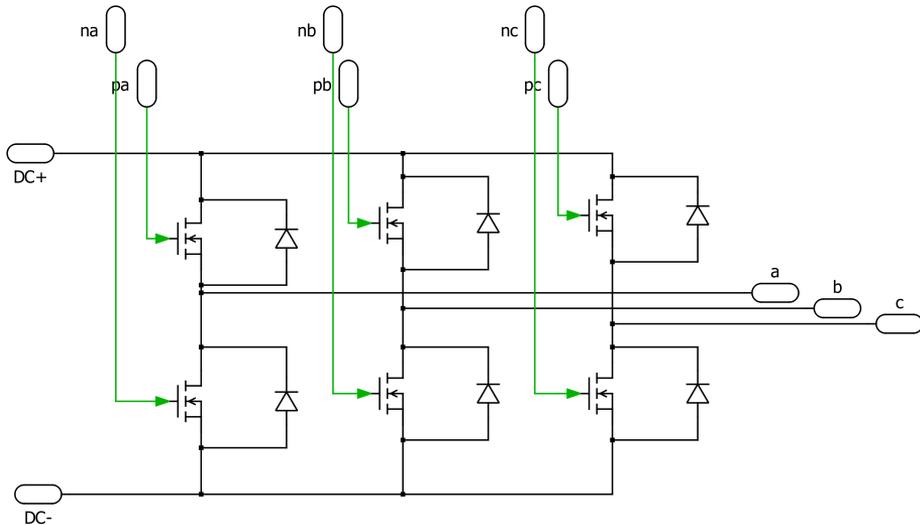


Figura 6.8: Modellistica del circuito elettrico dell'inverter in Plecs. In ingresso i morsetti del DC-link e in uscita i morsetti del motore trifase. In ingresso, inoltre, i sei comandi binari per gli switches.

6.5.2 Modulatore PWM

Il modulatore riceve in ingresso la portante PWM generata dal generatore di segnali PWM e i tre duty-cycles generati dal controllo. Inoltre é predisposto a ricevere anche comandi logici per disabilitare le tre gambe o una singola gamba nello specifico. Il modulatore genera i sei comandi che vanno a pilotare l'inverter.

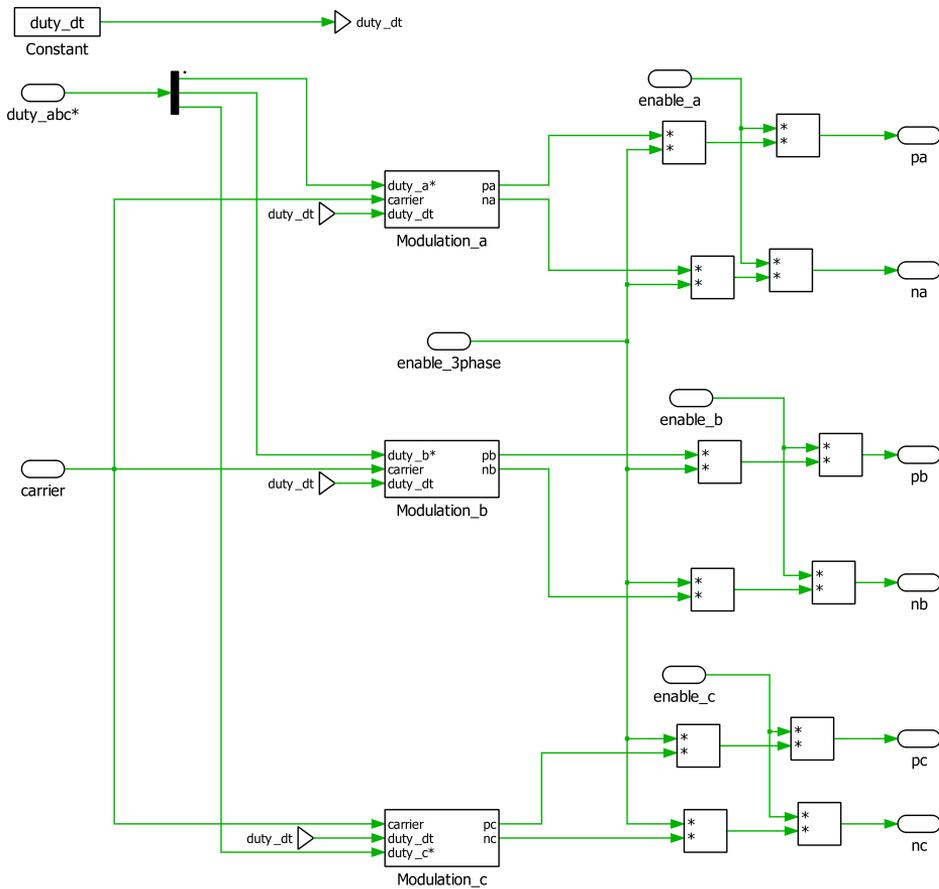


Figura 6.9: Modellistica del modulatore PWM in Plecs. I tre blocchi principali contengono le modulazioni PWM per le rispettive fasi. A destra una logica a blocchi per disabilitare tutte e tre le fasi o una singola.

Una gamba di inverter riceve due comandi per i due rispettivi switches. I due comandi per una corretta modulazione devono risultare complementari o quanto meno non devono mai risultare alzati contemporaneamente in un inverter a tensione impressa, per evitare di cortocircuitare il banco di condensatori del DC-link. Per ovviare questo problema si introduce un *dead-time* (passato all'interno del blocco, vedi Fig. 6.9) tra i due comandi, al fine di escludere i due comandi alzati nello stesso istante:

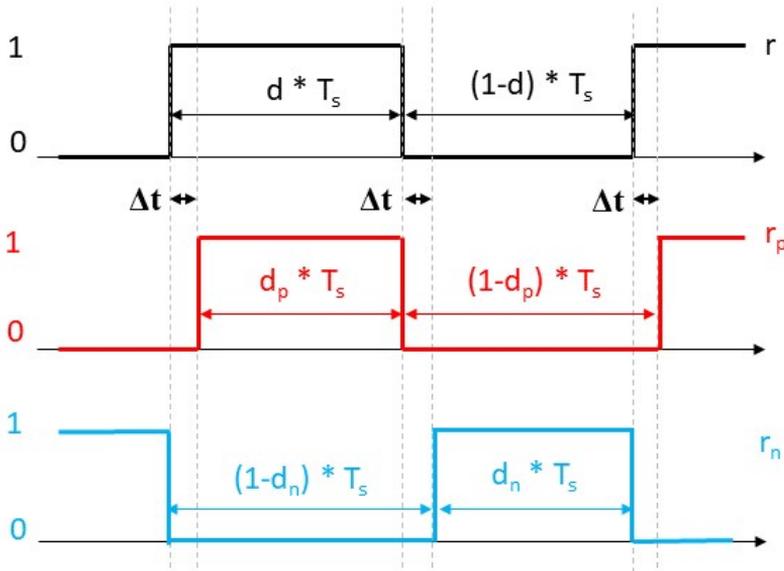


Figura 6.10: Il duty generato dal controllo corrisponde ad un comando logico r teorico (nero). Praticamente si pilota una gamba di inverter introducendo un dead-time Δt ottenendo un comando logico r_p per lo switch superiore (rosso) e un comando logico r_n per lo switch inferiore (azzurro).

Osservando Fig. 6.10 si nota come introdurre un dead-time di ritardo nei comandi, equivale a diminuire di un duty-cycle equivalente di dead-time ($d_{eq.dt}$) lo stesso duty generato dal controllo (d). In formule risulta:

$$d_{eq.dt} = \frac{\Delta t}{T_s} \quad (6.2)$$

$$d_p = d - d_{eq.dt}$$

$$d_n = (1 - d) - d_{eq.dt}$$

Nella realtà degli azionamenti elettrici, un valore indicativo per ottenere questo risultato é quello di impostare un dead-time pari all' 1% del periodo di commutazione.

In questa simulazione viene aggiunto, anziché sottratto, il duty-cycle equivalente per gli switches inferiori. Il risultato voluto rimane invariato se si inverte anche la comparazione logica tra portante e modulante solo per gli stessi switches inferiori.

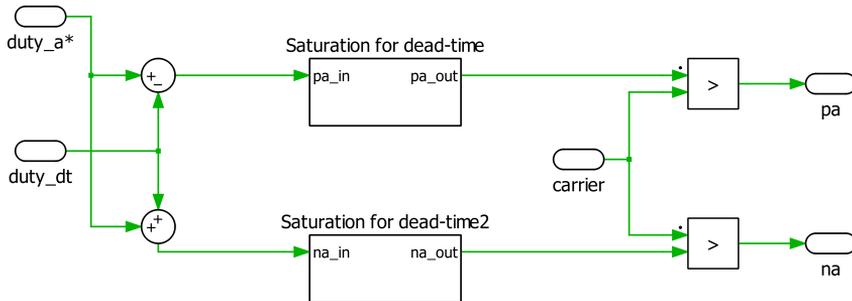


Figura 6.11: Modellistica per la modulazione PWM di una singola fase in PLECS. Il saturatore a valle della modulazione satura a 1 i duty-cycles maggiori di una certa soglia (es. 0,98) e a 0 analogamente i duty-cycles minori di un certo valore (es. 0,02).

6.6 Modello del DC-link e delle misure elettriche

Anche questa sezione risulterà molto breve in quando è tutto analogo a quello che è stato già presentato in ambiente Simulink. Unica differenza è che il DC-link, non più in una logica a blocchi, è modellizzato con un generatore di tensione costante in continua.

6.7 Algoritmo di controllo

Il codice in C del controllo DFVC è stato precedentemente testato e validato nella simulazione Simulink, quindi nel nuovo ambiente Plecs viene semplicemente importato in un blocco che ha una funzione analoga alla *S-Function* in Simulink.

Unica modifica necessaria al codice riguarda la parte iniziale, prima della logica a stati della macchina (Fig. 5.9), dove sono ricostruite le tre tensioni di fase a partire dai tre duty-cycles generati al passo precedente. In Plecs si è modellizzato un inverter reale che porta degli errori di tensioni tra le tensioni di riferimento del controllo e le tensioni di fase realmente applicate al motore. Ecco che di conseguenza bisogna modificare il codice, tenendo conto nel ricostruire le tensioni degli errori dovuti ad un inverter reale e non ideale.

6.7.1 Ricostruzione delle tensioni di fase

In Simulink, avendo nella simulazione un inverter completamente ideale (zero cadute di tensione e dead-time nulli), le tensioni di fase erano correttamente ricostruite a partire dai duty-cycles generati al passo precedente. Ricordando la formula per una generica fase, facendo riferimento alla (5.2):

$$v_a = \frac{V_{DC}}{3} (2d_a^* - d_b^* - d_c^*)$$

Ora però con un inverter reale, tra la tensione di riferimento e quella di fase, vi sono le cadute di tensione dell'inverter e una tensione equivalente dovuta al dead-time, ottenendo per una generica fase f :

$$d_f^* \neq d_f$$

$$v_a = \frac{V_{DC}}{3} (2d_a - d_b - d_c)$$

Quindi l'obiettivo è cercare di capire come ricostruire i duty-cycles reali di fase (d) a partire da quelli di riferimento generati dal controllo (d^*).

Chiamando sempre con M il punto medio virtuale del DC-link, si inizia a scrivere per una generica fase:

$$v_{fM}^* - v_{fM} = (V_{DC} d_{eq,dt} + V_0) \text{sign}(i_f) + R_{on} i_f = \Delta v \quad (6.3)$$

E con i rispettivi duty-cycles diventa (inglobando la resistenza equivalente degli interruttori con la resistenza serie di fase del motore):

$$d_f^* - d_f = \Delta d_{compensazione} \text{sign}(i_f) \quad (6.4)$$

In conclusione si giunge ad ottenere un'equazione che legghi i duty-cycles di riferimento a quelli reali imposti alle fasi del motore:

$$\begin{cases} d_f = d_f^* - \Delta d_{compensazione} & \text{se } i_f > 0 \\ d_f = d_f^* + \Delta d_{compensazione} & \text{se } i_f < 0 \end{cases} \quad (6.5)$$

Per aiutare la comprensione di queste ultime equazioni si riportano i seguenti grafici che fanno riferimento ad una generale gamba di un inverter.

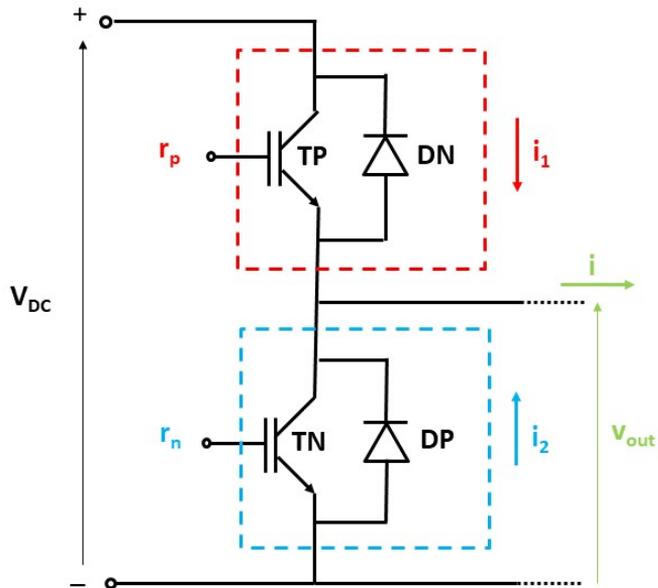


Figura 6.12: Schema di una generale gamba di inverter.

Si suppone nei grafici di intendere una tensione in uscita al convertitore che veda a valle la serie della resistenza di fase del motore (R_s) e di quella equivalente in conduzione dell'inverter (R_{on}):

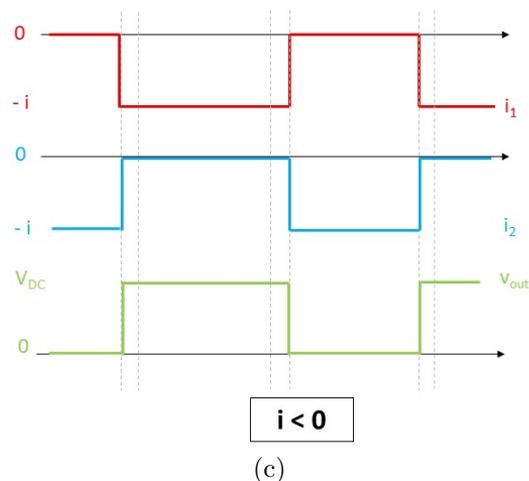
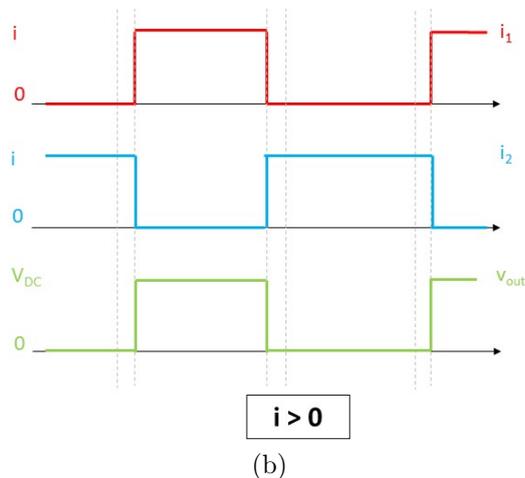
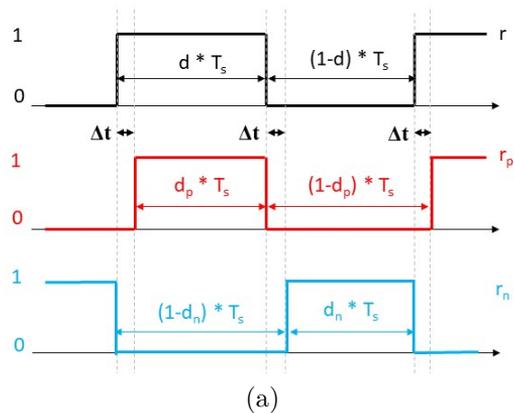


Figura 6.13: Dall'alto verso il basso: (a) in nero il comando teorico generato. In rosso e in celeste rispettivamente i comandi reali dello switch superiore e inferiore che si differenziano da quello teorico per un duty-cycle equivalente sottratto (nel rosso) o aggiunto (nel celeste); (b)-(c) andamenti di correnti e tensioni facenti riferimento alla Fig. 6.12 per una corrente in uscita rispettivamente positiva e negativa.

I grafici confermano l'equazione (6.5).

Ultimo passaggio, prima di riportare le modifiche apportate al codice, é spiegare come si puó ottenere una tabella che metta in relazione per ogni valore di corrente un duty-cycle equivalente per la compensazione.

La mappatura per la compensazione degli errori di tensione dovuti all’inverter segue fedelmente la procedura indicata in [7]. Unica differenza sta nel codice di controllo dell’inverter, perché nell’articolo esso stesso prevede una fase di *commissioning* iniziale in cui il controllo si costruisce la sua mappa, mentre in questo lavoro di tesi, solo per non appesantire ulteriormente il codice, si é ricavata la tabella in una simulazione gemella per poi essere caricata sulla simulazione principale.

Per ricavare la tabella del duty-cycle equivalente (per la compensazione) e la corrente di fase si controlla il motore in continua controllando una corrente di test in asse α . In queste condizioni, come si dimostra in [8], si puó scrivere:

$$\begin{aligned}\vec{v}_s^* &= \vec{v}_s + \Delta\vec{v}_{tot} = R_s \vec{I}_{test} + \Delta\vec{v}_{tot} \\ \Delta\vec{v}_{tot} &= \frac{4}{3} V_{th\,eq} \text{sign}(\vec{I}_{test}) + R_{on} \vec{I}_{test} \quad \text{con } V_{th\,eq} = V_{DC} d_{eq.\,dt} + V_0 \\ V_\alpha^* &= (R_s + R_{on}) I_{\alpha\,test} + \frac{4}{3} V_{th\,eq} (I_{\alpha\,test})\end{aligned}\tag{6.6}$$

Si pone attenzione come la caduta di soglia equivalente é funzione della corrente non solo per il segno, ma anche dal suo stesso valore che assume, a causa della non linearitá dell’inverter.

La prova sinteticamente si suddivide in una prima fase in cui si calcola la resistenza serie equivalente:

$$R'_s = R_s + R_{on} = \frac{V_{\alpha 2}^* - V_{\alpha 1}^*}{I_{\alpha 2\,test} - I_{\alpha 1\,test}}\tag{6.7}$$

E una seconda fase dove si costruisce la tabella finale, obiettivo principale:

$$\Delta d_{compensazione\,i} = \frac{3}{4} \frac{V_{\alpha i}^* - R'_s I_{\alpha\,test\,i}}{V_{DC}}\tag{6.8}$$

Questa procedura é stata ripetuta dodici volte sulla simulazione gemella e poi le tabelle ottenute sono state caricate sulla simulazione principale. Si ottengono dodici tabelle in quanto si sono combinate dodici diverse possibili configurazioni, con quattro valori di tensione al DC-link e tre valori possibili di dead-time. La corrente di test impiegata, invece, varia quadraticamente fino alla corrente massima ammessa dal controllo.

```
// TRASFORMAZIONI DI CLARKE DELLE CORRENTI E DELLE TENSIONI
// DI FASE

// Trasformazione per le correnti
Clarke (&iabc, &iabFbk);

// Look-Up-Table dei duty-cycles equivalenti per la compensazione
// in funzione del valore assoluto della corrente di fase.
// &invcomp_<Tensione DC-link [V]>_<dead-time [us]>
DTduty.a = ReadLut (&invcomp_300_075[0], &I_comp[0],
                   fabs(iabc.a));
DTduty.b = ReadLut (&invcomp_300_075[0], &I_comp[0],
                   fabs(iabc.b));
DTduty.c = ReadLut (&invcomp_300_075[0], &I_comp[0],
                   fabs(iabc.c));

// Ricostruzione dei duty-cycles reali applicati al motore
dt_duty (&iabc, &duty, DTduty);

// Ricostruzione delle tensione reali di fase applicate
// al motore
real_Vref (&vabcRec, duty, vdc);

// Trasformazione per le tensioni
Clarke (&vabcRec, &vabRec);
```

6.8 Risultati simulazione Plecs

In questa sezione si giunge all'obiettivo principale di questo lavoro di tesi, ovvero simulare il controllo in DFVC e l'intero azionamento in

ambiente Plecs.

Il test finale in Plecs sarà il medesimo di quello eseguito in Simulink, quindi imponendo gli andamenti di coppia e velocità mostrati in Fig. 5.11. Secondo test sarà verificare, con la modellistica adottata (vedi Fig. 6.1), le grandezze elettriche sul DC-link in condizioni di UGO alla massima velocità meccanica (20000 rpm).

Prima di presentare i due test si precisa solamente che le condizioni di simulazione sono le medesime della Tab. 5.1, con un DC-link anch'esso fissato a 300 V. Unica modifica riguarda la taratura dell'osservatore di flusso, della limitazione in deflussaggio e di un tempo di dead-time; tutti e tre a causa soprattutto della presenza di una modellistica di un inverter reale. La taratura dei primi due, al solito, non è legata ad un procedimento univoco, ma frutto dell'esperienza e di alcune prove a tentativo.

Per i due parametri in coda si rimanda (scritta a lato) all'equazione di riferimento.

Inverter	<i>Dead-time</i>	0,75	μs
Osservatore di flusso	g (3.8)	500	rad/s
Deflussaggio	α (3.17)	0,90	-

Tabella 6.1: In tabella si riportano solo le modifiche apportate in Plecs rispetto alla Tab. 5.1 della taratura dei parametri di controllo in Simulink.

Mentre i dati dell'azionamento elettrico naturalmente rimangono invariati, ad eccezione che nel codice si modifica la resistenza di fase con la resistenza serie equivalente, che tiene conto (in parte) delle cadute resistive degli switches, secondo anche il modello in riferimento a [7]:

$$R'_s = 80 m\Omega$$

6.8.1 Test in Plecs

Il test finale in Plecs prevede gli stessi riferimenti, in termini di coppia e di velocità, precedentemente illustrati per il test finale in ambiente Simulink (Fig. 5.11). Confrontando i risultati si nota subito come in Plecs il controllo appare molto piú disturbato causato sicuramente dalla presenza di una modellistica di un inverter reale.

Procedendo con il medesimo ordine si fa vedere il grafico riguardo il riferimento di coppia. Si precisa semplicemente che rispetto all'andamento in Simulink (Fig. 5.12), i comandi iniziali di partenza del motore hanno una sequenza temporale diversa, ma é comunque prevista una fase di *Flux Build* per far andare a regime l'osservatore nella sua inizializzazione.

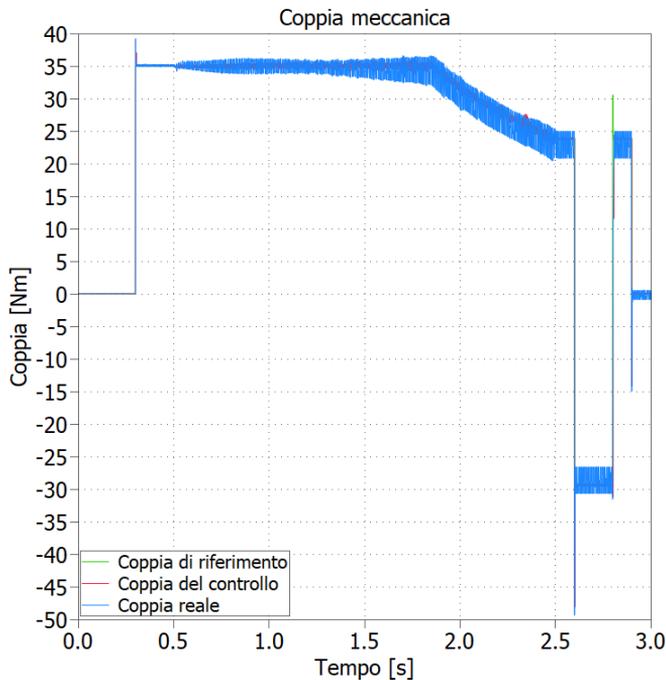


Figura 6.14: In questo plot si é considerata la coppia di riferimento già saturata all'interno del codice

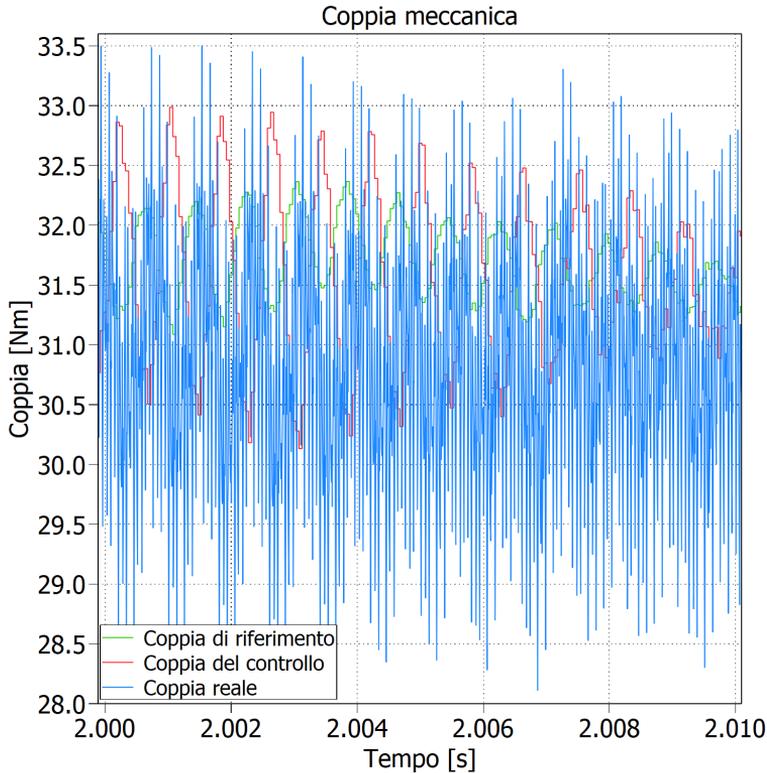


Figura 6.15

E' subito evidente (Fig. 6.14) come la presenza di un inverter reale renda il controllo molto piú disturbato all'aumentare della velocità, nonostante le accortezze prese nella modifica di alcune parti del codice come visto nella precedente sezione: saturazione della coppia di riferimento e ricostruzione delle tensioni di fase tenendo conto della compensazione dell'errore di tensione dovuto all'inverter.

Seguono i grafici principali del controllo, ovvero il flusso controllato in asse f e la corrente in asse τ . Per entrambi é subito evidente, come anche per la coppia, il fenomeno di un disturbo che cresce sensibilmente all'aumentare della velocità imposta al rotore:

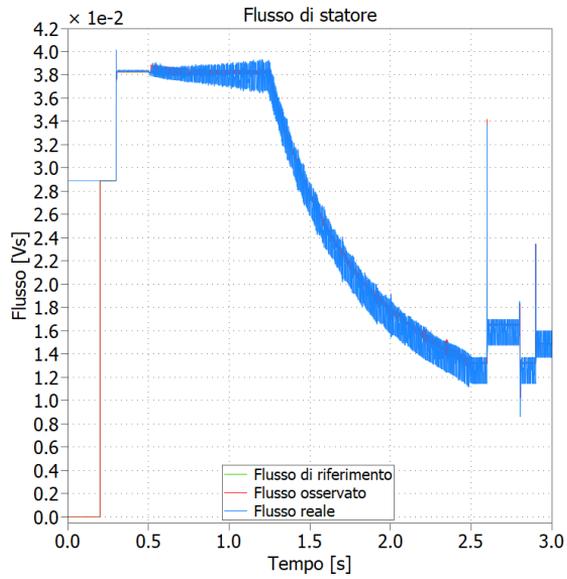


Figura 6.16: Flusso di statore controllato in asse f .

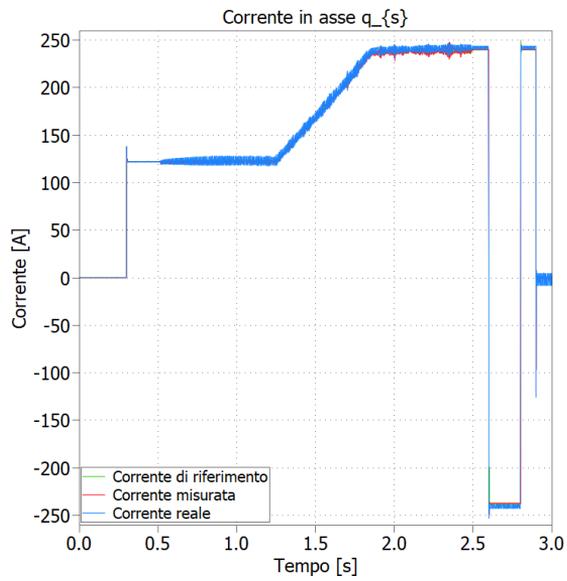


Figura 6.17: Corrente controllata in asse τ .

Procedendo nello stesso ordine si arriva alla potenza trasmessa all'albero. Interessante apprezzare ora il contributo dell'inverter in termini di perdite:

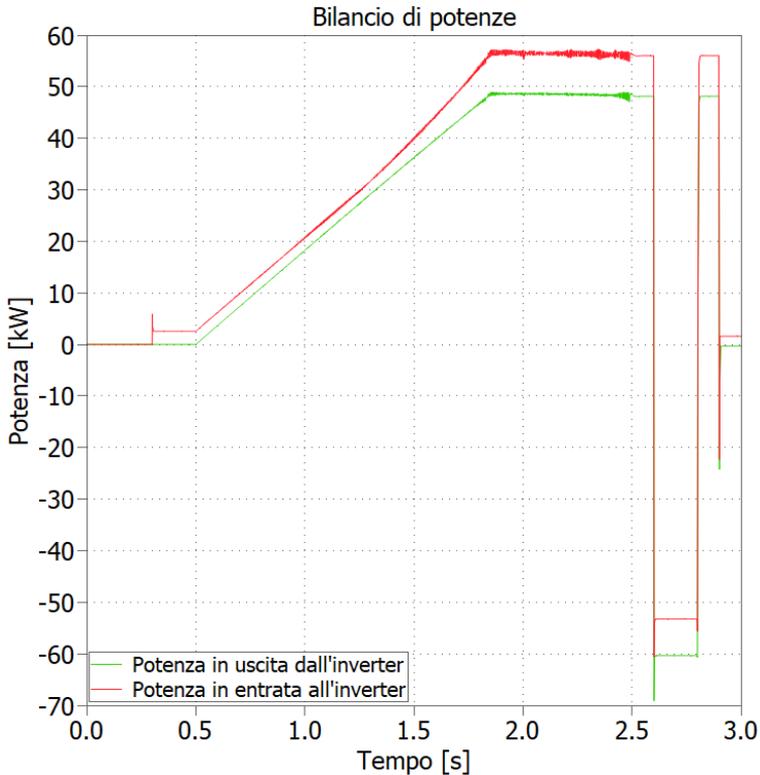


Figura 6.18

Il PLL lavora con la stessa efficacia già vista in Simulink, quindi si è preferito non riportare il suo grafico che non porterebbe nulla di ulteriormente significativo.

L'osservatore in Plecs, nonostante una modellistica più realistica continua a lavorare molto bene. Le figure mettono in evidenza la zona di transizione tra le due stime di flusso (ora $g = 500 \text{ rad/s}$, quindi la transizione si ha per una velocità meccanica di circa 955

rpm , ricordando sempre $p = 5$) e l'integrazione euleriana ritardata di un passo di campionamento:

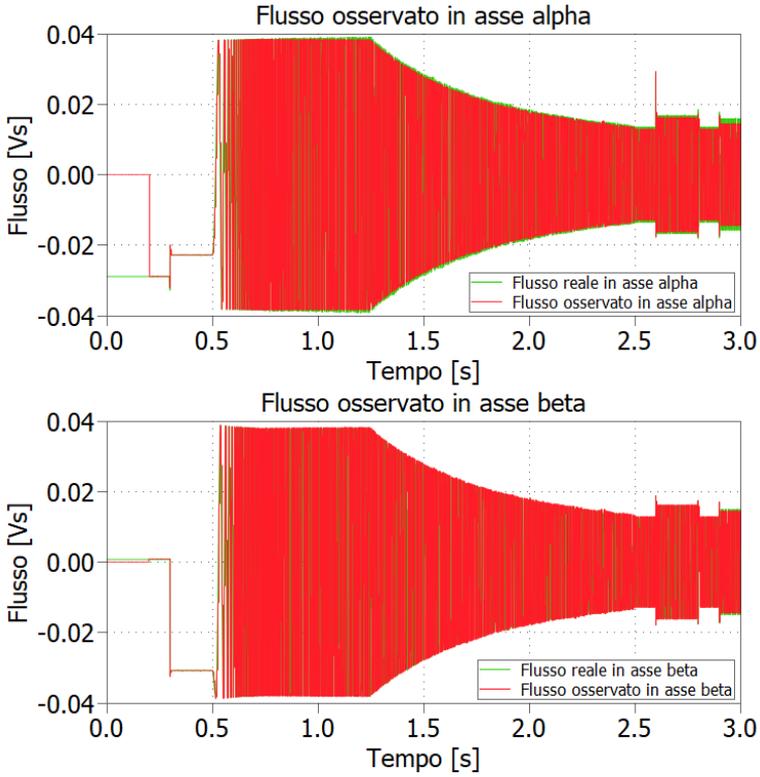


Figura 6.19

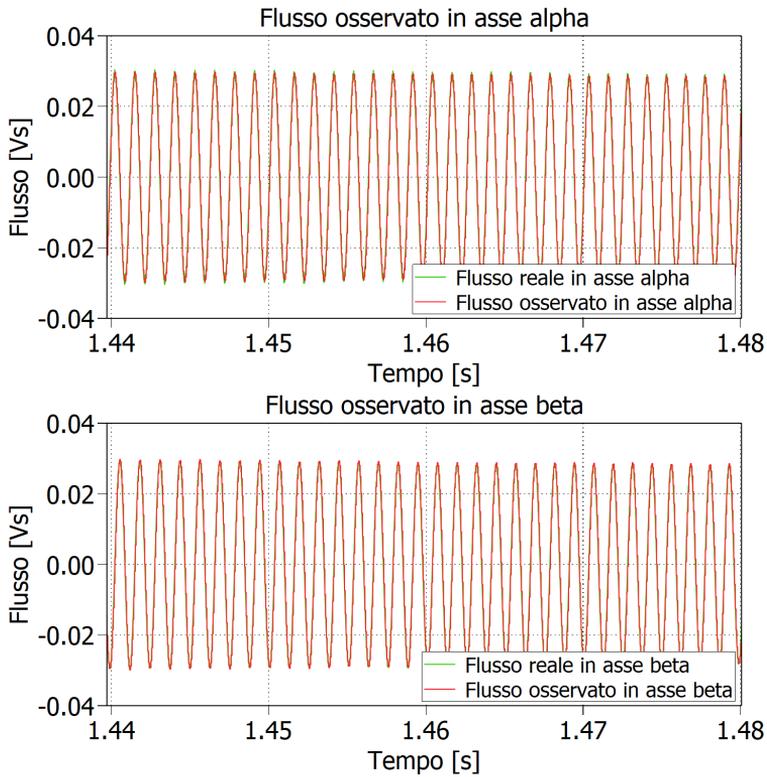


Figura 6.20: Zona di transizione tra le due diverse stime nell'osservatore di flusso (si ricorda $g = 500 \text{ rad/s}$).

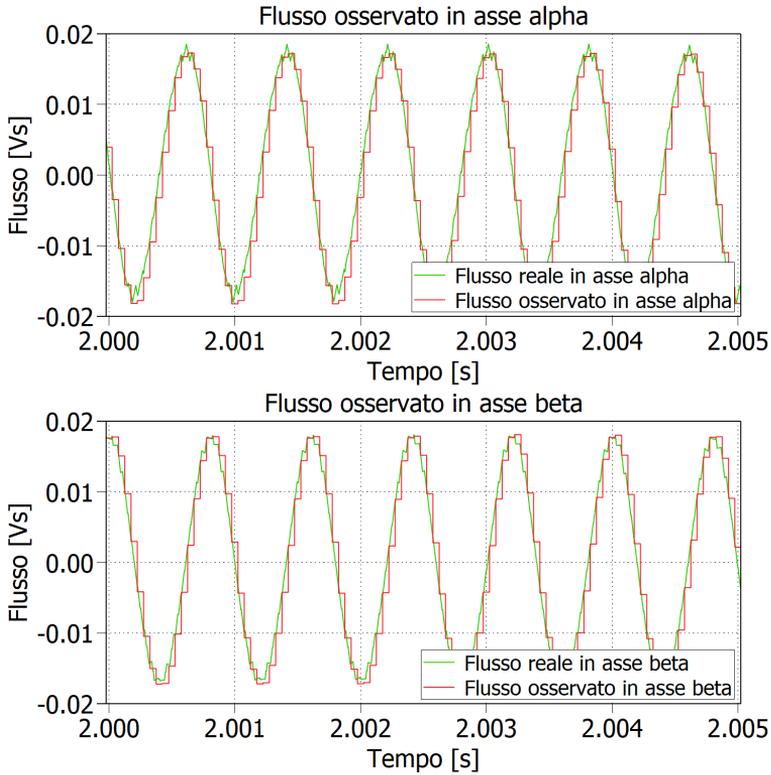


Figura 6.21: Nell’osservatore di flusso é implementata un’integrazione euleriana ritardata di un passo di campionamento.

Anche in Plecs viene confermato che questo motore non vada mai a lavorare sulla caratteristica di MTPV ricavata in linearit  magnetica ($\delta_{MAX} = 100^\circ$) come si pu  notare in Fig. 6.22. Questo risultato   un ulteriore verifica del fatto che l’anello di controllo per l’angolo di carico non entrer  mai in funzione.

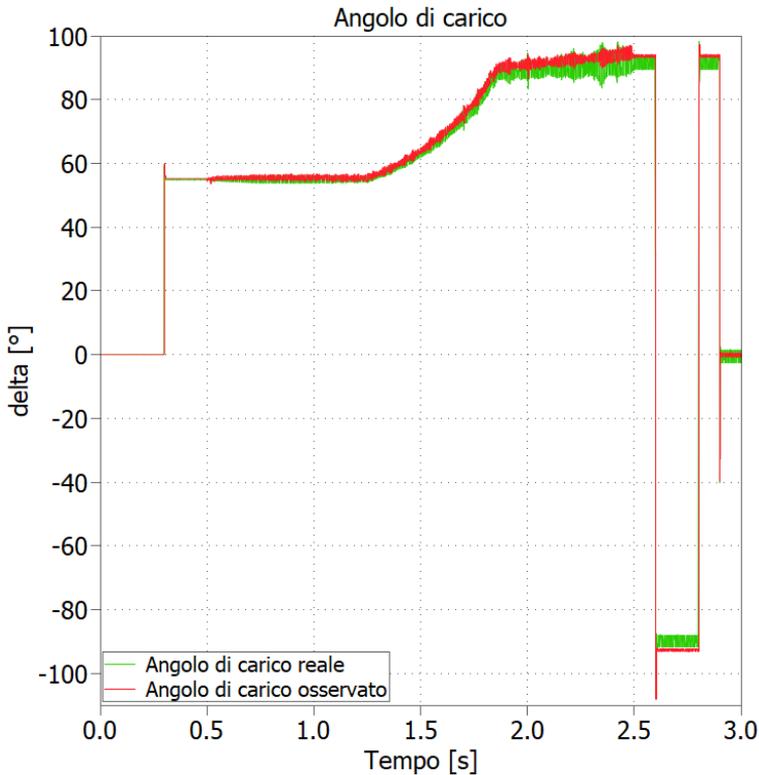
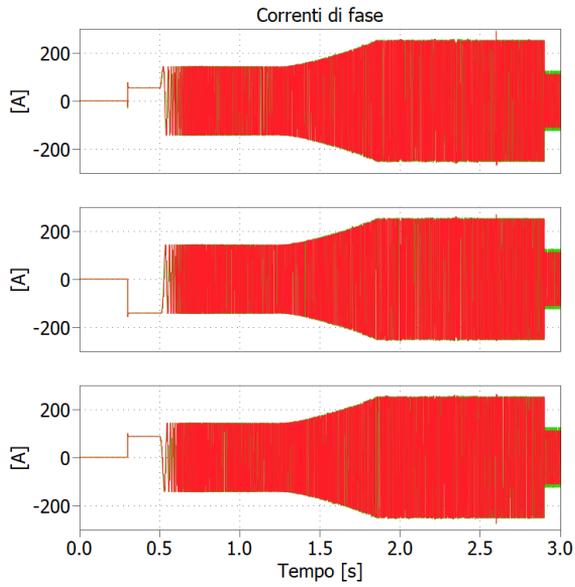
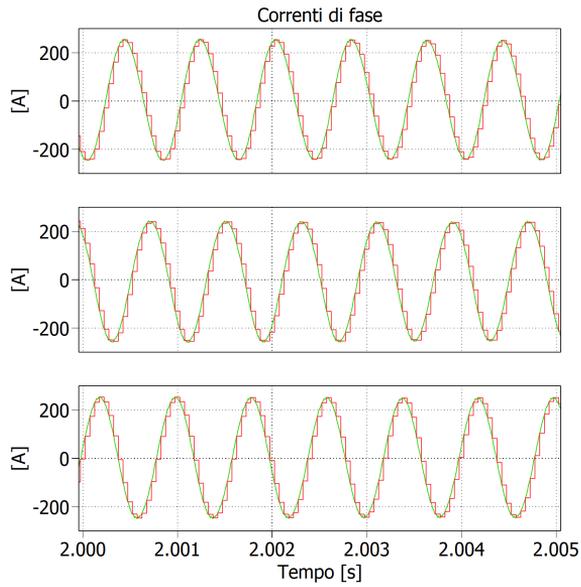


Figura 6.22

Segue una carrellata delle principali grandezze trifase dell'azionamento. Il controllo riesce a imporre delle correnti molto pulite, in termini di distorsione armonica, come si può anche vedere in Fig. 6.23, nonostante la presenza di un inverter reale e di una modulazione PWM. Mentre nella Fig. 6.25 si nota la *BEM*, come tecnica di modulazione implementata nel controllo, e l'importanza del coefficiente α (vedi Tab. 6.1), il quale evita la saturazione dei duty-cycles.



(a)



(b)

Figura 6.23: Nelle due rispettive figure: in verde le correnti reali di macchina e in rosso le correnti campionate per il controllo.

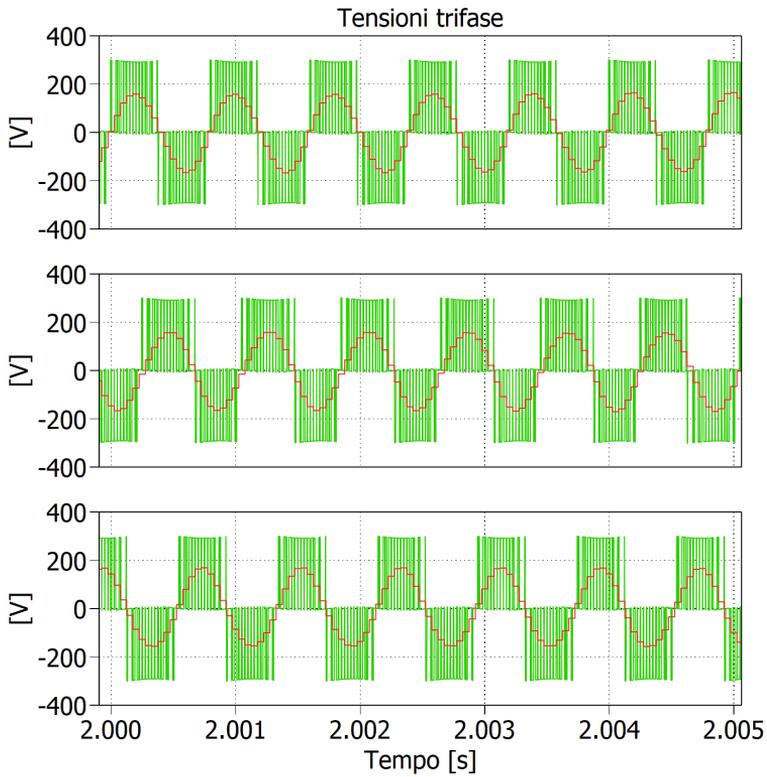
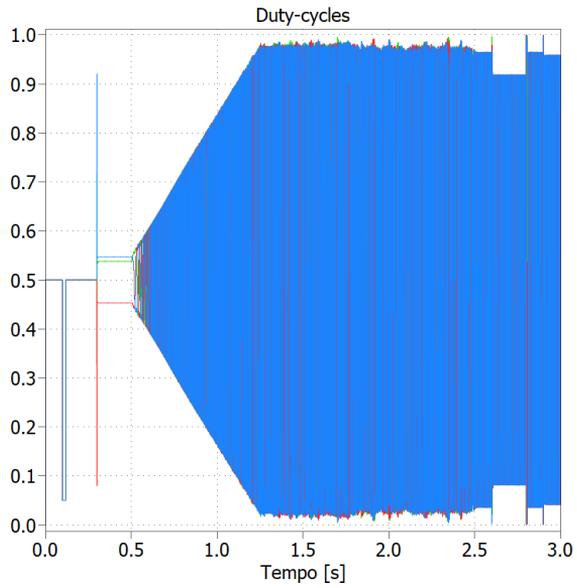
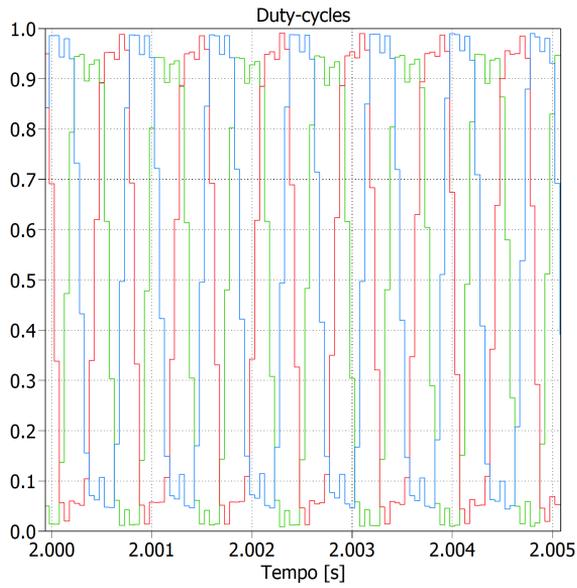


Figura 6.24: In verde le tensioni concatenate PWM imposte dall'inverter al motore trifase. In rosso le tensioni di fase di riferimento generate dal controllo.



(a)



(b)

Figura 6.25

Prima di concludere questa presentazione del test finale in Ples, si fa vedere un grafico che piú caratterizza la differenza tra i due ambienti di simulazione, ovvero i sei comandi logici per le tre gambe di inverter:

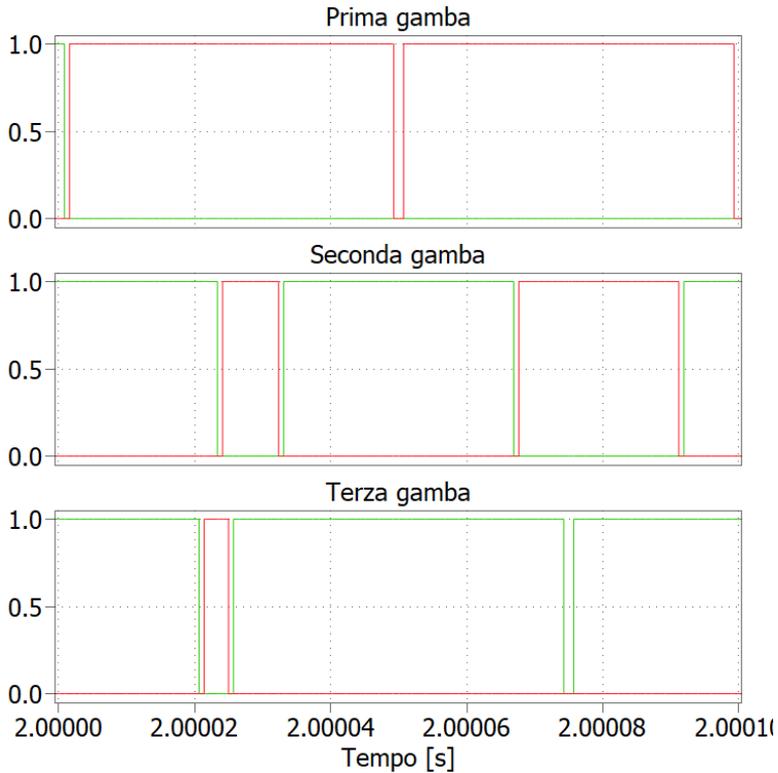


Figura 6.26: I sei comandi logici per le tre gambe di inverter. Per ogni plot si osservano i due comandi complementari di una singola gamba con un opportuno dead-time di ritardo (dichiarato in Tab. 6.1).

In chiusura si riporta, per confronto con il rispettivo grafico in Simulink, il grafico che mette in risalto l'utilità del controllo in *Phase Advancing* per azionamenti che prevedono elevate velocità.

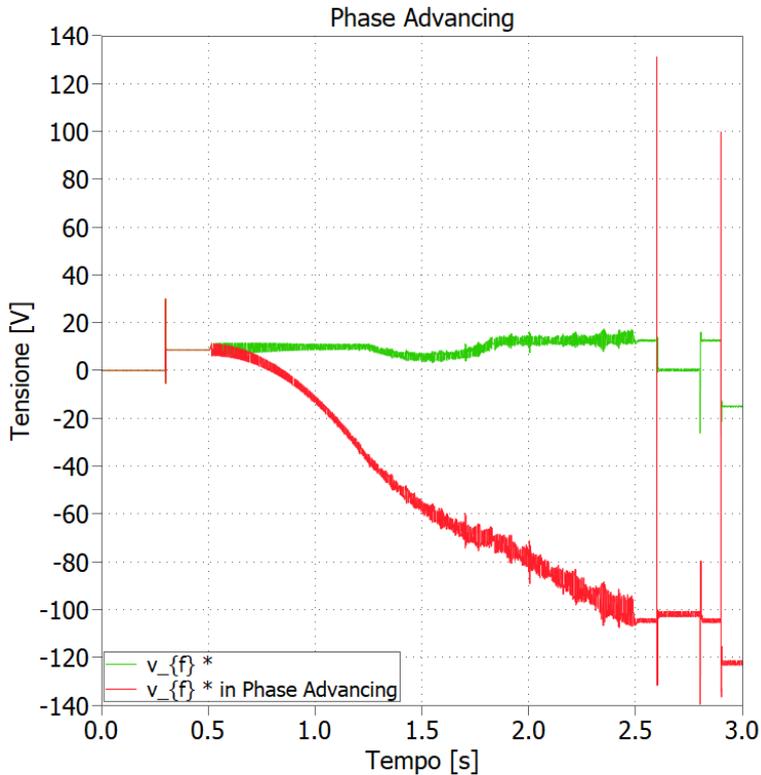


Figura 6.27

6.8.2 Test in condizioni di *UGO*

In ambiente Plecs, oltre ad eseguire il test ad elevatissima dinamica, si é potuto testare (potendo disporre di una modellistica elettrica) il comportamento dell'azionamento in condizioni di *UGO* (*Uncontroller Generator Operation*); ovvero perdita di controllo dell'azionamento, quindi dell'inverter, quando il rotore gira alla massima velocità.

In particolar modo si é voluto simulare il picco di corrente al il DC-link, causata da un motore che in questi istanti funzionerebbe da generatore con un convertitore che farebbe da raddrizzatore trifase a

commutazione naturale (lavorano solo i diodi), in quanto la concatenata al motore risulta essere maggiore della tensione al DC-link. Un test utile per chi si é occupato del progetto reale dell'inverter, dato che questa tesi ricade in un lavoro di team molto piú ampio come introdotto all'inizio di questa tesi.

La prova consiste di portare il motore in condizioni di massima velocità (20000 rpm) imponendo la coppia massima (35 Nm) ricavata dalla caratteristica di MTPA in linearitá magnetica. Giunti a questo punto di lavoro si disattiva il modulatore PWM aprendo istantaneamente gli interruttori. Si possono osservare i seguenti risultati:

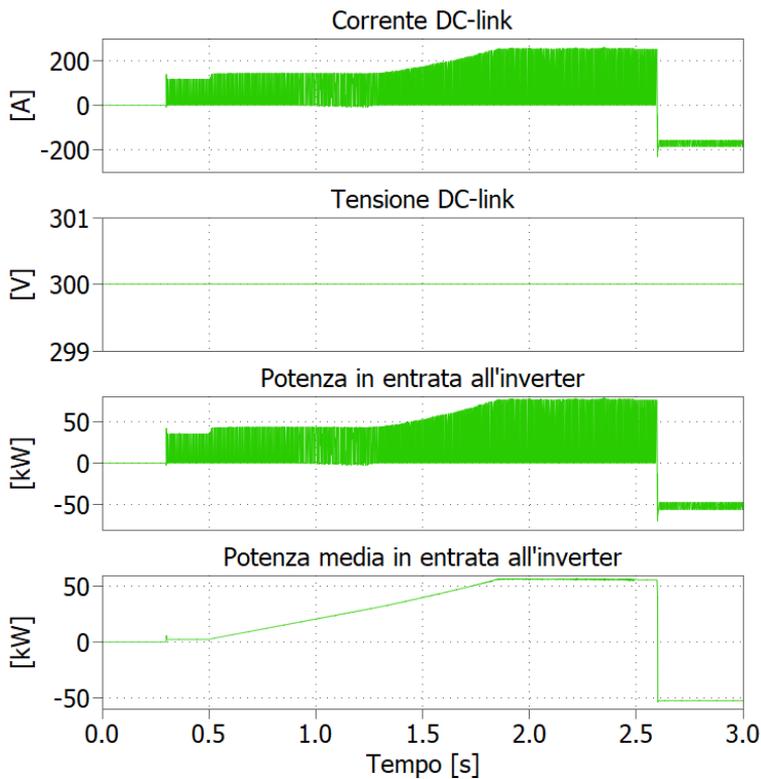


Figura 6.28

Naturalmente, nell'esaminare questi risultati, bisogna tenere conto della modellistica adottata, ovvero di un condensatore di DC-link modellizzato con un generatore di tensione; corretto se si vuole studiare una finestra temporale che non tenga conto del sovraccarico in tensione del condensatore, ma solo della sovracorrente dei primi istanti.

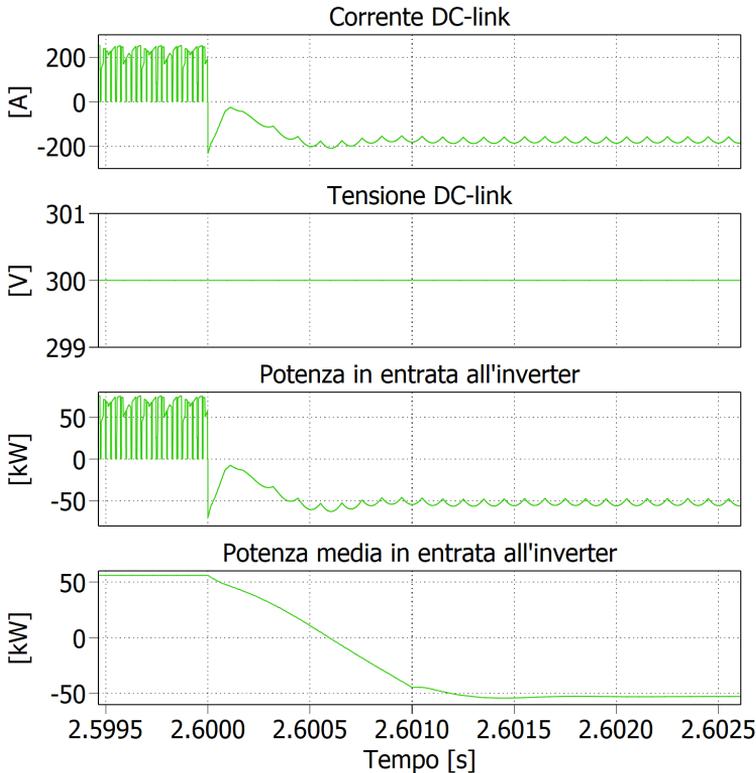


Figura 6.29: Ingrandimento della figura precedente in una ristretta finestra temporale. Si può notare, dopo la simulazione della perdita del controllo, come la corrente abbia una sesta armonica (avendo un raddrizzatore trifase naturale) della fondamentale, la quale corrisponde alla frequenza elettrica della velocità meccanica massima di 20000 rpm (ricordando $p=5$).

Capitolo 7

Conclusioni

Si giunge al termine di questo lavoro di tesi con un controllo in DFVC che ha brillantemente superato i vari test ad elevatissima dinamica, eseguiti in entrambe le simulazioni in Simulink e in Plecs. Sicuramente ha piú valore il test riuscito in Plecs in quanto ha una modellistica maggiormente vicina alla realtà, tuttavia la prima simulazione é stata necessaria per validare il codice. Ora il codice é pronto per essere testato sperimentalmente, però come anche anticipato nelle righe introduttive di questa tesi, per ritardi dovuti a complicazioni tecniche all'interno del lavoro in team, il codice sará testato su un motore alternativo della stessa tipologia di quello della Squadra Corse.

Il controllo in DFVC, volendo fare una previsione, potrà avere negli anni a venire un'importante diffusione nel campo degli azionamenti elettrici. Si é apprezzato come esso rientri in una logica di un controllo unificato di macchina, quindi applicabile ad ogni tipo di motore elettrico. Senza dubbio il suo punto di forza maggiore risulta essere l'elevata dinamica che conserva in punti di lavoro anche in deflussaggio spinto, grazie ad una semplice relazione algebrica che satura il flusso di riferimento da controllare. Questo aspetto emerge chiaramente nelle simulazioni dove il controllo riesce a gestire inversioni di coppia massima a gradino a velocità elevatissime; prestazioni difficilmente ottenibili da un controllo classico in FOC (Field Oriented

Control).

Infine si può fare un breve confronto tra i due diversi ambienti di simulazioni: Simulink e Plecs. Il primo vanta una potenza grafica e di elaborazione dati nettamente superiore, aiutato anche da un interfacciamento diretto su Matlab. Tuttavia dispone solamente l'implementazione di una logica a blocchi per questi tipi di lavori; infatti Plecs, se pur perdendo in grafica, si indirizza verso un nuovo concetto, dove in uno stesso ambiente di simulazione sono condensati più modellistiche implementabili, provenienti dal mondo elettrico, meccanico, termico e magnetico. Questo aspetto rende la modellizzazione sicuramente più veloce e più realistica.

Concludo questo elaborato di tesi parlando in prima persona manifestando tutta la mia soddisfazione per aver lavorato ad un progetto molto importante per la mia crescita, per la mia passione e per i miei sacrifici che in questi anni mi hanno guidato in questo percorso di studi.

Bibliografia

- [1] E. Armando, R. Bojoi, P. Guglielmi, G. Pellegrino e M. Pastorelli, "Experimental Identification of the Magnetic Model of Synchronous Machines", *IEEE Trans. Ind. Applicat.*, vol. 49, no. 5, 2013, pp. 2116-2125.
- [2] E. Armando, P. Guglielmi, G. Pellegrino e R. Bojoi, "Flux Linkage Maps Identification of Synchronous AC Motors Under Controlled Thermal Conditions", *IEEE International Electric Machines and Drives Conference (IEMDC)*, 2017.
- [3] G. Pellegrino, R. Bojoi e P. Guglielmi, "Unified Direct-Flux Vector Control for AC Motor Drives" *IEEE Trans. Ind. Applicat.*, vol. 47, no. 5, 2013, pp. 2093-2102.
- [4] G. Pellegrino, E. Armando e P. Guglielmi, "Direct Flux Field-Oriented Control of IPM Drives With Variable DC Link in the Field-Weakening Region" *IEEE Trans. Ind. Applicat.*, vol. 45, no. 5, 2009, pp. 1619-1627.
- [5] A. Vagati, G. Pellegrino e P. Guglielmi, "Comparison between SPM and IPM motor drives for EV application", *XIX International Conference on Electrical Machines (ICEM)*, Roma, 2010.
- [6] G. Pellegrino, A. Vagati, B. Boazzo e P. Guglielmi, "Comparison of Induction and PM Synchronous Motor Drives for EV Application Including Design Examples" *IEEE Trans. Ind. Applicat.*, vol. 48, no. 6, 2012, pp. 2322-2332.

- [7] R. Bojoi, E. Armando, G. Pellegrino e S. G. Rosu, "Self-commissioning of inverter non-linear effects in AC drives", Energy Conference and Exhibition (ENERGYCON), 2012 IEEE International, pp 213-218, 9-12 Sept. 2012
- [8] J. Holtz e J. Quan, "Sensorless Vector Control of Induction Motors at Very Low Speed Using a Nonlinear Inverter Model and Parameter Identification", IEEE Trans. Ind. Appl., vol. 38, no. 4, pp. 1087–1095, Jul./Aug. 2002.